

UNIVERSITY OF THE PHILIPPINES MANILA
COLLEGE OF ARTS AND SCIENCES
DEPARTMENT OF PHYSICAL SCIENCES AND MATHEMATICS

VECTOR-BORNE DISEASES MONITORING SYSTEM
(VBDMS)

A special problem in partial fulfillment
of the requirements for the degree of
Bachelor of Science in Computer Science

Submitted by:

Sherwin Keith B. Saringan

April 2014

Permission is given for the following people to have access to this SP:

Available to the general public	Yes
Available only after consultation with author/SP adviser	No
Available only to those bound by confidentiality agreement	No

ACCEPTANCE SHEET

The Special Problem entitled “Vector-Borne Diseases Monitoring System (VBDMS)” prepared and submitted by Sherwin Keith B. Saringan in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science has been examined and is recommended for acceptance.

Vincent Peter C. Magboo, M.D., M.Sc.
Adviser

EXAMINERS:

	Approved	Disapproved
1. Gregorio B. Baes, Ph.D. (<i>candidate</i>)	_____	_____
2. Avegail D. Carpio, M.Sc.	_____	_____
3. Aldrich Colin K. Co, M.Sc. (<i>candidate</i>)	_____	_____
4. Ma. Sheila A. Magboo, M.Sc.	_____	_____
5. Richard Bryann L. Chua, M.Sc.	_____	_____
6. Geoffrey A. Solano, M.Sc.	_____	_____
7. Bernie B. Terrado, M.Sc. (<i>candidate</i>)	_____	_____

Accepted and approved as partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science.

_____ Ma. Sheila A. Magboo, M.Sc. Unit Head Mathematical and Computing Sciences Unit Department of Physical Sciences and Mathematics	_____ Marcelina B. Lirazan, Ph.D. Chair Department of Physical Sciences and Mathematics
--	--

Alex C. Gonzaga, Ph.D., Dr.Eng.
Dean
College of Arts and Sciences

Abstract

Vector-borne diseases have been a major public health concern in most tropical areas, and an emerging threat for more developed countries. This study proposes a monitoring information system with the aid of geographic information system maps to help health officials in the surveillance of these diseases and for the immediate response and intervention.

Keywords: vector-borne diseases, chikungunya, dengue, malaria, vector-borne diseases mortality and morbidity, information system, disease surveillance, database management system, geographic information system

Contents

Acceptance Sheet	i
Abstract	ii
List of Figures	v
List of Tables	vii
I. Introduction	1
A. Background of the Study	1
B. Statement of the Problem	3
C. Objectives of the Study	4
D. Significance of the Project	7
E. Scope and Limitations	7
F. Assumptions	9
II. Review of Related Literature	10
III. Theoretical Framework	13
A. Vector-Borne Diseases	13
B. Department of Health (DOH)	16
C. Epidemiology and Disease Surveillance Systems	18
D. Information System / Database Systems	19
E. Geographic Information System (GIS)	20
IV. Design and Implementation	22
A. Context Diagram	22
B. Use Case Diagram	23
C. Flowchart	31
D. Entity Relationship Diagram	32
E. Data Dictionary	33

V.	Architecture	43
	A. System Architecture	43
	B. Technical Architecture	44
VI.	Results	45
	A. General View	45
	B. Administrator View / National Health Official View / Provincial Health Official View / Municipal/City Health Official View . . .	51
	C. Barangay Health Worker View / Municipal/City Health Worker View / Provincial Health Worker View	56
	D. Researcher View	59
	E. Generated Report	61
	F. Generated Dataset in Spreadsheet	62
	G. Generated SMS	63
VII.	Discussions	64
VIII.	Conclusions	66
IX.	Recommendations	67
X.	Bibliography	68
XI.	Appendix	72
	A. Source Code	72
	1. XML Files	72
	2. Controllers	73
	3. Models and DAOs	86
	4. Utilities	105
	5. Views	113
XII.	Acknowledgement	176

List of Figures

1	Researcher Reports Page, VBDMS	16
2	Level 0 Context Free Diagram	22
3	Use Case Diagram	23
4	Use Case Diagram for <i>Barangay Health Workers</i>	24
5	Use Case Diagram for <i>Municipal/City Health Workers</i>	25
6	Use Case Diagram for <i>Provincial Health Workers</i>	26
7	Use Case Diagram for <i>Municipal/City, Provincial and National Health Officials</i>	27
8	Use Case Diagram for <i>Researchers</i>	28
9	Use Case Diagram for <i>Public Users</i>	29
10	Use Case Diagram for <i>All System Administrators</i>	30
11	System Flow Chart	31
12	Entity Relationship Diagram	32
13	VBDMS System Architecture	43
14	Home Page, VBDMS	45
15	Login Page, VBDMS	46
16	Registration Form Page, VBDMS	47
17	Disease Page, VBDMS	48
18	Disease Statistics Page, VBDMS	49
19	Awareness Campaign Page, VBDMS	50
20	Alerts Page, VBDMS	51
21	Registration Page, VBDMS	52
22	Health Worker and Officials Page, VBDMS	53
23	Administrator, National, Provincial, Municipal/City Health Officials Reports Page, VBDMS	54
24	Administrator, National, Provincial, Municipal/City Health Officials Profile Page, VBDMS	55
25	Manage Records Page, VBDMS	56

26	Barangay, Municipal/City, Provincial Health Workers Reports Page, VBDMS	57
27	Barangay, Municipal/City, Provincial Health Workers Profile Page, VBDMS	58
28	Researcher Reports Page, VBDMS	59
29	Researcher Profile Page, VBDMS	60
30	Generated Report, VBDMS	61
31	Generated Spreadsheet, VBDMS	62
32	Generated SMS, VBDMS	63

List of Tables

1	User table	33
2	User_Logs table	33
3	Registrations table	34
4	Accounts table	35
5	BarangayRecords table	36
6	BarangayRecords table (Continuation)	37
7	MuncityRecords table	37
8	MuncityRecords table (Continuation)	38
9	Patients table	39
10	Patients table (Continuation)	40
11	Map table	41
12	Vector-Borne Diseases table	41
13	Alerts table	42

I. Introduction

A. Background of the Study

Vector-borne diseases are infectious diseases which can be transmitted by various vectors (e.g insects, ticks). They represent a major public health concern in most tropical areas, and an emerging threat for more developed countries [1]. In the country, these diseases are most commonly brought by mosquito vectors [2].

Vector-borne diseases specifically have been one of the leading causes of morbidity and mortality in human and animal populations with over 50 million that were accounted in 2004 for disability-adjusted life years (DALYs) [3]. Malaria, dengue and the Gambian form of human African trypanosomiasis, which are mosquito-borne diseases with only human hosts, all together affect over 250 millions people and kill around 900,000 humans every year. Other non-human host mosquito-borne diseases, like Japanese encephalitis and Chagas disease, are responsible for more than 50,000 human deaths a year, and incapacitate several hundred thousands people [1].

The vector-borne diseases with the most prevalence in the country are chikungunya, dengue and malaria. Chikungunya is a mosquito-borne disease which shares clinical signs with dengue but not as deadly. Recently, there was a reported outbreak of chikungunya in Patnongon, Antique having 313 cases from April to June 2013 with no reported deaths. [4]

Dengue ranks as the most important mosquito-borne viral disease in the world with up to 50 million infections occurring annually and 22, 000 deaths mainly among children [5]. About 80, 745 cases of dengue were identified and 496 cases of which led to death from the timeline of January 1 to August 18, 2012 only in the Philippines [6].

Malaria is one of the endemic vector-borne diseases not only in the country but in the world. In 2010, an estimated 216 million clinical episodes and 655, 000 deaths caused by malaria globally were recorded. And 6% of those recorded

deaths were from Southeast Asian Region [7]. In 2011, there were only 9, 642 malaria cases in the country which is a significant decrease compared to 43, 441 cases in 2003 [8].

Surveillance of these vector-borne diseases in the Philippines is conducted by health workers in a hierarchical order. Health workers from barangays conduct surveys and collect data of the incidence of these diseases in the community on a monthly basis. Health workers in municipalities are responsible for verifying these records sent by barangay health workers and keeping them. The information gathered will be sent to the provincial offices. Provincial health workers will then encode and put these data on the Philippine Integrated Disease Surveillance and Response (PIDSR) which is managed by the Department of Health (DOH). The local government has an active role of detecting diseases and undertaking response actions while the regional and national levels will provide the needed support, the policies, guidelines, and the training as well. [2]

PIDSR is a stand-alone information system which serves as the departments central repository of almost 22 diseases in the country including some vector-borne diseases like dengue and malaria. Its main purpose is data recording of patients which includes patient information (full name, age, sex, date of birth, complete address) and clinical information (attending hospital/clinic, type of hospital/clinic, hospital/clinic address, date admitted, date of onset, morbidity week, type and case classification, outcome and date died) having vector-borne diseases. As they populate the system with records of patients, they use a separate software application, EpiInfo, to compute for the statistics needed to be included in the reports. EpiInfo can also generate maps but this functionality isnt implemented by the DOH in their generation of reports of the diseases. Users of this system only include the DOH staffs mainly of the National Epidemiology and Surveillance Center and provincial health officials. It is deployed in selected province only. [2]

The Department of Health also implemented the Philippine Malaria Information System (PhilMIS) which was developed to provide information needed for

planning, implementation, monitoring and evaluation of malaria control program in a computerized form. The PhilMIS is also a stand-alone information system which basic operations include data recording and entry of records of malaria cases (age, sex, barangay, date of onset of disease, date died, and inclusive date of survey) and generation of statistical reports. In 2005, PhilMIS was deployed in a total of 17 provinces. PhilMIS software was revised in 2007, and is released as PhilMIS2. A total of 30 malaria endemic provinces were installed with PhilMIS2. The users of the PhilMIS2 include DOH staffs and provincial health officers only. It is deployed in provincial level but in selected provinces only as well. [9]

B. Statement of the Problem

Vector-borne diseases are endemic to tropical countries like Philippines. Monitoring these diseases is not easily done and information through monitoring is not that much accessible to the public.

Though there are many information systems being used by the Department of Health (DOH) to keep records of these diseases, some of those systems are deployed only in the machine or work stations of users and updates in the system are done by manually receiving patches from the main office via email. Its main users are the DOH officials and health officials in provincial levels only. The DOH, being responsible for informing the public about the health status in the country, provides information in their DOH website and also in the news. But the information with regard to the diseases, not only vector-borne diseases, is mostly out of date and not updated even in a monthly or quarterly basis.

DOH initially planned to develop web-based information systems for disease monitoring hence, their beta version but because of some program errors with the system, the implementation was cancelled. [2]

Researchers, epidemiologist and other disease analyst, may have difficulties in obtaining data for their researches as for these data are only exclusively provided by DOH. Ease of access of data will help researchers to easily focus on investigation

of possible outbreaks, disease surveillance and other health related actions.

Determining and manually mapping areas in the country where there is a high occurrence of a disease is tedious given that there can be thousands of persons who may have caught the disease since gathering information, like having survey from house-to-house or getting records from hospitals/clinic, is already pretty exhaustive for health workers especially when repeating the process monthly. With the unavailability of up-to-date information of these diseases, the public are much more likely to be unaware of the status of these diseases in the country until it becomes endemic or announced in the news. We have limited access to an updated and real-time monitoring of these diseases especially in areas which are more susceptible to these diseases.

These diseases cannot be exterminated in the near future. Besides global climate change can bring about possible changes in the prevalence of mosquito vector-borne diseases especially in temperate zones and higher altitudes in tropical countries like Philippines [10]. So as long as the vectors of the disease exist, there is always a chance that a person will be infected thus, the more needed it is to monitor these diseases.

C. Objectives of the Study

This study aims to develop an online monitoring information system for vector-borne diseases, specifically for chikungunya, dengue, and malaria, with the following functionalities:

1. Allow all levels of health workers and health officials to:
 - (a) Register to the system. He/She can log into the system once approved by the system administrator.
 - (b) Generate reports that may include specific disease statistics (morbidity, geographic distribution), GIS maps and charts (no. of cases by the specified date, disease case by age group and gender, disease case

fatality ratio by age group, disease case and death) from specific area/s and from specific date/s.

(c) Change his/her account password.

2. Allow barangay health workers to enter the data for the vector-borne diseases which will then be verified by the municipal health workers.

3. Allow municipal/city health workers to:

(a) Check the data submitted by the barangay health workers and approve these data after verification and checking.

(b) Send the data to provincial health workers after verification and approval.

4. Allow provincial health workers to:

(a) Check the data submitted by the municipal health workers and approve these data after verification and checking.

(b) Update the system after verification and checking of data gathered from lower tier health workers.

5. Allow municipal/city, provincial and national health officials to:

(a) Be notified whenever there is a near-case of epidemic of a disease in a certain area via SMS and message prompt in their user account

(b) Automatically redirect the GIS map into the users location through their assigned location and show the map of the area near it.

(c) Monitor/Manage the user accounts of health officials and health workers within their location.

6. Allow a researcher to:

(a) Register to the system. He/She can log into the system once approved by the system administrator.

- (b) Export in spreadsheet the data sets acquired by the health workers.
- (c) Generate reports that may include specific disease statistics (morbidity, geographic distribution), GIS maps and charts (no. of cases by the specified date, disease case by age group or/and gender, disease case fatality ratio by age group, disease case and death) from specific area/s and from specific date/s.
- (d) Change his/her account password.

7. Allow public users to:

- (a) Automatically redirect the GIS map into the users location through their IP address and show the map of the area near it.
- (b) View the statistics (morbidity, geographic distribution) of the specific disease in specific area/s and from specific date/s.
- (c) View awareness campaign by DOH about vector-borne diseases (preventions, symptoms and possible first-aid treatments).

8. Allow national, provincial and municipal/city system administrators to:

- (a) Activate/Deactivate researchers or health officials accounts within their location.
- (b) View logs of activities of the researcher and health officials within their location.
- (c) Export in spreadsheet the data sets acquired by the system that are input by health workers.
- (d) Generate reports that may include specific diseases statistics (morbidity, geographic distribution), GIS maps and charts (no. of cases by the specified date, disease case by age group and gender, disease case fatality ratio by age group, disease case and death) from specific area/s and from specific date/s.
- (e) Change his/her account password.

D. Significance of the Project

This monitoring information system will help in monitoring vector-borne diseases in real-time and provide up-to-date information regarding the incidence of a vector-borne disease which may come in handy to determine possible outbreaks.

It can also help researchers and epidemiologist to gather data for research since data on these diseases are not that much accessible. And reports generated that may include statistics of the disease will also help the researchers study the incidence of these diseases and their possible impacts to the community.

Real-time and easily accessible monitoring of vector-borne diseases can assist health officials to pinpoint areas where there is a possible endemic of a disease and make the necessary course of actions and interventions. Availability of monitored information of the said diseases especially online will raise the awareness of the public regarding the disease and will help them be more cautious about their surrounding environment.

The use of maps will assist in identifying areas with the most number of cases of a disease. The public can also make use of this map to help them see whether there are a lot of cases near them and can invoke necessary precautions and might as well inform the barangay officials to perform immediate response and interventions as deemed necessary by the situation at hand.

E. Scope and Limitations

For this study, the researcher will produce a system under the following conditions:

1. This project will initially consist of the vector-borne diseases: chikungunya, dengue, and malaria.
2. This project will only accept data sets input in the interface provided by the system in order to prevent non-uniform use of file types in the inputs and these data sets will include the patients age, sex, date of birth, complete address, attending hospital/clinic, type of hospital/clinic, hospital/clinic ad-

dress, date admitted, date of onset, type and case classification, outcome and date died.

3. This project will only export data sets including patients age, gender, barangay, disease type and patient status to protect the patients confidentiality.
4. This project will generate reports in a .pdf format.
5. The approval and deactivation of the researcher registering to the system will be solely decided by the national system administrator, and he/she informs the researcher regarding the approval of the registration.
6. This project will have a countrywide scope for the cases of the listed vector-borne diseases.
7. The map to be used in this project is static.
8. The project will reflect the cases of diseases in the map only for a month.
9. Data collection comes from surveys done by barangay health workers on households/barangay health center/hospitals for cases/incidence of diseases and is done on a monthly basis. Data gathered are sent to municipal health workers for checking and verification, and is sent to provincial health offices. Provincial health workers check and verify these data and update their records as well as the information systems and also send these data to DOH.
10. Other vector-borne diseases can be later added.

F. Assumptions

The following are the assumptions that the researcher includes in the study:

1. There is only one account (username and password) for the system administrator in the national level, and one system administrator for each province and municipality/city.
2. The system administrator uses as his/her primary basis for the approval of researcher accounts the following information provided by the researcher upon registration: the institution they belong to, significant people who have possibly referred them to the system and their research interest.
3. Barangay health workers regularly collect vector-borne disease data as mandated by law.

II. Review of Related Literature

Improvements in information technology (i.e mapping, and geographic information systems) also improves the capacity to predict, prevent and control vector-borne and other infectious diseases. [11, 12]

In Nicaragua, vector control begins at municipality level where public health workers from the Ministry of Health survey homes to calculate the mosquito infestation per neighborhood and identify sites of potential larval development. Data collected from the public health surveyors was then entered into the map and utilized to calculate several indices of larval infestation. Once information on the locations of larval development sites and dengue cases had been input on the map, spatial patterns among the known dengue cases, mosquito larval indices, and the previously identified larval development sites were visually analyzed for patterns. As a result, it is possible to prioritize control strategies and to target interventions to highest risk areas in order to eliminate the likely origin of the mosquito vector. [13]

In Saudi Arabia, researchers created models of areas with humans at risk of dengue fever prevalence based on spatial relationship between dengue fever cases and different socioeconomic parameters (e.g population numbers, population density, and neighborhood quality). These were analyzed using Geographically Weighted Regression (GWR) to create a prediction model identifying levels of risk of dengue and the diseases association with the fore-mentioned socioeconomic factors. Geographic Information Systems (GIS) was used to characterize the socioeconomic parameters like calculating population density and determining the number of dengue fever cases in that layer. [14]

Another study conducted in Jeddah, Saudi Arabia, which aimed to use geographical information systems to demonstrate the Dengue fever (DF) risk on a monthly basis in the said setting with the purpose to provide documentation to improve surveillance and monitor the *Aedis aegypti* mosquito vector. The data on clinical cases of DF were obtained from the Jeddah Health Affairs department

which keeps record of confirmed cases of DF since 2006. The records include age, sex, nationality, district, coordinates and the week of disease onset for each case. [15]

In North America, National Electronic Disease Surveillance Disease was implemented by the United States Centers for Disease Control and Prevention to promote rapid collection of standardized disease data. Another one is the multi-disease data management system platform with capacity for dengue and malaria which is capable of enhancing the users ability to carry out continuous surveillance, monitor interventions, evaluate control program performance, and engage in evidence-based decision making and includes production of maps and both standard and customized reports. [11]

The VBD-AIR tool by Huang, et al. [16] helps better define the roles of airlines in the transmission and spread of disease. Air travel likely moves vector-borne diseases (via infected passengers) than in moving the vector. Up to 8 % of travelers to the developing world become ill enough to seek healthcare upon returning home, with a significant proportion of these suffering from vector-borne infections. This study focuses only on four common vector-borne diseases namely malaria, dengue, yellow fever, and chikungunya. VBD-AIR integrates data sources from airport locations and air routes, disease and vector distributions, global climate data, and global land-based travel time data.

A study by Yang, et al. [17], a nationwide web-based automated system for outbreak detection and rapid response was developed in 2008 in China. The China Infectious Disease Automated-alert and Response System (CIDARS), developed by the Chinese Center for Disease Control, facilitates data analysis, aberration detection, signal dissemination, signal response and information communication needed by public health departments across the country. CIDARS has been used nationwide since 2008 and assists on early outbreak detection at the local level prompts reporting of unusual disease occurrences to CDCs throughout the country.

The study of Lars Eisen and Rebecca Eisen [12] focuses on how advances in

mapping, Geographic Information System, and Decision Support System technologies, and progress in spatial and space-time modeling, can be harnessed to prevent and control vector-borne diseases. Its major themes include importance of using high-quality and biologically or epidemiologically relevant data, incorporation of new technologies into operational vector and disease control programs, transfer of map-based information to stake-holders, and adaptation of technology solutions for use in resource-poor environments.

In the study done by Amoli [18], GIS is used to perform statistical and spatial analysis on a retrospective database, including the total number of incidence of Leishmaniasis in Isfahan, Iran between the year 2007 to 2009. It produces a map showing the relative risk of each area for this disease. The developed risk map can be used by the Public Health Department for the valuable information it provide can help in the planning of precautionary measures and for controlling the spread of Leishmaniasis.

This study will develop a web-based system which uses GIS maps so that health professionals can also record the information of vector-borne diseases into GIS system to visually display on the electronic map just like in the study done by Zhen, et al. [19] so that proper interventions can be adopted or the corresponding early warning program will be started, targeted to prevent further spread of epidemic.

III. Theoretical Framework

A. Vector-Borne Diseases

Vector-borne diseases are infections transmitted by the bite of infected arthropod species such as mosquitoes, ticks, etc. The arthropod that transmits these diseases through feeding activity is termed as vector [20, 21]. This study focuses on the vector-borne diseases chikungunya, dengue and malaria.

1. Chikungunya

Chikungunya is a viral disease that causes fever and severe joint pain. Sharing clinical signs with dengue, it is usually misdiagnosed in areas where dengue is common. It is transmitted by infected mosquitoes, usually the genus *Aedes*. [22]

(a) Signs and Symptoms

Aside from abrupt onset of fever frequently accompanied by joint pains, other signs and symptoms of chikungunya include muscle pain, headache, nausea, fatigue and rash. Joint pain usually lasts within a few days or weeks and while most patients recover fully from this, some cases may persist for several months, or even years. Occasionally, cases of eye, neurological and heart complications as well as gastrointestinal complaints have been reported. This disease not often cause serious complications but can contribute to death among older population. [22]

(b) Treatment

Since there are no specific drugs to cure the disease, relieving the symptoms including the joint pains are directed primarily. There is no commercial chikungunya vaccine. [22]

(c) Health Offices' Action/Advise to Public

Health officers provide information dissemination for residents to clean their surroundings and get rid of mosquito-breeding sites.

2. Dengue

Dengue is a mosquito-borne viral infection that causes flu-like illness, and occasionally develops into a potentially lethal complication called severe dengue. It is found in tropical and sub-tropical climates and mostly in urban and semi-urban areas. It is transmitted by an infected *Aedes* mosquito (*Aedes aegypti* or *Aedes albopictus*). [23]

(a) Signs and Symptoms

When running a high fever ($40^{\circ}\text{C}/100^{\circ}\text{F}$) and is accompanied by two symptoms, namely, severe headache, pain behind the eyes, muscle and joint pains, nausea, vomiting, swollen glands or rash, dengue should be suspected. These symptoms usually last for 2 to 7 days; that is after 4-10 days of incubation period after the bite from an infected mosquito. A potentially deadly complication from dengue, severe dengue, is due to plasma leaking, fluid accommodation, respiratory diseases, severe bleeding, or organ impairment. After 3-7 days when the first symptoms in conjunction with a decrease in temperature of below $38^{\circ}\text{C}/100^{\circ}\text{F}$, warning sign occurs that includes severe abdominal pain, persistent vomiting, rapid breathing, bleeding gums, fatigue, restlessness and blood in vomit. Proper medical care is needed to avoid complications and risk of death especially in the next 24-48 hours of the critical stage. [23]

(b) Treatment

There is no specific treatment for dengue fever. For severe dengue, medical care by physicians and nurses experienced with the effects and progression of the disease can save lives. Maintenance of the patients body fluid volume is critical to severe dengue care. [23]

(c) Health Offices' Action/Advise to Public

Health officers provide comprehensive information dissemination for residents to clean their surroundings and get rid of mosquito-breeding sites.

3. Malaria

Malaria, though preventable and curable, is a life-threatening disease caused by parasites that are transmitted to people through the bites of infected mosquitoes. The burden of malaria in many places has been dramatically reduced by increased malaria prevention and control measures. The Plasmodium parasite causes malaria to humans and is transmitted by Anopheles mosquitoes. [24]

(a) Signs and Symptoms

Malaria is an illness characterized with acute fever. Symptoms appear 7 days or more (10-15 days) for non-immune individuals after being bitten by an infected mosquito. The first symptoms include fever, headache, chills and vomiting which may be mild and difficult to be recognized as malaria. Residents in malaria endemic areas may develop partial immunity which allows asymptomatic infections to occur. [24]

(b) Treatment

Early diagnosis and treatment of malaria reduces disease and prevents deaths. It also contributes to reducing malaria transmission. The best available treatment, particularly for *P. falciparum* malaria, is artemisinin-based combination therapy (ACT). [24]

(c) Health Offices' Action/Advise to Public

Health officers provide comprehensive information dissemination for residents to clean their surroundings and get rid of mosquito-breeding sites.

B. Department of Health (DOH)

1. Organizational Structure

The organizational structure of the country's Department of Health (DOH) is shown below:

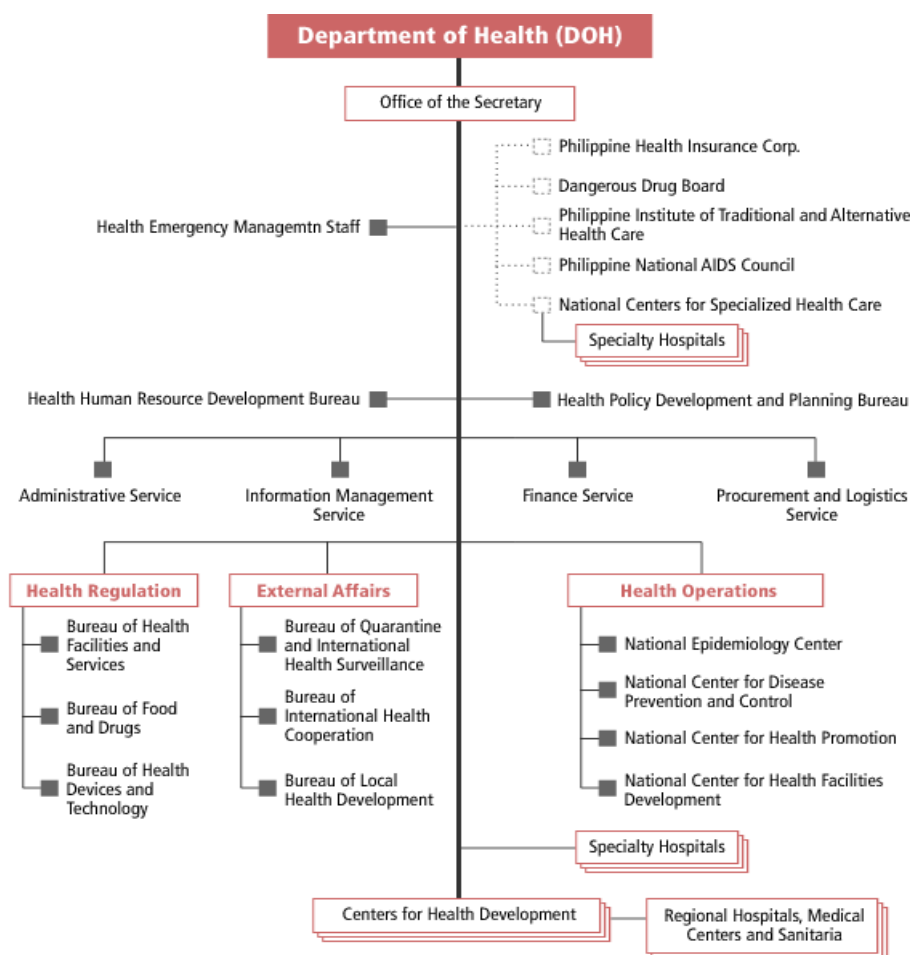


Figure 1: Researcher Reports Page, VBDMS

The Department of Health is headed by the Secretary and under the Office of the Secretary are the listed centers offices and services. The centers in charge of health operations are the National Epidemiology Center, National Center for Disease and Prevention Control, National Center for Health Promotion, and National Center for Health Facilities Development. There are Centers for Health Development in every region and affiliated with these centers are Regional Hospitals, Medical Centers and Sanitaria.

2. Data Collection

Data collection for cases of vector-borne diseases in the country are observed and recorded by the health workers in barangay level on a monthly basis. Collected data will be sent to municipal/city health office for checking. After verification, these data will be sent to provincial health offices. Provincial health officials will check the data, and input these on systems deployed in the office (like PIDS and PhilMIS) and will also be sent to national office which is the Department of Health (DOH). [2]

3. Prompt Health Actions

In the public sector, the Department of Health (DOH) delivers tertiary services, rehabilitative services and specialized healthcare, while the local government units (LGUs) deliver health promotion, disease prevention, primary, secondary, long-term care. Primary health services are delivered in barangay health stations, health centers, and at hospitals.

Case Management and Treatment: Health care providers in the Barangay Health Stations (BHS) and Rural Health Units (RHUs) were given training on Basic Malaria Management.

Vector Control: Vector control measures designed to prevent transmission of disease by aiming for the vectors were intensified through distribution of insecticide-treated nets (ITNs) with 1 net to 2 person ratio and provision of insecticide for complementary indoor residual spraying. [25]

4. Official Duty in Disease Surveillance

The countrys Bureau of Quarantine and International Health Surveillance under the DOH implements mandates to help in accomplishing a mission of having an effective surveillance and control measures of infectious diseases and other health concerns with global impact through local and international networking. [26]

One of these mandates, adopted on July 23, 1969 during the 22nd WHA, is the International Health Regulations (2005) of the WHO. This IHR covers activities ranging from surveillance and response, to notification and verification to WHO of certain public health events and risks, to rules on application of health measures and development of minimum public health capacities for surveillance, assessment, response, and reporting for a broad range of risks throughout the territories of all States Parties. [27]

Another one is the Republic Act 3573 or the Law of Reporting of Communicable Diseases which requires all individuals and health facilities to report notifiable diseases to local and national health authorities. [28]

The Resolution WHA48.13 year 1995 urges Member States to strengthen national and local programs of active surveillance for infectious diseases, ensuring that efforts were directed towards early detection of epidemics and prompt identification of new, emerging and re-emerging infectious diseases. [29]

Administrative Order No. 2005-0023 or the Implementing Guidelines for Fourmula One for Health as Framework for Health Reforms Section C2.c.iii which states that, Disease surveillance shall be intensified to ensure that the targets for disease elimination, prevention and control are attained. [30]

C. Epidemiology and Disease Surveillance Systems

The National Epidemiology Center of the Department of Health is primarily responsible for assessing the health status of Filipinos, detecting or confirming disease outbreaks and implementing outbreak control measures including but not limited to rapid containment. They are also in charge of collecting, analyzing and disseminating reliable and timely information on the health status of the population in the country. Investigation of disease outbreaks and other threats to the public health are also done by them, as well as development and evaluation of

surveillance and other health information systems. Other responsibilities of the National Epidemiology Center include advising the Secretary and Undersecretary of Health on matters pertaining to epidemiology and health surveillance. [26]

Epidemiology is the study concerned with the distribution and determinants of health-related states, events or disease and the application of this study to the control of diseases and other health problems. There are various methods that are used in carrying out epidemiological investigations and one of these is through surveillance. [31]

Disease surveillance is a continuous and systematic process with the primary activities of collection, analysis and dissemination of various data that describe the occurrence and distribution of disease, events or condition. Surveillance includes the collection of relevant data for a specified population, time period and/or geographic area, meaningful analysis of data and dissemination of data with accompanying interpretation. Disease surveillance is an important process that determines the extent of infections and the risk of disease transmission so that prevention and control measures can be applied effectively and efficiently to minimize the burden of illness. To accurately reflect the occurrence and distribution of disease, surveillance data are needed to be timely and complete. [32]

D. Information System / Database Systems

An information system is a set of interacting components (people, procedures and technologies) that together collect, process, store, and distribute information to support control, decision-making and management in organizations. Information systems personnel include all the people who manage, run, program, and maintain the system. Users include people who work with information systems to get results. The strategies, policies, methods and rules for using information system are known to be procedures. It includes the operation, maintenance and security of the computer. And the technologies, including all the software, hardware databases, telecommunications, people and procedures, are configured to collect, manipulate,

store, and process data into information. [33, 34]

In improving the quality and efficiency of healthcare, developing secure and efficient information systems for the healthcare industry plays a vital role. Healthcare information systems are an information intensive industry in which reliable and timely information is a critical resource for the planning and monitoring of service provision. Health information system collects data as a part of the patient care process. This data can be used for many different purposes across a number of systems. The patient data must be subject to confidentiality and must integrate with data from other facilities and must meet the needs of various professional groups. With the improvements in healthcare information up to this day, healthcare delivery has been more effective and far more efficient. [35]

A database system is a system for storing, processing and retrieving data or information with in a central item, the database, and its components include the software, hardware and users. A DBMS is a collection of interrelated data and a set of programs that manage or control the use of data and provides an interface between the database and its users. Its primary objective is to provide means to organize data in storage media and to enable users with easy and efficient access mechanisms. Major roles of a modern database management system are data structure definition for data storage, providing suitable mechanism for data access and data manipulation, maintain system integrity, provide safety and security measures for data, and to provide mechanism for data sharing among users concurrently. [36]

E. Geographic Information System (GIS)

An organized collection of computer hardware and software, people, money and organizational infrastructure that makes possible the acquisition and storage of geographic and related attribute data is called a geographic information system (GIS). It serves the purpose of retrieval, analysis, synthesis, and display to promote understanding and to assist in decision making. It is a digital map-based

technology commonly integrated with a database management system (DBMS) to display information, and spatial analysis. GIS combined with other information systems makes a powerful tool for location mapping, and decision making. [37, 38, 39]

Geographic information systems are being used in public health to model where people live and the environments they experience. GIS provide a digital lens for exploring the dynamic connections between people, their health and well-being, and changing physical and social environments. GIS provide ideal platforms for the convergence of disease-specific information and their analyses in relation to population settlements, surrounding social and health services and the natural environment. However, at present, most of the applications are static, which means that the process of data collection, management, analysis, and result publication is based on a fixed and cable connected network. [40, 41]

IV. Design and Implementation

A. Context Diagram

Below is the Level 0 Context Diagram of the Vector-Borne Diseases Monitoring System (VBDMS):

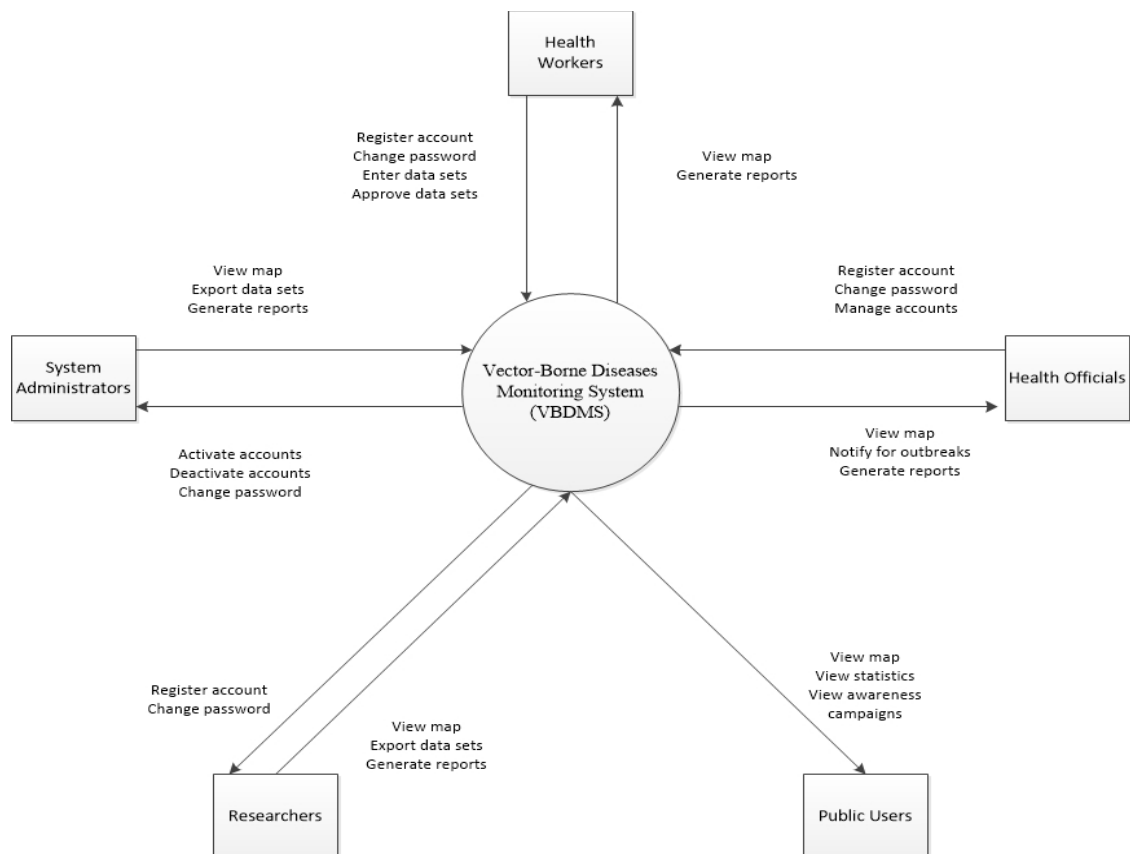


Figure 2: Level 0 Context Free Diagram

There are five external entities: health workers, health officials, researchers, public users and the system administrators. The system allows the barangay health workers to enter data sets and the municipal/city and provincial health workers to approve these data sets; they can also view the map and generate reports. The health officials can view the map, generate reports and will be notified by the system once an outbreak is possible to occur. The researchers can view the map, generate reports and export data sets. The public users can also view the map, the statistics of diseases and the awareness campaigns regarding the

diseases. The system administrators can view the map, export data sets, generate reports as well as activate/deactivate user accounts.

B. Use Case Diagram

This project will have seven classification of user namely the system administrators, barangay health workers, municipal/city health workers, provincial health workers, the municipal/city, provincial and national health officials, researchers and public users. All but the public users are able to register an account, change their account password, and generate results.

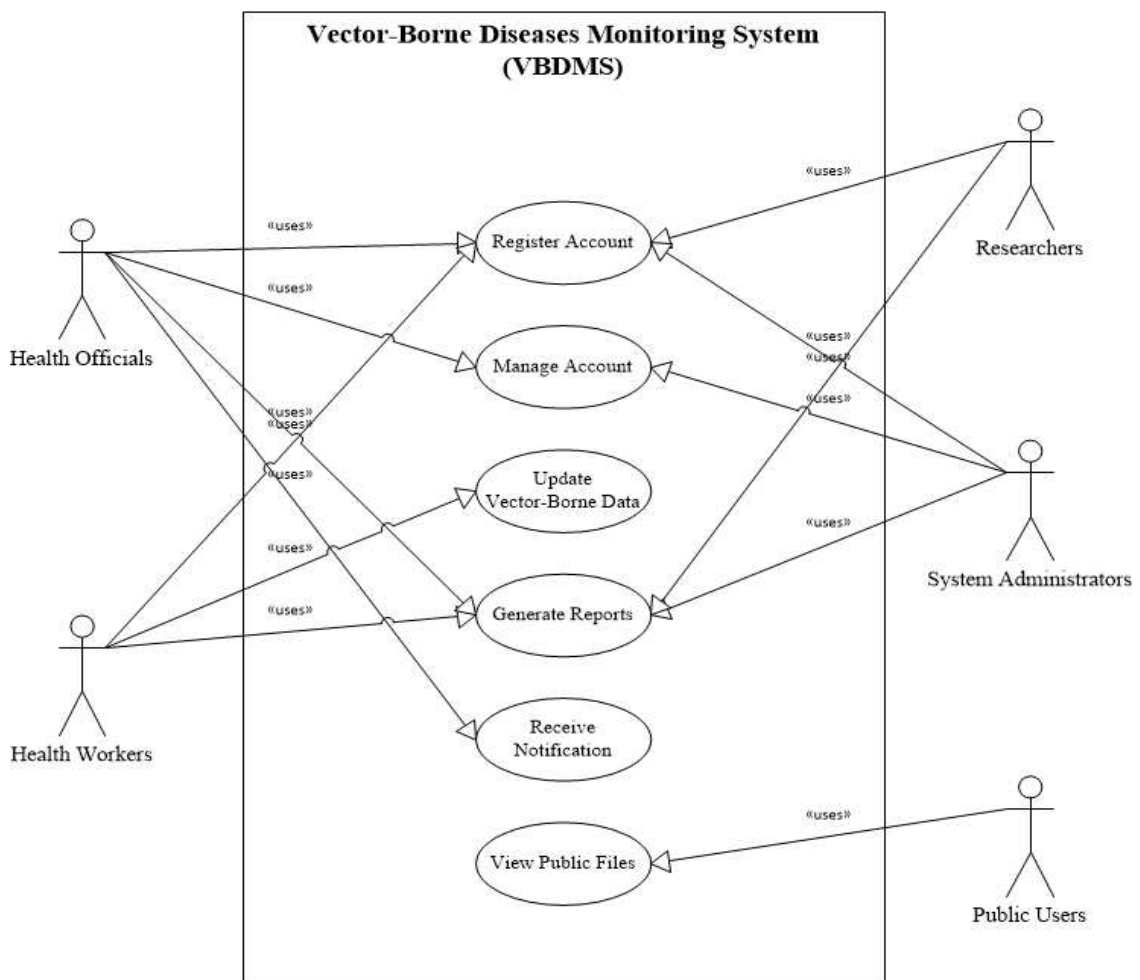


Figure 3: Use Case Diagram

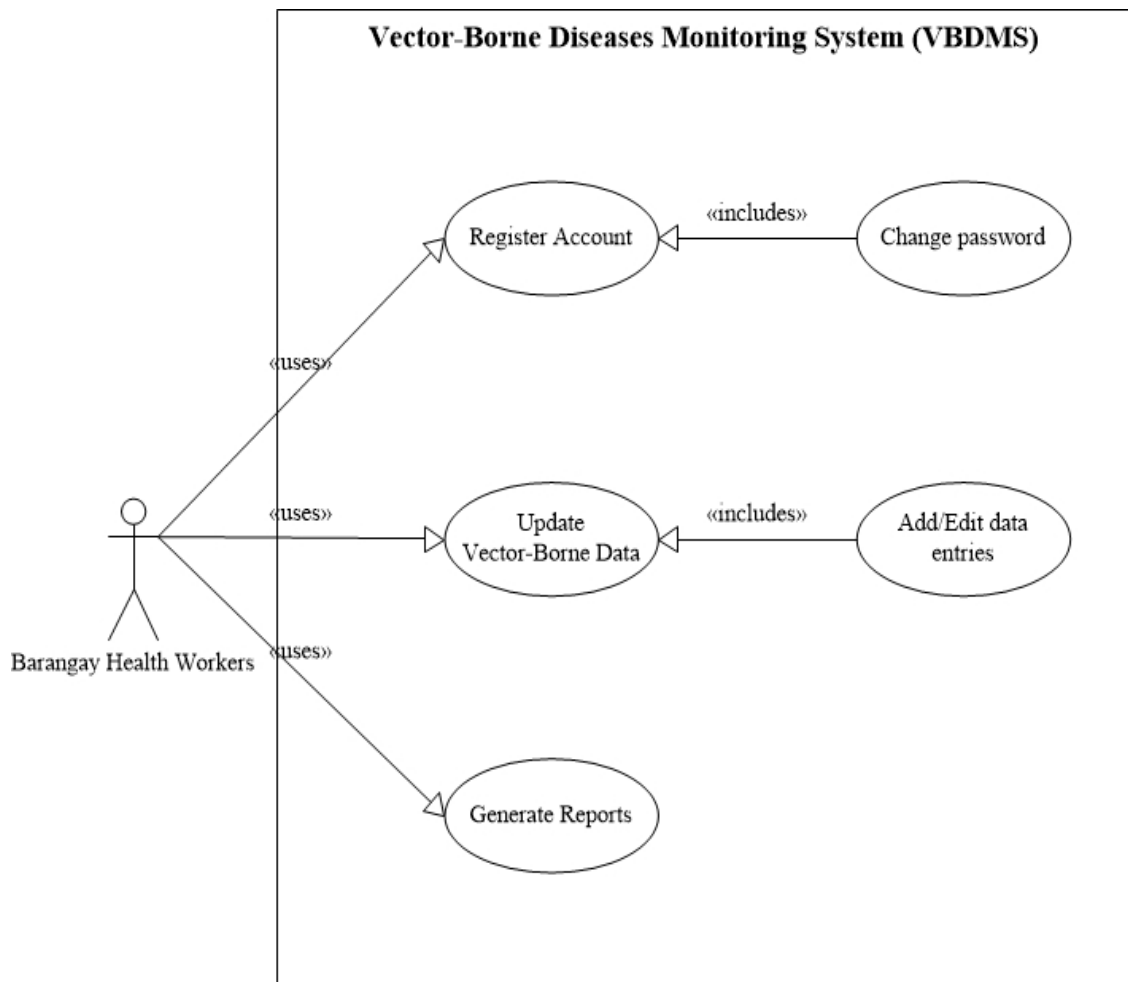


Figure 4: Use Case Diagram for *Barangay Health Workers*

The barangay health workers will have the feature of inputting the data acquired from the surveys directly into the system. After which, the input data will be submitted to the municipal/city health workers account for verification and approval.

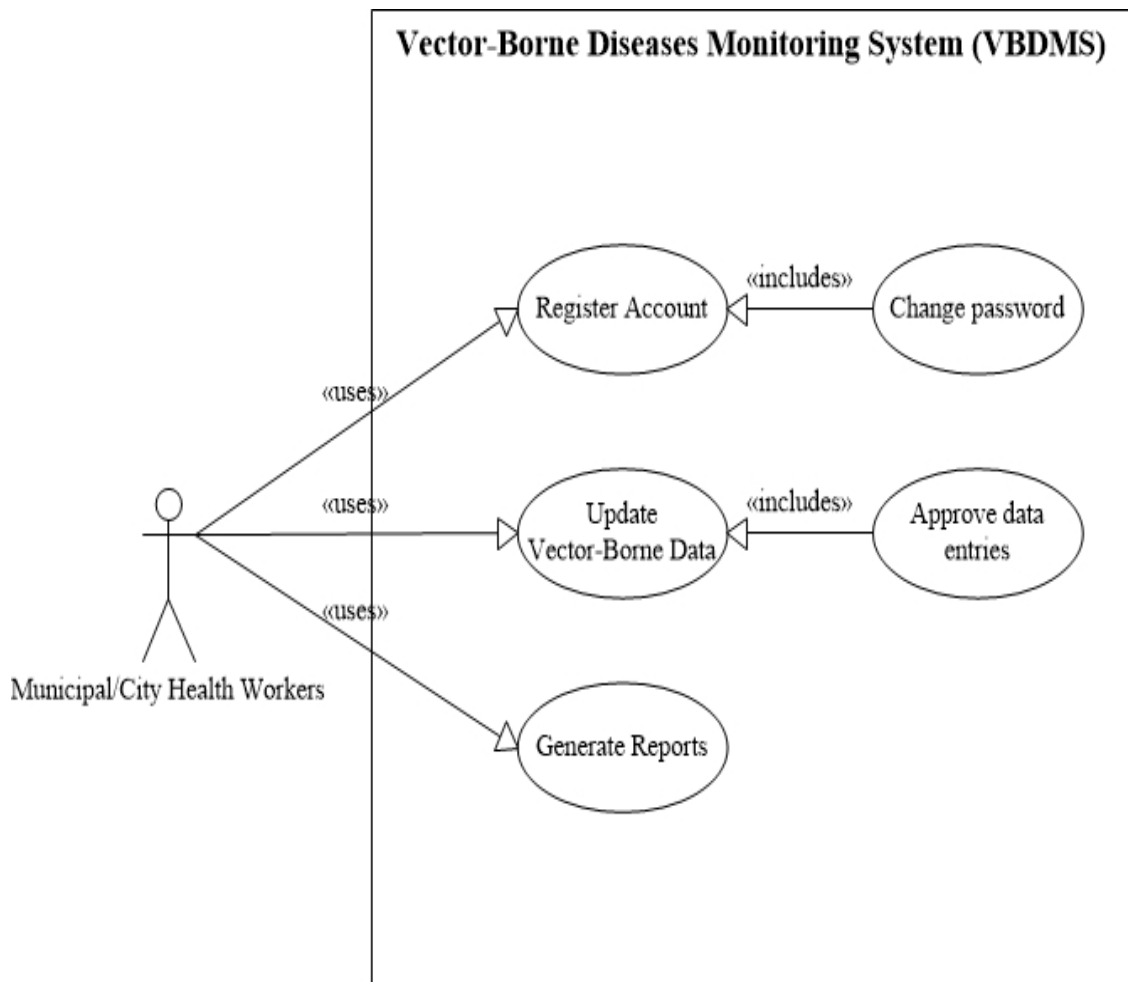


Figure 5: Use Case Diagram for *Municipal/City Health Workers*

The municipal/city health workers have the responsibility to go over the data records submitted by the barangay health workers under their municipality/city for checking and verification. Once these data are verified, they can send these data to provincial health workers account.

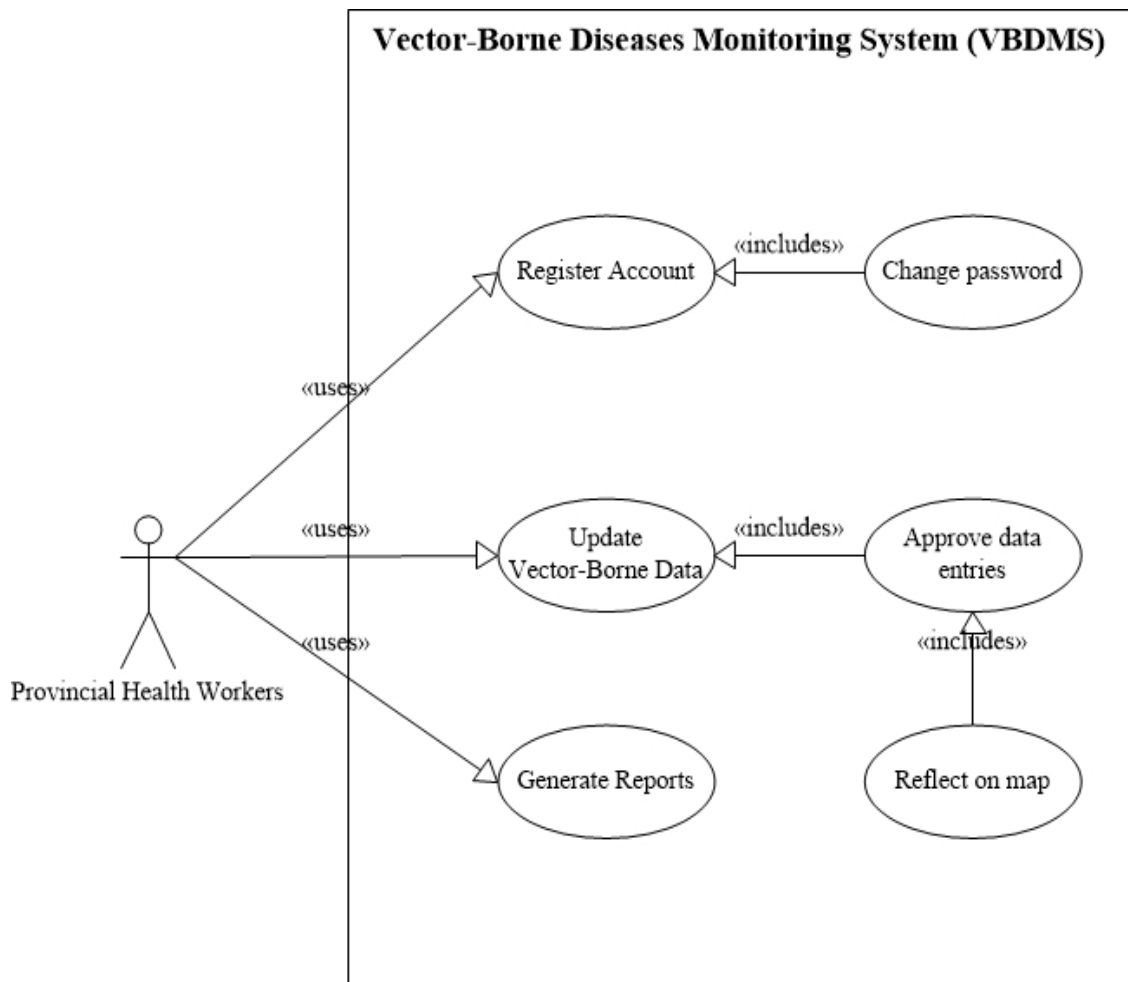


Figure 6: Use Case Diagram for *Provincial Health Workers*

The provincial health workers will again check the data submitted by the municipal/city health workers and after approval they can submit these data to update the system and send these data to national health officials as well.

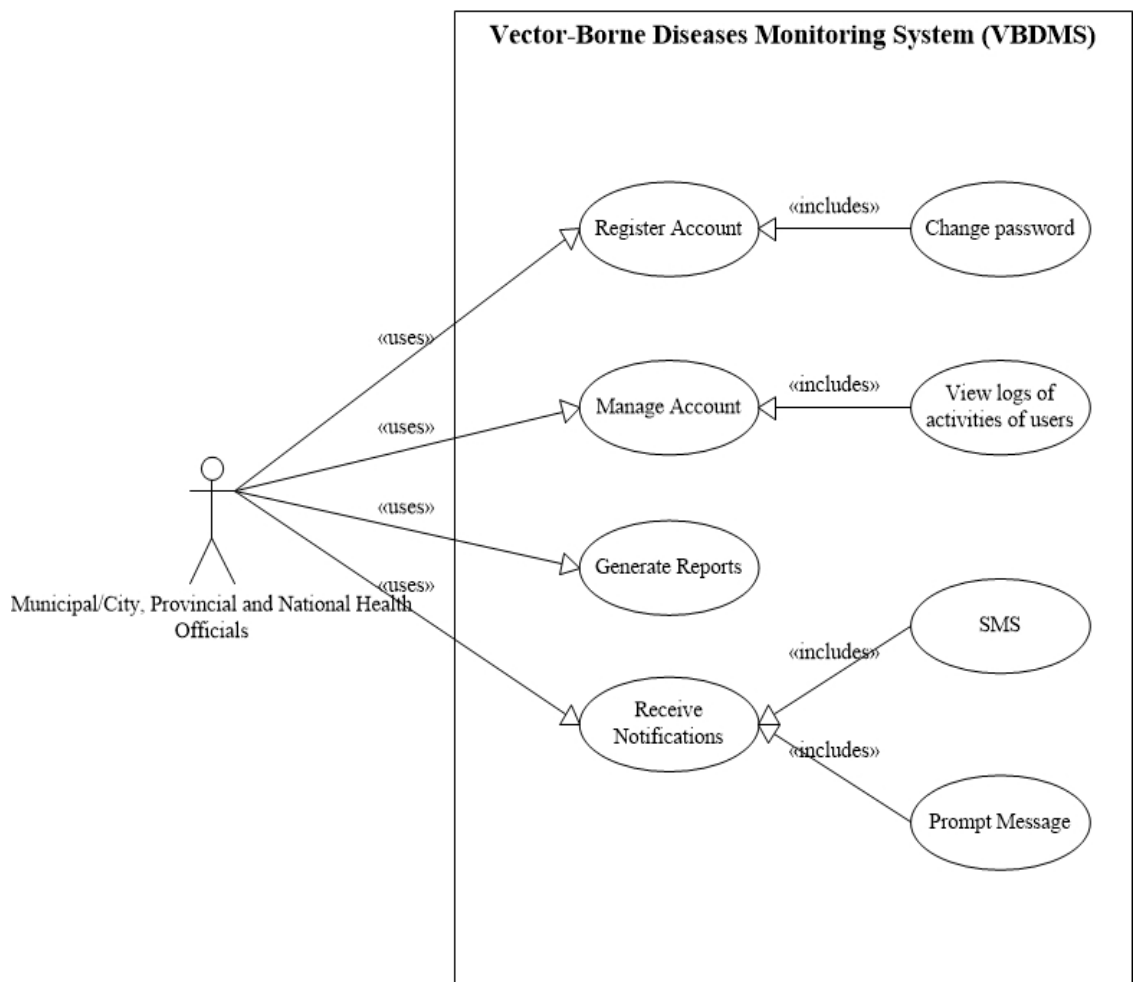


Figure 7: Use Case Diagram for *Municipal/City, Provincial and National Health Officials*

The municipal/city, provincial and national health officials will be notified when an outbreak of a vector-borne disease is likely to happen or occurs. They also have the privilege to manage or monitor the accounts of the health workers/officials under them in case of staff changes or other circumstances.

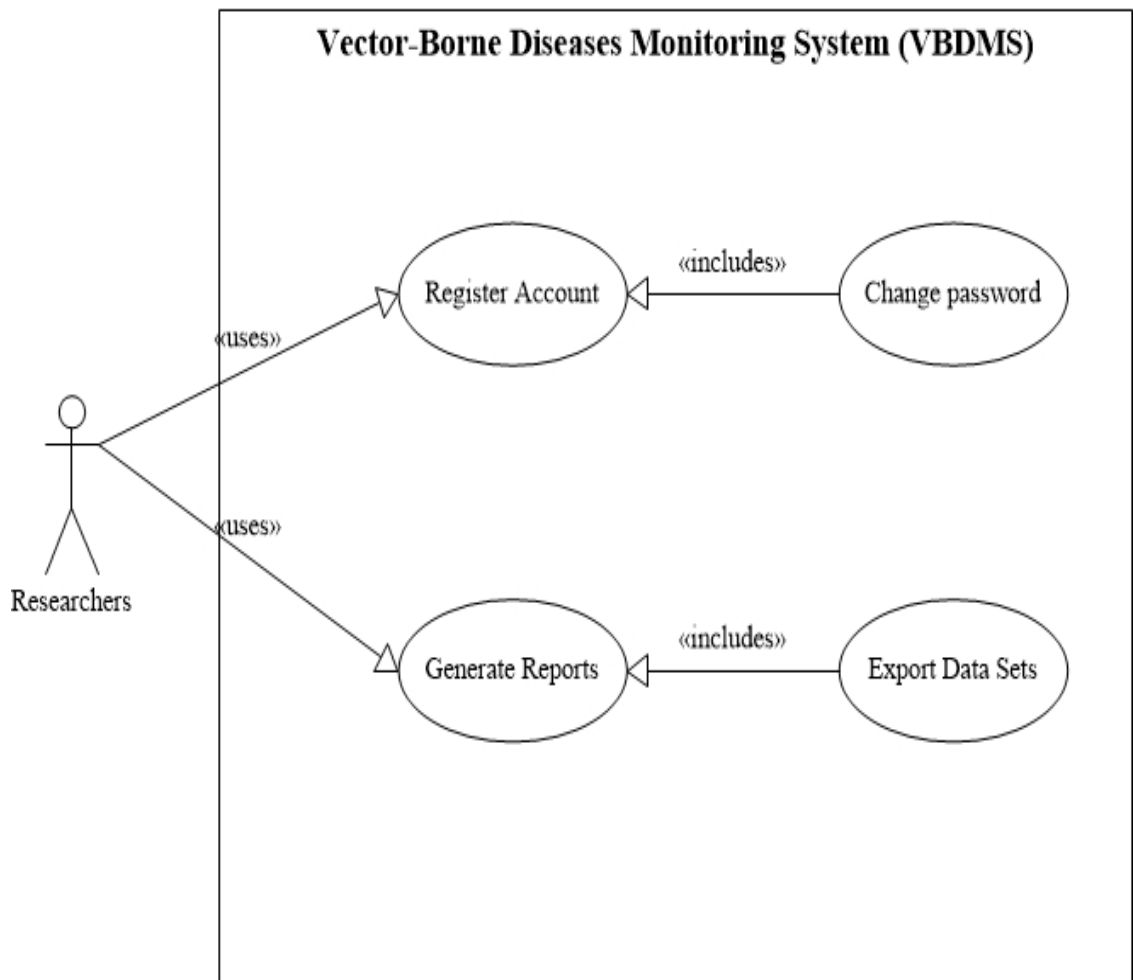


Figure 8: Use Case Diagram for *Researchers*

The researchers can download data sets, the records of patients with vector-borne diseases, in excel format for their research.

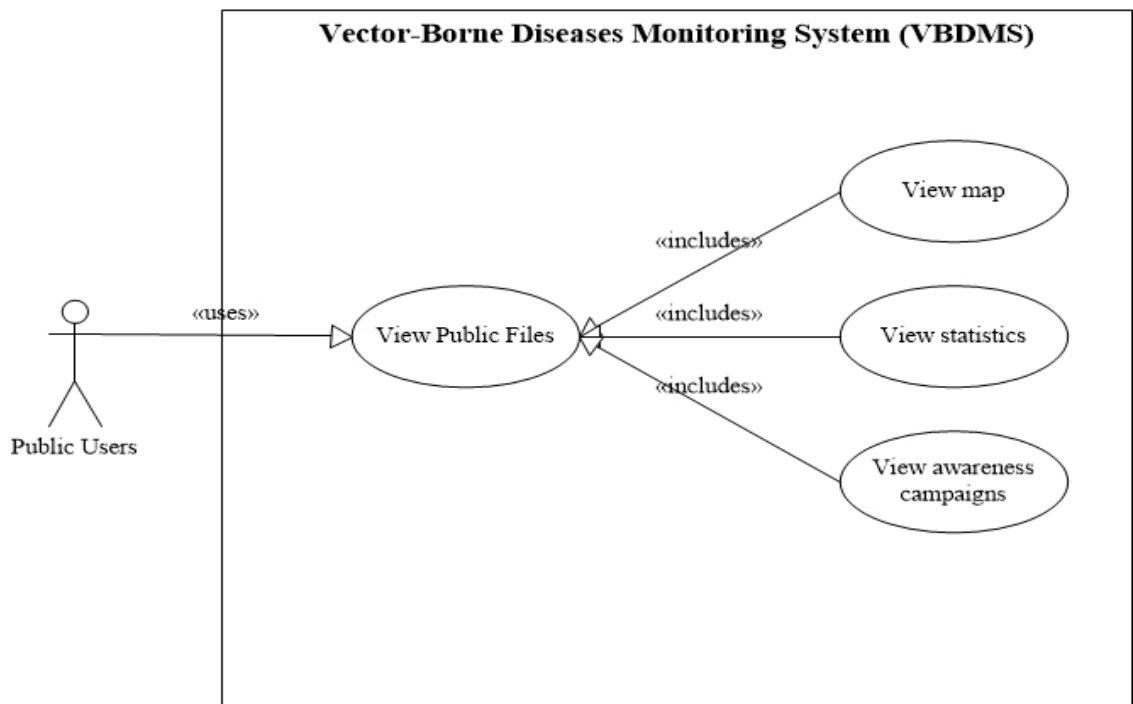


Figure 9: Use Case Diagram for *Public Users*

The public users will be automatically redirected to the map in which they are located. They can also view statistics of diseases in specific area/s within specific date/s. And for information dissemination purposes, there will also be a page for awareness campaign for vector-borne diseases. The figure below shows the use case diagram of the project.

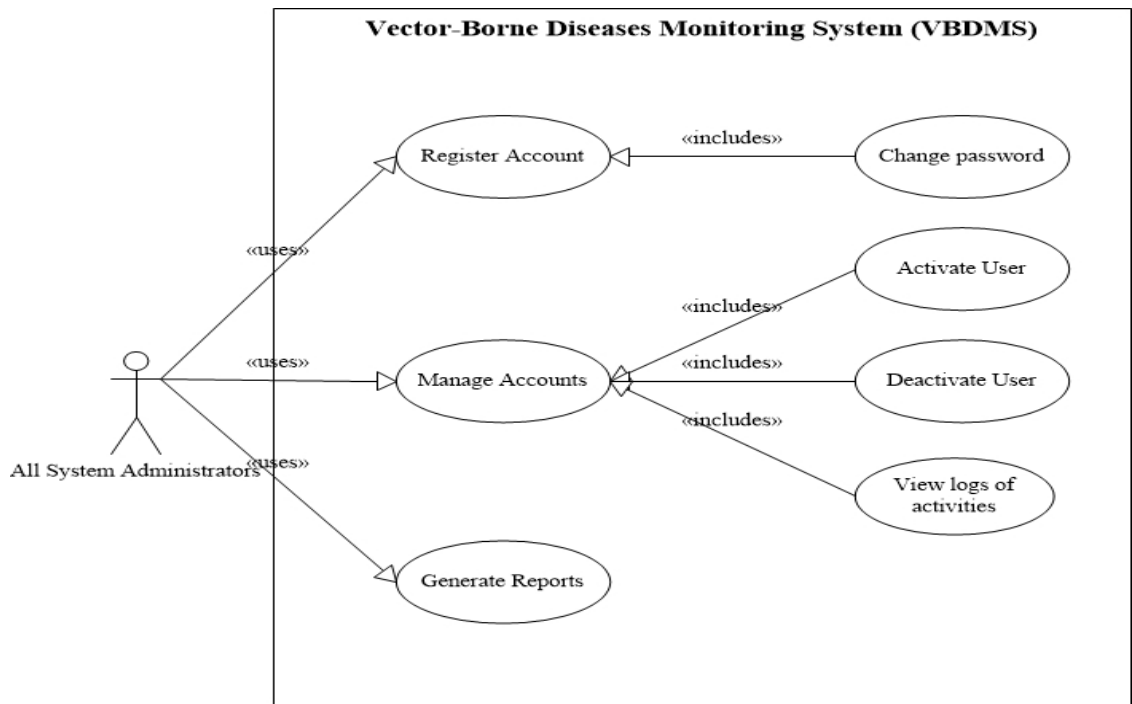


Figure 10: Use Case Diagram for *All System Administrators*

The system administrators, which consist of national, provincial and municipal health official/s and the system and database administrator, have the privilege to activate/deactivate the user account of researchers or health workers/officials within their location. The system and database administrator can view the logs of activities done by its users. All levels of system administrators can also download the data sets, the records of patients with vector-borne diseases, in excel format as well as update the system which includes updating the maps.

C. Flowchart

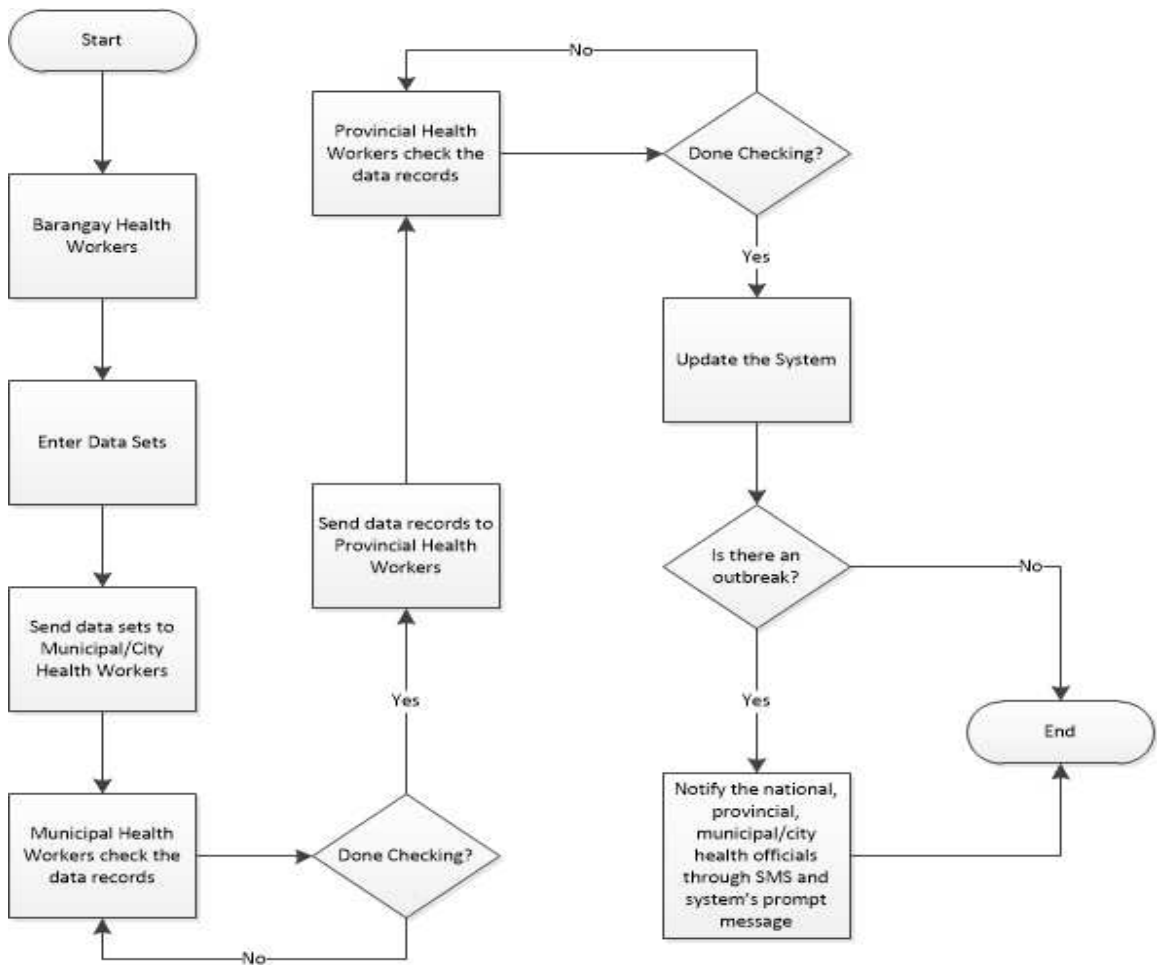


Figure 11: System Flow Chart

This flow chart shows the main flow of the system. The barangay health workers will enter the data sets gathered from their surveys into the system. After entering these data sets, they will send it to the Municipal/City Health workers who will check and validate the data gathered. After checking the data records, they will send these data to Provincial Health Workers. The provincial health workers will check the data records and once done, they are able to update the system. If there is an outbreak, the system will notify the national, provincial, municipal/city health officials via SMS and prompt message from the system for the needed interventions.

D. Entity Relationship Diagram

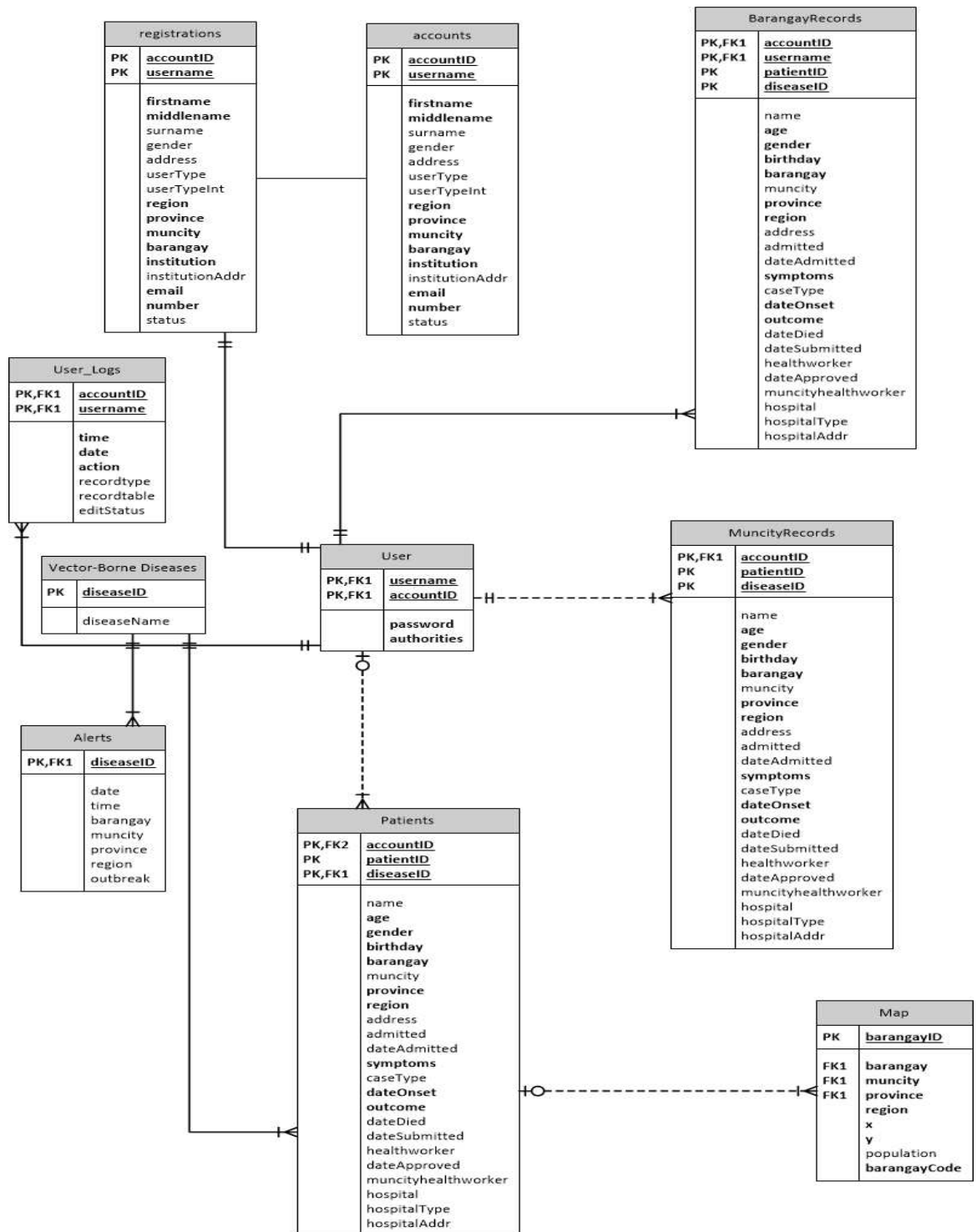


Figure 12: Entity Relationship Diagram

E. Data Dictionary

Table 1: User table

Attribute	Data Type	Description
accountID	integer	Primary key; Unique identifier of a user
username	integer	Primary key; Username
password	varchar(50)	Password
authorities	varchar(30)	User's Authorities

Table 2: User_Logs table

Attribute	Data Type	Description
accountID	integer	Primary and foreign key; Unique identifier of a user
time	Datetime	Primary key; Time that the action was done
date	Date	Primary key; Date that the action was done
action	varchar(20)	Action done by the user
recordtype	varchar(20)	Type of record
recordtable	varchar(20)	Type of record table
editStatus	integer	Edit status of a record

Table 3: Registrations table

Attribute	Data Type	Description
accountID	integer	Primary key; Unique identifier of a user
username	integer	Primary key; Unique identifier of the health office
firstname	varchar(50)	User's First Name
middlename	varchar(30)	User's Middle Name
surname	varchar(30)	User's Surame
gender	varchar(30)	User's Gender
userType	varchar(30)	Type of User
userTypeInt	integer	Type of User (in Integer)
region	varchar(30)	Region of User's affiliated Institution
province	varchar(30)	Province of User's affiliated Institution
muncity	varchar(30)	Municipality/City of User's affiliated Institution
barangay	varchar(30)	Barangay of User's affiliated Institution
institution	text	User's affiliated Institution
institutionAddr	varchar(30)	User's affiliated Institution Address
email	varchar(30)	User's email
number	varchar(30)	User's contact number
status	integer	User's Registration Status Ex: 0 = pending, 1 = approved

Table 4: Accounts table

Attribute	Data Type	Description
accountID	integer	Primary key; Unique identifier of a user
username	integer	Primary key; Unique identifier of the health office
firstname	varchar(50)	User's First Name
middlename	varchar(30)	User's Middle Name
surname	varchar(30)	User's Surame
gender	varchar(30)	User's Gender
userType	varchar(30)	Type of User
userTypeInt	integer	Type of User (in Integer)
region	varchar(30)	Region of User's affiliated Institution
province	varchar(30)	Province of User's affiliated Institution
muncity	varchar(30)	Municipality/City of User's affiliated Institution
barangay	varchar(30)	Barangay of User's affiliated Institution
institution	text	User's affiliated Institution
institutionAddr	varchar(30)	User's affiliated Institution Address
email	varchar(30)	User's email
number	varchar(30)	User's contact number

Table 5: BarangayRecords table

Attribute	Data Type	Description
accountID	integer	Primary key; Unique identifier of a patient
patientID	integer	Primary and foreign key; Unique identifier of a vecto-borne disease
diseaseID	integer	Patient's name
name	varchar(50)	Patient's name
age	integer	Patient's Age
sex	varchar(10)	Patient's Sex
address	varchar(50)	Patient's Complete Address
region	varchar(4)	Patient's regional address
province	varchar(20)	Patient's provincial address
muncity	varchar(20)	Patient's municipal or city address
barangay	varchar(20)	Patient's barangay address
symptoms	varchar(50)	Patient's disease symptoms
caseType	varchar(50)	Patient's disease case classification
dateOnset	Date	Date onset of the disease
outcome	varchar(30)	Patient's outcome
dateDied	Date	Date of death
dateSubmitted	Date	Date submitted by Barangay Health Worker
healthworker	varchar(30)	Barangay Health Worker

Table 6: BarangayRecords table (Continuation)

Attribute	Data Type	Description
dateApproved	Date	Date approved by Municipal/City Health Worker
muncityhealthworker	varchar(30)	Municipal/City Health Worker
admitted	varchar(30)	Is patient admitted? Yes or No
dateAdmitted	Date	Date admitted in the hospital
hospital	varchar(30)	Hospital where patient's admitted
hospitalType	varchar(30)	Type of Hospital
hospitalAddr	varchar(30)	Hospital's address

Table 7: MuncityRecords table

Attribute	Data Type	Description
accountID	integer	Primary key; Unique identifier of a patient
patientID	integer	Primary and foreign key; Unique identifier of a vecto-borne disease
diseaseID	integer	Patient's name
name	varchar(50)	Patient's name
age	integer	Patient's Age
sex	varchar(10)	Patient's Sex
address	varchar(50)	Patient's Complete Address

Table 8: MuncityRecords table (Continuation)

Attribute	Data Type	Description
region	varchar(4)	Patient's regional address
province	varchar(20)	Patient's provincial address
muncity	varchar(20)	Patient's municipal or city address
barangay	varchar(20)	Patient's barangay address
symptoms	varchar(50)	Patient's disease symptoms
caseType	varchar(50)	Patient's disease case classification
dateOnset	Date	Date onset of the disease
outcome	varchar(30)	Patient's outcome
dateDied	Date	Date of death
dateSubmitted	Date	Date submitted by Barangay Health Worker
healthworker	varchar(30)	Barangay Health Worker
dateApproved	Date	Date approved by Municipal/City Health Worker
muncityhealthworker	varchar(30)	Municipal/City Health Worker
admitted	varchar(30)	Is patient admitted? Yes or No
dateAdmitted	Date	Date admitted in the hospital
hospital	varchar(30)	Hospital where patient's admitted
hospitalType	varchar(30)	Type of Hospital
hospitalAddr	varchar(30)	Hospital's address

Table 9: Patients table

Attribute	Data Type	Description
accountID	integer	Primary key; Unique identifier of a patient
patientID	integer	Primary and foreign key; Unique identifier of a vecto-borne disease
diseaseID	integer	Patient's name
name	varchar(50)	Patient's name
age	integer	Patient's Age
sex	varchar(10)	Patient's Sex
address	varchar(50)	Patient's Complete Address
region	varchar(4)	Patient's regional address
province	varchar(20)	Patient's provincial address
muncity	varchar(20)	Patient's municipal or city address
barangay	varchar(20)	Patient's barangay address
symptoms	varchar(50)	Patient's disease symptoms
caseType	varchar(50)	Patient's disease case classification
dateOnset	Date	Date onset of the disease
outcome	varchar(30)	Patient's outcome
dateDied	Date	Date of death
dateSubmitted	Date	Date submitted by Barangay Health Worker
healthworker	varchar(30)	Barangay Health Worker

Table 10: Patients table (Continuation)

Attribute	Data Type	Description
dateApproved	Date	Date approved by Municipal/City Health Worker
muncityhealthworker	varchar(30)	Municipal/City Health Worker
dateApproved2	Date	Date approved by Provincial Health Worker
provincehealthworker	varchar(30)	Province Health Worker
admitted	varchar(30)	Is patient admitted? Yes or No
dateAdmitted	Date	Date admitted in the hospital
hospital	varchar(30)	Hospital where patient's admitted
hospitalType	varchar(30)	Type of Hospital
hospitalAddr	varchar(30)	Hospital's address

Table 11: Map table

Attribute	Data Type	Description
barangayID	integer	Primary and foreign key; Unique identifier of a barangay
barangay	varchar(50)	Barangay
muncity	varchar(50)	Muncity
province	varchar(50)	Province
region	varchar(50)	Region
x	double	Barangay's X coordinate
y	double	Barangay's Y coordinate
population	double	Barangay's population

Table 12: Vector-Borne Diseases table

Attribute	Data Type	Description
diseaseID	integer	Primary key; Unique identifier of a vecto-borne disease
diseaseName	varchar(20)	Vector-borne disease name (Chikungunya, Dengue, Malaria)

Table 13: Alerts table

Attribute	Data Type	Description
diseaseID	integer	Primary and foreign key; Unique identifier of a vector-borne disease
time	Datetime	Primary key; Time for Outbreak record
date	Date	Primary key; Date for Outbreak record
barangay	varchar(50)	Barangay with Outbreak
muncity	varchar(50)	Muncity with Outbreak
province	varchar(50)	Province with Outbreak
region	varchar(50)	Region with Outbreak
outbreak	double	Outbreak percentage
status	int	Alert status, Ex: 0 = unseen, 1 seen

V. Architecture

A. System Architecture

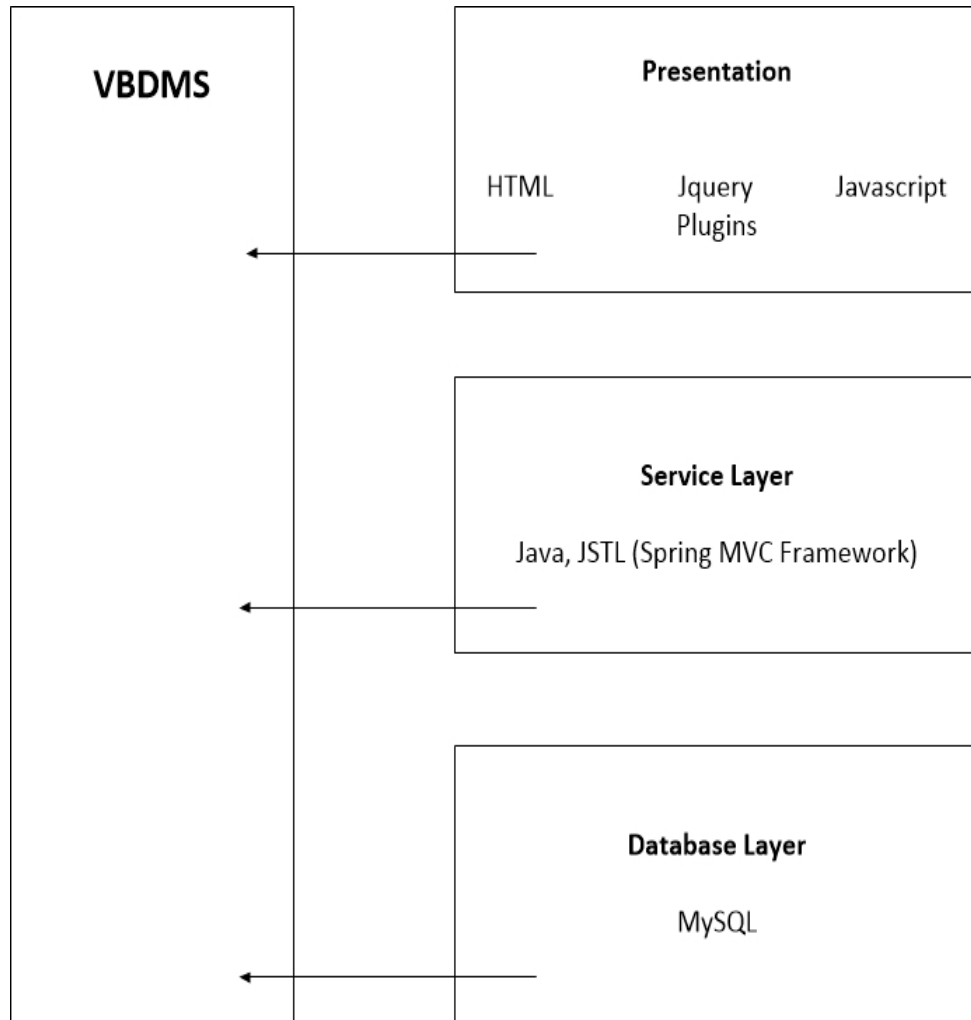


Figure 13: VBDMS System Architecture

In this figure, it shows the three main domains of VBDMS. Its web interface is encoded in HTML and Javascript. It is linked to the Service Layer using Spring MVC Framework and JSTL. VBDMS implements the MVC, Model View Controller, pattern of the Spring MVC Framework. VBDMS uses Java language to connect to its database and perform other services as well. VBDMS uses MySQL to connect to its database.

B. Technical Architecture

The hardware requirements for the server:

- 2 GHz processor or higher
- 512 MB of RAM or higher
- Up to 2GB of free disk space

The hardware requirements for the client:

- 2 GHz processor or higher
- 512 MB of RAM or higher
- 300 MB of free disk space
- Consistent internet connectivity
- Web browser (Google Chrome, Mozilla Firefox, other google map compatible browsers)

VI. Results

A. General View

1. Home Page

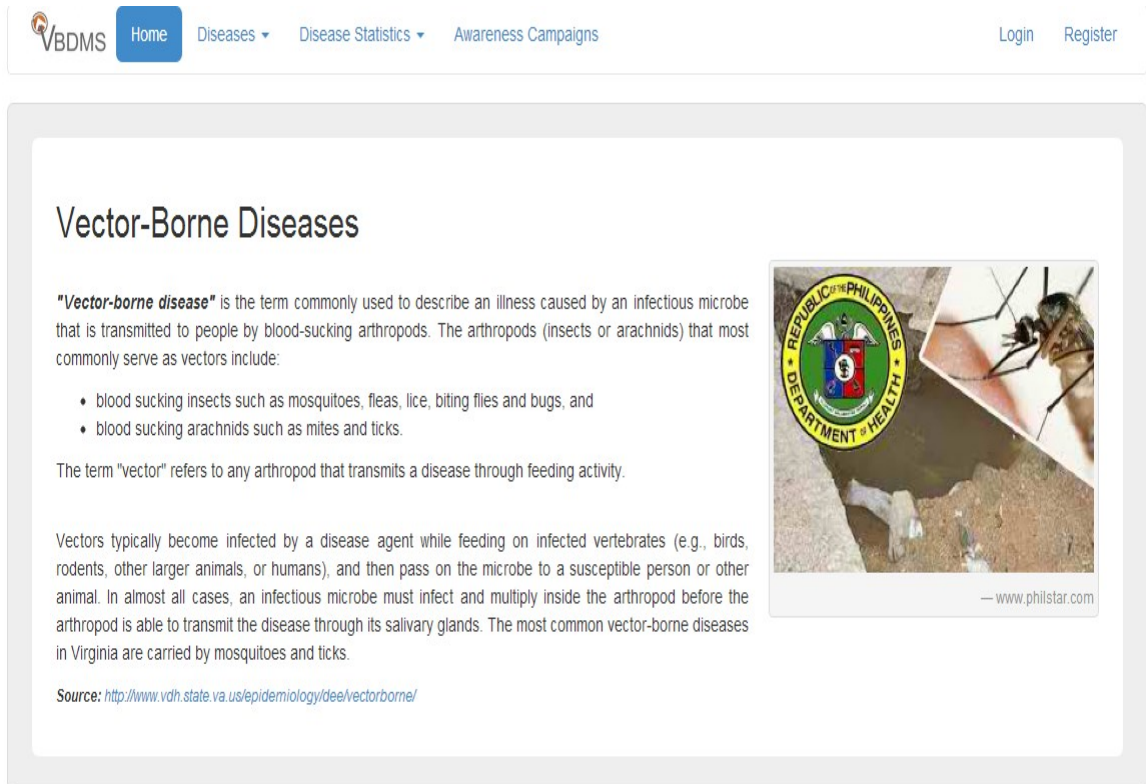


Figure 14: Home Page, VBDMS

Figure 14 shows the home page of Vector-Borne Diseases Monitoring System (VBDMS). This home page contains brief information such as signs and symptoms, mode of transmission, incubation period, communicability, and treatment of Vector-Borne Diseases. It also includes disease statistics and current awareness campaigns.

2. Login Page

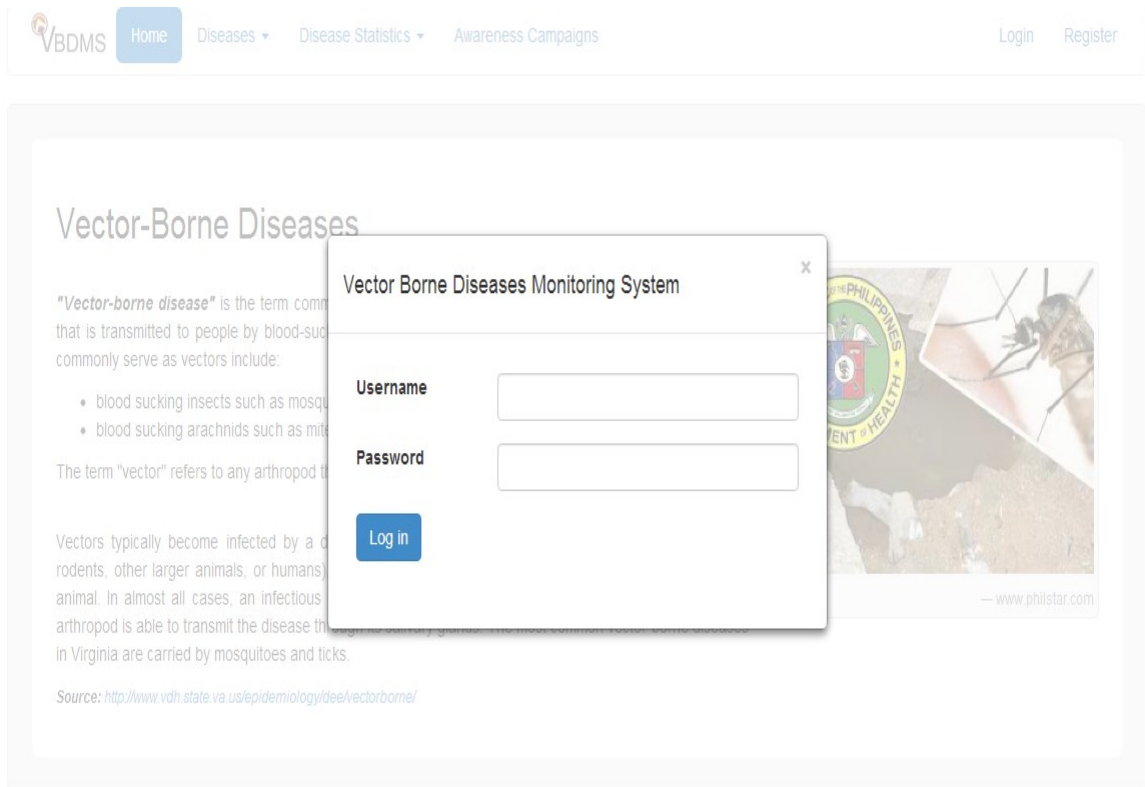


Figure 15: Login Page, VBDMS

The administrator, all levels of health workers and health officials, and researchers can login to their respective accounts through this page. (See Figure 15)

3. Registration Page

VBDMS Home Diseases Disease Statistics Awareness Campaigns

Registration Form

* Required fields

First Name * Middle Name

Surname * Gender

Home Address

Register as * Username *

Region * Province *

City/Municipality * Barangay *

Institution *

Figure 16: Registration Form Page, VBDMS

Registration form for interested users or researchers and health officials and health workers. (See Figure 16)

4. Disease Page

VBDMS Home **Chikungunya** Disease Statistics Awareness Campaigns Login Register

Chikungunya

Chikungunya is a viral disease that causes fever and severe joint pain. Sharing clinical signs with dengue, it is usually misdiagnosed in areas where dengue is common. It is transmitted by infected mosquitoes, usually the genus *Aedes*.

a. Signs and Symptoms

Aside from abrupt onset of fever frequently accompanied by joint pains, other signs and symptoms of chikungunya include muscle pain, headache, nausea, fatigue and rash. Joint pain usually lasts within a few days or weeks and while most patients recover fully from this, some cases may persist for several months, or even years. Occasionally, cases of eye, neurological and heart complications as well as gastrointestinal complaints have been reported. This disease not often cause serious complications but can contribute to death among older population.

b. Mode Of Transmission

Chikungunya virus is transmitted to humans from infected non-human primates and other humans by the bite of *Aedes* mosquitoesFootnote 1. Evidence exists that CHIKV can also be passed from an infected mother to a developing foetusFootnote 2 Footnote 13. Furthermore, inhalation of aerosolised chikungunya virus in a laboratory setting may lead to chikungunya virus

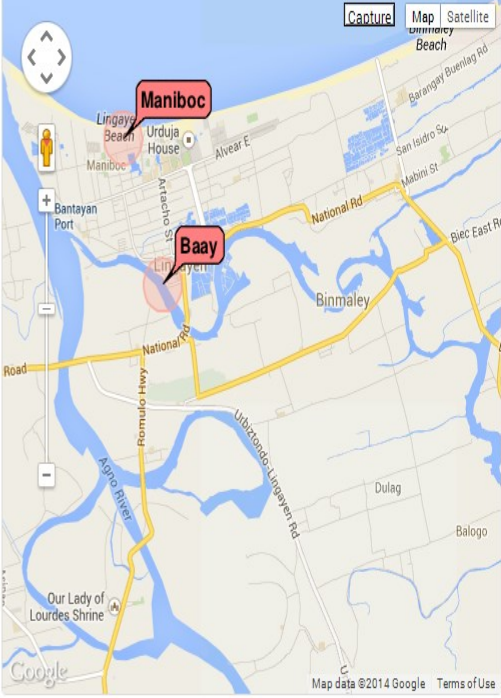


Figure 17: Disease Page, VBDMS

Figure 17 shows where all system users, as well as public users, can view information such as signs and symptoms, mode of transmission, incubation period, communicability, and treatment on vector-borne diseases specified in the system; namely, chikungunya, dengue and malaria. This page also contains a map showing markers of barangays where there are cases of the selected disease.

5. Disease Statistics Page

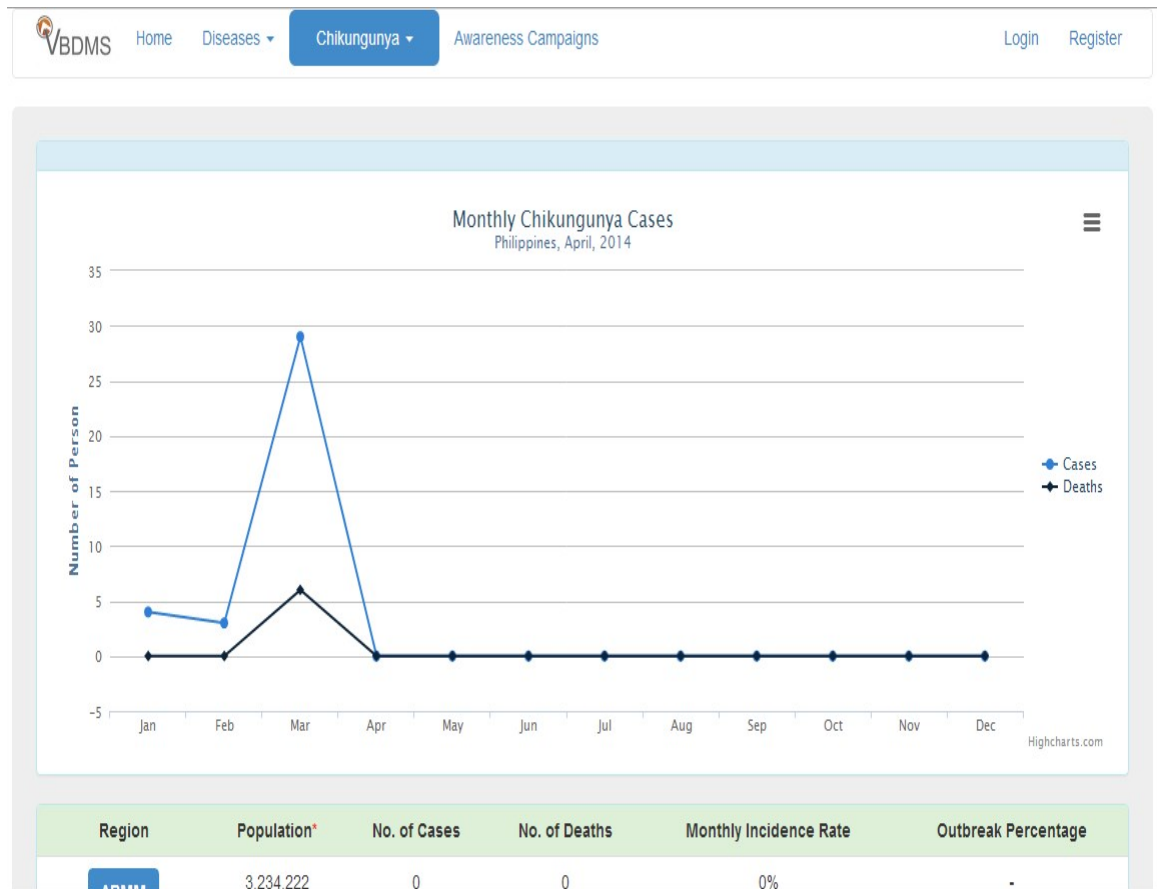


Figure 18: Disease Statistics Page, VBDMS

In Figure 18, all users can view statistics about the chosen vector-borne disease. A sample chart portraying the monthly cases of the disease (April 2014) is shown, as well as incidence rate, and outbreak percentage on regional and municipal level.

6. Awareness Campaign Page

VBDMS Home Disease ▾ Disease Statistics ▾ Awareness Campaigns Login Register

Chikungunya Dengue Malaria

To prevent the spread of dengue fever, you must first prevent the breeding of its vector, the Aedes mosquito. The Aedes mosquito is easily identifiable by its distinctive black and white stripes on their body. It prefers to breed in clean, stagnant water easily found in our homes. You can get rid of the Aedes mosquito by frequently checking and removing stagnant water in your premises. The guidelines below will give you an overview of how you can prevent the Aedes mosquito from breeding.

- Keep your homes and offices protected against mosquitoes.
- Keep homes and offices airy, bright and safe from moisture.
- Fix nets on doors and windows.
- Wear full sleeves clothes.
- Use mosquito nets while sleeping.
- Don't leave the overhead tanks open.
- Don't keep water in containers for more than a week. Instead, empty them every week, let them dry and then fill again.
- Don't let the water falling from the overhead tanks to accumulate permanently. Instead dry it.
- Don't let the water accumulate in any case both inside or outside the home.
- Be mindful of your home's cleanliness.
- Be mindful of your home's cleanliness.
- Keep the fence and hedge boundaries duly cut both inside and outside the home, and spray over them with insecticides, particularly in the evening.
- Don't let the water stay all the time in the flower pots, gamias of plants. Instead water them only in the morning every alternate day.



Dengue Sanhi: Dengue Virus
Paano Nakukuha: Mula sa kagat ng lamok na may dalang dengue virus

Mga Palatandaan

- Mataas na lagnat sa loob ng 2-7 na araw
- Pananakit ng kasaksaan at kalamnan
- Panghina
- Pagkawala ng ganang kumain
- Pamumula ng balat
- Pananakit ng tiyan at pagsusuka
- Pagdurugo ng ilong at gilagid

Pag-iwas at pagsugo:

- Unsin at talpan ang mga dram at iba pang imbakan ng tubig.
- Iacob ang mga bote, lata at iba pang maaring mapag-iyunan ng tubig at mapanghiyugan ng mga lamok.
- Magsuot ng mahahabang kasutan upang hindi madaling makagat ng lamok.
- Maaari din gumamit ng insect repellent.
- Magpatinig sa health center.

HEALTH ADVISORY

— pinoykidsmd.com

Figure 19: Awareness Campaign Page, VBDMS

Shown in Figure 19 are the different awareness campaigns for vector-borne diseases instituted by the Department of Health (DOH) and other international health organizations.

B. Administrator View / National Health Official View / Provincial Health Official View / Municipal/City Health Official View

1. Notification Page

(a) Alerts Page

VBDMS Home Diseases ▾ Disease Statistics ▾ Awareness Campaigns Logged in as: Administrator

Notifications Health Workers and Officials Reports Profile

Alerts 0 Registrations ▾

Date	Disease	Outbreak Percentage	Barangay	Municipality	Province	Region
2014-03-30	Chikungunya	311.11111%	Baay	LINGAYEN	Pangasinan	Region I
2014-03-29	Chikungunya	271.42857%	Baay	LINGAYEN	Pangasinan	Region I
2014-03-29	Chikungunya	244.44444%	Baay	LINGAYEN	Pangasinan	Region I
2014-03-29	Chikungunya	244.44444%	Baay	LINGAYEN	Pangasinan	Region I
2014-03-29	Chikungunya	255.55556%	Baay	LINGAYEN	Pangasinan	Region I
2014-03-29	Chikungunya	266.66667%	Baay	LINGAYEN	Pangasinan	Region I
2014-03-29	Chikungunya	277.77778%	Baay	LINGAYEN	Pangasinan	Region I

Showing 1 to 7 of 7 entries

« 1 »

Figure 20: Alerts Page, VBDMS

System alerts are shown in Figure 20. These alerts, aside from SMS alerts, are to notify the health officials and administrators if there are outbreaks in a certain barangay whenever the provincial health worker updates the system.

(b) Registration Page

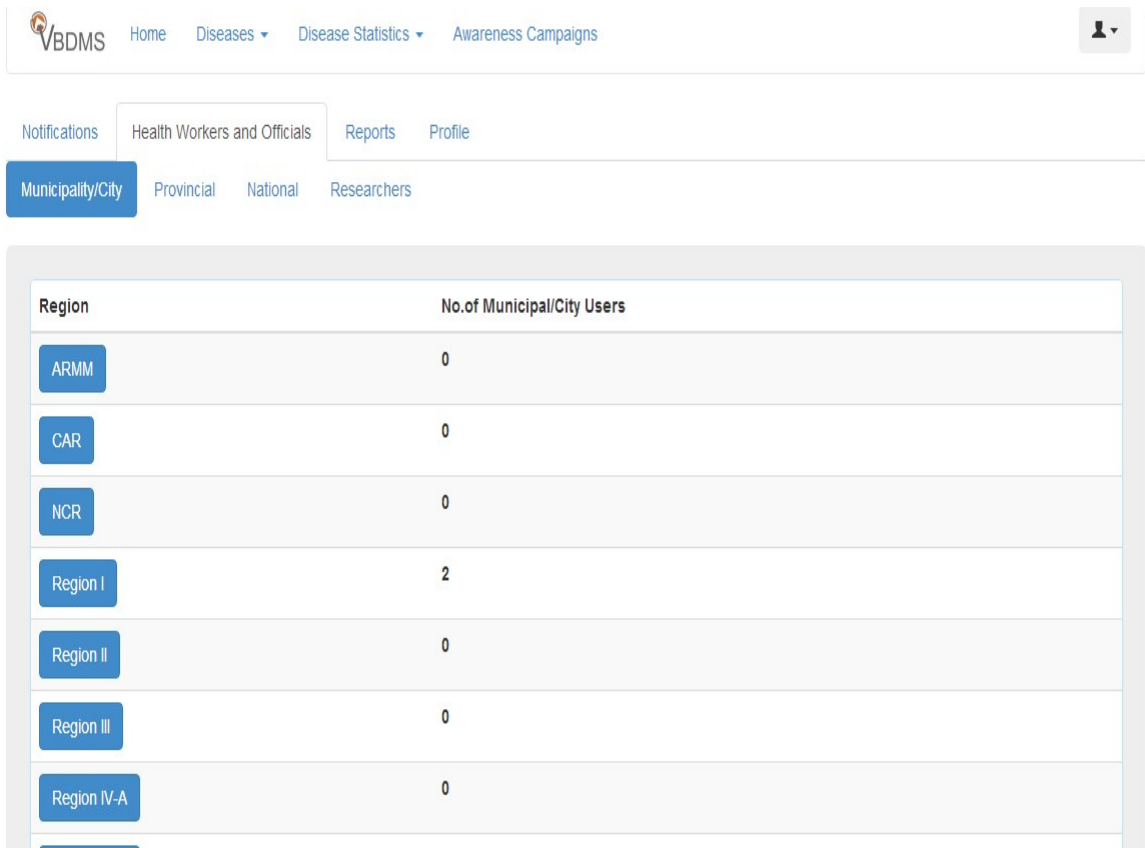
The screenshot displays the VBDMS (Vigilance and Biomedical Data Management System) interface. At the top, the navigation bar includes the VBDMS logo, a home button, and dropdown menus for Diseases, Disease Statistics, and Awareness Campaigns. The user is logged in as Administrator. Below the navigation bar, there are tabs for Notifications, Health Workers and Officials, Reports, and Profile. A dropdown menu for Alerts shows 0 alerts, and another dropdown menu for Registrations is open, showing options for Municipal/City, Provincial, National, and Researchers. The main content area is titled "Pending Registrations" and contains a table with two columns: "Region" and "No. of Municipal/City Users".

Region	No. of Municipal/City Users
ARMM	0
CAR	0
NCR	0
Region I	1
Region II	0
Region III	0

Figure 21: Registration Page, VBDMS

The registrations to the system can be viewed by the administrator, national, provincial, municipal/city health officials. The figure shows the registrations pending for approval, approved registrations and rejected registrations. (See Figure 21)

2. Health Worker and Officials Page



Region	No. of Municipal/City Users
ARMM	0
CAR	0
NCR	0
Region I	2
Region II	0
Region III	0
Region IV-A	0

Figure 22: Health Worker and Officials Page, VBDMS

All users registered to the VBDMS is shown in Figure 22. They are arranged according to their scope of operations: barangay, municipal/city, provincial, national. The researchers are also shown here as well. Through this page, the administrator, national, provincial, municipal/city health officials can view the activities and profile of all the users of the system.

3. Reports Page

The screenshot shows the VBDMS Reports Page. At the top, there is a navigation bar with the VBDMS logo, 'Home', 'Diseases', 'Disease Statistics', and 'Awareness Campaigns'. On the right, it says 'Logged in as: Administrator' with a user icon. Below this is a secondary navigation bar with 'Notifications', 'Health Workers and Officials', 'Reports' (which is highlighted), and 'Profile'. Under 'Reports', there are two buttons: 'Export Reports' (highlighted in blue) and 'Export Data'. The main content area is titled 'Generate reports' and contains the following fields:

- INCLUDE DATES**
 - From:** 2014-01-01 (with a calendar icon)
 - To:** 2014-01-01 (with a calendar icon)
- Disease:** A dropdown menu.
- Include following charts:**
 - Disease Case by Age Group
 - Disease Case by Gender Group
 - Cases and Death by Municipality/City
 - Cases Fatality Ratio by Age Group
 - Cases and Death by Region
 - A dropdown menu at the bottom right of the chart selection area.

A blue button is visible at the bottom center of the form area.

Figure 23: Administrator, National, Provincial, Municipal/City Health Officials Reports Page, VBDMS

The reports page is where the reports for chosen vector-borne disease can be generated and exported to pdf as seen in Figure 23. The user can also export data sets of VBDMS patient records in xls format.

4. Profile Page

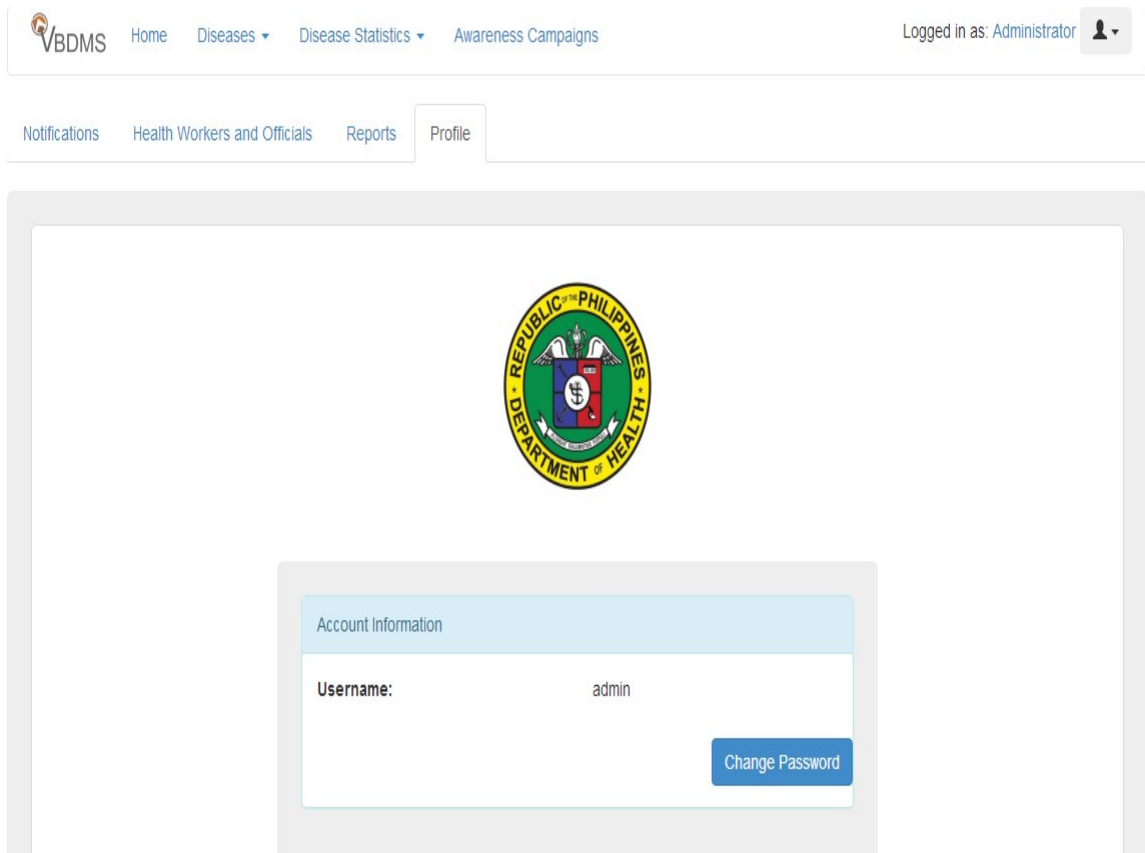


Figure 24: Administrator, National, Provincial, Municipal/City Health Officials Profile Page, VBDMS

The user currently logged in can view his profile in this page, and can change some of his/her information as well. (See Figure 24)

C. Barangay Health Worker View / Municipal/City Health Worker View / Provincial Health Worker View

1. Manage Records Page

The screenshot displays the VBDMS interface for managing records. At the top, the navigation bar includes the VBDMS logo, a home icon, and menu items for 'Diseases', 'Disease Statistics', and 'Awareness Campaigns'. The user is logged in as 'Barangay Health Worker'. Below the navigation bar, there are tabs for 'Manage Records', 'Reports', and 'Profile'. The main content area is titled 'Inserting Barangay Patient Record' and contains two sections: 'Patient's Record Information' and 'Patient's General Information'. The 'Patient's Record Information' section includes fields for 'Default Region' (Region I), 'Default Province' (Pangasinan), 'Municipality/City' (dropdown), and 'Barangay' (dropdown). The 'Patient's General Information' section includes fields for 'Name', 'Complete Address', 'Date of Birth' (three dropdowns), and 'Gender' (dropdown).

Figure 25: Manage Records Page, VBDMS

The barangay health worker can add new barangay patient records which will be sent to municipal/city health workers for approval. Municipal/City health workers will receive these patient records and verifies them. After verification and approval of these patient records, they will be sent to the provincial health workers. Health workers page in various levels are similar to Figure 25, 26 and 27.

2. Reports Page

The screenshot displays the VBDMS Reports Page. At the top, the navigation bar includes the VBDMS logo, 'Home', 'Diseases', 'Disease Statistics', and 'Awareness Campaigns'. The user is logged in as 'Barangay Health Worker'. Below the navigation bar, there are tabs for 'Manage Records', 'Reports', and 'Profile'. Under the 'Reports' tab, there are buttons for 'Export Reports' and 'Export Data'. The main content area is titled 'Generate reports' and contains the following form fields:

- INCLUDE DATES**
 - From:** 2014-01-01
 - To:** 2014-01-01
- Disease:** (Dropdown menu)
- Include following charts:**
 - Disease Case by Age Group
 - Disease Case by Gender Group
 - Cases and Death by Municipality/City
 - Cases Fatality Ratio by Age Group
 - Cases and Death by Region

A blue button is visible at the bottom of the form area.

Figure 26: Barangay, Municipal/City, Provincial Health Workers Reports Page, VBDMS

Figure 26 shows where the reports for chosen vector-borne disease can be generated and exported to pdf. The user can also export data sets of VBDMS patient records in xls format.

3. Profile Page

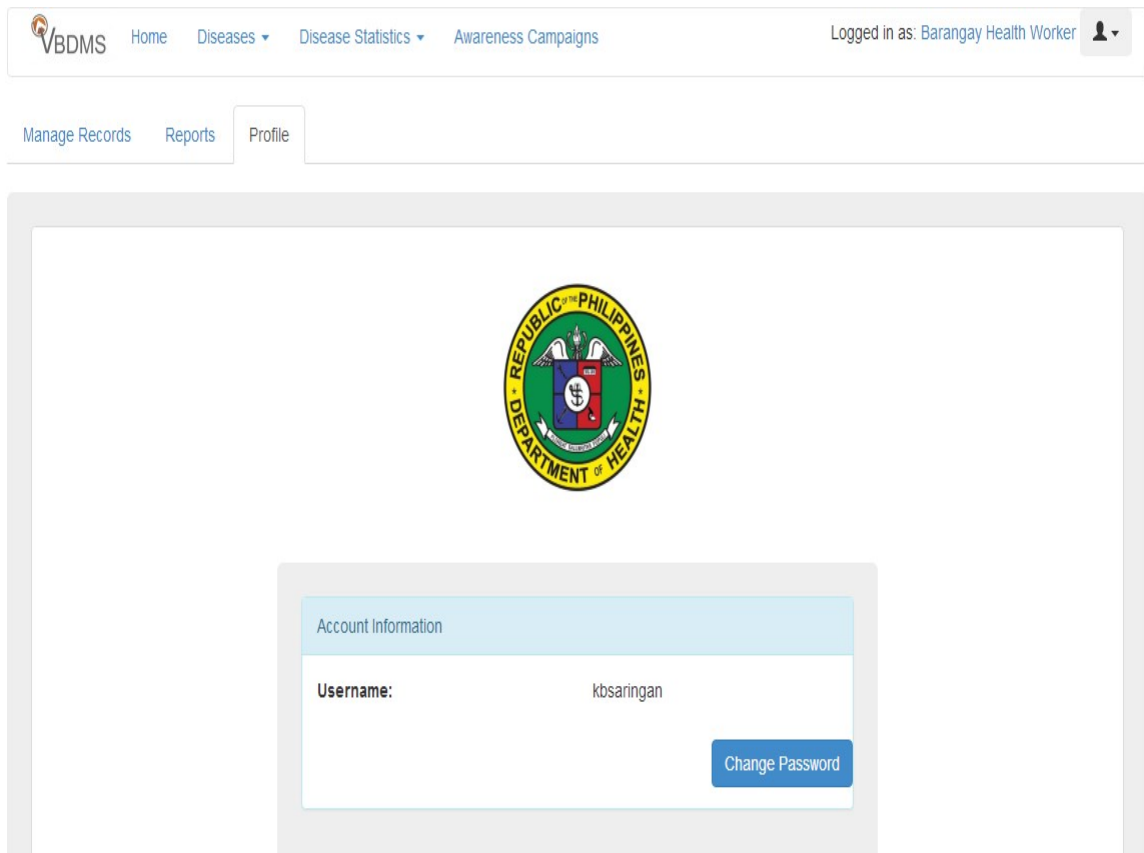


Figure 27: Barangay, Municipal/City, Provincial Health Workers Profile Page, VBDMS

The user currently logged in can view his profile in Figure 27, and can change some of his/her information as well.

D. Researcher View

1. Reports Page

VBDMS Home Diseases Disease Statistics Awareness Campaigns Logged in as: Barangay Health Worker

Manage Records Reports Profile

Export Reports Export Data

Generate reports

INCLUDE DATES

From: 2014-01-01 **To:** 2014-01-01

Disease

Include following charts:

Disease Case by Age Group Cases Fatality Ratio by Age Group

Disease Case by Gender Group Cases and Death by Region

Cases and Death by Municipality/City

Figure 28: Researcher Reports Page, VBDMS

Figure 28 shows where the reports for chosen vector-borne disease can be generated and exported to pdf. The user can also export data sets of VBDMS patient records in xls format.

2. Profile Page

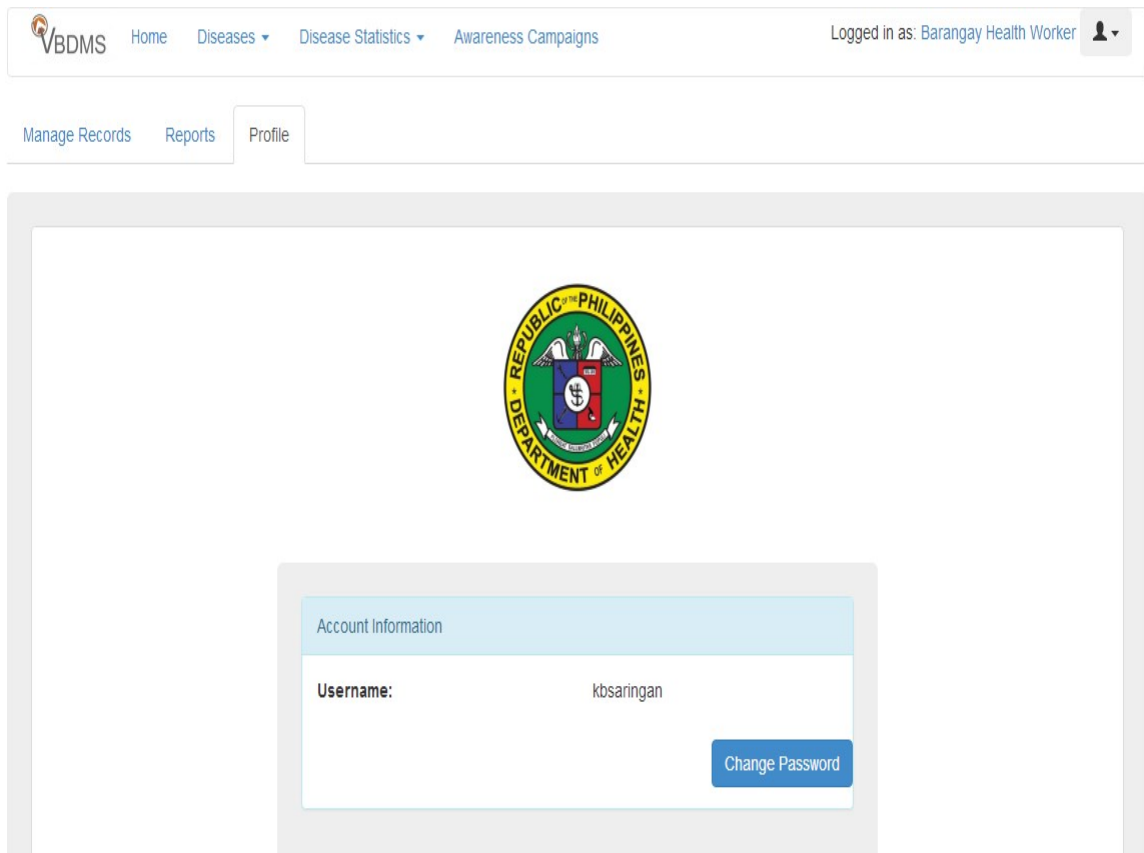


Figure 29: Researcher Profile Page, VBDMS

The user currently logged in can view his profile in Figure 29, and can change some of his/her information as well.

E. Generated Report

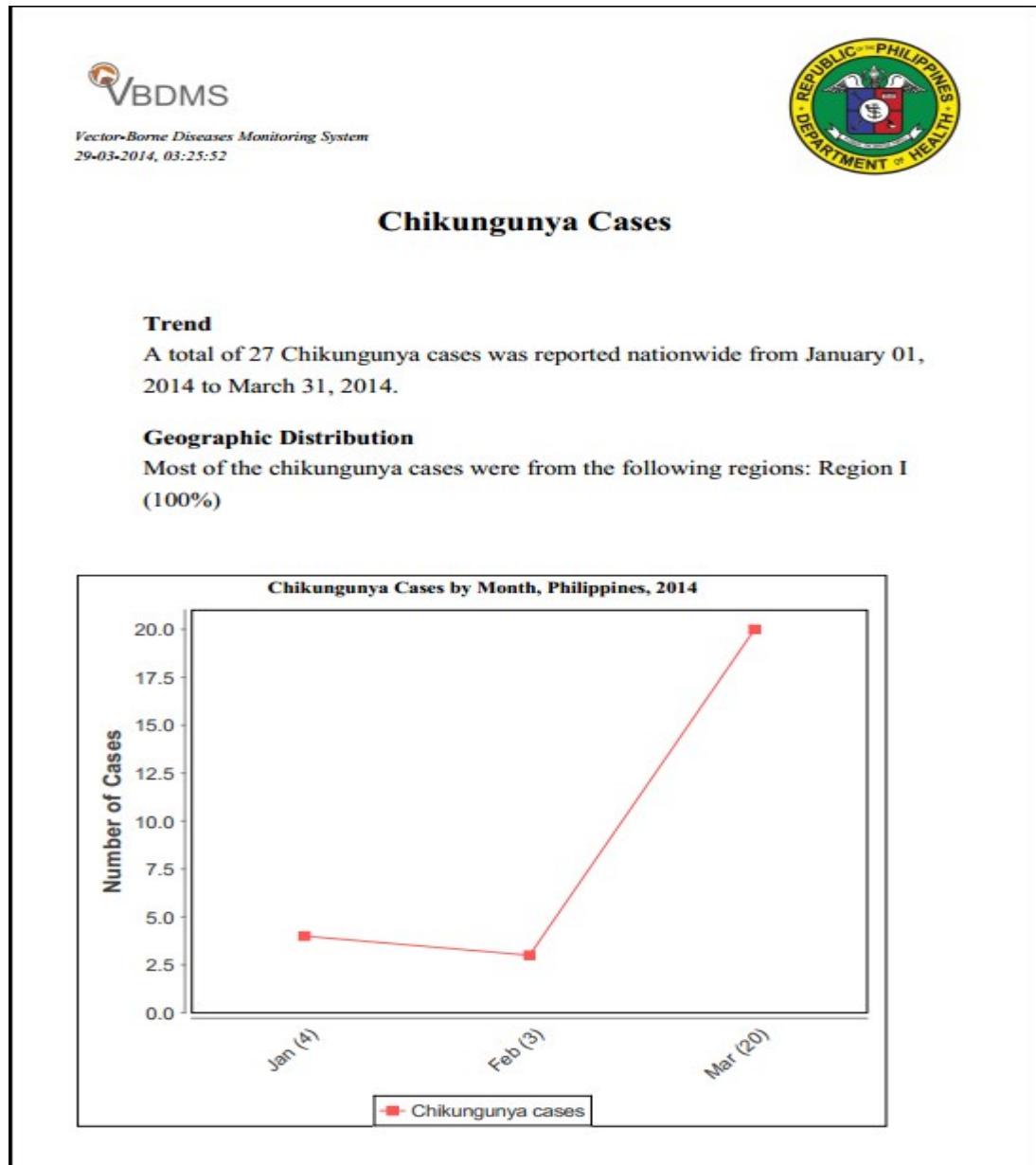


Figure 30: Generated Report, VBDMS

A sample generated report for vector-borne disease Chikungunya is shown in Figure 30 while sample generated dataset in spreadsheet is seen in Figure 31. Figure 32 on the other hand shows the sample generated SMS for an outbreak.

F. Generated Dataset in Spreadsheet

Microsoft Excel - VBDMS dataset(1) [Compatibility Mode]														
	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Disease:	Chikungunya												
2	Record No	Age	Gender	Symptoms	Date Onset	Outcome	Date Died	Barangay	Muncity	Province	Region			
3	1	21	Male	High fever	2014-03-02	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
4	2	41	Female	High fever	2014-03-02	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
5	3	23	Male	High fever	2014-03-09	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
6	4	26	Female	High fever	2014-03-06	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
7	5	41	Female	High fever	2014-03-03	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
8	6	37	Female	High fever	2014-03-06	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
9	7	21	Male	High fever	2013-03-09	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
10	8	22	Male	High fever	2014-03-02	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
11	9	22	Male	High fever	2014-03-02	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
12	10	22	Male	High fever	2014-03-02	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
13	11	22	Male	High fever	2014-03-02	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
14	12	22	Male	High fever	2014-03-02	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
15	13	22	Male	High fever	2014-03-02	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
16	14	10	Female	High fever	2014-03-11	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
17	15	10	Female	High fever	2014-03-11	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
18	16	10	Female	High fever	2014-03-11	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
19	17	10	Female	High fever	2014-03-11	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
20	18	29	Male	High fever	2014-03-03	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
21	19	21	Male	High fever	2014-03-03	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
22	20	21	Male	High fever	2014-03-02	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
23	21	41	Female	High fever	2014-03-02	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
24	22	23	Male	High fever	2014-03-09	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
25	23	21	Male	High fever	2013-03-02	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
26	24	41	Female	High fever	2013-03-02	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
27	25	21	Male	High fever	2013-03-02	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
28	26	41	Female	High fever	2013-03-02	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
29	27	41	Female	High fever	2013-03-02	Recovered		Baay	LINGAYEN	Pangasinan	Region I			
30														
31	Vector-Borne Disease Monitoring System													
32														
33														

Figure 31: Generated Spreadsheet, VBDMS

A sample generated dataset in spreadsheet for vector-borne disease Chikungunya is shown in Figure 31.

G. Generated SMS

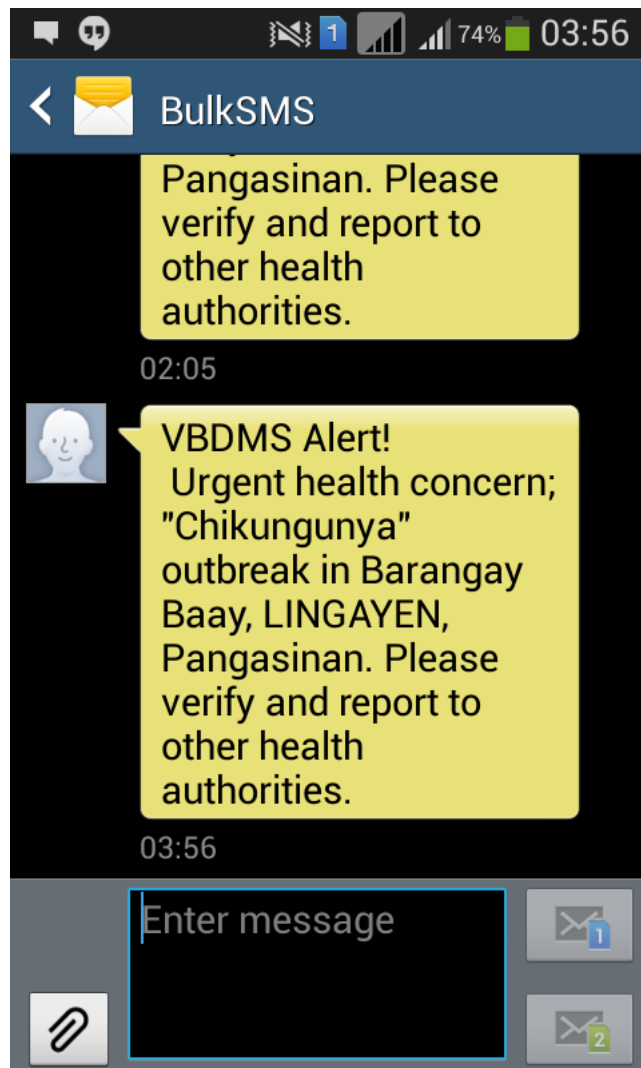


Figure 32: Generated SMS, VBDMS

A sample generated SMS sent to health officials when there's an outbreak is shown in Figure 32.

VII. Discussions

The Vector-Borne Diseases Monitoring System (VBDMS) is a web application which aims to help in monitoring vector-borne diseases, to provide up-to-date information regarding the incidence of a vector-borne disease and to determine barangays where outbreaks of a vector-borne disease occur. The web application developed was able to function just like how it was designed to and was able to meet the main and other objectives that were set by the developer.

Using Spring MVC Framework made the development of this web application easier in some ways. There are many spring resources which greatly helped the researcher in developing the web application without requiring too much effort. Problems also arised in configuring the xml files needed for the web application. Lot of patience and research was done by the developer in order to implement spring resources like spring-security which is an essential tool for web applications especially for systems with many different users and user types like this system.

The programming language used in writing the controllers is Java, and since Java has many widely known API (Application Programming Interface) and libraries which made some functionalities easier to be implemented. But in the view pages, using JSTL somehow caused difficulty for the researcher. Merging JSTL with javascript requires other medium, JSON, to be able to make the programming language work smoothly together. The developer, being unfamiliar with JSON, decided to use other means that worked for the system like calling the script internally within the page for the compiler to be able to read JSTL codes with the javascript.

The system has many types of users to carry out the system's objective or main tasks as well as to monitor the activities of other users in the system. Many types of system user means that the view pages for each type of user will be a little bit similar or will be mostly different especially for tasks, functionalities or areas unique or within the vicinity of the type of user. Creating different pages for different type of users is quite tedious even if they have the same functionalities.

The developer also made use of nested modals and data table to let the users easily navigate areas within their vicinity but using these and dealing with a nationwide area up to the barangay level caused the browser to not respond having large load of work on rendering so many modals and data tables. The large number of records for mapping barangays also made the running time of the web application to be slow. This may also be caused by the complexity of the code of the developer and the condition of the processor of the computer running the server.

Another problem encountered by the developer is the unavailability of a free SMS Gateway provider in the country. The SMS Gateway used in the system, BulkSMS, was used to send SMS to health officials notifying them when theres an outbreak. This module of the system is of utmost importance and one of the main objective of the system. But a disadvantage of this SMS Gateway is that it charges the user registered to the SMS Gateway provider for every message sent.

Comparing VBDMS with systems being used by the DOH, it may lack in functionalities since the researcher didn't have a first hand experience in using the systems they use such as the Philippine Integrated Disease Surveillance and Response (PIDSRS) and the Philippine Malaria Information System (PhilMIS). But even if VBDMS may lack some functionalities, it makes up for it through its powerful tools such as the barangay mapping and statistics of the diseases, the SMS notification to health officials and the built-in reports generation and dataset exporting which cannot be found on the forementioned systems being used by the DOH in monitoring diseases in the country.

VIII. Conclusions

Vector-Borne Disease Monitoring System (VBDMS) provides aid in vector-borne disease surveillance. It helps in tracking down barangays with cases of certain vector-borne disease and notifies the health officials for proper courses of action.

Patient records, which are gathered through household surveying and through patients visiting the barangay health center, will be added by Barangay Health Workers to the system. After adding these barangay patient records, they will be automatically sent to municipal/city health workers.

The Municipal/City Health Workers will check if the sent barangay patient records are credible enough and may also pay a visit in the barangay health center to see if the situation based on the received records are true. Once these patient records are approved, they will be sent to the provincial health workers.

The Provincial Health Workers will approve the records sent by the municipal/city health workers, and once approved, they will be added to the systems patients database records. These will also reflect on the map, and as well as on the disease statistics page.

After the approval of the provincial health workers of the records, the system will check if there's an outbreak in the barangay. The system will send an alert to all the health officials under that barangay, municipality, province and even the national health officials, and they will also receive a SMS as well to notify that a certain area has a vector-borne-disease outbreak. Now, it would be on the hands of the health officials to take the necessary course of actions or protocols and proper interventions regarding the outbreak of vector-borne disease in that area.

IX. Recommendations

This special project monitors the vector-borne diseases chikungunya, dengue and malaria. These three diseases were the initially proposed diseases since they are the most widely known cases in the country and the vector carrying these diseases are mosquito which is in every Filipino household. To have a more extensive monitoring of diseases in the country, adding other vector-borne diseases in the system with other vectors aside from mosquitoes would be beneficial once added to the system.

It would also be good to have a system messaging or chat function among the users so information gathering and disseminating would be a lot easier. There will be instant communication of users within the system where they can interchange disease information, disease notifications as well as discuss proper interventions and seek help from other health officials and workers easily by being connected to the system.

X. Bibliography

- [1] G. Rascalou, D. Pontier, F. Menu, and S. Gourbiere, “Emergence and prevalence of human vector-borne diseases in sink vector populations,” *PLoS ONE*, vol. 7, 2012.
- [2] V. E. P. Jr. Personal Interview, September 2013.
- [3] N. C. Nieto, K. Khan, G. Uhliah, and M. B. Teglas, “The emergence and maintenance of vector-borne diseases in the khyber pakhtunkhwa province, and the federally administered tribal areas of pakistan,” *Frontiers in Physiology*, vol. 3, 2012.
- [4] L. Bacani, “Doh confirms dengue-like virus outbreak in antique town.” <http://www.philstar.com/nation/2013/06/28/959267/doh-confirms-dengue-virus-outbreak-antique-town>, September 2013.
- [5] “Impact of dengue.” <http://www.who.int/csr/disease/dengue/impact/en/>, September 2013.
- [6] “Mosquito bites.” <http://www.doh.gov.ph/content/mosquito-bites.html>, September 2013.
- [7] “Impact of malaria,” September 2013.
- [8] “Doh: Malaria cases reach record low.” <http://www.gov.ph/2012/08/24/doh-malaria-cases-reach-record-low/>, September 2013.
- [9] “Philippine malaria information system (philmis),” September 2013.
- [10] R. Ramasamy and S. N. Surendran, “Possible impact of rising sea levels on vector-borne infectious diseases,” *BMC Infectious Diseases*, vol. 11, 2011.
- [11] L. Eisen, M. Coleman, S. Lozano-Fuentes, N. McEachen, M. Orleans, and M. Coleman, “Multi-disease data management system platform for vector-borne diseases,” *Neglected Tropical Diseases*, vol. 5, 2011.

- [12] L. Eisen and R. J. Eisen, “Using geographic information system and decision support systems for the prediction, prevention, and control of vector-borne diseases,” *Annual Review of Epidemiology*, vol. 56, pp. 41–61, 2010.
- [13] A. Y. Chang, M. E. Parrales, J. Jimenez, M. E. Sobieszczyk, S. M. Hammer, D. J. Copenhaver, and R. P. Kulkarni, “Combining google earth and gis mapping technologies in a dengue surveillance system for developing countries,” *International Journal of Health Geographics*, vol. 8, 2009.
- [14] H. M. Khormi and L. Kumar, “Modeling dengue fever risk based on socioeconomic parameters, nationality and age groups: Gis and remote sensing based case study,” *Science of the Total Environment*, 2011.
- [15] H. M. K. and, “Modeling spatio-temporal risk changes in the incidence of dengue fever in saudi arabia: a geographical information case study,” *Geospatial Health*, vol. 6, pp. 77–84, 2011.
- [16] Z. Huang, A. Das, Y. Qiu, and A. J. Tatem, “Web-based gis: the vector-borne diseases airline importation risk (vbd-air) tool,” *International Journal of Health Geographics*, vol. 11, 2012.
- [17] W. Yang, Z. Li, Y. Lan, J. Wang, J. Ma, L. Jin, Q. Sun, W. Lv, S. Lai, Y. Liao, and W. Hu, “A nationwide web-based automated system for outbreak detection and rapid response in china,” *WPSAR*, vol. 2, 2011.
- [18] G. M. Amoli, “Gis-based risk map analysis of leishmaniasis disease in isfahan, iran,” 2011.
- [19] Z. Zhen, J. Jing-min, and L. Fang, “The application of geographic information system (gis) in the field of public health,” 2010.
- [20] “Vector-borne diseases.,” October 2013.
- [21] “Vector-borne disease control.” <http://www.vdh.state.va.us/epidemiology/dee/vectorborne>, October 2013.

- [22] “Chikungunya.” <http://www.who.int/mediacentre/factsheets/fs327/en/>, October 2013.
- [23] “Dengue and severe dengue.” <http://www.who.int/mediacentre/factsheets/fs117/en/>, October 2013.
- [24] “Malaria.” <http://www.who.int/mediacentre/factsheets/fs094/en/>, October 2013.
- [25] “Malaria medium term development plan.” <http://www.doh.gov.ph/sites/default/files/FinalSeptember> 2013.
- [26] “National epidemiology center.” <http://www.doh.gov.ph/content/national-epidemiology-center.html>, October 2013.
- [27] “The proceedings and monitoring.” <http://www.who.int/ihr/procedures/en/>, October 2013.
- [28] “Republic act no. 3573.” <http://www.chanrobles.com/republicactno3573.htm>006C, October 2013.
- [29] “Resolution wha48.13,” October 2013.
- [30] “Administrative order 2005-0023,” October 2013.
- [31] “Epidemiology.” <http://www.who.int/topics/epidemiology/en/>, October 2013.
- [32] “Communicable disease surveillance.” <http://www.gov.mb.ca/health/publichealth/cdc/prot> October 2013.
- [33] R. M. Stair and G. W. Reynolds, *Fundamentals of Information System*. Cengage Learning, January 1, 2011.
- [34] R. Vidgen, *Developing Web Information Systems: From Strategy to Implementation*. Butterworth-Heinemann, September 25, 2002.

- [35] J. Rodrigues, *Health Information Systems: Concepts, Methodology, Tools and Application*. Idea Group Inc. (IGI), December 31, 2009.
- [36] M. K. Pakhira, *DATABASE MANAGEMENT SYSTEM*. PHI Learning Pvt. Ltd., October 30, 2012.
- [37] M. D. Kennedy, *Introducing Geographic Information Systems with ArcGIS: A Workbook Approach to Learning GIS*. John Wiley & Sons, March 20, 2013.
- [38] M. H. B. Selamat, A. B. Selamat, M. S. B. Othman, N. H. B. M. Shamsuddin, and N. I. B. M. Zukepli, "A review on geographical information system (gis) in town planning: Malaysia experience," *Geoinformatica: An International Journal*, vol. 2, 2012.
- [39] S. E. houssaini, "Development of a gis-based monitoring system for road network," *JOURNAL OF COMPUTING*, vol. 2, 2012.
- [40] W. Wang, G. Jianhua, L. Zhang, J. Zhang, F. Liqun, and C. Wuchun, "Design and implementation of mobile geospatial information system for public health emergency," *IEEE*, 2009.
- [41] E. K. Cromley and S. McLafferty, *GIS and Public Health*. Guildord Press, 2012.

XI. Appendix

A. Source Code

1. XML Files

Listing 1: web.xml

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <web-app xmlns:xsi="http://www.w3.org/2001/
  XMLSchema-instance"
3   xmlns="http://java.sun.com/xml/ns/javaee"
4   xsi:schemaLocation="http://java.sun.com/xml/
  ns/javaee http://java.sun.com/xml/ns/javaee
  /web-app_3.0.xsd" id="WebApp_ID" version
  ="3.0">
5
6   <display-name>VectorBorne</display-name>
7
8   <welcome-file-list>
9     <welcome-file>home.jsp</welcome-file>
10  </welcome-file-list>
11
12  <servlet>
13    <servlet-name>Disease</servlet-name>
14    <servlet-class>
15      org.springframework.web.servlet.
        DispatcherServlet
16    </servlet-class>
17    <load-on-startup>1</load-on-startup>
18  </servlet>
19
20  <servlet-mapping>
21    <servlet-name>Disease</servlet-name>
22    <url-pattern>/</url-pattern>
23  </servlet-mapping>
24
25  <servlet-mapping>
26    <servlet-name>default</servlet-name>
27    <url-pattern>*.css</url-pattern>
28  </servlet-mapping>
29
30  <servlet-mapping>
31    <servlet-name>default</servlet-name>
32    <url-pattern>*.js</url-pattern>
33  </servlet-mapping>
34
35  <servlet-mapping>
36    <servlet-name>default</servlet-name>
37    <url-pattern>*.jpg</url-pattern>
38  </servlet-mapping>
39
40  <servlet-mapping>
41    <servlet-name>default</servlet-name>
42    <url-pattern>*.png, *.svg, *.wotf, *.ttf
43  </url-pattern>
44  </servlet-mapping>
45
46  <servlet-mapping>
47    <servlet-name>default</servlet-name>
48    <url-pattern>*.svg</url-pattern>
49  </servlet-mapping>
50
51  <servlet>
52    <servlet-name>ongetParameterValues</
53    servlet-name>
54    <servlet-class>arraying.
55      OngetParameterValues</servlet-class>
56  </servlet>
57
58  <servlet-mapping>
59    <servlet-name>ongetParameterValues</
60    servlet-name>
61    <url-pattern>/onGPV</url-pattern>
62  </servlet-mapping>
63
64  <context-param>
65    <param-name>contextConfigLocation</param
66    name>
67    <param-value>
68      /WEB-INF/Disease-servlet.xml
69      /WEB-INF/security-context.xml
70    </param-value>
71  </context-param>
72
73  <!-- Filter for the Spring Security -->
74  <filter>
75    <filter-name>springSecurityFilterChain</
76    filter-name>
77    <filter-class>org.springframework.web.
78      filter.DelegatingFilterProxy</filter-class>
79  </filter>
80
81  <filter-mapping>
82    <filter-name>springSecurityFilterChain</
83    filter-name>
84    <url-pattern>/*</url-pattern>
85  </filter-mapping>
86
87  <listener>
88    <listener-class>org.springframework.web.
89      context.ContextLoaderListener</listener-
90      class>
91  </listener>
92
93  <listener>
94    <listener-class>
95      org.springframework.security.web.session.
96      HttpSessionEventPublisher
97  </listener-class>
98  </listener>
99
100 <error-page>
101   <exception-type>misc.CustomGenericException</
102   exception-type>
103   <location>/WEB-INF/views/custom_error.jsp</
104   location>
105 </error-page>
106
107 <error-page>
108   <exception-type>java.lang.Exception</exception
109   -type>
110   <location>/WEB-INF/views/generic_error.jsp</
111   location>
112 </error-page>
113 </web-app>
```

Listing 2: Disease-servlet.xml

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <beans xmlns="http://www.springframework.org/
  schema/beans"
3   xmlns:xsi="http://www.w3.org/2001/XMLSchema-
  instance"
4   xmlns:p="http://www.springframework.org/
  schema/p"
5   xmlns:context="http://www.springframework.
  org/schema/context"
6   xmlns:mvc="http://www.springframework.org/
  schema/mvc"
7   xsi:schemaLocation="
8     http://www.springframework.org/schema/beans
9     http://www.springframework.org/schema/beans/
  spring-beans-3.2.xsd
10    http://www.springframework.org/schema/mvc
11    http://www.springframework.org/schema/mvc/spring
  -mvc-3.2.xsd
12    http://www.springframework.org/schema/context
13    http://www.springframework.org/schema/context/
  spring-context-3.2.xsd">
14
15   <context:component-scan
16     base-package="controllers" />
17
18   <mvc:annotation-driven />
19   <bean id="viewResolver"
20     class="org.springframework.web.servlet.
21     view.UrlBasedViewResolver">
22     <property name="viewClass"
23       value="org.springframework.web.
24       servlet.view.JstlView" />
25     <property name="prefix" value="/WEB-INF/
26     views/" />
27     <property name="suffix" value=".jsp" />
28   </bean>
29
30   <bean id="messageSource"
31     class="org.springframework.context.support.
32     ResourceBundleMessageSource">
33     <property name="basename" value="messages" />
34   </bean>
35
36   <!-- for datasource -->
```



```

33 <bean id="dataSource"
34     class="org.springframework.jdbc.datasource
        .DriverManagerDataSource">
35     <property name="driverClassName" value="
        com.mysql.jdbc.Driver"/>
36     <property name="url" value="jdbc:mysql://
        localhost:3306/vbd?zeroDateTimeBehavior=
        convertToNull"/>
37     <property name="username" value="root"/>
38     <property name="password" value=""/>
39 </bean>
40
41 <bean id="barangayRecordsDAO" class="dao.
        BarangayRecordsDAO" />
42 <bean id="diseaseDAO" class="dao.DiseaseDAO
        " />
43 <bean id="mapsDAO" class="dao.MapsDAO" />
44
45 <bean id="newUserDAO" class = "dao.
        NewUserDAO" />
46 <bean id="patientsDAO" class="dao.
        PatientsDAO" />
47 <bean id="registrationDAO" class="dao.
        RegistrationDAO" />
48 <bean id="reportsDAO" class="dao.ReportsDAO"
        />
49 <bean id="userProfileDAO" class="dao.
        UserProfileDAO" />
50
51 <bean id="BCryptPasswordEncoder" class="org.
        springframework.security.crypto.bcrypt.
        BCryptPasswordEncoder" />
52 </beans>

```

Listing 3: security-context.xml

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <beans:beans xmlns="http://www.springframework.
        org/schema/security"
3     xmlns:beans="http://www.springframework.org/
        schema/beans" xmlns:xsi="http://www.w3.org
        /2001/XMLSchema-instance"
4     xsi:schemaLocation="http://www.springframework.
        org/schema/beans http://www.springframework
        .org/schema/beans/spring-beans-3.1.xsd
5         http://www.
        springframework.org/schema/security http://
        www.springframework.org/schema/security/
        spring-security-3.1.xsd">
6
7     <http auto-config="true" use-expressions = "
        true">
8         <intercept-url pattern="/pages*" access="
            hasAnyRole('ROLE_USER', 'ROLE_BHWorker', '
            ROLE_MCHWorker', 'ROLE_MCOfficial', '
            ROLE_PHWorker', 'ROLE_PHOOfficial', '
            ROLE_NHOOfficial')" />
9         <!-- requires-channel="https" -->
10        <intercept-url pattern="/pages/insertRecords*"
            access="hasAnyRole('ROLE_USER', '
            ROLE_BHWorker', 'ROLE_MCHWorker', '
            ROLE_MCOfficial', 'ROLE_PHWorker', '
            ROLE_PHOOfficial', 'ROLE_NHOOfficial')" />
11        <form-login
12            login-page="/home"
13            default-target-url="/loginSuccess"
14            authentication-failure-url="/loginfailed"
15        />
16        <logout logout-success-url="/home"
17        />
18    </http>
19
20    <beans:bean id="encoder" class="org.
        springframework.security.crypto.bcrypt.
        BCryptPasswordEncoder"/>
21
22    <authentication-manager>
23    <authentication-provider>
24        <password-encoder ref = "encoder"/>
25        <jdbc-user-service data-source-ref="
26            dataSource"
27            users-by-username-query = "SELECT
                username, password, true FROM users WHERE
                username = ?"
28            authorities-by-username-query = "SELECT
                username, authorities FROM users WHERE
                username = ?"
29        />
30    </authentication-provider>
31 </authentication-manager>
32
33 </beans:beans>

```

2. Controllers

Listing 4: HomeController.java

```

1 package controllers;
2
3 import java.net.URL;
4 import java.sql.SQLException;
5 import java.text.DateFormat;
6 import java.text.SimpleDateFormat;
7 import java.util.ArrayList;
8 import java.util.Calendar;
9 import java.util.Date;
10 import java.util.List;
11 import java.util.Map;
12
13 import javax.mail.internet.InternetAddress;
14 import javax.servlet.ServletOutputStream;
15 import javax.servlet.http.HttpServlet;
16 import javax.servlet.http.HttpServletRequest;
17 import javax.servlet.http.HttpServletResponse;
18 import javax.validation.Valid;
19
20 import misc.SendEmail;
21 import misc.SendSMS;
22
23 import org.apache.catalina.connector.Request;
24 import org.jfree.ui.RefineryUtilities;
25 import org.springframework.beans.factory.
        annotation.Autowired;
26 import org.springframework.core.io.
        ClassPathResource;
27 import org.springframework.core.io.
        FileSystemResource;
28 import org.springframework.core.io.Resource;
29 import org.springframework.http.HttpRequest;
30 import org.springframework.security.core.
        Authentication;
31 import org.springframework.security.core.context.
        SecurityContextHolder;
32 import org.springframework.security.crypto.
        bcrypt.BCryptPasswordEncoder;
33 import org.springframework.security.crypto.
        password.PasswordEncoder;
34 import org.springframework.stereotype.Controller
        ;
35 import org.springframework.ui.ModelMap;
36 import org.springframework.validation.
        BindingResult;
37 import org.springframework.web.bind.annotation.
        ModelAttribute;
38 import org.springframework.web.bind.annotation.
        PathVariable;
39 import org.springframework.web.bind.annotation.
        RequestMapping;
40 import org.springframework.web.bind.annotation.
        RequestMethod;
41 import org.springframework.web.bind.annotation.
        RequestParam;
42 import org.springframework.web.bind.annotation.
        SessionAttributes;
43 import org.springframework.web.servlet.
        ModelAndView;
44 import org.springframework.web.servlet.mvc.
        support.RedirectAttributes;
45
46 import com.itextpdf.text.DocumentException;
47
48 import dao.BarangayRecordsDAO;
49 import dao.DiseaseDAO;
50 import dao.MapsDAO;
51 import dao.NewUserDAO;
52 import dao.PatientsDAO;
53 import dao.RegistrationDAO;
54 import dao.ReportsDAO;
55 import dao.UserProfileDAO;
56 import objects.BarangayRecords;
57 import objects.Diseases;
58 import objects.FormStruct;
59 import objects.Maps;
60 import objects.NewUser;
61 import objects.Patients;
62 import objects.PatientsForm;
63 import objects.Place;
64 import objects.Registrants;
65 import objects.Reports;
66 import objects.UserChange;
67 import objects.UserProfile;
68 import pdf.GenerateReports;
69

```

```

70 @Controller
71 @SessionAttributes({"registers"})
72 public class HomeController extends HttpServlet
73 {
74     @Autowired
75     DiseaseDAO diseasesDAO;
76     @Autowired
77     NewUserDAO newUserDAO;
78     @Autowired
79     UserProfileDAO userProfileDAO;
80     @Autowired
81     RegistrationDAO registrationDAO;
82     @Autowired
83     BarangayRecordsDAO brgyRecordsDAO;
84     @Autowired
85     PatientsDAO patientsDAO;
86     @Autowired
87     MapsDAO mapsDAO;
88     @Autowired
89     ReportsDAO reportsDAO;
90
91     @RequestMapping(value = "/home", method =
92         RequestMethod.GET)
93     public String home(ModelMap model) throws
94         Exception {
95         ProfileController pr = new ProfileController()
96         ;
97         if(pr.alertCheck == 0) {
98             pr.alertCheck = 1;
99             diseasesDAO.updateAlerts();
100         }
101         List<Diseases> dis = diseasesDAO.listDiseases
102         ();
103         model.addAttribute("disease", dis);
104         model.addAttribute("pages", "home");
105         return "home";
106     }
107
108     @RequestMapping(value = "/disease/{diseaseID}",
109         method = {RequestMethod.GET, RequestMethod
110         .POST})
111     public String chikungunya(@PathVariable("
112         diseaseID") String disease,
113         ModelMap model) throws Exception {
114         ProfileController pr = new ProfileController()
115         ;
116         if(pr.alertCheck == 0) {
117             pr.alertCheck = 1;
118             diseasesDAO.updateAlerts();
119         }
120         List<Diseases> dis = diseasesDAO.listDiseases
121         ();
122         if(disease.equalsIgnoreCase("Chikungunya")) {
123             List<Maps> map = mapsDAO.listOfCPatients();
124             model.addAttribute("maps", map);
125             System.out.println(map.size());
126         } else if(disease.equalsIgnoreCase("Dengue"))
127         {
128             List<Maps> map = mapsDAO.listOfDPatients();
129             model.addAttribute("maps", map);
130         } else if(disease.equalsIgnoreCase("Malaria"))
131         {
132             List<Maps> map = mapsDAO.listOfMPatients();
133             model.addAttribute("maps", map);
134         }
135         model.addAttribute("disease", dis);
136         model.addAttribute("diseaseName", disease);
137         return "disease";
138     }
139
140     @RequestMapping(value = "/statistics/{disease
141         }", method = {RequestMethod.GET,
142         RequestMethod.POST})
143     public ModelAndView stat(@PathVariable("disease
144         ") String disease, ModelMap model) throws
145         Exception {
146         ProfileController pr = new ProfileController()
147         ;
148         if(pr.alertCheck == 0) {
149             pr.alertCheck = 1;
150             diseasesDAO.updateAlerts();
151         }
152         model.addAttribute("pages", "statistics");
153
154         List<Patients> patients = patientsDAO.
155         listAllPatients();
156         List<Patients> patients2 = patientsDAO.
157         listOldPatients();
158         String[] regions = {"ARMM", "CAR", "NCR", "
159         Region I", "Region II", "Region III",
160         "Region IV-A", "Region IV-B", "Region V", "
161         Region VI", "Region VII",
162         "Region VIII", "Region IX", "Region X", "
163         Region XI", "Region XII", "Region XIII"};
164         List<List<Maps>> map = new ArrayList<List<Maps
165         >>(17);
166
167         for(int i = 0; i < 17; i++) {
168             map.add(i, mapsDAO.listAllMuncity(regions[i]
169             ));
170         }
171
172         int[] popula = new int[17];
173         int mCases = 0;
174         int mDeaths = 0;
175         int oCases = 0;
176         int oDeaths = 0;
177         for(int i = 0; i < 17; i++) {
178             popula[i] = 0;
179             for(Maps ms : map.get(i)) {
180                 popula[i] += ms.getPopulation();
181                 mCases = 0;
182                 mDeaths = 0;
183                 for(Patients pat : patients) {
184                     if(pat.getDisease().equalsIgnoreCase(
185                     disease)) {
186                         if(pat.getRegion().equalsIgnoreCase(
187                         regions[i])) {
188                             if(pat.getMuncity().equalsIgnoreCase(ms.
189                             getMuncity())) {
190                                 mCases++;
191                                 if(pat.getDeceased().equalsIgnoreCase("
192                                 Died"))
193                                     mDeaths++;
194                             }
195                         }
196                     }
197                 }
198             }
199             for(Patients pat : patients2) {
200                 if(pat.getDisease().equalsIgnoreCase(
201                 disease)) {
202                     if(pat.getRegion().equalsIgnoreCase(
203                     regions[i])) {
204                         if(pat.getMuncity().equalsIgnoreCase(ms.
205                         getMuncity())) {
206                             oCases++;
207                             if(pat.getDeceased().equalsIgnoreCase("
208                             Died"))
209                                 oDeaths++;
210                         }
211                     }
212                 }
213             }
214             ms.setCountMCases(mCases);
215             ms.setCountMDeaths(mDeaths);
216             ms.setCountOCases(oCases);
217             ms.setCountODeaths(oDeaths);
218         }
219         model.put("map", map);
220
221         int cases[] = new int[17];
222         int deaths[] = new int[17];
223         int oldCases[] = new int[17];
224         int oldDeaths[] = new int[17];
225
226         for(Patients pat : patients) {
227             for(int i = 0; i < regions.length; i++) {
228                 if(pat.getDisease().equalsIgnoreCase(disease
229                 )) {
230                     if(pat.getRegion().equalsIgnoreCase(regions
231                     [i])) {
232                         cases[i]++;
233                         if(pat.getDeceased().equalsIgnoreCase("Yes
234                         "))
235                             deaths[i]++;
236                     }
237                 }
238             }
239         }
240
241         for(Patients pat : patients2) {
242             for(int i = 0; i < regions.length; i++) {
243                 if(pat.getDisease().equalsIgnoreCase(disease
244                 )) {
245                     if(pat.getRegion().equalsIgnoreCase(regions
246                     [i])) {
247                         oldCases[i]++;
248                     }
249                 }
250             }
251         }
252
253         DateFormat dateFormat = new SimpleDateFormat("
254         yyyy-MM-dd");
255         Date date = new Date();
256         Date date2 = new Date();
257         date.setMonth(0);
258         date.setDate(1);
259         date2.setMonth(12);
260         date2.setDate(31);
261
262         List<PatientsForm> allPatients = patientsDAO.
263         listAllPatients(dateFormat.format(date),

```

```

229         dateFormat.format(date2));
230     int yearCases[] = new int[12];
231     int yearDeaths[] = new int[12];
232
233     for(PatientsForm pat : allPatients) {
234         String parts[] = pat.getDateSubmitted().split
235             ("-");
236         for(int i = 0; i < 12; i++) {
237             String days = "";
238             if(i < 9)
239                 days = "0" + (i+1);
240             else
241                 days = "" + (i+1);
242             if(pat.getDisease().equalsIgnoreCase(disease
243             )) {
244                 if(parts[i].equalsIgnoreCase(days)) {
245                     yearCases[i]++;
246                     if(pat.getOutcome().equalsIgnoreCase("Died
247                     "))
248                         yearDeaths[i]++;
249                 }
250             }
251         }
252     }
253     model.addAttribute("pat", patients);
254     model.addAttribute("regions", regions);
255     model.addAttribute("cases", cases);
256     model.addAttribute("deaths", deaths);
257     model.addAttribute("oldCases", oldCases);
258     model.addAttribute("popula", popula);
259     String op = "[";
260     String dat1 = "", dat2 = "";
261     for(int i = 0; i < 12; i++) {
262         dat1 += yearCases[i] + ", ";
263         dat2 += yearDeaths[i] + ", ";
264     }
265     dat1 = op + dat1 + "]";
266     dat2 = op + dat2 + "]";
267     model.addAttribute("data1", dat1);
268     model.addAttribute("data2", dat2);
269     model.addAttribute("diseaseName", disease);
270     String text = "Monthly " + disease + " Cases";
271     model.addAttribute("titles", text);
272     model.addAttribute("disease", diseasesDAO.
273         listDiseases());
274     return new ModelAndView("statistics");
275 }
276
277 @RequestMapping(value = "/campaigns", method =
278     RequestMethod.GET)
279 public ModelAndView camps(ModelMap model) {
280     ProfileController pr = new ProfileController()
281     ;
282     if(pr.alertCheck == 0) {
283         pr.alertCheck = 1;
284         diseasesDAO.updateAlerts();
285     }
286     List<Diseases> dis = diseasesDAO.listDiseases
287     ();
288     model.addAttribute("disease", dis);
289     return new ModelAndView("campaigns");
290 }
291
292 @RequestMapping(value = "/register")
293 public ModelAndView reg(ModelMap model) throws
294     Exception {
295     List<Diseases> dis = diseasesDAO.listDiseases
296     ();
297     model.addAttribute("disease", dis);
298     model.addAttribute("pages", "register");
299     model.put("registers", new NewUser());
300     model.addAttribute("userNames", userProfileDAO
301         .getAllUserNames());
302     model.addAttribute("userNumbers",
303         userProfileDAO.getAllUserNumbers());
304     return new ModelAndView("register");
305 }
306
307 @RequestMapping(value = "/registerUser", method
308     = {RequestMethod.POST, RequestMethod.GET})
309 public ModelAndView registerUser(
310     @ModelAttribute("registers") @Valid NewUser
311     user,
312     BindingResult result, ModelMap model) {
313     boolean err = false;
314     for(String u : userProfileDAO.getAllUserNames
315     ()) {
316         if(user.getUsername().equalsIgnoreCase(u)) {
317             model.addAttribute("nameTaken", "Username
318             already taken!");
319             err = true;
320         }
321     }
322     for(String u : userProfileDAO.getAllRegNumbers
323     ()) {
324         if(user.getNumber().equalsIgnoreCase(u)) {
325             model.addAttribute("numTaken", "Contact
326             number should be unique.");
327             err = true;
328         }
329     }
330     for(String u : userProfileDAO.getAllRegNumbers
331     ()) {
332         if(user.getNumber().equalsIgnoreCase(u)) {
333             model.addAttribute("numTaken", "Contact
334             number should be unique.");
335             err = true;
336         }
337     }
338     if(result.hasErrors() || err) {
339         model.addAttribute("userNames",
340             userProfileDAO.getAllUserNames());
341         model.addAttribute("userNumbers",
342             userProfileDAO.getAllUserNumbers());
343         model.addAttribute("pages", "register");
344         return new ModelAndView("register");
345     } else {
346         newUserDAO.insert(user.getFirstname(), user.
347             getMiddleName(),
348             user.getSurname(), user.getGender(), user.
349             getAddress(),
350             user.getUserType(), user.getUserTypeInt(),
351             user.getRegion(), user.getProvince(), user.
352             getMuncity(),
353             user.getBarangay(), user.getInstitution(),
354             user.getInstitutionAddr(), user.getEmail(),
355             user.getNumber());
356         return new ModelAndView("successRegister");
357     }
358 }
359
360 @RequestMapping(value = "/successRegister")
361 public ModelAndView success(ModelMap model) {
362     List<Diseases> dis = diseasesDAO.listDiseases
363     ();
364     model.addAttribute("disease", dis);
365     return new ModelAndView("successRegister");
366 }
367
368 @RequestMapping(value = "/loginfailed", method
369     = {RequestMethod.GET, RequestMethod.POST})
370 public String errorLogin(ModelMap model) {
371     List<Diseases> dis = diseasesDAO.listDiseases
372     ();
373     model.addAttribute("disease", dis);
374     model.addAttribute("pages", "loginfailed");
375     return "loginfailed";
376 }
377
378 @RequestMapping(value = "/"
379     changeCredentialsSuccessful")
380 public ModelAndView changeCredit(ModelMap model
381 ) {
382     List<Diseases> dis = diseasesDAO.listDiseases
383     ();
384     model.addAttribute("disease", dis);
385     return new ModelAndView("
386     changeCredentialsSuccessful");
387 }
388
389 @RequestMapping(value = "/loginSuccess")
390 public ModelAndView loginSuccess(ModelMap model
391 ) {
392     List<Diseases> dis = diseasesDAO.listDiseases
393     ();
394     model.addAttribute("disease", dis);
395     Authentication auth = SecurityContextHolder.
396     getContext()
397     .getAuthentication();
398     String name = auth.getName();
399     UserProfile prof = userProfileDAO.getUser(name
400     );
401     if(prof.getUserType().equalsIgnoreCase("
402     Barangay Health Worker") ||
403     prof.getUserType().equalsIgnoreCase("
404     Municipal/City Health Worker") ||
405     prof.getUserType().equalsIgnoreCase("
406     Provincial Health Worker")) {
407         return new ModelAndView("redirect:pages/
408         system");
409     }
410     else if(prof.getUserType().equalsIgnoreCase("
411     National Health Official") ||
412     prof.getUserType().equalsIgnoreCase("Municipal
413     /City Health Official") ||
414     prof.getUserType().equalsIgnoreCase("
415     Provincial Health Official") ||
416     prof.getUserType().equalsIgnoreCase("

```

```

384     Administrator")) {
385     return new ModelAndView("redirect:pages/
386     notifications");
387 }
388 else
389 }
390 }
391 }

```

ProfileController.java

Listing 5: ProfileController.java

```

1  package controllers;
2
3  import java.awt.Image;
4  import java.awt.Rectangle;
5  import java.io.ByteArrayOutputStream;
6  import java.io.File;
7  import java.io.FileInputStream;
8  import java.io.FileNotFoundException;
9  import java.io.IOException;
10 import java.io.InputStream;
11 import java.net.URL;
12 import java.nio.ByteBuffer;
13 import java.sql.SQLException;
14 import java.text.DateFormat;
15 import java.text.SimpleDateFormat;
16 import java.util.ArrayList;
17 import java.util.Calendar;
18 import java.util.Date;
19 import java.util.HashMap;
20 import java.util.Iterator;
21 import java.util.List;
22 import java.util.Map;
23 import java.util.Set;
24
25 import javax.servlet.ServletOutputStream;
26 import javax.servlet.http.HttpServlet;
27 import javax.servlet.http.HttpServletRequest;
28 import javax.servlet.http.HttpServletResponse;
29 import javax.swing.ImageIcon;
30 import javax.swing.JFrame;
31 import javax.swing.JLabel;
32 import javax.validation.Valid;
33
34 import misc.SendEmail;
35 import misc.SendSMS;
36
37 import org.apache.catalina.connector.Request;
38 import org.jfree.ui.RefineryUtilities;
39 import org.springframework.beans.factory.
40     annotation.Autowired;
41 import org.springframework.core.io.
42     ClassPathResource;
43 import org.springframework.core.io.Resource;
44 import org.springframework.security.core.
45     Authentication;
46 import org.springframework.security.core.context.
47     SecurityContextHolder;
48 import org.springframework.security.crypto.
49     bcrypt.BCryptPasswordEncoder;
50 import org.springframework.security.crypto.
51     password.PasswordEncoder;
52 import org.springframework.stereotype.Controller;
53 ;
54 import org.springframework.ui.ModelMap;
55 import org.springframework.validation.
56     BindingResult;
57 import org.springframework.validation.FieldError;
58 ;
59 import org.springframework.web.bind.annotation.
60     ModelAttribute;
61 import org.springframework.web.bind.annotation.
62     PathVariable;
63 import org.springframework.web.bind.annotation.
64     RequestMapping;
65 import org.springframework.web.bind.annotation.
66     RequestMethod;
67 import org.springframework.web.bind.annotation.
68     RequestParam;
69 import org.springframework.web.bind.annotation.
70     SessionAttributes;
71 import org.springframework.web.servlet.
72     ModelAndView;
73 import org.springframework.web.servlet.mvc.
74     support.RedirectAttributes;
75
76 import com.itextpdf.text.DocumentException;
77 import com.sun.pdfview.PDFFile;
78 import com.sun.pdfview.PDFPage;
79
80 import dao.BarangayRecordsDAO;
81 import dao.DiseaseDAO;
82 import dao.MapsDAO;
83 import dao.NewUserDAO;
84 import dao.PatientsDAO;
85 import dao.RegistrationDAO;
86 import dao.ReportsDAO;
87 import dao.UserProfileDAO;
88 import objects.BarangayRecords;

```

```

387 return new ModelAndView("pages/reports");
388
389 }
390
391 }
}

import objects.ContactsChange;
import objects.Diseases;
import objects.FormStruct;
import objects.Maps;
import objects.NewUser;
import objects.Patients;
import objects.PatientsForm;
import objects.Place;
import objects.Registrants;
import objects.Reports;
import objects.UserChange;
import objects.UserLogs;
import objects.UserProfile;
import pdf.GenerateExcel;
import pdf.GenerateReports;

@Controller
@SessionAttributes({"changeUser", "
changeContacts", "cBrgyForm", "addNewUser",
"forReports", "diseases", "forExcel",
"name", "age", "gender", "deceased", "
patientForm"})

public class ProfileController extends
    HttpServlet {
    public int alertCheck = -1;
    @Autowired
    DiseaseDAO diseasesDAO;
    @Autowired
    NewUserDAO newUserDAO;
    @Autowired
    UserProfileDAO userProfileDAO;
    @Autowired
    RegistrationDAO registrationDAO;
    @Autowired
    BarangayRecordsDAO brgyRecordsDAO;
    @Autowired
    MapsDAO mapsDAO;
    @Autowired
    ReportsDAO reportsDAO;
    @Autowired
    PatientsDAO patientsDAO;

    @RequestMapping(value = "/pages/profile",
        method = {RequestMethod.GET, RequestMethod.
        POST})
    public ModelAndView toProfile(ModelMap model)
        throws Exception {
        Authentication auth = SecurityContextHolder.
        getContext()
        .getAuthentication();
        String name = auth.getName();
        System.out.println(name);
        UserProfile prof = userProfileDAO.getUser(name
        );

        model.put("disease", diseasesDAO.listDiseases
        ());
        model.addAttribute("user", prof);
        model.addAttribute("userDetails",
        userProfileDAO.getUserDetails(name));
        model.addAttribute("thePage", "profile");
        model.addAttribute("us", new UserChange());
        model.addAttribute("insF", new FormStruct());
        model.addAttribute("reports", new Reports());
        model.put("changeUser", new UserChange());
        model.put("changeContacts", new ContactsChange
        ());
        model.put("cBrgyForm", new FormStruct());

        if(prof.getStatus() == 1)
            return new ModelAndView("pages/profile");
        else {
            return new ModelAndView("pages/errors/
            userDeactivated");
        }
    }

    @RequestMapping(value = "/pages/messages",
        method = {RequestMethod.GET, RequestMethod.
        POST})
    public ModelAndView toMessages(ModelMap model)
    {
        Authentication auth = SecurityContextHolder.
        getContext()
        .getAuthentication();
        String name = auth.getName();
        UserProfile prof = userProfileDAO.getUser(name
    )

```

```

    );
144 model.addAttribute("disease", diseasesDAO.
    listDiseases());
145 model.addAttribute("user", prof);
146 model.addAttribute("userDetails",
    userProfileDAO.getUserDetails(name));
147 model.addAttribute("thePage", "messages");
148
149 return new ModelAndView("pages/messages");
150 }
151
152 @RequestMapping(value = "/pages/notifications",
    method = {RequestMethod.GET, RequestMethod
    .POST})
153 public ModelAndView toNotifications(ModelMap
    model) {
154
155     if(alertCheck == -1)
156         alertCheck = 0;
157
158     Authentication auth = SecurityContextHolder.
    getContext()
159         .getAuthentication();
160     String name = auth.getName();
161     UserProfile prof = userProfileDAO.getUser(name
    );
162     model.addAttribute("disease", diseasesDAO.
    listDiseases());
163     model.addAttribute("user", prof);
164     model.addAttribute("userDetails",
    userProfileDAO.getUserDetails(name));
165     model.addAttribute("thePage", "notifications")
    ;
166     String[] regions = {"ARMH", "CAR", "NCR", "
    Region I", "Region II", "Region III",
167     "Region IV-A", "Region IV-B", "Region V", "
    Region VI", "Region VII",
168     "Region VIII", "Region IX", "Region X", "
    Region XI", "Region XII", "Region XIII"};
169
170     System.out.println(prof.getUserType());
171     int classify = 0;
172     if(prof.getUserType().equalsIgnoreCase("
    Municipal/City Health Official")) {
173         model.addAttribute("pageVisit", "mofficial");
174         classify = 2;
175         model.addAttribute("allUsers", userProfileDAO
    .listAllUsers(classify));
176         model.addAttribute("brgy", mapsDAO.
    getAllBarangay(prof.getMuncity(), prof.
    getProvince(), prof.getRegion());
177     }
178     else if(prof.getUserType().equalsIgnoreCase("
    Provincial Health Official")) {
179         model.addAttribute("pageVisit", "pofficial");
180         classify = 4;
181         model.addAttribute("allUsers", userProfileDAO
    .listAllUsers(classify));
182         List<List<Place>> prov = new ArrayList<List<
    Place>>();
183         List<List<Place>> mun = new ArrayList<List<
    Place>>();
184         List<List<Place>> brg = new ArrayList<List<
    Place>>();
185         int count = 0, count2 = 0;
186         String theRegion = "";
187         for(int i = 0; i < regions.length; i++) {
188             if(regions[i].equalsIgnoreCase(prof.
    getRegion()))
189                 theRegion = prof.getRegion();
190             else
191                 theRegion = "";
192             prov.add(i, mapsDAO.getAllProvince(theRegion
    ));
193             for(Place p : prov.get(i)) {
194                 mun.add(count, mapsDAO.
    getAllMuncityFromProv(p.getProvince(),
    theRegion));
195                 for(Place m : mun.get(count)) {
196                     brg.add(count2, mapsDAO.getAllBarangay(m.
    getMuncity(), p.getProvince(), theRegion));
197                     count2++;
198                 }
199                 count++;
200             }
201         }
202         model.addAttribute("prov", prov);
203         model.addAttribute("mun", mun);
204         model.addAttribute("brgy", brg);
205     }
206     else if(prof.getUserType().equalsIgnoreCase("
    National Health Official")) {
207         model.addAttribute("pageVisit", "nofficial");
208         classify = 5;
209         model.addAttribute("allUsers", userProfileDAO
    .listAllUsers(classify));
210         List<List<Place>> prov = new ArrayList<List<
    Place>>();
211         List<List<Place>> mun = new ArrayList<List<
    Place>>();
212         List<List<Place>> brg = new ArrayList<List<
    Place>>();
213         int count = 0, count2 = 0;
214         for(int i = 0; i < regions.length; i++) {
215             prov.add(i, mapsDAO.getAllProvince(regions[i
    ]));
216             for(Place p : prov.get(i)) {
217                 mun.add(count, mapsDAO.
    getAllMuncityFromProv(p.getProvince(),
    regions[i]));
218                 count++;
219             }
220         }
221         model.addAttribute("prov", prov);
222         model.addAttribute("mun", mun);
223         model.addAttribute("brgy", brg);
224     }
225     else if(prof.getUserType().equalsIgnoreCase("
    Administrator")) {
226         model.addAttribute("pageVisit", "admin");
227         classify = 7;
228         model.addAttribute("allUsers", userProfileDAO
    .listAllUsers(classify));
229         List<List<Place>> prov = new ArrayList<List<
    Place>>();
230         List<List<Place>> mun = new ArrayList<List<
    Place>>();
231         List<List<Place>> brg = new ArrayList<List<
    Place>>();
232         int count = 0, count2 = 0;
233         for(int i = 0; i < regions.length; i++) {
234             prov.add(i, mapsDAO.getAllProvince(regions[i
    ]));
235             for(Place p : prov.get(i)) {
236                 mun.add(count, mapsDAO.
    getAllMuncityFromProv(p.getProvince(),
    regions[i]));
237                 count++;
238             }
239         }
240         model.addAttribute("prov", prov);
241         model.addAttribute("mun", mun);
242     }
243     if(prof.getUserType().equalsIgnoreCase("
    Municipal/City Health Official")) {
244         classify = 2;
245         model.addAttribute("registrations",
    registrationDAO.listAllRegistrations(
    classify));
246     }
247     else if(prof.getUserType().equalsIgnoreCase("
    Provincial Health Official")) {
248         classify = 4;
249         model.addAttribute("registrations",
    registrationDAO.listAllRegistrations(
    classify));
250     }
251     else if(prof.getUserType().equalsIgnoreCase("
    National Health Official")) {
252         classify = 5;
253         model.addAttribute("registrations",
    registrationDAO.listAllRegistrations(
    classify));
254     }
255     else if(prof.getUserType().equalsIgnoreCase("
    Administrator")) {
256         classify = 7;
257         model.addAttribute("registrations",
    registrationDAO.listAllRegistrations(
    classify));
258     }
259     if(prof.getUserTypeInt() == 2 || prof.
    getUserTypeInt() == 4 || prof.
    getUserTypeInt() == 5 || prof.
    getUserTypeInt() == 7) {
260         model.addAttribute("pending", registrationDAO
    .countPending(classify));
261         model.addAttribute("approved",
    registrationDAO.countApproved(classify));
262         model.addAttribute("rejected",
    registrationDAO.countRejected(classify));
263     }
264     model.addAttribute("regions", regions);
265     model.addAttribute("alerts", diseasesDAO.
    getAlerts());
266     model.addAttribute("numAlerts", diseasesDAO.
    getCountAlerts());
267     return new ModelAndView("pages/notifications")
    ;
268 }
269
270 @RequestMapping(value = "/pages/addUser/{regID
    }", method = {RequestMethod.GET,

```

```

RequestMethod.POST})
276 public ModelAndView addUser(@PathVariable("
    regID") int regID, ModelMap model) {
277
278     if(alertCheck == 0) {
279         alertCheck = 1;
280         diseasesDAO.updateAlerts();
281     }
282
283     Authentication auth = SecurityContextHolder.
        getContext()
284         .getAuthentication();
285     String name = auth.getName();
286     UserProfile prof = userProfileDAO.getUser(name
        );
287     model.addAttribute("disease", diseasesDAO.
        listDiseases());
288     model.addAttribute("thePage", "notifications")
        ;
289
290     String yow = newUserDAO.getUser(regID);
291
292     int classify = 0;
293     if(prof.getUserType().equalsIgnoreCase("
        Municipal/City Health Official")) {
294         classify = 2;
295         model.addAttribute("registrations",
            registrationDAO.listAllRegistrations(
                classify));
296     }
297     else if(prof.getUserType().equalsIgnoreCase("
        Provincial Health Official")) {
298         classify = 4;
299         model.addAttribute("registrations",
            registrationDAO.listAllRegistrations(
                classify));
300     }
301     else if(prof.getUserType().equalsIgnoreCase("
        National Health Official")) {
302         classify = 5;
303         model.addAttribute("registrations",
            registrationDAO.listAllRegistrations(
                classify));
304     }
305     else if(prof.getUserType().equalsIgnoreCase("
        Administrator")) {
306         classify = 7;
307         model.addAttribute("registrations",
            registrationDAO.listAllRegistrations(
                classify));
308     }
309
310     if(prof.getUserTypeInt() == 2 || prof.
        getUserTypeInt() == 4 || prof.
        getUserTypeInt() == 6 || prof.
        getUserTypeInt() == 7) {
311         model.addAttribute("pending", registrationDAO
            .countPending(classify));
312         model.addAttribute("approved",
            registrationDAO.countApproved(classify));
313         model.addAttribute("rejected",
            registrationDAO.countRejected(classify));
314     }
315
316     if (!yow.isEmpty()) {
317         UserChange user = newUserDAO.getAccount(regID
            );
318         UserProfile prof2 = userProfileDAO.getUser(
            regID);
319         SendEmail sends = new SendEmail();
320         sends.sendmail(prof2.getEmail(), prof2.
            getUsername(), yow, prof2.getUserType(),
            true);
321
322         model.addAttribute("success", "Adding New
            User Successful!");
323         return new ModelAndView("pages/notifications
            ");
324     }
325
326     return null;
327 }
328
329 @RequestMapping(value = "/pages/rejectUser/{
    regID}", method = {RequestMethod.GET,
    RequestMethod.POST})
330 public ModelAndView rejectUser(@PathVariable("
    regID") int regID, ModelMap model) {
331     if(alertCheck == 0) {
332         alertCheck = 1;
333         diseasesDAO.updateAlerts();
334     }
335     Authentication auth = SecurityContextHolder.
        getContext()
336         .getAuthentication();
337     String name = auth.getName();
338     UserProfile prof = userProfileDAO.getUser(name
        );
339     model.addAttribute("disease", diseasesDAO.
        listDiseases());
340
341     int classify = 0;
342     if(prof.getUserType().equalsIgnoreCase("
        Municipal/City Health Official")) {
343         classify = 2;
344         model.addAttribute("registrations",
            registrationDAO.listAllRegistrations(
                classify));
345     }
346     else if(prof.getUserType().equalsIgnoreCase("
        Provincial Health Official")) {
347         classify = 4;
348         model.addAttribute("registrations",
            registrationDAO.listAllRegistrations(
                classify));
349     }
350     else if(prof.getUserType().equalsIgnoreCase("
        National Health Official")) {
351         classify = 5;
352         model.addAttribute("registrations",
            registrationDAO.listAllRegistrations(
                classify));
353     }
354     else if(prof.getUserType().equalsIgnoreCase("
        Administrator")) {
355         classify = 7;
356         model.addAttribute("registrations",
            registrationDAO.listAllRegistrations(
                classify));
357     }
358
359     if(prof.getUserTypeInt() == 2 || prof.
        getUserTypeInt() == 4 || prof.
        getUserTypeInt() == 6 || prof.
        getUserTypeInt() == 7) {
360         model.addAttribute("pending", registrationDAO
            .countPending(classify));
361         model.addAttribute("approved",
            registrationDAO.countApproved(classify));
362         model.addAttribute("rejected",
            registrationDAO.countRejected(classify));
363     }
364     newUserDAO.rejectUser(regID);
365     UserProfile prof2 = userProfileDAO.
        getRejectedUser(regID);
366     SendEmail sends = new SendEmail();
367     sends.sendmail(prof2.getEmail(), "", "", prof2
        .getUserType(), false);
368
369     System.out.println("Success!!!");
370     model.addAttribute("success", "Rejecting User
        Successful!");
371     return new ModelAndView("pages/notifications")
        ;
372 }
373
374 @RequestMapping(value = "/pages/reports",
    method = {RequestMethod.GET, RequestMethod.
    POST})
375 public ModelAndView toReports(ModelMap model) {
376     if(alertCheck == 0) {
377         alertCheck = 1;
378         diseasesDAO.updateAlerts();
379     }
380     model.addAttribute("disease", diseasesDAO.
        listDiseases());
381     model.addAttribute("thePage", "reports");
382     model.put("forReports", new Reports());
383     model.put("forExcel", new Reports());
384     return new ModelAndView("pages/reports");
385 }
386
387 @RequestMapping(value = "/pages/system", method
    = {RequestMethod.GET, RequestMethod.POST})
388 public ModelAndView toSystem(ModelMap model)
    throws Exception {
389
390     Authentication auth = SecurityContextHolder.
        getContext()
391         .getAuthentication();
392     String name = auth.getName();
393     System.out.println(name);
394     UserProfile prof = userProfileDAO.getUser(name
        );
395
396     model.addAttribute("disease", diseasesDAO.
        listDiseases());
397     model.addAttribute("user", prof);
398     model.addAttribute("userDetails",
        userProfileDAO.getUserDetails(name));
399     model.addAttribute("thePage", "system");
400     model.addAttribute("patientForm", new
        PatientsForm());
401
402     //if(prof.getUserTypeInt() == 0) {
403
404

```

```

405 List<Place> muncies = mapsDAO.
    getAllMuncityFromProv(prof.getProvince(),
    prof.getRegion());
406
407 List<List<Place>> brgys = new ArrayList<List<
    Place>>(muncies.size());
408
409 for(int i = 0; i < muncies.size(); i++) {
410     brgys.add(i, mapsDAO.getAllBarangay(muncies.
    get(i).getMuncity(), prof.getProvince(),
    prof.getRegion()));
411 }
412 model.addAttribute("munci", muncies);
413 model.addAttribute("brgys", brgys);
414 //}
415
416 if(prof.getUserType().equalsIgnoreCase("
    Municipal/City Health Worker"))
417     model.addAttribute("allBRecords",
    brgyRecordsDAO.listOfAllBrgyRecords(prof.
    getMuncity(),
    prof.getProvince(), prof.getRegion());
418 }
419 else if(prof.getUserType().equalsIgnoreCase("
    Provincial Health Worker"))
420     model.addAttribute("allMRecords",
    brgyRecordsDAO.listOfAllMunCityRecords(
    userProfileDAO.getUser(name).getProvince(),
    prof.getRegion());
421
422 return new ModelAndView("pages/system");
423 }
424 }
425
426 @RequestMapping(value = "/pages/system/
    toInsertRecord", method = {RequestMethod.
    GET, RequestMethod.POST})
427 public ModelAndView insertNewRec(
    @ModelAttribute("patientForm") @Valid
    PatientsForm p,
    BindingResult results, ModelMap model) {
428
429     model.addAttribute("disease", diseasesDAO.
    listDiseases());
430
431     Authentication auth = SecurityContextHolder.
    getContext()
432     .getAuthentication();
433     String name = auth.getName();
434     UserProfile prof = userProfileDAO.getUser(name
    );
435
436     model.addAttribute("disease", diseasesDAO.
    listDiseases());
437     model.addAttribute("user", prof);
438     model.addAttribute("userDetails",
    userProfileDAO.getUserDetails(name));
439     model.addAttribute("thePage", "system");
440
441     List<Place> muncies = mapsDAO.
    getAllMuncityFromProv(prof.getProvince(),
    prof.getRegion());
442
443     List<List<Place>> brgys = new ArrayList<List<
    Place>>(muncies.size());
444
445     for(int i = 0; i < muncies.size(); i++) {
446         brgys.add(i, mapsDAO.getAllBarangay(muncies.
    get(i).getMuncity(), prof.getProvince(),
    prof.getRegion()));
447     }
448     model.addAttribute("munci", muncies);
449     model.addAttribute("brgys", brgys);
450
451     boolean withErr = false;
452
453     if(p.getDisease().equalsIgnoreCase("Dengue")
    && p.getCaseType().isEmpty()) {
454         model.addAttribute("eCaseType", "Case Type is
    required.");
455         withErr = true;
456     }
457
458     if(p.getAdmitted().equalsIgnoreCase("Yes")) {
459         if(p.getDateAdmitted().isEmpty()) {
460             model.addAttribute("eDateAdmitted", "Date
    Admitted is required.");
461             withErr = true;
462         }
463         if(p.getHospital().isEmpty()) {
464             model.addAttribute("eHospital", "Hospital/
    Clinic is required.");
465             withErr = true;
466         }
467         if(p.getHospitalAddr().isEmpty()) {
468             model.addAttribute("eHospitalAddr", "
    Hospital Address is required.");
469             withErr = true;
470         }
471         if(p.getHospitalType().isEmpty()) {
472             model.addAttribute("eHospitalType", "
    Hospital Type is required.");
473
474             withErr = true;
475         }
476     }
477     if(p.getOutcome().equalsIgnoreCase("Died")) {
478         if(p.getDateDied().isEmpty()) {
479             model.addAttribute("eDateDied", "Date Died
    is required.");
480             withErr = true;
481         }
482     }
483     if (results.hasErrors() || withErr) {
484         model.addAttribute("disease", diseasesDAO.
    listDiseases());
485         System.out.println("With Errors");
486         return new ModelAndView("pages/systems/
    recordInsertion");
487     } else {
488         brgyRecordsDAO.insertBrgyRecords(p, prof,
    name);
489         model.addAttribute("disease", diseasesDAO.
    listDiseases());
490         model.addAttribute("success", "Adding record
    successful!");
491         model.addAttribute("pages", "system");
492         model.put("patientForm", new PatientsForm());
493         return new ModelAndView("pages/system");
494     }
495 }
496 }
497
498 @RequestMapping(value = "/pages/system/
    addToMunCityRecords/{recID}", method = {
    RequestMethod.GET, RequestMethod.POST})
499 public ModelAndView addRecordMunCity(
    @PathVariable("recID") int recID, ModelMap
    model) {
500
501     model.addAttribute("disease", diseasesDAO.
    listDiseases());
502
503     Authentication auth = SecurityContextHolder.
    getContext()
504     .getAuthentication();
505     String names = auth.getName();
506     UserProfile prof = userProfileDAO.getUser(
    names);
507
508     model.addAttribute("disease", diseasesDAO.
    listDiseases());
509     model.addAttribute("user", prof);
510     model.addAttribute("userDetails",
    userProfileDAO.getUserDetails(names));
511     model.addAttribute("thePage", "system");
512
513     List<Place> muncies = mapsDAO.
    getAllMuncityFromProv(prof.getProvince(),
    prof.getRegion());
514
515     List<List<Place>> brgys = new ArrayList<List<
    Place>>(muncies.size());
516
517     for(int i = 0; i < muncies.size(); i++) {
518         brgys.add(i, mapsDAO.getAllBarangay(muncies.
    get(i).getMuncity(), prof.getProvince(),
    prof.getRegion()));
519     }
520     model.addAttribute("munci", muncies);
521     model.addAttribute("brgys", brgys);
522
523     if (brgyRecordsDAO.insertMunCityRecords(recID,
    prof.getFirstname() + " " + prof.
    getMiddlename() + " " + prof.getSurname(),
    names) {
524         model.addAttribute("disease", diseasesDAO.
    listDiseases());
525         model.put("success", "Adding barangay patient
    record successful!");
526         model.addAttribute("allBRecords",
    brgyRecordsDAO.listOfAllBrgyRecords(prof.
    getMuncity(),
    prof.getProvince(), prof.getRegion());
527     }
528     return new ModelAndView("pages/system");
529 }
530
531 @RequestMapping(value = "/pages/system/
    addToProvRecords/{recID}", method = {
    RequestMethod.GET, RequestMethod.POST})
532 public ModelAndView addRecordProvincial(
    @PathVariable("recID") int recID, ModelMap
    model) throws Exception {
533
534     model.addAttribute("disease", diseasesDAO.
    listDiseases());
535
536     Authentication auth = SecurityContextHolder.

```

```

540         getContext();
541         .getAuthentication();
542         String names = auth.getName();
543         UserProfile prof = userProfileDAO.getUser(
544             names);
545
546     model.addAttribute("disease", diseasesDAO.
547         listDiseases());
548     model.addAttribute("user", prof);
549     model.addAttribute("userDetails",
550         userProfileDAO.getUserDetails(names));
551     model.addAttribute("thePage", "system");
552
553     List<Place> muncies = mapsDAO.
554         getAllMuncityFromProv(prof.getProvince(),
555         prof.getRegion());
556     List<List<Place>> brgys = new ArrayList<List<
557         Place>>(muncies.size());
558
559     for(int i = 0; i < muncies.size(); i++) {
560         brgys.add(i, mapsDAO.getAllBarangay(muncies.
561             get(i).getMuncity(), prof.getProvince(),
562             prof.getRegion()));
563     }
564     model.addAttribute("munci", muncies);
565     model.addAttribute("brgys", brgys);
566
567     if (brgyRecordsDAO.insertProvRecords(recID,
568         prof.getFirstname() + " " + prof.
569         getMiddlename() + " " + prof.getSurname(),
570         names)) {
571         model.addAttribute("disease", diseasesDAO.
572             listDiseases());
573         model.addAttribute("success", "Adding
574             municipality/city patient record successful
575             !");
576         model.addAttribute("allMRecords",
577             brgyRecordsDAO.listOfAllMunCityRecords(prof.
578                 getProvince(),
579                 prof.getRegion());
580     }
581     PatientsForm pF = brgyRecordsDAO.getRecord(
582         recID);
583     checkOutbreak(pF.getDisease(), pF.getMuncity(),
584         prof.getProvince(), prof.getRegion());
585
586     return new ModelAndView("pages/system");
587 }
588
589 @RequestMapping(value = "/pages/system/
590     rejectMunCityRecords/{recID}", method = {
591     RequestMethod.GET, RequestMethod.POST})
592 public ModelAndView rejMunCity(@PathVariable("
593     recID") int recID, ModelMap model) {
594     model.addAttribute("disease", diseasesDAO.
595         listDiseases());
596
597     Authentication auth = SecurityContextHolder.
598         getContext()
599         .getAuthentication();
600     String names = auth.getName();
601     UserProfile prof = userProfileDAO.getUser(
602         names);
603
604     model.addAttribute("disease", diseasesDAO.
605         listDiseases());
606     model.addAttribute("user", prof);
607     model.addAttribute("userDetails",
608         userProfileDAO.getUserDetails(names));
609     model.addAttribute("thePage", "system");
610     model.addAttribute("rejected", "yes");
611
612     List<Place> muncies = mapsDAO.
613         getAllMuncityFromProv(prof.getProvince(),
614         prof.getRegion());
615     List<List<Place>> brgys = new ArrayList<List<
616         Place>>(muncies.size());
617
618     for(int i = 0; i < muncies.size(); i++) {
619         brgys.add(i, mapsDAO.getAllBarangay(muncies.
620             get(i).getMuncity(), prof.getProvince(),
621             prof.getRegion()));
622     }
623     model.addAttribute("munci", muncies);
624     model.addAttribute("brgys", brgys);
625
626     if (brgyRecordsDAO.rejectProvRecords(recID,
627         prof.getFirstname() + " " + prof.
628         getMiddlename() + " " + prof.getSurname(),
629         names)) {
630         model.addAttribute("disease", diseasesDAO.
631             listDiseases());
632         model.addAttribute("rej", "Record
633             successfully rejected!");
634         model.addAttribute("allMRecords",
635             brgyRecordsDAO.listOfAllMunCityRecords(
636                 userProfileDAO.getUser(names).getProvince()
637                 ,
638                 prof.getRegion());
639     }
640     return new ModelAndView("pages/system");
641 }
642
643 @RequestMapping(value = "/pages/system/
644     editRecord/{recID}", method = {
645     RequestMethod.POST, RequestMethod.GET})
646 public ModelAndView toEditRecord(@PathVariable
647     ("recID") int recID, ModelMap model) {
648     Authentication auth = SecurityContextHolder.
649         getContext()
650         .getAuthentication();
651     String name = auth.getName();
652     System.out.println(name);
653     UserProfile prof = userProfileDAO.getUser(name
654         );
655     model.addAttribute("disease", diseasesDAO.
656         listDiseases());
657     model.addAttribute("user", prof);
658     model.addAttribute("userDetails",
659         userProfileDAO.getUserDetails(name));
660     model.addAttribute("thePage", "system");
661
662     if (prof.getUserType().equalsIgnoreCase("
663         Municipal/City Health Worker"))
664         model.addAttribute("patientForm",
665             brgyRecordsDAO.getRecord(recID));
666     else if (prof.getUserType().equalsIgnoreCase("
667         Provincial Health Worker"))
668         model.addAttribute("patientForm",
669             brgyRecordsDAO.getMunCityRecord(recID));
670
671     List<Place> muncies = mapsDAO.
672         getAllMuncityFromProv(prof.getProvince(),
673         prof.getRegion());

```



```

664 List<List<Place>> brgys = new ArrayList<List<
Place>>(muncies.size());
665
666 for(int i = 0; i < muncies.size(); i++) {
667     brgys.add(i, mapsDAO.getAllBarangay(muncies.
get(i).getMuncity(), prof.getProvince(),
prof.getRegion()));
668 }
669     model.addAttribute("munci", muncies);
670     model.addAttribute("brgys", brgys);
671
672     return new ModelAndView("pages/systems/
editingRecords");
673 }
674
675 @RequestMapping(value = "/pages/system/
toEditRecord", method = {RequestMethod.GET,
RequestMethod.POST})
676 public ModelAndView editNewRec(@ModelAttribute
("patientForm") @Valid PatientsForm p,
BindingResult results, ModelMap model) {
677
678     model.addAttribute("disease", diseasesDAO.
listDiseases());
679
680     Authentication auth = SecurityContextHolder.
getContext()
681     .getAuthentication();
682     String name = auth.getName();
683     UserProfile prof = userProfileDAO.getUser(name
);
684
685     model.addAttribute("disease", diseasesDAO.
listDiseases());
686     model.addAttribute("user", prof);
687     model.addAttribute("userDetails",
userProfileDAO.getUserDetails(name));
688     model.addAttribute("thePage", "system");
689
690     List<Place> muncies = mapsDAO.
getAllMuncityFromProv(prof.getProvince(),
prof.getRegion());
691     List<List<Place>> brgys = new ArrayList<List<
Place>>(muncies.size());
692
693     for(int i = 0; i < muncies.size(); i++) {
694         brgys.add(i, mapsDAO.getAllBarangay(muncies.
get(i).getMuncity(), prof.getProvince(),
prof.getRegion()));
695     }
696     model.addAttribute("munci", muncies);
697     model.addAttribute("brgys", brgys);
698
699     boolean withErr = false;
700
701     if(p.getDisease().equalsIgnoreCase("Dengue")
&& p.getCaseType().isEmpty()) {
702         model.addAttribute("eCaseType", "Case Type is
required.");
703         withErr = true;
704     }
705
706     if(p.getAdmitted().equalsIgnoreCase("Yes")) {
707         if(p.getDateAdmitted().isEmpty()) {
708             model.addAttribute("eDateAdmitted", "Date
Admitted is required.");
709             withErr = true;
710         }
711         if(p.getHospital().isEmpty()) {
712             model.addAttribute("eHospital", "Hospital/
Clinic is required.");
713             withErr = true;
714         }
715         if(p.getHospitalAddr().isEmpty()) {
716             model.addAttribute("eHospitalAddr", "
Hospital Address is required.");
717             withErr = true;
718         }
719         if(p.getHospitalType().isEmpty()) {
720             model.addAttribute("eHospitalType", "
Hospital Type is required.");
721             withErr = true;
722         }
723     }
724     if(p.getOutcome().equalsIgnoreCase("Died")) {
725         if(p.getDateDied().isEmpty()) {
726             model.addAttribute("eDateDied", "Date Died
is required.");
727             withErr = true;
728         }
729     }
730     if (results.hasErrors() || withErr) {
731
732         model.addAttribute("disease", diseasesDAO.
listDiseases());
733         System.out.println("With Errors");
734         return new ModelAndView("pages/systems/
editingRecords");
735     }
736 } else {
737
738     model.addAttribute("disease", diseasesDAO.
listDiseases());
739     model.addAttribute("success", "Editing record
successful!");
740     model.addAttribute("pages", "system");
741     if(prof.getUserType().equalsIgnoreCase("
Municipal/City Health Worker")) {
742         brgyRecordsDAO.insertEditBrgyRecords(p, prof
, name);
743         model.addAttribute("allBRecords",
brgyRecordsDAO.listOfAllBrgyRecords(prof.
getMuncity(),
prof.getProvince(), prof.getRegion()));
744     }
745     else if(prof.getUserType().equalsIgnoreCase("
Provincial Health Worker")) {
746         brgyRecordsDAO.insertEditMunCityRecords(p,
prof, name);
747         model.addAttribute("allMRecords",
brgyRecordsDAO.listOfAllMunCityRecords(prof
.getProvince(), prof.getRegion()));
748     }
749     return new ModelAndView("pages/system");
750 }
751 }
752 }
753 }
754 }
755
756 @RequestMapping(value = "/pages/
changeUserDetails", method = {RequestMethod
.POST, RequestMethod.GET})
757 public ModelAndView changeUserDetails(
@ModelAttribute("changeUser") @Valid
UserChange change, BindingResult result,
ModelMap model) {
758     if(alertCheck == 0) {
759         alertCheck = 1;
760         diseasesDAO.updateAlerts();
761     }
762     Authentication auth = SecurityContextHolder.
getContext()
763     .getAuthentication();
764     String name = auth.getName();
765     UserProfile prof = new UserProfile();
766     prof = userProfileDAO.getUser(name);
767
768     model.addAttribute("disease", diseasesDAO.
listDiseases());
769     model.addAttribute("user", userProfileDAO.
getUser(name));
770     model.addAttribute("userDetails",
userProfileDAO.getUserDetails(name));
771     model.addAttribute("thePage", "profile");
772
773     PasswordEncoder passwordEncoder = new
BCryptPasswordEncoder();
774
775     if (result.hasErrors() || !passwordEncoder.
matches(change.getAccountPassword(),
userProfileDAO.getUserDetails(name).
getAccountPassword()) ||
!change.getAccountPassword2().equals(change.
getAccountPassword3()) {
776         if(!passwordEncoder.matches(change.
getAccountPassword(),
userProfileDAO.getUserDetails(name).
getAccountPassword()))
777             model.addAttribute("error", "Old password is
incorrect!");
778         if(!change.getAccountPassword2().equals(
change.getAccountPassword3()))
779             model.addAttribute("error2", "New passwords
didn't match!");
780         change.setAccountPassword("");
781         change.setAccountPassword2("");
782         change.setAccountPassword3("");
783         model.addAttribute("pages", "profile");
784         return new ModelAndView("pages/errors/
changeCredentialsError");
785     } else {
786         newUserDAO.editUser(name, change);
787         return new ModelAndView("
changeCredentialsSuccessful");
788     }
789 }
790 }
791 }
792 }
793 }
794 }
795 }
796 }
797 }
798 }
799
800 @RequestMapping(value = "/pages/
changeUserContacts", method = {
RequestMethod.POST, RequestMethod.GET})
801 public ModelAndView changeUserContacts(
@ModelAttribute("changeContacts") @Valid
ContactsChange change, BindingResult result
,
ModelMap model) {
802     if(alertCheck == 0) {
803         alertCheck = 1;

```

```

804     diseasesDAO.updateAlerts();
805 }
806 Authentication auth = SecurityContextHolder.
    .getContext()
807     .getAuthentication();
808 String name = auth.getName();
809 UserProfile prof = new UserProfile();
810 prof = userProfileDAO.getUser(name);
811
812 model.addAttribute("disease", diseasesDAO.
    listDiseases());
813 model.addAttribute("user", userProfileDAO.
    getUser(name));
814 model.addAttribute("userDetails",
    userProfileDAO.getUserDetails(name));
815 model.addAttribute("thePage", "profile");
816
817
818 if (result.hasErrors()) {
819     model.addAttribute("pages", "profile");
820     return new ModelAndView("pages/errors/
        changeContactsError");
821
822 } else {
823
824     newUserDAO.editUserContacts(name, change.
        getEmail(), change.getNumber());
825     prof = userProfileDAO.getUser(name);
826     model.addAttribute("success", "Changing
        contact details successful!");
827     return new ModelAndView("pages/profile");
828 }
829 }
830
831 @RequestMapping(value = "/pages/generateReports
    ", method = {RequestMethod.GET,
    RequestMethod.POST})
832 public ModelAndView toGenerateReports(
    @ModelAttribute("forReports") @Valid
    Reports report, BindingResult result,
833     HttpServletResponse res, ModelMap model)
    throws DocumentException, IOException {
834
835     if(alertCheck == 0) {
836         alertCheck = 1;
837         diseasesDAO.updateAlerts();
838     }
839     model.addAttribute("disease", diseasesDAO.
        listDiseases());
840     System.out.println(report.getRegion());
841     if(result.hasErrors() || (report.
        isCaseAndDeath() && report.getRegion().
        isEmpty())) {
842         if(report.isCaseAndDeath() && report.
            getRegion().isEmpty())
843             model.addAttribute("error", "Please select
                region!");
844
845         for(int i = 0; i < result.getErrorCount(); i
            ++){
846             for(FieldError e : result.getFieldErrors())
847                 System.out.println(e.getDefaultMessage());
848         }
849     }
850     model.addAttribute("location", "reports");
851     System.out.println("Errors!");
852
853     return new ModelAndView("pages/reports");
854 }
855 else {
856
857     List<PatientsForm> patients = patientsDAO.
        listAllPatients(report.getDateFrom(),
        report.getDateTo(), report.getDisease());
858
859     for(PatientsForm pat : patients) {
860         System.out.println("One: " + pat.getAge());
861     }
862     Date date = new Date();
863     DateFormat dateFormat =
        new SimpleDateFormat("yyyy-MM-dd");
864     System.out.println((date.getYear() - 1)
        + "-01-01" + " " + (date.getYear() - 1) +
        "-12-31");
865     System.out.println(date.getYear() +
        "-01-01" + " " + date.getYear() +
        "-12-31");
866     String d1[] = dateFormat.format(date).
        split("-");
867     List<PatientsForm> ps1 = patientsDAO.
        listAllPatients((Integer.parseInt(d1[0]) -
        1) + "-01-01", (Integer.parseInt(d1[0]) -
        1) + "-12-31", report.getDisease());
868     List<PatientsForm> ps2 = patientsDAO.
        listAllPatients(Integer.parseInt(d1[0]) +
        "-01-01", Integer.parseInt(d1[0]) +
        "-12-31", report.getDisease());
869
870
871
872 GenerateReports rep = new GenerateReports();
873 List<Place> place = null;
874
875 if(report.isCaseAndDeath())
876     place = mapsDAO.getAllMuncity(report.
        getRegion());
877
878
879     ByteArrayOutputStream bits = rep.
        generateReports(patients, report, place,
        ps2, ps1);
880     String filename = "VBDMs reports.pdf";
881
882     res.setHeader("Cache-Control", "max-age=30");
883     res.setContentType("application/pdf");
884     res.setHeader("Content-disposition", "
        inline; filename=" + filename);
885
886     ServletOutputStream sos;
887     sos = res.getOutputStream();
888
889     try {
890         bits.writeTo(sos);
891     } catch (NullPointerException npe) {
892         npe.printStackTrace();
893     }
894     sos.flush();
895
896     Authentication auth =
        SecurityContextHolder.getContext()
897         .getAuthentication();
898     String name = auth.getName();
899
900     return null;
901 }
902 }
903
904 @RequestMapping(value = "/pages/generateExcel",
    method = {RequestMethod.GET, RequestMethod.
    .POST})
905 public ModelAndView toGenerateExcel(
    @ModelAttribute("forExcel") @Valid Reports
    report, BindingResult result,
906     HttpServletResponse res, ModelMap model)
    throws DocumentException, IOException {
907     if(alertCheck == 0) {
908         alertCheck = 1;
909         diseasesDAO.updateAlerts();
910     }
911     model.addAttribute("disease", diseasesDAO.
        listDiseases());
912
913     if(result.hasErrors() || (report.isSpecRegion
        () && report.getRegion().isEmpty())) {
914         if(report.isCaseAndDeath() && report.
            getRegion().isEmpty())
915             model.addAttribute("error", "Please select
                region!");
916         model.addAttribute("location", "reports");
917         model.addAttribute("reportsPage", "Excel");
918         System.out.println("Errors!");
919
920         return new ModelAndView("pages/reports");
921     }
922     else {
923         List<PatientsForm> patients = null;
924
925         if(report.isSpecRegion())
926             patients = patientsDAO.listAllPatients(
                report.getDateFrom(), report.getDateTo(),
                report.getRegion());
927         else
928             patients = patientsDAO.listAllPatients(
                report.getDateFrom(), report.getDateTo());
929
930         Date date = new Date();
931         DateFormat dateFormat =
            new SimpleDateFormat("yyyy-MM-dd");
932
933         GenerateExcel exc = new GenerateExcel();
934
935         ByteArrayOutputStream bits = exc.
            generateExcel(patients, report);
936
937         String filename = "VBDMs dataset.xls";
938
939         res.setHeader("Cache-Control", "max-age=30");
940         res.setContentType("application/xls");
941         res.setHeader("Content-disposition", "
            inline; filename=" + filename);
942
943         ServletOutputStream sos;
944         sos = res.getOutputStream();
945
946         try {
947             bits.writeTo(sos);
948         } catch (NullPointerException npe) {
949             npe.printStackTrace();
950         }

```

```

951     }
952     sos.flush();
953     return null;
954 }
955 }
956
957
958 @RequestMapping(value = "/pages/workers",
959     method = {RequestMethod.GET, RequestMethod.
960     POST})
961 public ModelAndView toWorkers(ModelMap model)
962     throws Exception {
963     if(alertCheck == 0) {
964         alertCheck = 1;
965         diseasesDAO.updateAlerts();
966     }
967     Authentication auth = SecurityContextHolder.
968     getContext()
969     .getAuthentication();
970     String name = auth.getName();
971     UserProfile prof = userProfileDAO.getUser(name
972     );
973     model.addAttribute("disease", diseasesDAO.
974     listDiseases());
975     model.addAttribute("user", prof);
976     model.addAttribute("userDetails",
977     userProfileDAO.getUserDetails(name));
978     model.addAttribute("thePage", "workers");
979     String[] regions = {"ARMM", "CAR", "NCR", "
980     Region I", "Region II", "Region III",
981     "Region IV-A", "Region IV-B", "Region V", "
982     Region VI", "Region VII",
983     "Region VIII", "Region IX", "Region X", "
984     Region XI", "Region XII", "Region XIII"};
985     int classify = 0;
986     if(prof.getUserType().equalsIgnoreCase("
987     Municipal/City Health Official")) {
988         model.addAttribute("pageVisit", "mofficial");
989         classify = 2;
990         model.addAttribute("allUsers", userProfileDAO
991         .listAllUsers(classify));
992         model.addAttribute("brgy", mapsDAO.
993         getAllBarangay(prof.getMuncity(), prof.
994         getProvince(), prof.getRegion()));
995         model.addAttribute("brgR", brgyRecordsDAO.
996         listOfAllBrgyRecords(prof.getMuncity(),
997         prof.getProvince(), prof.getRegion()));
998         model.addAttribute("munR", brgyRecordsDAO.
999         listOfAllMunCityRecords(prof.getProvince(),
1000         prof.getRegion()));
1001         model.addAttribute("editB", brgyRecordsDAO.
1002         listOfAllEditBrgyRecords(prof.getMuncity(),
1003         prof.getProvince(), prof.getRegion()));
1004     }
1005     else if(prof.getUserType().equalsIgnoreCase("
1006     Provincial Health Official")) {
1007         model.addAttribute("pageVisit", "pofficial");
1008         classify = 4;
1009         model.addAttribute("allUsers", userProfileDAO
1010         .listAllUsers(classify));
1011         List<List<Place>> prov = new ArrayList<List<
1012         Place>>();
1013         List<List<Place>> mun = new ArrayList<List<
1014         Place>>();
1015         List<List<Place>> brg = new ArrayList<List<
1016         Place>>();
1017         List<List<PatientsForm>> munR = new ArrayList
1018         <List<PatientsForm>>();
1019         List<List<PatientsForm>> provR = new
1020         ArrayList<List<PatientsForm>>();
1021         List<List<PatientsForm>> munR = new ArrayList
1022         <List<PatientsForm>>();
1023         List<List<PatientsForm>> provR = new
1024         ArrayList<List<PatientsForm>>();
1025         List<List<PatientsForm>> editB = new
1026         ArrayList<List<PatientsForm>>();
1027         int count = 0, count2 = 0;
1028         String theRegion = "";
1029         for(int i = 0; i < regions.length; i++) {
1030             if(regions[i].equalsIgnoreCase(prof.
1031             getRegion()))
1032                 theRegion = prof.getRegion();
1033             else
1034                 theRegion = "";
1035             prov.add(i, mapsDAO.getAllProvince(theRegion
1036             ));
1037             for(Place p : prov.get(i)) {
1038                 provR.add(count, brgyRecordsDAO.
1039                 listOfAllProvincialRecords(p.getProvince(),
1040                 theRegion));
1041                 mun.add(count, mapsDAO.
1042                 getAllMuncityFromProv(p.getProvince(),
1043                 theRegion));
1044                 for(Place m : mun.get(count)) {
1045                     editB.add(count2, brgyRecordsDAO.
1046                     listOfAllEditBrgyRecords(m.getMuncity(), p.
1047                     getProvince(), theRegion));
1048                     brg.add(count2, mapsDAO.getAllBarangay(m.
1049                     getMuncity(), p.getProvince(), theRegion));
1050                     brgR.add(count2, brgyRecordsDAO.
1051                     listOfAllBrgyRecords(m.getMuncity(), p.
1052                     listOfAllBrgyRecords(m.getMuncity(), p.
1053                     getProvince(), theRegion));
1054                     count2++;
1055                 }
1056             }
1057             munR.add(count, brgyRecordsDAO.
1058             listOfAllMunCityRecords(p.getProvince(),
1059             regions[i]));
1060             count++;
1061         }
1062         model.addAttribute("prov", prov);
1063         model.addAttribute("mun", mun);
1064         model.addAttribute("brgy", brg);
1065         model.addAttribute("brgR", munR);
1066         model.addAttribute("provR", munR);
1067         model.addAttribute("editB", editB);
1068     }
1069     else if(prof.getUserType().equalsIgnoreCase("
1070     National Health Official")) {
1071         model.addAttribute("pageVisit", "nofficial");
1072         classify = 5;
1073         model.addAttribute("allUsers", userProfileDAO
1074         .listAllUsers(classify));
1075         List<List<Place>> prov = new ArrayList<List<
1076         Place>>();
1077         for(int i = 0; i < regions.length; i++) {
1078             prov.add(i, mapsDAO.getAllProvince(regions[i
1079             ]));
1080             model.addAttribute("prov", prov);
1081             List<Place> mun = mapsDAO.
1082             getAllMuncityFromProv(prof.getProvince(),
1083             prof.getRegion());
1084             model.addAttribute("mun", mun);
1085             List<List<Place>> brg = new ArrayList<List<
1086             Place>>();
1087             int count = 0;
1088             for(Place m : mun) {
1089                 brg.add(count, mapsDAO.getAllBarangay(prof.
1090                 getMuncity(), prof.getProvince(), prof.
1091                 getRegion()));
1092                 count++;
1093             }
1094             model.addAttribute("brgy", brg);
1095         }
1096     }
1097     else if(prof.getUserType().equalsIgnoreCase("
1098     Administrator")) {
1099         model.addAttribute("pageVisit", "admin");
1100         classify = 7;
1101         model.addAttribute("allUsers", userProfileDAO
1102         .listAllUsers(classify));
1103         List<List<Place>> prov = new ArrayList<List<
1104         Place>>();
1105         List<List<Place>> mun = new ArrayList<List<
1106         Place>>();
1107         List<List<Place>> brg = new ArrayList<List<
1108         Place>>();
1109         List<List<PatientsForm>> munR = new ArrayList
1110         <List<PatientsForm>>();
1111         List<List<PatientsForm>> provR = new
1112         ArrayList<List<PatientsForm>>();
1113         List<List<PatientsForm>> editB = new
1114         ArrayList<List<PatientsForm>>();
1115         int count = 0, count2 = 0;
1116         for(int i = 0; i < regions.length; i++) {
1117             prov.add(i, mapsDAO.getAllProvince(regions[i
1118             ]));
1119             for(Place p : prov.get(i)) {
1120                 provR.add(count, brgyRecordsDAO.
1121                 listOfAllProvincialRecords(p.getProvince(),
1122                 regions[i]));
1123                 mun.add(count, mapsDAO.
1124                 getAllMuncityFromProv(p.getProvince(),
1125                 regions[i]));
1126                 for(Place m : mun.get(count)) {
1127                     editB.add(count2, brgyRecordsDAO.
1128                     listOfAllBrgyRecords(m.getMuncity(), p.
1129                     getProvince(), regions[i]));
1130                     count2++;
1131                 }
1132             }
1133             munR.add(count, brgyRecordsDAO.
1134             listOfAllMunCityRecords(p.getProvince(),
1135             regions[i]));
1136             count++;
1137         }
1138         model.addAttribute("prov", prov);
1139         model.addAttribute("mun", mun);
1140         model.addAttribute("brgy", brg);
1141         model.addAttribute("brgR", munR);
1142         model.addAttribute("provR", munR);
1143         model.addAttribute("editB", editB);
1144     }
1145 }

```

```

1081 model.addAttribute("regions", regions);          1140
1082 model.addAttribute("allLogs", userProfileDAO.    1141
    allLogs());
1083
1084 return new ModelAndView("pages/workers");        1142
1085 }
1086
1087 @RequestMapping(value = "/pages/deactivateUser    1143
    /{regID}", method = {RequestMethod.GET,
    RequestMethod.POST})
1088 public ModelAndView deactivateUser(              1144
    @PathVariable("regID") int regID, ModelMap    1145
    model) throws Exception {                    1146
1089
1090     if(alertCheck == 0) {                      1147
1091         alertCheck = 1;                       1148
1092         diseasesDAO.updateAlerts();           1149
1093     }                                          1150
1094     Authentication auth = SecurityContextHolder.  1151
    getContext()
1095     .getAuthentication();                    1153
1096     String name = auth.getName();             1154
1097     UserProfile prof = userProfileDAO.getUser(   1155
    name
    );                                          1156
1098     model.addAttribute("disease", diseasesDAO.  1157
    listDiseases());
1099     model.addAttribute("user", prof);          1159
1100     model.addAttribute("userDetails",          1160
    userProfileDAO.getUserDetails(name));
1101     model.addAttribute("thePage", "workers");  1161
1102
1103
1104     if(userProfileDAO.deactivateUser(regID,     1163
    name)
    {
1105         String[] regions = {"ARMM", "CAR", "NCR", "  1164
    Region I", "Region II", "Region III",
1106         "Region IV-A", "Region IV-B", "Region V", "  1165
    Region VI", "Region VII",
1107         "Region VIII", "Region IX", "Region X", "  1166
    Region XI", "Region XII", "Region XIII"};
1108         int classify = 0;                      1167
1109         if(prof.getUserType().equalsIgnoreCase("  1169
    Municipal/City Health Official")) {
1110             model.addAttribute("pageVisit", "mof  1170
    ficial")
            ;
1111             classify = 2;                      1171
1112             model.addAttribute("allUsers",      1172
    userProfileDAO.listAllUsers(classify));
1113             model.addAttribute("brgy", mapsDAO.  1173
    getAllBarangay(prof.getMuncity(), prof.
    getProvince(), prof.getRegion());
1114             model.addAttribute("brgR", brgyRecor  1174
    dsDAO.
    listOfAllBrgyRecords(prof.getMuncity(),
    prof.getProvince(), prof.getRegion());
1115             model.addAttribute("munR", brgyRecor  1177
    dsDAO.
    listOfAllMunCityRecords(prof.getProvince(),
    prof.getRegion());
1116             model.addAttribute("editB", brgyRecor  1180
    dsDAO.
    listOfAllEditBrgyRecords(prof.getMuncity(),
    prof.getProvince(), prof.getRegion());
1117         }
1118         else if(prof.getUserType().equalsIgnore  1183
    Case("
    Provincial Health Official")) {
1119             model.addAttribute("pageVisit", "pof  1184
    ficial")
            ;
1120             classify = 4;                      1185
1121             model.addAttribute("allUsers",      1186
    userProfileDAO.listAllUsers(classify));
1122             List<List<Place>> prov = new ArrayLi  1187
    st<List<
    Place>>();
1123             List<List<Place>> mun = new ArrayLi  1188
    st<List<
    Place>>();
1124             List<List<Place>> brg = new ArrayLi  1189
    st<List<
    Place>>();
1125             List<List<PatientsForm>> brgR = new  1190
    ArrayList<List<PatientsForm>>();
1126             List<List<PatientsForm>> munR = new  1191
    ArrayList<List<PatientsForm>>();
1127             List<List<PatientsForm>> provR = new  1192
    ArrayList<List<PatientsForm>>();
1128             List<List<PatientsForm>> editB = new  1193
    ArrayList<List<PatientsForm>>();
1129             int count = 0, count2 = 0;
1130             String theRegion = "";
1131             for(int i = 0; i < regions.length; i  1195
    ++){
1132                 if(regions[i].equalsIgnoreCase  1196
    (prof.
    getRegion()))
1133                 theRegion = prof.getRegion();    1197
1134             else
1135                 theRegion = "";
1136             prov.add(i, mapsDAO.getAllProvince(  1198
    theRegion));
1137             for(Place p : prov.get(i)) {
1138                 provR.add(count, brgyRecordsDA  1199
    O.
    listOfAllProvincialRecords(p.getProvince(),
    theRegion));
1139                 mun.add(count, mapsDAO.
    getAllMuncityFromProv(p.getProvince(),
    theRegion));
1140                 for(Place m : mun.get(count)) {
1141                     editB.add(count2, brgyRecor  1142
    dsDAO.
    listOfAllEditBrgyRecords(m.getMuncity(), p.
    getProvince(), theRegion));
1143                     brg.add(count2, mapsDAO.
    getAllBarangay(m.
    getMuncity(), p.getProvince(), theRegion));
1144                     brgR.add(count2, brgyRecor  1145
    dsDAO.
    listOfAllBrgyRecords(m.getMuncity(), p.
    getProvince(), theRegion));
1146                     count2++;
1147                 }
1148                 munR.add(count, brgyRecordsDA  1149
    O.
    listOfAllMunCityRecords(p.getProvince(),
    theRegion));
1150                 count++;
1151             }
1152             model.addAttribute("prov", prov);
1153             model.addAttribute("mun", mun);
1154             model.addAttribute("brgy", brg);
1155             model.addAttribute("brgR", brgR);
1156             model.addAttribute("munR", munR);
1157             model.addAttribute("provR", munR);
1158             model.addAttribute("editB", editB);
1159         }
1160         else if(prof.getUserType().equalsIgnore  1161
    Case("
    National Health Official")) {
1162             model.addAttribute("pageVisit", "no  1162
    ficial")
            ;
1163             classify = 5;
1164             model.addAttribute("allUsers",      1163
    userProfileDAO.listAllUsers(classify));
1165             List<List<Place>> prov = new ArrayLi  1164
    st<List<
    Place>>();
1166             for(int i = 0; i < regions.length; i  1165
    ++){
1167                 prov.add(i, mapsDAO.getAllProv  1166
    ince(regions[
    i]));
1168             }
1169             model.addAttribute("prov", prov);
1170             List<Place> mun = mapsDAO.
    getAllMuncityFromProv(prof.getProvince(),
    prof.getRegion());
1171             model.addAttribute("mun", mun);
1172             List<List<Place>> brg = new ArrayLi  1171
    st<List<
    Place>>();
1173             int count = 0;
1174             for(Place m : mun) {
1175                 brg.add(count, mapsDAO.
    getAllBarangay(prof.
    getMuncity(), prof.getProvince(), prof.
    getRegion());
1176                 count++;
1177             }
1178             model.addAttribute("brgy", brg);
1179         }
1180         else if(prof.getUserType().equalsIgnore  1180
    Case("
    Administrator")) {
1181             model.addAttribute("pageVisit", "ad  1181
    min")
            ;
1182             classify = 7;
1183             model.addAttribute("allUsers",      1182
    userProfileDAO.listAllUsers(classify));
1184             List<List<Place>> prov = new ArrayLi  1183
    st<List<
    Place>>();
1185             List<List<Place>> mun = new ArrayLi  1184
    st<List<
    Place>>();
1186             List<List<Place>> brg = new ArrayLi  1185
    st<List<
    Place>>();
1187             List<List<PatientsForm>> munR = new  1186
    ArrayList<List<PatientsForm>>();
1188             List<List<PatientsForm>> provR = new  1187
    ArrayList<List<PatientsForm>>();
1189             List<List<PatientsForm>> editB = new  1188
    ArrayList<List<PatientsForm>>();
1190             int count = 0, count2 = 0;
1191             for(int i = 0; i < regions.length; i  1189
    ++){
1192                 prov.add(i, mapsDAO.getAllProv  1190
    ince(regions[
    i]));
1193                 for(Place p : prov.get(i)) {
1194                     provR.add(count, brgyRecor  1191
    dsDAO.
    listOfAllProvincialRecords(p.getProvince(),
    regions[i]));
1195                     mun.add(count, mapsDAO.
    getAllMuncityFromProv(p.getProvince(),
    regions[i]));
1196                     for(Place m : mun.get(count  1192
    )) {
1197                         brg.add(count2, mapsDA  1193
    O.
    getAllBarangay(m.
    getMuncity(), p.getProvince(), regions[i]
    ));
1198                     ;
1199                     editB.add(count2, brgyRecor  1194
    dsDAO.
    listOfAllBrgyRecords(m.getMuncity(), p.
    getProvince(), regions[i]));
1200                     count2++;
1201                 }
1202                 munR.add(count, brgyRecordsDA  1195
    O.
    listOfAllMunCityRecords(p.getProvince(),
    regions[i]));
1203                 count++;
1204             }
1205         }
1206     }
1207 }

```

```

1203     }
1204
1205     }
1206     model.addAttribute("prov", prov);
1207     model.addAttribute("mun", mun);
1208     model.addAttribute("brgy", brg);
1209     model.addAttribute("munR", munR);
1210     model.addAttribute("provR", munR);
1211     model.addAttribute("editB", editB);
1212
1213 }
1214
1215 model.addAttribute("regions", regions);
1216 model.addAttribute("allLogs", userProfileDAO.
allLogs());
1217
1218 System.out.println("Deactivation Success!!!")
;
1219 model.addAttribute("successD", "Deactivating
User Successful!");
1220 return new ModelAndView("pages/workers");
1221
1222 }
1223 return null;
1224 }
1225
1226 @RequestMapping(value = "/pages/activateUser/{
regID}", method = {RequestMethod.GET,
RequestMethod.POST})
1227 public ModelAndView activateUser(@PathVariable
("regID") int regID, ModelMap model) throws
Exception {
1228
1229 if(alertCheck == 0) {
1230     alertCheck = 1;
1231     diseasesDAO.updateAlerts();
1232 }
1233 Authentication auth = SecurityContextHolder.
getContext()
1234     .getAuthentication();
1235 String name = auth.getName();
1236 UserProfile prof = userProfileDAO.getUser(name
);
1237 model.addAttribute("disease", diseasesDAO.
listDiseases());
1238 model.addAttribute("user", prof);
1239 model.addAttribute("userDetails",
userProfileDAO.getUserDetails(name));
1240 model.addAttribute("thePage", "workers");
1241
1242
1243 if(userProfileDAO.activateUser(regID, name)) {
1244
1245     String[] regions = {"ARMM", "CAR", "NCR", "
Region I", "Region II", "Region III",
1246     "Region IV-A", "Region IV-B", "Region V", "
Region VI", "Region VII",
1247     "Region VIII", "Region IX", "Region X", "
Region XI", "Region XII", "Region XIII"};
1248     int classify = 0;
1249     if(prof.getUserType().equalsIgnoreCase("
Municipal/City Health Official")) {
1250         model.addAttribute("pageVisit", "mofficial")
;
1251         classify = 2;
1252         model.addAttribute("allUsers",
userProfileDAO.listAllUsers(classify));
1253         model.addAttribute("brgy", mapsDAO.
getAllBarangay(prof.getMuncity(), prof.
getProvince(), prof.getRegion());
1254         model.addAttribute("brgR", brgyRecordsDAO.
listOfAllBrgyRecords(prof.getMuncity(),
prof.getProvince(), prof.getRegion());
1255         model.addAttribute("munR", brgyRecordsDAO.
listOfAllMunCityRecords(prof.getProvince(),
prof.getRegion());
1256         model.addAttribute("editB", brgyRecordsDAO.
listOfAllEditBrgyRecords(prof.getMuncity(),
prof.getProvince(), prof.getRegion());
1257     }
1258     else if(prof.getUserType().equalsIgnoreCase("
Provincial Health Official")) {
1259         model.addAttribute("pageVisit", "pofficial")
;
1260         classify = 4;
1261         model.addAttribute("allUsers",
userProfileDAO.listAllUsers(classify));
1262         List<List<Place>> prov = new ArrayList<List<
Place>>();
1263         List<List<Place>> mun = new ArrayList<List<
Place>>();
1264         List<List<Place>> brg = new ArrayList<List<
Place>>();
1265         List<List<PatientsForm>> brgR = new
ArrayList<List<PatientsForm>>();
1266         List<List<PatientsForm>> munR = new
ArrayList<List<PatientsForm>>();
1267         List<List<PatientsForm>> provR = new
ArrayList<List<PatientsForm>>();
1268
1269         List<List<PatientsForm>> editB = new
ArrayList<List<PatientsForm>>();
1270         int count = 0, count2 = 0;
1271         String theRegion = "";
1272         for(int i = 0; i < regions.length; i++) {
1273             if(regions[i].equalsIgnoreCase(prof.
getRegion()))
1274                 theRegion = prof.getRegion();
1275             else
1276                 theRegion = "";
1277             prov.add(i, mapsDAO.getAllProvince(
theRegion));
1278             for(Place p : prov.get(i)) {
1279                 provR.add(count, brgyRecordsDAO.
listOfAllProvincialRecords(p.getProvince(),
theRegion));
1280                 mun.add(count, mapsDAO.
getAllMuncityFromProv(p.getProvince(),
theRegion));
1281                 for(Place m : mun.get(count)) {
1282                     editB.add(count2, brgyRecordsDAO.
listOfAllEditBrgyRecords(m.getMuncity(), p.
getProvince(), theRegion));
1283                     brg.add(count2, mapsDAO.
getAllBarangay(m.
getMuncity(), p.getProvince(), theRegion));
1284                     brgR.add(count2, brgyRecordsDAO.
listOfAllBrgyRecords(m.getMuncity(), p.
getProvince(), theRegion));
1285                     count2++;
1286                 }
1287                 munR.add(count, brgyRecordsDAO.
listOfAllMunCityRecords(p.getProvince(),
theRegion));
1288                 count++;
1289             }
1290         }
1291         model.addAttribute("prov", prov);
1292         model.addAttribute("mun", mun);
1293         model.addAttribute("brgy", brg);
1294         model.addAttribute("brgR", brgR);
1295         model.addAttribute("munR", munR);
1296         model.addAttribute("provR", provR);
1297         model.addAttribute("editB", editB);
1298     }
1299     else if(prof.getUserType().equalsIgnoreCase("
National Health Official")) {
1300         model.addAttribute("pageVisit", "nofficial")
;
1301         classify = 5;
1302         model.addAttribute("allUsers",
userProfileDAO.listAllUsers(classify));
1303         List<List<Place>> prov = new ArrayList<List<
Place>>();
1304         List<List<Place>> mun = new ArrayList<List<
Place>>();
1305         List<List<Place>> brg = new ArrayList<List<
Place>>();
1306         List<List<PatientsForm>> munR = new
ArrayList<List<PatientsForm>>();
1307         List<List<PatientsForm>> provR = new
ArrayList<List<PatientsForm>>();
1308         List<List<PatientsForm>> editB = new
ArrayList<List<PatientsForm>>();
1309         int count = 0, count2 = 0;
1310         for(int i = 0; i < regions.length; i++) {
1311             prov.add(i, mapsDAO.getAllProvince(
regions[
i]));
1312             for(Place p : prov.get(i)) {
1313                 provR.add(count, brgyRecordsDAO.
listOfAllProvincialRecords(p.getProvince(),
regions[
i]));
1314                 mun.add(count, mapsDAO.
getAllMuncityFromProv(p.getProvince(),
regions[
i]));
1315                 for(Place m : mun.get(count)) {
1316                     editB.add(count2, brgyRecordsDAO.
listOfAllBrgyRecords(m.getMuncity(), p.
getProvince(), regions[
i]));
1317                     count2++;
1318                 }
1319                 munR.add(count, brgyRecordsDAO.
listOfAllMunCityRecords(p.getProvince(),
regions[
i]));
1320                 count++;
1321             }
1322         }
1323         model.addAttribute("prov", prov);
1324         model.addAttribute("mun", mun);
1325         model.addAttribute("brgy", brg);
1326         model.addAttribute("munR", munR);
1327         model.addAttribute("provR", provR);
1328         model.addAttribute("editB", editB);
1329     }
1330     else if(prof.getUserType().equalsIgnoreCase("
Administrator")) {
1331         model.addAttribute("pageVisit", "admin");
1332     }
1333 }

```

```

1334     classify = 7;                                1390
1335     model.addAttribute("allUsers",              1391     HashMap bar = new HashMap();
1336     List<List<Place>> prov = new ArrayList<List< 1392     HashMap bars = new HashMap();
1337     Place>>());                                  1393
1338     List<List<Place>> mun = new ArrayList<List< 1394     int cou = 0;
1339     Place>>());                                  1395     for(Place mp : mapsDAO.getAllBarangay(muncity,
1340     List<List<PatientsForm>> munR = new          1396     province, region)) {
1341     ArrayList<List<PatientsForm>>());             1397     bars.put(cou, mp.getBarangay());
1342     List<List<PatientsForm>> provR = new          1398     cou++;
1343     ArrayList<List<PatientsForm>>());             1399     }
1344     List<List<PatientsForm>> editB = new          1400     //bar.put(muncity, mapsDAO.getAllBarangay(
1345     ArrayList<List<PatientsForm>>());             1401     muncity, province, region));
1346     int count = 0, count2 = 0;                   1402     double[] cases = new double[bars.size()];
1347     for(int i = 0; i < regions.length; i++) {    1403     double[] oCases = new double[bars.size()];
1348     prov.add(i, mapsDAO.getAllProvince(regions[ 1404     int counterS = 0;
1349     i]));                                         1405     for(PatientsForm p : newPatients) {
1350     for(Place p : prov.get(i)) {                 1406     if(p.getDisease().equalsIgnoreCase(disease
1351     provR.add(count, brgyRecordsDAO.            1407     )) {
1352     listOfAllProvincialRecords(p.getProvince(), 1408     if(p.getRegion().equalsIgnoreCase(region))
1353     regions[i]);                                  1409     {
1354     mun.add(count, mapsDAO.                      1410     if(p.getProvince().equalsIgnoreCase(
1355     getAllMuncityFromProv(p.getProvince(),       1411     province)) {
1356     regions[i]));                                  1412     if(p.getMuncity().equalsIgnoreCase(
1357     for(Place m : mun.get(count)) {              1413     muncity)) {
1358     editB.add(count2, brgyRecordsDAO.           1414     for(int i = 0; i < bars.size(); i++) {
1359     listOfAllBrgyRecords(m.getMuncity(), p.     1415     System.out.println(p.getBarangay());
1360     getProvince(), regions[i]));                1416     System.out.println(bars.get(i).
1361     count2++;                                     1417     toString());
1362     munR.add(count, brgyRecordsDAO.             1418     if(p.getBarangay().equalsIgnoreCase(
1363     listOfAllMunCityRecords(p.getProvince(),     1419     bars.get(i).toString())
1364     regions[i]));                                  1420     cases[i]++;
1365     count++;                                       1421     }
1366     }                                             1422     }
1367     }                                             1423     }
1368     }                                             1424     }
1369     model.addAttribute("prov", prov);           1425     }
1370     model.addAttribute("mun", mun);             1426     }
1371     model.addAttribute("brgy", brg);            1427     }
1372     model.addAttribute("munR", munR);           1428     }
1373     model.addAttribute("provR", provR);         1429     }
1374     model.addAttribute("editB", editB);         1430     }
1375     }                                             1431     }
1376     }                                             1432     }
1377     model.addAttribute("regions", regions);     1433     }
1378     model.addAttribute("allLogs", userProfileDAO. 1434     }
1379     allLogs());                                  1435     }
1380     }                                             1436     }
1381     }                                             1437     }
1382     model.addAttribute("successD", "Activating  1438     String brg = "";
1383     User Successful!");                          1439     for(int i = 0; i < cases.length; i++) {
1384     return new ModelAndView("pages/workers");    1440     if(cases[i] > 0 && oCases[i] != 0) {
1385     }                                             1441     if(cases[i]/oCases[i] > 0.40) {
1386     }                                             1442     brg = bars.get(i).toString();
1387     }                                             1443     diseasesDAO.insertAlert(brg, muncity,
1388     }                                             1444     province, region, cases[i]/oCases[i],
1389     }                                             1445     disease);
1390     }                                             1446     SendSMS sms = new SendSMS();
1391     }                                             1447     List<UserProfile> allU = userProfileDAO.
1392     }                                             1448     allUsers(brg, muncity, province, region);
1393     }                                             1449     sms.msgsend(disease, brg, muncity,
1394     }                                             1450     province, allU);
1395     }                                             1451     }
1396     }                                             1452     }
1397     }                                             1453     }
1398     }                                             1454     }
1399     }                                             1455     }

```

3. Models and DAOs

Listing 6: BarangayRecordsDAO.java 8

```

1 package dao;
2
3 import java.sql.DriverManager;
4 import java.sql.SQLException;
5 import java.text.DateFormat;
6 import java.text.SimpleDateFormat;
7 import java.util.Date;
8 import java.util.List;
9
10 import mapper.BarangayRecordsMapper;
11 import mapper.DiseaseMapper;
12 import mapper.PatientsFormMapper;
13 import objects.BarangayRecords;
14 import objects.Diseases;
15 import objects.PatientsForm;
16 import objects.UserProfile;

```

```

17
18 import org.springframework.beans.factory.
    annotation.Autowired;
19 import org.springframework.jdbc.core.
    JdbcTemplate;
20 import org.springframework.jdbc.core.
    PreparedStatementCreator;
21 import org.springframework.jdbc.support.
    GeneratedKeyHolder;
22 import org.springframework.jdbc.support.
    KeyHolder;
23
24 import com.mysql.jdbc.Connection;
25 import com.mysql.jdbc.PreparedStatement;
26 import com.mysql.jdbc.Statement;
27
28 public class BarangayRecordsDAO {
29
30     @Autowired
31     private javax.sql.DataSource dataSource;
32     private JdbcTemplate jdbcTemplateObject;
33
34     public void insertBrgyRecords(final
        PatientsForm pat, final UserProfile prof,
        String username) {
35
36         jdbcTemplateObject = new JdbcTemplate(
            dataSource);
37
38         final DateFormat dateFormat = new
            SimpleDateFormat("yyyy-MM-dd");
39         final Date date = new Date();
40
41         DateFormat timeFormat = new SimpleDateFormat("
            HH:mm:ss");
42         Date time =new Date();
43
44
45         String SQL = "select accountID from users
            where username ='"+ username + "'";
46
47         final int accountID = jdbcTemplateObject.
            queryForInt(SQL);
48
49         int diseaseID = -1;
50
51         if(pat.getDisease().equals("Chikungunya"))
52             diseaseID = 0;
53         else if(pat.getDisease().equals("Dengue"))
54             diseaseID = 1;
55         else if(pat.getDisease().equals("Malaria"))
56             diseaseID = 2;
57
58         final int disID = diseaseID;
59         PatientsForm pats = pat;
60         if(pat.getDateAdmitted() == null)
61             pats.setDateAdmitted("0000-00-00");
62         if(pat.getDateDied() == null)
63             pats.setDateDied("0000-00-00");
64         if(pat.getHospital() == null)
65             pats.setHospital("");
66         if(pat.getHospitalType() == null)
67             pats.setHospitalType("");
68         if(pat.getHospitalAddr() == null)
69             pats.setHospitalAddr("");
70         if(pat.getCaseType() == null)
71             pats.setCaseType("");
72
73         final PatientsForm pat2 = pats;
74
75         final String SQL2 = "insert into
            barangayrecords (accountID, name, age,
            gender, birthDate, address, admitted,
            dateAdmitted, "
76         + "outcome, dateDied, hospital, hospitalType,
            hospitalAddr, region, province, muncity,
            barangay, "
77         + "diseaseID, symptoms, dateOnset,
            healthworker, muncityhealthworker,
            dateSubmitted) "
78         + "values (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?,
            ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)";
79
80         KeyHolder keyHolder = new GeneratedKeyHolder();
81         jdbcTemplateObject.update(
            new PreparedStatementCreator() {
82             @Override
83             public java.sql.PreparedStatement
                createPreparedStatement(
84                 java.sql.Connection connection) throws
                    SQLException {
85                 PreparedStatement pS = (PreparedStatement)
                    connection.prepareStatement(SQL2, Statement.
                        RETURN_GENERATED_KEYS);
86                 pS.setDouble(1, accountID);
87                 pS.setString(2, pat2.getName());
88                 pS.setDouble(3, pat2.getAge());
89                 pS.setString(4, pat2.getGender());
90                 pS.setString(5, pat2.getDateofbirth());
91                 pS.setString(6, pat2.getAddress());
92                 pS.setString(7, pat2.getAdmitted());
93                 pS.setString(8, pat2.getDateAdmitted());
94                 pS.setString(9, pat2.getOutcome());
95                 pS.setString(10, pat2.getDateDied());
96                 pS.setString(11, pat2.getHospital());
97                 pS.setString(12, pat2.getHospitalType());
98                 pS.setString(13, pat2.getHospitalAddr());
99                 pS.setString(14, pat2.getRegion());
100                pS.setString(15, pat2.getProvince());
101                pS.setString(16, pat2.getMuncity());
102                pS.setString(17, pat2.getBarangay());
103                pS.setDouble(18, disID);
104                pS.setString(19, pat2.getSymptoms());
105                pS.setString(20, pat2.getDateOnset());
106                pS.setString(21, prof.getFirstname() + " " +
                    prof.getMiddlename() + " " +
                    prof.getSurname());
107                pS.setString(22, "");
108                pS.setString(23, dateFormat.format(date));
109                return pS;
110            }
111        },
        keyHolder);
112
113         String action = "Inserted barangay record.";
114         SQL = "insert into userlogs (accountID, action,
            date, time, recordID, recordtable,
            recordtype) values (?, ?, ?, ?, ?, ?)";
115
116         jdbcTemplateObject.update(SQL, new Object[]{
            accountID, action, dateFormat.format(date),
            timeFormat.format(time), keyHolder.getKey
            ().intValue(), "barangayrecords", 1});
117
118     }
119
120     public void insertEditBrgyRecords(PatientsForm
        pat, UserProfile prof, String username) {
121
122         jdbcTemplateObject = new JdbcTemplate(
            dataSource);
123
124         DateFormat dateFormat = new SimpleDateFormat("
            yyyy-MM-dd");
125         Date date = new Date();
126
127         DateFormat timeFormat = new SimpleDateFormat("
            HH:mm:ss");
128         Date time =new Date();
129
130         String SQL = "select accountID from users where
            username ='"+ username + "'";
131
132         int accountID = jdbcTemplateObject.queryForInt(
            SQL);
133
134         int diseaseID = -1;
135
136         if(pat.getDisease().equals("Chikungunya"))
137             diseaseID = 0;
138         else if(pat.getDisease().equals("Dengue"))
139             diseaseID = 1;
140         else if(pat.getDisease().equals("Malaria"))
141             diseaseID = 2;
142
143         if(pat.getDateAdmitted() == null)
144             pat.setDateAdmitted("0000-00-00");
145         if(pat.getDateDied() == null)
146             pat.setDateDied("0000-00-00");
147         if(pat.getDateOnset() == null)
148             pat.setDateOnset("0000-00-00");
149         if(pat.getHospital() == null)
150             pat.setHospital("");
151         if(pat.getHospitalType() == null)
152             pat.setHospitalType("");
153         if(pat.getHospitalAddr() == null)
154             pat.setHospitalAddr("");
155         if(pat.getCaseType() == null)
156             pat.setCaseType("");
157
158         SQL = "SELECT * FROM barangayrecords WHERE
            recordID = '" + pat.getRecordID() + "'";
159         List<PatientsForm> pGS = jdbcTemplateObject.
            query(SQL, new PatientsFormMapper());
160
161         SQL = "DELETE FROM editbarangayrecords WHERE
            recordID = '" + pat.getRecordID() + "'";
162         jdbcTemplateObject.update(SQL);
163
164         SQL = "insert into editbarangayrecords (
            accountID, name, age, gender, birthDate,
            address, admitted, dateAdmitted, "
165         + "outcome, dateDied, hospital, hospitalType,
            hospitalAddr, region, province, muncity,
            barangay, "
166         + "diseaseID, dateOnset, symptoms,

```

```

healthworker, muncityhealthworker,
dateSubmitted, recordID) "
172 + "values (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)";
173
174 for(PatientsForm pG : pGS) {
175     if(pG.getDateAdmitted() == null)
176         pG.setDateAdmitted("0000-00-00");
177     if(pG.getDateDied() == null)
178         pG.setDateDied("0000-00-00");
179     if(pG.getDateOnset() == null)
180         pG.setDateOnset("0000-00-00");
181     jdbcTemplateObject.update(SQL, new Object[]{
182         accountID, pG.getName(), pG.getAge(), pG.
183         getGender(), pG.getDateofbirth(), pG.
184         getAddress(), pG.getAdmitted(), pG.
185         getDateAdmitted(), pG.getOutcome(), pG.
186         getDateDied(), pG.getHospital(),
187         pG.getHospitalType(), pG.getHospitalAddr(), pG.
188         .getRegion(), pG.getProvince(), pG.
189         getMuncity(), pG.getBarangay(),
190         diseaseID, pG.getDateOnset(), pG.getSymptoms()
191         , pG.getHealthworker(), "", pG.
192         getDateSubmitted(), pG.getRecordID());
193 }
194 SQL = "DELETE FROM barangayrecords WHERE
195 recordID = '" + pat.getRecordID() + "'";
196 jdbcTemplateObject.update(SQL);
197
198 SQL = "insert into barangayrecords (accountID,
199 name, age, gender, birthDate, address,
200 admitted, dateAdmitted, "
201 + "outcome, dateDied, hospital, hospitalType,
202 hospitalAddr, region, province, muncity,
203 barangay, "
204 + "diseaseID, dateOnset, symptoms, healthworker
205 , muncityhealthworker, dateSubmitted,
206 edittdBy, dateEdited, recordID, editStatus
207 ) "
208 + "values (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?,
209 ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)";
210
211 jdbcTemplateObject.update(SQL, new Object[]{
212     accountID, pat.getName(), pat.getAge(), pat.
213     .getGender(), pat.getDateofbirth(),
214     pat.getAddress(), pat.getAdmitted(), pat.
215     getDateAdmitted(), pat.getOutcome(), pat.
216     getDateDied(), pat.getHospital(),
217     pat.getHospitalType(), pat.getHospitalAddr(),
218     pat.getRegion(), pat.getProvince(), pat.
219     getMuncity(), pat.getBarangay(),
220     diseaseID, pat.getDateOnset(), pat.getSymptoms
221     (), pat.getHealthworker(), "", pat.
222     getDateSubmitted(),
223     prof.getFirstname() + " " + prof.getMiddlename
224     () + " " + prof.getSurname(), dateFormat.
225     format(date), pat.getRecordID(), 1});
226
227 String action = "Edited a barangay patient
228 record.";
229
230 SQL = "insert into userlogs (accountID, action,
231 date, time, recordID, editStatus,
232 recordtable, recordtype) values (?, ?, ?,
233 ?, ?, ?, ?, ?)";
234
235 jdbcTemplateObject.update(SQL, new Object[]{
236     accountID, action, dateFormat.format(date),
237     timeFormat.format(time), pat.getRecordID()
238     , 1, "editbarangayrecords", 0});
239
240 }
241
242 public boolean insertMunCityRecords(int id,
243 String name, String username) {
244     jdbcTemplateObject = new JdbcTemplate(
245         dataSource);
246
247     DateFormat dateFormat = new SimpleDateFormat("
248     yyyy-MM-dd");
249     Date date = new Date();
250
251     DateFormat timeFormat = new SimpleDateFormat("
252     HH:mm:ss");
253     Date time =new Date();
254
255     String SQL = "select accountID from users where
256     username ='" + username + "'";
257
258     int accountID = jdbcTemplateObject.queryForInt(
259     SQL);
260
261     int diseaseID = -1;
262
263     if(pat.getDisease().equals("Chikungunya"))
264         diseaseID = 0;
265     else if(pat.getDisease().equals("Dengue"))
266         diseaseID = 1;
267     else if(pat.getDisease().equals("Malaria"))
268         diseaseID = 2;
269
270     SQL = "select accountID from users where
271     username ='" + username + "'";
272
273     int accountID = jdbcTemplateObject.queryForInt(
274     SQL);
275
276     int diseaseID = -1;
277
278     if(pf.getDisease().equals("Chikungunya"))
279         diseaseID = 0;
280     else if(pf.getDisease().equals("Dengue"))
281         diseaseID = 1;
282     else if(pf.getDisease().equals("Malaria"))
283         diseaseID = 2;
284
285     SQL = "insert into muncityrecords (accountID,
286     name, age, gender, birthDate, address,
287     admitted, dateAdmitted, "
288     + "outcome, dateDied, hospital, hospitalType,
289     hospitalAddr, region, province, muncity,
290     barangay, "
291     + "diseaseID, symptoms, healthworker,
292     muncityhealthworker, dateSubmitted,
293     recordID, dateApproved, dateOnset) "
294     + "values (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?,
295     ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)";
296
297     jdbcTemplateObject.update(SQL, new Object[]{
298     accountID, pf.getName(), pf.getAge(), pf.
299     getGender(), pf.getDateofbirth(), pf.
300     getAddress(),
301     pf.getAdmitted(), pf.getDateAdmitted(), pf.
302     getOutcome(), pf.getDateDied(), pf.
303     getHospital(), pf.getHospitalType(),
304     pf.getHospitalAddr(), pf.getRegion(), pf.
305     getProvince(), pf.getMuncity(), pf.
306     getBarangay(), diseaseID, pf.getSymptoms(),
307     pf.getHealthworker(), name, pf.getDateSubmitted
308     (), pf.getRecordID(), dateFormat.format(
309     date), pf.getDateOnset());
310
311     SQL = "UPDATE barangayrecords SET status = ?,
312     dateApproved = ?, muncityhealthworker = ?
313     WHERE recordID = ?";
314     jdbcTemplateObject.update(SQL, 1, dateFormat.
315     format(date), name, id);
316
317     String action = "Approved a barangay patient
318     record. Inserted into municipal/city
319     patient record.";
320
321     SQL = "insert into userlogs (accountID, action,
322     date, time, recordID, recordtable,
323     recordtype) values (?, ?, ?, ?, ?, ?, ?)";
324     jdbcTemplateObject.update(SQL, new Object[]{
325     accountID, action, dateFormat.format(date),
326     timeFormat.format(time), pf.getRecordID(),
327     "muncityrecords", 1});
328
329     return true;
330 }
331
332 public void insertEditMunCityRecords(
333     PatientsForm pat, UserProfile prof, String
334     username) {
335     jdbcTemplateObject = new JdbcTemplate(
336         dataSource);
337
338     DateFormat dateFormat = new SimpleDateFormat("
339     yyyy-MM-dd");
340     Date date = new Date();
341
342     DateFormat timeFormat = new SimpleDateFormat("
343     HH:mm:ss");
344     Date time =new Date();
345
346     String SQL = "select accountID from users where
347     username ='" + username + "'";
348
349     int accountID = jdbcTemplateObject.queryForInt(
350     SQL);
351
352     int diseaseID = -1;
353
354     if(pat.getDisease().equals("Chikungunya"))
355         diseaseID = 0;
356     else if(pat.getDisease().equals("Dengue"))
357         diseaseID = 1;
358     else if(pat.getDisease().equals("Malaria"))
359         diseaseID = 2;

```



```

290
291 if(pat.getDateAdmitted() == null)
292 pat.setDateAdmitted("0000-00-00");
293 if(pat.getDateDied() == null)
294 pat.setDateDied("0000-00-00");
295
296 SQL = "SELECT * FROM muncityrecords WHERE
recordID = '" + pat.getRecordID() + "'";
297 List<PatientsForm> pGS = jdbcTemplateObject.
query(SQL, new PatientsFormMapper());
298
299 SQL = "DELETE FROM editmuncityrecords WHERE
recordID = '" + pat.getRecordID() + "'";
300 jdbcTemplateObject.update(SQL);
301
302 SQL = "insert into editmuncityrecords (
accountID, name, age, gender, birthDate,
address, admitted, dateAdmitted, "
303 + "outcome, dateDied, hospital, hospitalType,
hospitalAddr, region, province, muncity,
barangay, "
304 + "diseaseID, symptoms, healthworker,
muncityhealthworker, dateSubmitted,
recordID, dateApproved, dateOnset) "
305 + "values (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?,
?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)";
306
307 for(PatientsForm pG : pGS) {
308 if(pG.getDateAdmitted() == null)
309 pG.setDateAdmitted("0000-00-00");
310 if(pG.getDateDied() == null)
311 pG.setDateDied("0000-00-00");
312
313 jdbcTemplateObject.update(SQL, new Object[]{
accountID, pG.getName(), pG.getAge(), pG.
getGender(), pG.getDateofbirth(),
314 pG.getAddress(), pG.getAdmitted(), pG.
getDateAdmitted(), pG.getOutcome(), pG.
getDateDied(), pG.getHospital(),
315 pG.getHospitalType(), pG.getHospitalAddr(), pG
.getRegion(), pG.getProvince(), pG.
getMuncity(), pG.getBarangay(),
316 diseaseID, pG.getSymptoms(), pG.
getHealthworker(), pG.getMuncityHW(), pG.
getDateSubmitted(), pG.getRecordID(),
317 pG.getDateApproved(), pG.getDateOnset()});
318 }
319 SQL = "DELETE FROM muncityrecords WHERE recordID
= '" + pat.getRecordID() + "'";
320 jdbcTemplateObject.update(SQL);
321
322 SQL = "insert into muncityrecords (accountID,
name, age, gender, birthDate, address,
admitted, dateAdmitted, "
323 + "outcome, dateDied, hospital, hospitalType,
hospitalAddr, region, province, muncity,
barangay, "
324 + "diseaseID, symptoms, healthworker,
muncityhealthworker, dateSubmitted,
recordID, dateApproved, dateOnset,"
325 + "dateEdited, editedBy, editStatus) "
326 + "values (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?,
?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)";
327
328 jdbcTemplateObject.update(SQL, new Object[]{
accountID, pat.getName(), pat.getAge(), pat
.getGender(), pat.getDateofbirth(),
329 pat.getAddress(), pat.getAdmitted(), pat.
getDateAdmitted(), pat.getOutcome(), pat.
getDateDied(), pat.getHospital(),
330 pat.getHospitalType(), pat.getHospitalAddr(),
pat.getRegion(), pat.getProvince(), pat.
getMuncity(), pat.getBarangay(),
331 diseaseID, pat.getSymptoms(), pat.
getHealthworker(), pat.getMuncityHW(), pat.
getDateSubmitted(), pat.getRecordID(),
332 pat.getDateApproved(), pat.getDateOnset(),
dateFormat.format(date),
333 prof.getFirstname() + " " + prof.getMiddlename
() + " " + prof.getSurname(), 1});
334
335 String action = "Edited a municipal/city patient
record.";
336
337 SQL = "insert into userlogs (accountID, action,
date, time, recordID, editStatus,
recordtable, recordtype) values (?, ?, ?,
?, ?, ?, ?, ?)";
338
339 jdbcTemplateObject.update(SQL, new Object[]{
accountID, action, dateFormat.format(date),
timeFormat.format(time), pat.getRecordID()
, 1, "editmuncityrecords", 1});
340
341
342 }
343
344 public boolean rejectMunCityRecords(int id,
String name, String username) {
345
346 jdbcTemplateObject = new JdbcTemplate(
dataSource);
347
348 String SQL = "UPDATE barangayrecords SET status
= ?, dateApproved = ?, muncityhealthworker
= ? WHERE recordID = ?";
349
350 DateFormat dateFormat = new SimpleDateFormat("
yyyy-MM-dd");
351 Date date = new Date();
352
353 DateFormat timeFormat = new SimpleDateFormat("
HH:mm:ss");
354 Date time =new Date();
355
356 jdbcTemplateObject.update(SQL, 2, dateFormat.
format(date), name, id);
357
358 SQL = "select accountID from users where
username ='" + username + "'";
359
360 int accountID = jdbcTemplateObject.queryForInt(
SQL);
361
362 String action = "Rejected a barangay patient
record.";
363
364 SQL = "insert into userlogs (accountID, action,
date, time, recordID, recordtable,
recordtype) values (?, ?, ?, ?, ?, ?)";
365
366 jdbcTemplateObject.update(SQL, new Object[]{
accountID, action, dateFormat.format(date),
timeFormat.format(time), id, "
barangayrecords", 1});
367
368 return true;
369 }
370
371 public boolean insertProvRecords(int id, String
name, String username) throws Exception {
372
373 jdbcTemplateObject = new JdbcTemplate(
dataSource);
374
375 String SQL = "SELECT * FROM muncityrecords
WHERE recordID = ?";
376
377 PatientsForm pf = jdbcTemplateObject.
queryForObject(SQL, new Object[]{id}, new
PatientsFormMapper());
378
379 DateFormat dateFormat = new SimpleDateFormat("
yyyy-MM-dd");
380 Date date = new Date();
381
382 DateFormat timeFormat = new SimpleDateFormat("
HH:mm:ss");
383 Date time =new Date();
384
385 SQL = "select accountID from users where
username ='" + username + "'";
386
387 int accountID = jdbcTemplateObject.queryForInt(
SQL);
388
389 int diseaseID = -1;
390 if(pf.getDisease().equals("Chikungunya"))
diseaseID = 0;
391 else if(pf.getDisease().equals("Dengue"))
diseaseID = 1;
392 else if(pf.getDisease().equals("Malaria"))
diseaseID = 2;
393
394 if(pf.getDateDied() == null)
pf.setDateDied("0000-00-00");
395
396 if(pf.getDateAdmitted() == null)
pf.setDateAdmitted("0000-00-00");
397
398 SQL = "insert into patientrecords (accountID,
name, age, gender, birthDate, address,
admitted, dateAdmitted, "
399 + "outcome, dateDied, hospital, hospitalType,
hospitalAddr, region, province, muncity,
barangay, "
400 + "diseaseID, symptoms, healthworker,
muncityhealthworker, recordID, dateApproved
, dateSubmitted, "
401 + "dateOnset, dateApproved2,
provincehealthworker) "
402 + "values (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?,
?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)";
403
404
405
406
407
408
409
410
411
}

```

```

411
412 jdbcTemplateObject.update(SQL, new Object[]{
    accountID, pf.getName(), pf.getAge(), pf.
    getGender(), pf.getDateofbirth(), pf.
    getAddress(),
413 pf.getAdmitted(), pf.getDateAdmitted(), pf.
    getOutcome(), pf.getDateDied(), pf.
    getHospital(), pf.getHospitalType(),
414 pf.getHospitalAddr(), pf.getRegion(), pf.
    getProvince(), pf.getMuncity(), pf.
    getBarangay(), diseaseID, pf.getSymptoms(),
415 pf.getHealthworker(), pf.getMuncityHW(), pf.
    getRecordID(), pf.getDateApproved(), pf.
    getDateSubmitted(), pf.getDateOnset(),
    dateFormat.format(date), name});
416
417 SQL = "UPDATE muncityrecords SET status = ?,
    dateApproved2 = ?, provincehealthworker = ?
    WHERE recordID = ?";
418 jdbcTemplateObject.update(SQL, 1, dateFormat.
    format(date), name, id);
419
420
421 String action = "Approved municipality/city
    patient record. Inserted into patient
    records.";
422 SQL = "insert into userlogs (accountID, action,
    date, time, recordID, recordtable,
    recordtype) values (?, ?, ?, ?, ?, ?)";
423 jdbcTemplateObject.update(SQL, new Object[]{
    accountID, action, dateFormat.format(date),
    timeFormat.format(time), pf.getRecordID(),
    "muncityrecords", 2});
424
425
426 return true;
427
428 }
429
430 public boolean rejectProvRecords(int id, String
    name, String username) {
431
432 jdbcTemplateObject = new JdbcTemplate(
    dataSource);
433
434 String SQL = "UPDATE muncityrecords SET status
    = ?, dateApproved = ?, provincehealthworker
    = ? WHERE recordID = ?";
435
436 DateFormat dateFormat = new SimpleDateFormat("
    yyyy-MM-dd");
437 Date date = new Date();
438
439 DateFormat timeFormat = new SimpleDateFormat("
    HH:mm:ss");
440 Date time = new Date();
441
442 jdbcTemplateObject.update(SQL, 2, dateFormat.
    format(date), name, id);
443
444 SQL = "select accountID from users where
    username ='"+ username + "'";
445
446 int accountID = jdbcTemplateObject.queryForInt(
    SQL);
447
448 String action = "Rejected a municipal/city
    patient record.";
449 SQL = "insert into userlogs (accountID, action,
    date, time, recordID, recordtable,
    recordtype) values (?, ?, ?, ?, ?, ?)";
450 jdbcTemplateObject.update(SQL, new Object[]{
    accountID, action, dateFormat.format(date),
    timeFormat.format(time), id, "
    muncityrecords", 1});
451
452 return true;
453 }
454
455 public PatientsForm getRecord(int id) {
456
457 jdbcTemplateObject = new JdbcTemplate(
    dataSource);
458
459 String SQL = "SELECT * FROM barangayrecords
    WHERE recordID = ?";
460
461 PatientsForm pf = jdbcTemplateObject.
    queryForObject(SQL, new Object[]{id}, new
    PatientsFormMapper());
462
463 return pf;
464
465 }
466 public PatientsForm getMunCityRecord(int id) {
467
468 jdbcTemplateObject = new JdbcTemplate(
    dataSource);
469
470 String SQL = "SELECT * FROM muncityrecords
    WHERE recordID = ?";
471
472 PatientsForm pf = jdbcTemplateObject.
    queryForObject(SQL, new Object[]{id}, new
    PatientsFormMapper());
473
474 return pf;
475 }
476
477 public static Connection getMySQLConnection()
    throws Exception {
478
479 Connection conn = null;
480 String driver = "com.mysql.jdbc.Driver";
481 String url = "jdbc:mysql://localhost:3306/vbd?
    zeroDateTimeBehavior=convertToNull";
482 String username = "root";
483 String password = "";
484
485 Class.forName(driver);
486
487 conn = (Connection) DriverManager.getConnection
    (url, username, password);
488
489 return conn;
490 }
491
492 public List<PatientsForm> listOfOrigBrgyRecords(
    String muncity, String username) {
493 jdbcTemplateObject = new JdbcTemplate(
    dataSource);
494
495 String SQL = "select * from editbarangayrecords
    where muncity = ? ORDER BY dateSubmitted,
    barangay, recordID";
496
497 List<PatientsForm> pF = jdbcTemplateObject.
    query(SQL,
    new Object[]{muncity}, new PatientsFormMapper
    ());
498
499 return pF;
500 }
501
502 public List<PatientsForm> listOfEditBrgyRecords(
    String muncity, String username) {
503 jdbcTemplateObject = new JdbcTemplate(
    dataSource);
504
505 String SQL = "select * from barangayrecords
    where muncity = ? AND editStatus = ? ORDER
    BY dateSubmitted, barangay, recordID";
506
507 List<PatientsForm> pF = jdbcTemplateObject.
    query(SQL,
    new Object[]{muncity, 1}, new
    PatientsFormMapper());
508
509 return pF;
510 }
511
512 public List<PatientsForm> listOfAllBrgyRecords(
    String muncity, String province, String
    region) {
513 jdbcTemplateObject = new JdbcTemplate(
    dataSource);
514
515 String SQL = "select * from barangayrecords
    where muncity = ? AND province = ? AND
    region = ? ORDER BY dateSubmitted, muncity,
    barangay, recordID";
516
517 List<PatientsForm> muncy = jdbcTemplateObject.
    query(SQL,
    new Object[]{muncity, province, region}, new
    PatientsFormMapper());
518
519 return muncy;
520 }
521
522 public List<PatientsForm>
    listOfAllEditBrgyRecords(String muncity,
    String province, String region) {
523 jdbcTemplateObject = new JdbcTemplate(
    dataSource);
524
525 String SQL = "select * from barangayrecords
    where muncity = ? AND province = ? AND
    region = ? ORDER BY dateSubmitted, muncity,
    barangay, recordID";
526
527 List<PatientsForm> muncy = jdbcTemplateObject.
    query(SQL,
    new Object[]{muncity, province, region}, new
    PatientsFormMapper());
528
529 return muncy;
530 }
531
532 public List<PatientsForm> listOfAllEditBrgyRecords(
    String muncity, String province, String region) {
533 jdbcTemplateObject = new JdbcTemplate(
    dataSource);
534
535 String SQL = "select * from barangayrecords
    where muncity = ? AND province = ? AND
    region = ? ORDER BY dateSubmitted, muncity,
    barangay, recordID";
536
537 List<PatientsForm> muncy = jdbcTemplateObject.
    query(SQL,
    new Object[]{muncity, province, region}, new
    PatientsFormMapper());
538
539 return muncy;
540 }

```

```

535
536     return muncy;
537 }
538 }
539
540 public List<PatientsForm>
    listOfAllMunCityRecords(String province,
        String region) {
541     jdbcTemplateObject = new JdbcTemplate(
        dataSource);
542
543     String SQL = "select * from muncityrecords
        where province = ? AND region = ? ORDER BY
        dateSubmitted, muncity, barangay, recordID
        ";
544
545     List<PatientsForm> muncy = jdbcTemplateObject.
        query(SQL,
546         new Object[]{province, region}, new
        PatientsFormMapper());
547
548     return muncy;
549 }
550
551
552 public List<PatientsForm>
    listOfAllProvincialRecords(String province,
        String region) {
553     jdbcTemplateObject = new JdbcTemplate(
        dataSource);
554
555     String SQL = "select * from muncityrecords
        where province = ? AND region = ? AND
        dateApproved2 > ? ORDER BY dateSubmitted,
        muncity, barangay, recordID";
556
557     List<PatientsForm> muncy = jdbcTemplateObject.
        query(SQL,
558         new Object[]{province, region, "0000-00-00"},
        new PatientsFormMapper());
559
560     return muncy;
561
562 }
563
564 public List<PatientsForm>
    listOfOrigMunCityRecords(String province,
        String username) {
565     jdbcTemplateObject = new JdbcTemplate(
        dataSource);
566
567     String SQL = "select * from editmuncityrecords
        where province = ? ORDER BY dateSubmitted,
        barangay, recordID";
568
569     List<PatientsForm> pF = jdbcTemplateObject.
        query(SQL,
570         new Object[]{province}, new PatientsFormMapper
        ());
571
572     return pF;
573 }
574
575
576 public List<PatientsForm>
    listOfEditMunCityRecords(String province,
        String username) {
577     jdbcTemplateObject = new JdbcTemplate(
        dataSource);
578
579     String SQL = "select * from muncityrecords
        where province = ? AND editStatus = ? ORDER
        BY dateSubmitted, barangay, recordID";
580
581     List<PatientsForm> pF = jdbcTemplateObject.
        query(SQL,
582         new Object[]{province, 1}, new
        PatientsFormMapper());
583
584     return pF;
585 }
586
587 }
588 }

```

Listing 7: DiseaseDAO.java

```

1 package dao;
2
3 import java.text.DateFormat;
4 import java.text.SimpleDateFormat;
5 import java.util.Date;
6 import java.util.List;
7
8 import mapper.AlertsMapper;
9 import mapper.DiseaseMapper;
10 import objects.Alerts;
11 import objects.Diseases;
12
13 import org.springframework.beans.factory.
    annotation.Autowired;
14 import org.springframework.jdbc.core.
    JdbcTemplate;
15
16 public class DiseaseDAO {
17
18     @Autowired
19     private javax.sql.DataSource dataSource;
20     private JdbcTemplate jdbcTemplateObject;
21
22
23     public List<Diseases> listDiseases() {
24         jdbcTemplateObject = new JdbcTemplate(
            dataSource);
25
26         String SQL = "select * from diseases";
27
28         List<Diseases> diseases = jdbcTemplateObject.
            query(SQL,
29             new DiseaseMapper());
30
31         return diseases;
32     }
33
34
35     public Diseases getDisease(int id) {
36
37         jdbcTemplateObject = new JdbcTemplate(
            dataSource);
38
39         String SQL = "SELECT * FROM diseases WHERE
            diseaseID = '"+ id + "'";
40
41         String ans = "";
42         Diseases disease = jdbcTemplateObject.
            queryForObject(SQL, new DiseaseMapper());
43
44         return disease;
45     }
46
47     public List<Alerts> getAlerts(String barangay,
        String muncity, String province) {
48
49         jdbcTemplateObject = new JdbcTemplate(
            dataSource);
50
51         String SQL = "SELECT * FROM alerts WHERE (
            barangay = ? OR muncity = ?) AND province =
            ?";
52
53         List<Alerts> alerts = jdbcTemplateObject.query
            (SQL, new Object[]{barangay, muncity,
            province}, new AlertsMapper());
54
55         return alerts;
56     }
57
58
59     public void insertAlert(String barangay, String
        muncity, String province, String region,
        double outbreak, String disease) {
60
61         jdbcTemplateObject = new JdbcTemplate(
            dataSource);
62
63         DateFormat dateFormat = new SimpleDateFormat("
            yyyy-MM-dd");
64         Date date = new Date();
65
66         int diseaseID = -1;
67
68         if(disease.equals("Chikungunya"))
69             diseaseID = 0;
70         else if(disease.equals("Dengue"))
71             diseaseID = 1;
72         else if(disease.equals("Malaria"))
73             diseaseID = 2;
74         String SQL = "INSERT INTO alerts (barangay,
            muncity, province, region, outbreak, date,
            disease, diseaseID) VALUES (?, ?, ?, ?, ?,
            ?, ?, ?)";
75
76         jdbcTemplateObject.update(SQL, new Object[]{
            barangay, muncity, province, region,
            outbreak, dateFormat.format(date),
            disease, diseaseID});
77
78
79
80 }
81
82 public List<Alerts> getAlerts() {
83
84         jdbcTemplateObject = new JdbcTemplate(
            dataSource);
85

```

```

86 String SQL = "SELECT * FROM alerts ORDER BY
87     date DESC";
88 List<Alerts> alerts = jdbcTemplateObject.query
89     (SQL, new AlertsMapper());
90 return alerts;
91 }
92 public int getCountAlerts() {
93     jdbcTemplateObject = new JdbcTemplate(
94         dataSource);
95 String SQL = "SELECT COUNT(*) FROM alerts
96     WHERE status = 0";
97
98
99     return jdbcTemplateObject.queryForInt(SQL);
100 }
101 }
102 public void updateAlerts() {
103     jdbcTemplateObject = new JdbcTemplate(
104         dataSource);
105 String SQL = "UPDATE alerts SET status = 1
106     WHERE status = 0";
107     jdbcTemplateObject.update(SQL);
108 }
109 }
110 }
111 }
112 }

```

Listing 8: MapsDAO.java

```

1 package dao;
2
3 import java.sql.DriverManager;
4 import java.sql.ResultSet;
5 import java.text.DateFormat;
6 import java.text.SimpleDateFormat;
7 import java.util.ArrayList;
8 import java.util.Calendar;
9 import java.util.Date;
10 import java.util.List;
11
12 import mapper.BarangayRecordsMapper;
13 import mapper.MapsMapper;
14 import mapper.PlaceMapper;
15 import objects.BarangayRecords;
16 import objects.Maps;
17 import objects.Place;
18
19 import org.springframework.beans.factory.
20     annotation.Autowired;
21 import org.springframework.jdbc.core.
22     JdbcTemplate;
23
24 import com.mysql.jdbc.Connection;
25 import com.mysql.jdbc.PreparedStatement;
26
27 public class MapsDAO {
28     @Autowired
29     private javax.sql.DataSource dataSource;
30     private JdbcTemplate jdbcTemplateObject;
31
32     public List<Maps> listAllMuncity(String region)
33     throws Exception {
34         jdbcTemplateObject = new JdbcTemplate(
35             dataSource);
36         Connection dbConnect = getMySQLConnection();
37         PreparedStatement pSt = null;
38
39         String SQL = "SELECT DISTINCT muncity FROM
40             maps where region = '" + region + "' ORDER
41             BY muncity";
42         pSt = (PreparedStatement) dbConnect.
43             prepareStatement(SQL);
44         ResultSet rs = pSt.executeQuery();
45
46         DateFormat dateFormat = new SimpleDateFormat("
47             yyyy-MM-dd");
48         Date date = new Date();
49         Date late = new Date();
50         late.setDate(1);
51         Calendar cal = Calendar.getInstance();
52         date.setDate(cal.getActualMaximum(Calendar.
53             DAY_OF_MONTH));
54
55         List<Maps> map = new ArrayList<Maps>();
56
57         while(rs.next()) {
58             Maps maps = new Maps();
59             maps.setMuncity(rs.getString("muncity"));
60             maps.setPopulation(getTotalPop(rs.getString("
61                 muncity"), region));
62             map.add(maps);
63         }
64         pSt.close();
65         dbConnect.close();
66
67         return map;
68     }
69
70     public int getTotalPop(String muncity, String
71     region) throws Exception {
72         jdbcTemplateObject = new JdbcTemplate(
73             dataSource);
74         int n = muncity.indexOf("'");
75
76         if(n > 0) {
77             String ne = "";
78             ne += muncity.substring(0, n);
79             ne += "\\\"";
80             ne += muncity.substring(n);
81             muncity = ne;
82         }
83         String SQL = "SELECT population FROM maps
84             where region = '" + region + "' AND muncity
85             = '" + muncity + "'";
86         List<Maps> map = jdbcTemplateObject.query(SQL,
87             new MapsMapper());
88         int total = 0;
89         for(Maps m : map) {
90             total += m.getPopulation();
91         }
92         return total;
93     }
94
95     public List<Maps> listOfCPatients() throws
96     Exception {
97         jdbcTemplateObject = new JdbcTemplate(
98             dataSource);
99         Connection dbConnect = getMySQLConnection();
100         PreparedStatement pSt = null;
101
102         DateFormat dateFormat = new SimpleDateFormat("
103             yyyy-MM-dd");
104         Date date = new Date();
105         Date late = new Date();
106         late.setDate(1);
107         Calendar cal = Calendar.getInstance();
108         date.setDate(cal.getActualMaximum(Calendar.
109             DAY_OF_MONTH));
110
111         String SQL = "SELECT * FROM patientrecords
112             NATURAL LEFT JOIN maps where diseaseID =
113             '" + 0 + "' AND dateSubmitted "
114             + ">= '" + dateFormat.format(late) + "' AND
115             dateSubmitted <= '" + dateFormat.format(
116                 date) + "' GROUP BY barangay ORDER BY
117             dateSubmitted DESC";
118
119         pSt = (PreparedStatement) dbConnect.
120             prepareStatement(SQL);
121         ResultSet rs = pSt.executeQuery();
122         List<Maps> map = new ArrayList<Maps>();
123
124         while(rs.next()) {
125             Maps maps = new Maps();
126             maps.setBarangay(rs.getString("barangay"));
127             try {
128                 maps.setMuncity(rs.getString("muncity"));
129                 maps.setProvince(rs.getString("province"));
130                 maps.setRegion(rs.getString("region"));
131             }
132             catch(Exception es) {
133             }
134
135             SQL = "SELECT COUNT(*) AS count FROM
136                 patientrecords WHERE barangay = '" + rs.
137                 getString("barangay") + "' "
138                 + "AND dateSubmitted >= '" + dateFormat.
139                 format(late) + "' AND dateSubmitted <= '" +
140                 dateFormat.format(date) + "'";
141             maps.setCountCases(jdbcTemplateObject.
142                 queryForInt(SQL));
143             SQL = "SELECT COUNT(*) AS count FROM
144                 patientrecords WHERE barangay = '" + rs.
145                 getString("barangay") + "' AND "
146                 + "outcome = 'Died' AND dateSubmitted >= '" +
147                 dateFormat.format(late) + "' AND
148                 dateSubmitted <= '" + dateFormat.format(
149                     date) + "'";
150             maps.setCountDeaths(jdbcTemplateObject.
151                 queryForInt(SQL));

```

```

123     maps.setX(rs.getDouble("x"));
124     maps.setY(rs.getDouble("y"));
125     maps.setPopulation(rs.getInt("population"));
126     map.add(maps);
127 }
128
129 pSt.close();
130 dbConnect.close();
131
132 return map;
133
134 }
135
136 public List<Maps> listOfDPatients() throws
137     Exception {
138     jdbcTemplateObject = new JdbcTemplate(
139         dataSource);
140     Connection dbConnect = getMySQLConnection();
141     PreparedStatement pSt = null;
142     String SQL = "SELECT * FROM patientrecords
143         NATURAL LEFT JOIN maps where diseaseID =
144         '"+ 1 + "' GROUP BY barangay ORDER BY
145         dateSubmitted DESC";
146     pSt = (PreparedStatement) dbConnect.
147         prepareStatement(SQL);
148     ResultSet rs = pSt.executeQuery();
149
150     List<Maps> map = new ArrayList<Maps>();
151
152     while(rs.next()) {
153         Maps maps = new Maps();
154         maps.setBarangay(rs.getString("barangay"));
155         maps.setMuncity(rs.getString("muncity"));
156         maps.setProvince(rs.getString("province"));
157         maps.setRegion(rs.getString("region"));
158
159         SQL = "SELECT COUNT(*) AS count FROM
160             patientrecords WHERE barangay = '" + rs.
161             getString("barangay") + "'";
162         maps.setCountCases(jdbcTemplateObject.
163             queryForInt(SQL));
164         SQL = "SELECT COUNT(*) AS count FROM
165             patientrecords WHERE barangay = '" + rs.
166             getString("barangay") + "' AND "
167             + "outcome = 'Died'";
168         maps.setCountDeaths(jdbcTemplateObject.
169             queryForInt(SQL));
170         maps.setX(rs.getDouble("x"));
171         maps.setY(rs.getDouble("y"));
172         maps.setPopulation(rs.getInt("population"));
173         map.add(maps);
174     }
175     pSt.close();
176     dbConnect.close();
177     return map;
178 }
179
180 public List<Maps> listOfMPatients() throws
181     Exception {
182     jdbcTemplateObject = new JdbcTemplate(
183         dataSource);
184     Connection dbConnect = getMySQLConnection();
185     PreparedStatement pSt = null;
186     String SQL = "SELECT * FROM patientrecords
187         NATURAL LEFT JOIN maps where diseaseID =
188         '"+ 2 + "' GROUP BY barangay ORDER BY
189         dateSubmitted DESC";
190     pSt = (PreparedStatement) dbConnect.
191         prepareStatement(SQL);
192     ResultSet rs = pSt.executeQuery();
193
194     List<Maps> map = new ArrayList<Maps>();
195
196     while(rs.next()) {
197         Maps maps = new Maps();
198         maps.setBarangay(rs.getString("barangay"));
199         maps.setMuncity(rs.getString("muncity"));
200         maps.setProvince(rs.getString("province"));
201         maps.setRegion(rs.getString("region"));
202
203         SQL = "SELECT COUNT(*) AS count FROM
204             patientrecords WHERE barangay = '" + rs.
205             getString("barangay") + "'";
206         maps.setCountCases(jdbcTemplateObject.
207             queryForInt(SQL));
208         SQL = "SELECT COUNT(*) AS count FROM
209             patientrecords WHERE barangay = '" + rs.
210             getString("barangay") + "' AND "
211             + "outcome = 'Died'";
212         maps.setCountDeaths(jdbcTemplateObject.
213             queryForInt(SQL));
214         maps.setX(rs.getDouble("x"));
215         maps.setY(rs.getDouble("y"));
216         maps.setPopulation(rs.getInt("population"));
217         map.add(maps);
218     }
219     pSt.close();
220     dbConnect.close();
221     return map;
222 }
223
224 public static Connection getMySQLConnection()
225     throws Exception {
226     Connection conn = null;
227     String driver = "com.mysql.jdbc.Driver";
228     String url = "jdbc:mysql://localhost:3306/vbd?
229         zeroDateTimeBehavior=convertToNull";
230     String username = "root";
231     String password = "";
232
233     Class.forName(driver);
234     conn = (Connection) DriverManager.
235         getConnection(url, username, password);
236     return conn;
237 }
238
239 public List<Place> getAllMuncity(String region)
240     {
241     jdbcTemplateObject = new JdbcTemplate(
242         dataSource);
243     String SQL = "SELECT DISTINCT muncity FROM
244         maps WHERE region = '" + region + "'";
245     List<Place> place = jdbcTemplateObject.query(
246         SQL, new PlaceMapper());
247     return place;
248 }
249
250 public List<Place> getAllProvince(String region)
251     {
252     jdbcTemplateObject = new JdbcTemplate(
253         dataSource);
254     String SQL = "SELECT DISTINCT province FROM
255         maps WHERE region = '" + region + "'";
256     List<Place> place = jdbcTemplateObject.query(
257         SQL, new PlaceMapper());
258     for(Place p : place) {
259         p.setRegion(region);
260     }
261     return place;
262 }
263
264 public List<Place> getAllMuncityFromProv(String
265     province, String region) {
266     jdbcTemplateObject = new JdbcTemplate(
267         dataSource);
268     String SQL = "SELECT DISTINCT muncity FROM
269         maps WHERE province = '" + province + "'
270         AND region = '" + region + "' ORDER BY
271         muncity";
272     List<Place> place = jdbcTemplateObject.query(
273         SQL, new PlaceMapper());
274     for(Place p : place) {
275         p.setProvince(province);
276         p.setRegion(region);
277     }
278     return place;
279 }
280
281 public List<Place> getAllBarangay(String
282     muncity, String province, String region) {
283     jdbcTemplateObject = new JdbcTemplate(
284         dataSource);
285     int n = muncity.indexOf("");
286     if(n > 0) {
287         String ne = "";
288         ne += muncity.substring(0, n);
289         ne += "\\\"";
290         ne += muncity.substring(n);

```

```

278     muncity = ne;
279 }
280 String SQL = "SELECT DISTINCT barangay FROM
    maps WHERE muncity = '" + muncity + "' AND
    province = '" + province + "' AND region =
    '" + region + "' ORDER BY barangay";
281
282 List<Place> place = jdbcTemplateObject.query(
    SQL, new PlaceMapper());
283 for(Place p : place) {
284     p.setMuncity(muncity);
285     p.setProvince(province);
286     p.setRegion(region);
287 }
288 return place;
289 }
290
291
292 }

```

Listing 9: NewUserDAO.java

```

1 package dao;
2
3 import java.math.BigInteger;
4 import java.security.SecureRandom;
5
6 import mapper.RegistrationMapper;
7 import mapper.UserChangeMapper;
8 import mapper.UserProfileMapper;
9 import objects.Registrants;
10 import objects.UserChange;
11 import objects.UserProfile;
12
13 import org.springframework.beans.factory.
    annotation.Autowired;
14 import org.springframework.jdbc.core.
    JdbcTemplate;
15 import org.springframework.security.crypto.
    bcrypt.BCryptPasswordEncoder;
16 import org.springframework.security.crypto.
    password.PasswordEncoder;
17
18 public class NewUserDAO {
19
20     @Autowired
21     private javax.sql.DataSource dataSource;
22     private JdbcTemplate jdbcTemplateObject;
23     private SecureRandom random;
24
25     public void insert(String firstname, String
        middlename, String surname, String gender,
        String address,
26     String userType, int userTypeInt, String
        region, String province,
27     String muncity, String barangay, String
        institution,
28     String instAddr, String email, String number)
        {
29
30     jdbcTemplateObject = new JdbcTemplate(
        dataSource);
31
32     int status = 0;
33
34     String SQL = "insert into registrations (
        firstname, middlename, surname, gender,
        address, userType, userTypeInt, region,
        province, muncity, barangay, institution,
        instAddr, email, number, status) values (?,
        ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?,
        ?)";
35
36     jdbcTemplateObject.update(SQL, new Object
        []{firstname, middlename, surname, gender,
        address, userType, userTypeInt,
        region, province, muncity, barangay,
        institution, instAddr, email, number,
        status});
37
38 }
39 public String getUser(int id) {
40
41     jdbcTemplateObject = new JdbcTemplate(
        dataSource);
42
43     String SQL2 = "select * from registrations
        where regID = ?";
44
45     Registrants reg = jdbcTemplateObject.
        queryForObject(SQL2, new Object[]{id}, new
        RegistrationMapper());
46
47
48
49     String x = insertNewUser(reg);
50
51
52     return x;
53 }
54
55 public void rejectUser(int id) {
56
57     jdbcTemplateObject = new JdbcTemplate(
        dataSource);
58
59     String SQL = "UPDATE registrations SET status
        = ? where regID = ?";
60
61     jdbcTemplateObject.update(SQL, 2, id);
62
63 }
64
65 public UserChange getAccount(int id) {
66
67     jdbcTemplateObject = new JdbcTemplate(
        dataSource);
68
69     String SQL = "select * from users where
        accountID = ?";
70
71     UserChange user = jdbcTemplateObject.
        queryForObject(SQL, new Object[]{id}, new
        UserChangeMapper());
72
73     return user;
74 }
75
76 public String insertNewUser(Registrants reg) {
77
78     random = new SecureRandom();
79     String pass = new BigInteger(130, random).
        toString(32);
80     pass = pass.substring(0, 10);
81
82     PasswordEncoder passwordEncoder = new
        BCryptPasswordEncoder();
83     String encodedPassword = passwordEncoder.
        encode(pass);
84
85     jdbcTemplateObject = new JdbcTemplate(
        dataSource);
86     String SQL = "insert into accounts (
        accountID, firstname, middlename, surname,
        username, gender, address, userType,
        userTypeInt, region, province, muncity,
        barangay, institution, instAddr, email,
        number) values (?, ?, ?, ?, ?, ?, ?, ?, ?,
        ?, ?, ?, ?, ?, ?, ?)";
87
88     jdbcTemplateObject.update(SQL, new Object
        []{reg.getAccountID(), reg.getFirstname(),
        reg.getMiddlename(), reg.getSurname(), reg.
        getUsername(), reg.getGender(),
92     reg.getAddress(), reg.getUserType(), reg.
        getUserTypeInt(), reg.getRegion(), reg.
        getProvince(), reg.getMuncity(),
        reg.getBarangay(), reg.getInstitution(),
        reg.getInstitutionAddr(), reg.getEmail(),
        reg.getNumber()});
93
94     SQL = "UPDATE registrations SET status = ?
        where regID = ?";
95     int status = 1;
96
97     jdbcTemplateObject.update(SQL, status, reg.
        getAccountID());
98
99     String authorities = "";
100     if(reg.getUserTypeInt() == 0)
101         authorities = "ROLE_BHWorker";
102     else if(reg.getUserTypeInt() == 1)
103         authorities = "ROLE_MCHWorker";
104     else if(reg.getUserTypeInt() == 2)
105         authorities = "ROLE_MCOfficial";
106     else if(reg.getUserTypeInt() == 3)
107         authorities = "ROLE_PHWorker";
108     else if(reg.getUserTypeInt() == 4)
109         authorities = "ROLE_PHOfficial";
110     else if(reg.getUserTypeInt() == 5)
111         authorities = "ROLE_NHOfficial";
112     else if(reg.getUserTypeInt() == 6)
113         authorities = "ROLE_Researcher";
114     else
115         authorities = "ROLE_USER";
116
117     SQL = "insert into users (accountID,
        username, password, authorities) values (?,
        ?, ?, ?)";
118
119     jdbcTemplateObject.update(SQL, new Object

```

```

    []{reg.getAccountID(), reg.getUsername(),
    encodedPassword, authorities});
122
123
124     return pass;
125
126 }
127
128 public boolean editUser(String name, UserChange
    user) {
129
130     PasswordEncoder passwordEncoder = new
    BCryptPasswordEncoder();
131     String encodedPassword = passwordEncoder.
    encode(user.getAccountPassword2());
132
133     jdbcTemplateObject = new JdbcTemplate(
    dataSource);
134
135
136     String SQL = "UPDATE users SET username =
    ?, password = ? where username = ?";
137
138     jdbcTemplateObject.update(SQL, user.
    getAccountName(), encodedPassword, name);
139
140     return true;
141
142 }
143
144 public boolean editUserContacts(String name,
    String email, String number) {
145
146     jdbcTemplateObject = new JdbcTemplate(
    dataSource);
147
148     String SQL = "UPDATE accounts SET email = ?,
    number = ? where username = ?";
149
150     jdbcTemplateObject.update(SQL, email,
    number, name);
151
152     return true;
153
154 }
155
156 }

```

Listing 10: PatientsDAO.java

```

1 package dao;
2
3 import java.text.DateFormat;
4 import java.text.SimpleDateFormat;
5 import java.util.Calendar;
6 import java.util.Date;
7 import java.util.List;
8
9 import mapper.PatientsFormMapper;
10 import mapper.PatientsMapper;
11 import objects.Patients;
12 import objects.PatientsForm;
13
14 import org.springframework.beans.factory.
    annotation.Autowired;
15 import org.springframework.jdbc.core.
    JdbcTemplate;
16
17 public class PatientsDAO {
18
19     @Autowired
20     private javax.sql.DataSource dataSource;
21     private JdbcTemplate jdbcTemplateObject;
22
23     public List<Patients> listAllPatients() {
24
25         jdbcTemplateObject = new JdbcTemplate(
            dataSource);
26         DateFormat dateFormat = new SimpleDateFormat("
            yyyy-MM-dd");
27         Date date = new Date();
28         Date late = new Date();
29         late.setDate(1);
30         Calendar cal = Calendar.getInstance();
31         date.setDate(cal.getActualMaximum(Calendar.
            DAY_OF_MONTH));
32         String SQL = "select * from patientrecords
            where dateSubmitted >= '"+ dateFormat.
            format(late) + "' AND dateSubmitted <= '" +
            dateFormat.format(date) + "'";
33
34         List<Patients> allPatients =
            jdbcTemplateObject.query(SQL,
            new PatientsMapper());
35
36         return allPatients;
37     }
38
39     public List<Patients> listOldPatients() {
40
41         jdbcTemplateObject = new JdbcTemplate(
            dataSource);
42         DateFormat dateFormat = new SimpleDateFormat("
            yyyy-MM-dd");
43         Date date = new Date();
44         Date late = new Date();
45         late.setDate(1);
46         late.setYear(113);
47
48         Calendar cal = Calendar.getInstance();
49         date.setDate(cal.getActualMaximum(Calendar.
            DAY_OF_MONTH));
50         date.setYear(113);
51         String SQL = "select * from patientrecords
            where dateSubmitted >= '"+ dateFormat.
            format(late) + "' AND dateSubmitted <= '" +
            dateFormat.format(date) + "'";
52
53         List<Patients> allPatients =
            jdbcTemplateObject.query(SQL,
            new PatientsMapper());
54
55         return allPatients;
56     }
57
58     public List<PatientsForm> listAllPatients(
        String dateFrom, String dateTo, String
        disease) {
59
60         jdbcTemplateObject = new JdbcTemplate(
            dataSource);
61         int diseaseID = -1;
62         if(disease.equalsIgnoreCase("Chikungunya"))
            diseaseID = 0;
63         else if(disease.equalsIgnoreCase("Dengue"))
            diseaseID = 1;
64         else if(disease.equalsIgnoreCase("Malaria"))
            diseaseID = 2;
65
66         String SQL = "select * from patientrecords
            where dateSubmitted >= '"+ dateFrom + "'
            AND dateSubmitted <= '"+ dateTo + "' AND
            diseaseID = '" + diseaseID + "'";
67
68         List<PatientsForm> allPatients =
            jdbcTemplateObject.query(SQL,
            new PatientsFormMapper());
69
70         return allPatients;
71     }
72
73     public List<PatientsForm> listAllPatients(
        String dateFrom, String dateTo) {
74
75         jdbcTemplateObject = new JdbcTemplate(
            dataSource);
76         String SQL = "select * from patientrecords
            where dateSubmitted >= '"+ dateFrom + "'
            AND dateSubmitted <= '"+ dateTo + "'";
77
78         List<PatientsForm> allPatients =
            jdbcTemplateObject.query(SQL,
            new PatientsFormMapper());
79
80         return allPatients;
81     }
82
83     public List<PatientsForm> listAllPatients(
        int classify) {
84
85         jdbcTemplateObject = new JdbcTemplate(
            dataSource);
86         String SQL = "select * from patientrecords
            where dateSubmitted >= '"+ dateFrom + "'
            AND dateSubmitted <= '"+ dateTo + "'";
87
88         List<PatientsForm> allPatients =
            jdbcTemplateObject.query(SQL,
            new PatientsFormMapper());
89
90         return allPatients;
91     }
92
93     }

```

Listing 11: RegistrationDAO.java

```

1 package dao;
2
3 import java.util.List;
4
5 import mapper.RegistrationMapper;
6 import objects.Registrants;
7
8 import org.springframework.beans.factory.
    annotation.Autowired;
9 import org.springframework.jdbc.core.
    JdbcTemplate;
10
11 public class RegistrationDAO {
12
13     @Autowired
14     private javax.sql.DataSource dataSource;
15     private JdbcTemplate jdbcTemplateObject;
16
17     public List<Registrants> listAllRegistrations(
        int classify) {
18
19         jdbcTemplateObject = new JdbcTemplate(
            dataSource);
20
21         String SQL = "select * from registration
            where dateSubmitted >= '"+ dateFrom + "'
            AND dateSubmitted <= '"+ dateTo + "'";
22
23         List<Registrants> allRegistrations =
            jdbcTemplateObject.query(SQL,
            new RegistrantsMapper());
24
25         return allRegistrations;
26     }
27
28     }

```

```

        dataSource);
20 String SQL = "";
21 if(classify != 5)
22     SQL = "select * from registrations where
23         userTypeInt < '"+ classify +"';
24 else
25     SQL = "select * from registrations where
26         userTypeInt < '"+ classify +" OR
27         userTypeInt = 6";
28
29 List<Registrants> allRegs = jdbcTemplateObject
30     .query(SQL,
31         new RegistrationMapper());
32
33 return allRegs;
34 }
35
36 public int countPending(int classify) {
37     jdbcTemplateObject = new JdbcTemplate(
38         dataSource);
39     String SQL = "";
40
41     if(classify != 5 && classify != 7)
42         SQL = "select count(*) from registrations
43             where status = '0' AND userTypeInt < '"+
44                 classify +"';
45     else
46         SQL = "select count(*) from registrations
47             where status = '0' AND userTypeInt <> 0 AND
48             ( userTypeInt < '"+ classify +" OR
49             userTypeInt = 6 )";
50
51     int rowCount = jdbcTemplateObject.queryForInt(
52         SQL);
53
54 return rowCount;
55 }
56
57 public int countApproved(int classify) {
58     jdbcTemplateObject = new JdbcTemplate(
59         dataSource);
60     String SQL = "";
61
62     if(classify != 5 && classify != 7)
63         SQL = "select count(*) from registrations
64             where status = '1' AND userTypeInt < '"+
65                 classify +"';
66     else
67         SQL = "select count(*) from registrations
68             where status = '1' AND userTypeInt <> 0 AND
69             ( userTypeInt < '"+ classify +" OR
70             userTypeInt = 6 )";
71
72     int rowCount = jdbcTemplateObject.queryForInt(
73         SQL);
74
75 return rowCount;
76 }
77
78 public int countRejected(int classify) {
79     jdbcTemplateObject = new JdbcTemplate(
80         dataSource);
81     String SQL = "";
82
83     if(classify != 5 && classify != 7)
84         SQL = "select count(*) from registrations
85             where status = '2' AND userTypeInt < '"+
86                 classify +"';
87     else
88         SQL = "select count(*) from registrations
89             where status = '2' AND userTypeInt <> 0 AND
90             ( userTypeInt < '"+ classify +" OR
91             userTypeInt = 6 )";
92
93     int rowCount = jdbcTemplateObject.queryForInt(
94         SQL);
95
96 return rowCount;
97 }
98
99 }

```

Listing 12: UserProfileDAO.java

```

1 package dao;
2
3 import java.sql.DriverManager;
4 import java.sql.ResultSet;
5 import java.text.DateFormat;
6 import java.text.SimpleDateFormat;
7 import java.util.ArrayList;
8 import java.util.Date;
9 import java.util.List;
10 import java.util.Map;
11
12 import mapper.DiseaseMapper;
13 import mapper.RegistrationMapper;
14 import mapper.UserChangeMapper;
15 import mapper.UserProfileMapper;
16 import objects.Diseases;
17 import objects.Maps;
18 import objects.Registrants;
19 import objects.UserChange;
20 import objects.UserLogs;
21 import objects.UserProfile;
22
23 import org.springframework.beans.factory.
24     annotation.Autowired;
25 import org.springframework.jdbc.core.
26     JdbcTemplate;
27 import org.springframework.security.core.
28     Authentication;
29 import org.springframework.security.core.context.
30     SecurityContextHolder;
31
32 import com.mysql.jdbc.Connection;
33 import com.mysql.jdbc.PreparedStatement;
34
35 public class UserProfileDAO {
36
37     @Autowired
38     private javax.sql.DataSource dataSource;
39     private JdbcTemplate jdbcTemplateObject;
40
41     public List<String> getAllUserNames() {
42         jdbcTemplateObject = new JdbcTemplate(
43             dataSource);
44         String SQL = "select username from users";
45         List<String> userN = jdbcTemplateObject.
46             queryForList(SQL, String.class);
47
48         return userN;
49     }
50
51     public List<String> getAllRegNames() {
52         jdbcTemplateObject = new JdbcTemplate(
53             dataSource);
54         String SQL = "select username from
55             registrations";
56         List<String> userN = jdbcTemplateObject.
57             queryForList(SQL, String.class);
58
59         return userN;
60     }
61
62     public List<String> getAllUserNumbers() {
63         jdbcTemplateObject = new JdbcTemplate(
64             dataSource);
65         String SQL = "select number from accounts";
66         List<String> numbers = jdbcTemplateObject.
67             queryForList(SQL, String.class);
68
69         return numbers;
70     }
71
72     public List<String> getAllRegNumbers() {
73         jdbcTemplateObject = new JdbcTemplate(
74             dataSource);
75         String SQL = "select number from registrations
76             ";
77         List<String> numbers = jdbcTemplateObject.
78             queryForList(SQL, String.class);
79
80         return numbers;
81     }
82
83     public List<String> getAllEmails() {
84         jdbcTemplateObject = new JdbcTemplate(
85             dataSource);
86         String SQL = "select email from accounts";
87         List<String> email = jdbcTemplateObject.
88             queryForList(SQL, String.class);
89
90         return email;
91     }
92
93     public List<UserProfile> allUsers(String
94         barangay, String muncity, String province,

```



```

    String region) {
88
89     jdbcTemplateObject = new JdbcTemplate(
        dataSource);
90
91     String SQL = "select * from accounts where (
        barangay = ? OR muncity = ?) AND province =
        ? AND region = ?";
92
93     List<UserProfile> allU = jdbcTemplateObject.
        query(SQL, new Object[]{barangay, muncity,
94         province, region}, new UserProfileMapper());
95
96     return allU;
97 }
98
99 public int getUserAccID(String username) {
100
101     jdbcTemplateObject = new JdbcTemplate(
        dataSource);
102
103     String SQL = "select accountID from users
        where username ='"+ username + "'";
104
105     int accountID = jdbcTemplateObject.queryForInt
        (SQL);
106
107     return accountID;
108 }
109
110 public UserChange getUserDetails(String
    username) {
111
112     jdbcTemplateObject = new JdbcTemplate(
        dataSource);
113
114     String SQL = "select * from users where
        username = ?";
115
116     UserChange user = jdbcTemplateObject.
        queryForObject(SQL, new Object[]{username},
        new UserChangeMapper());
117
118     return user;
119 }
120
121 public String changeUserDetails(UserProfile
    prof, UserChange user) {
122
123     jdbcTemplateObject = new JdbcTemplate(
        dataSource);
124
125     String SQL = "UPDATE users SET username = ?,
        password = ? where accountID = ?";
126
127     jdbcTemplateObject.update(SQL, user.
        getAccountName(), user.getAccountPassword(),
        prof.getAccID());
128
129     return user.getAccountName();
130
131 }
132
133
134
135 public UserProfile getUser(String username) {
136
137     jdbcTemplateObject = new JdbcTemplate(
        dataSource);
138
139     String SQL = "select accountID from users
        where username ='"+ username + "'";
140
141     int accountID = jdbcTemplateObject.queryForInt
        (SQL);
142
143     String SQL2 = "select * from accounts where
        accountID = ?";
144
145     UserProfile prof = jdbcTemplateObject.
        queryForObject(SQL2, new Object[]{accountID
        }, new UserProfileMapper());
146
147     return prof;
148 }
149
150
151 public UserProfile getUser(int id) {
152
153     jdbcTemplateObject = new JdbcTemplate(
        dataSource);
154
155     String SQL2 = "select * from accounts where
        accountID = ?";
156
157     UserProfile prof = jdbcTemplateObject.
        queryForObject(SQL2, new Object[]{id}, new
        UserProfileMapper());
158
        return prof;
159     }
160
161     public UserProfile getRejectedUser(int id) {
162
163         jdbcTemplateObject = new JdbcTemplate(
164             dataSource);
165
166         String SQL2 = "select * from registrations
167             where regID = ?";
168
169         UserProfile prof = jdbcTemplateObject.
170             queryForObject(SQL2, new Object[]{id}, new
171             UserProfileMapper());
172
173         return prof;
174     }
175
176     public List<UserProfile> listAllUsers(int
177         classify) {
178
179         jdbcTemplateObject = new JdbcTemplate(
180             dataSource);
181         String SQL = "";
182         if(classify != 5)
183             SQL = "select * from accounts where
184                 userTypeInt < '"+ classify +"'";
185         else
186             SQL = "select * from accounts where
187                 userTypeInt < '"+ classify +"' OR
188                 userTypeInt = 6";
189
190         List<UserProfile> allProf = jdbcTemplateObject.
191             query(SQL,
192             new UserProfileMapper());
193
194         return allProf;
195     }
196
197     public boolean deactivateUser(int id, String
198         username) {
199
200         jdbcTemplateObject = new JdbcTemplate(
201             dataSource);
202
203         String SQL = "UPDATE accounts SET status = 0
204             where accountID = ?";
205
206         jdbcTemplateObject.update(SQL, id);
207
208         SQL = "UPDATE users SET authorities = '
209             ROLE_NONE' where accountID = ?";
210
211         jdbcTemplateObject.update(SQL, id);
212
213         DateFormat dateFormat = new SimpleDateFormat("
214             yyyy-MM-dd");
215         Date date = new Date();
216
217         DateFormat timeFormat = new SimpleDateFormat("
218             HH:mm:ss");
219         Date time =new Date();
220
221         SQL = "select accountID from users where
222             username ='"+ username + "'";
223
224         int accountID = jdbcTemplateObject.queryForInt
225             (SQL);
226
227         //String action = "Deactivated a user.";
228         //SQL = "insert into userlogs (accountID,
229             action, date, time) values (?, ?, ?, ?)";
230         //jdbcTemplateObject.update(SQL, new Object[]{
231             accountID, action, dateFormat.format(date),
232             dateFormat.format(time)});
233
234         return true;
235     }
236
237     public boolean activateUser(int id, String
238         username) {
239
240         jdbcTemplateObject = new JdbcTemplate(
241             dataSource);
242
243         String SQL = "UPDATE accounts SET status = 1
244             where accountID = ?";
245
246         jdbcTemplateObject.update(SQL, id);
247
248         SQL = "SELECT userTypeInt FROM accounts WHERE
249             accountID = '" + id + "'";
250
251         int x = jdbcTemplateObject.queryForInt(SQL);
252
253         String authorities = "";
254         if(x == 0)
255             authorities = "ROLE_BHWorker";
256         else if(x == 1)

```

```

233     authorities = "ROLE_MCHWorker";
234     else if(x == 2)
235     authorities = "ROLE_MCOfficial";
236     else if(x == 3)
237     authorities = "ROLE_PHWorker";
238     else if(x == 4)
239     authorities = "ROLE_PHOfficial";
240     else if(x == 5)
241     authorities = "ROLE_NHOfficial";
242     else if(x == 6)
243     authorities = "ROLE_Researcher";
244     else
245     authorities = "ROLE_USER";
246
247     SQL = "UPDATE users SET authorities = '" +
        authorities + "' WHERE accountID = '" + id
        + "'";
248
249     jdbcTemplateObject.update(SQL);
250
251     DateFormat dateFormat = new SimpleDateFormat("
252     yyyy-MM-dd");
253     Date date = new Date();
254
255     DateFormat timeFormat = new SimpleDateFormat("
256     HH:mm:ss");
257     Date time =new Date();
258
259     SQL = "select accountID from users where
        username='"+ username + "'";
260
261     int accountID = jdbcTemplateObject.queryForInt
        (SQL);
262
263     //String action = "Activated a user.";
264     //SQL = "insert into userlogs (accountID,
        action, date, time) values (?, ?, ?, ?)";
265     //jdbcTemplateObject.update(SQL, new Object[]{
        accountID, action, dateFormat.format(date),
        dateFormat.format(time)});
266
267     return true;
268 }
269 public List<UserLogs> allLogs() throws
        Exception {
270
271     jdbcTemplateObject = new JdbcTemplate(
        dataSource);
272     Connection dbConnect = getMySQLConnection();
273     PreparedStatement pSt = null;
274
275     String SQL = "SELECT * FROM userlogs";
276     pSt = (PreparedStatement) dbConnect.
        prepareStatement(SQL);
277     ResultSet rs = pSt.executeQuery();
278
279     List<UserLogs> logs = new ArrayList<UserLogs
        >();
280
281     while(rs.next()) {
282         UserLogs log = new UserLogs();
283         log.setAccountID(rs.getInt("accountID"));
284         log.setAction(rs.getString("action"));
285         log.setDate(rs.getString("date"));
286         log.setTime(rs.getString("time"));
287         log.setRecordID(rs.getInt("recordID"));
288         log.setRecordtable(rs.getString("recordtable
        "));
289         log.setEditStatus(rs.getInt("editStatus"));
290         log.setRecordtype(rs.getInt("recordtype"));
291         logs.add(log);
292     }
293
294     pSt.close();
295     dbConnect.close();
296
297     return logs;
298
299 }
300
301 public void insertReportLog(String username) {
302     jdbcTemplateObject = new JdbcTemplate(
        dataSource);
303
304     String SQL = "select accountID from users
        where username ='"+ username + "'";
305
306     int accountID = jdbcTemplateObject.queryForInt
        (SQL);
307
308     DateFormat dateFormat = new SimpleDateFormat("
309     yyyy-MM-dd");
310     Date date = new Date();
311
312     DateFormat timeFormat = new SimpleDateFormat("
313     HH:mm:ss");
314     Date time =new Date();
315
316     String action = "Generated report.";
317     SQL = "insert into userlogs (accountID, action
        , date, time, recordID, recordtable,
        recordtype) values (?, ?, ?, ?, ?, ?)";
318
319     jdbcTemplateObject.update(SQL, new Object[]{
        accountID, action, dateFormat.format(date),
        timeFormat.format(time), "", "", ""});
320
321 }
322
323 public void insertDataSetLog(String username) {
324     jdbcTemplateObject = new JdbcTemplate(
        dataSource);
325
326     String SQL = "select accountID from users
        where username ='"+ username + "'";
327
328     int accountID = jdbcTemplateObject.queryForInt
        (SQL);
329
330     DateFormat dateFormat = new SimpleDateFormat("
331     yyyy-MM-dd");
332     Date date = new Date();
333
334     DateFormat timeFormat = new SimpleDateFormat("
335     HH:mm:ss");
336     Date time =new Date();
337
338     String action = "Exported dataset.";
339     SQL = "insert into userlogs (accountID, action
        , date, time, recordID, recordtable,
        recordtype) values (?, ?, ?, ?, ?, ?)";
340
341     jdbcTemplateObject.update(SQL, new Object[]{
        accountID, action, dateFormat.format(date),
        timeFormat.format(time), "", "", ""});
342
343 }
344
345 public static Connection getMySQLConnection()
        throws Exception {
346     Connection conn = null;
347     String driver = "com.mysql.jdbc.Driver";
348     String url = "jdbc:mysql://localhost:3306/vbd?
        zeroDateTimeBehavior=convertToNull";
349     String username = "root";
350     String password = "";
351
352     Class.forName(driver);
353
354     conn = (Connection) DriverManager.
        getConnection(url, username, password);
355     return conn;
356 }
357 }

```

Listing 13: BarangayRecordsMapper.java

```

1 package mapper;
2
3 import java.sql.ResultSet;
4 import java.sql.SQLException;
5
6 import org.springframework.jdbc.core.RowMapper;
7
8 import objects.BarangayRecords;
9
10 public class BarangayRecordsMapper implements
        RowMapper {
11
12     public BarangayRecords mapRow(ResultSet rs, int
        rowNum) throws SQLException {
13
14         BarangayRecords brgy = new BarangayRecords();
15         brgy.setAccID(rs.getInt("accountID"));
16         brgy.setRegion(rs.getString("region"));
17         brgy.setProvince(rs.getString("province"));
18         brgy.setMuncity(rs.getString("munCity"));
19         brgy.setBarangay(rs.getString("barangay"));
20         brgy.setName(rs.getString("name"));
21         brgy.setAge(rs.getString("age"));
22         brgy.setGender(rs.getString("gender"));
23         brgy.setDeceased(rs.getString("deceased"));
24         brgy.setDate(rs.getString("date"));
25         brgy.setNumEntries(rs.getInt("numEntries"))
        ;
26         brgy.setDiseaseID(rs.getInt("diseaseID"));
27         brgy.setSymptoms(rs.getString("symptoms"));
28         brgy.setDateApproved(rs.getString("
        dateApproved"));

```

```

29     brgy.setStatus(rs.getInt("status"));
30     brgy.setHealthworker(rs.getString("
healthworker"));
31     brgy.setDisease(rs.getInt("diseaseID"));
32     brgy.setRecordID(rs.getInt("recordID"));
33     try {
34         brgy.setMuncityHW(rs.getString("
muncityhealthworker"));
35         brgy.setDateAppr(rs.getString("
dateApproved2"));

```

Listing 14: DiseaseMapper.java

```

1 package mapper;
2
3 import java.sql.ResultSet;
4 import java.sql.SQLException;
5
6 import org.springframework.jdbc.core.RowMapper;
7
8 import objects.Diseases;
9

```

Listing 15: PatientsFormMapper.java

```

1 package mapper;
2
3 import java.sql.ResultSet;
4 import java.sql.SQLException;
5
6 import org.springframework.jdbc.core.RowMapper;
7
8 import objects.PatientsForm;
9
10 public class PatientsFormMapper implements
    RowMapper {
11
12     public PatientsForm mapRow(ResultSet rs, int
        rowNum) throws SQLException {
13         PatientsForm patient = new PatientsForm();
14
15         patient.setName(rs.getString("name"));
16         patient.setDateofbirth(rs.getString("birthDate
            "));
17         patient.setGender(rs.getString("gender"));
18         patient.setAddress(rs.getString("address"));
19
20         patient.setAdmitted(rs.getString("admitted"));
21         patient.setDateAdmitted(rs.getString("
            dateAdmitted"));
22         patient.setOutcome(rs.getString("outcome"));
23         patient.setDateDied(rs.getString("dateDied"));
24
25         if(rs.getInt("diseaseID") == 0)
26             patient.setDisease("Chikungunya");
27         else if(rs.getInt("diseaseID") == 1)
28             patient.setDisease("Dengue");
29         else if(rs.getInt("diseaseID") == 2)
30             patient.setDisease("Malaria");
31
32         patient.setDateOnset(rs.getString("dateOnset")
            );
33         patient.setSymptoms(rs.getString("symptoms"));
34
35         patient.setCaseType("");
36         patient.setHospital(rs.getString("hospital"));
37         patient.setHospitalType(rs.getString("
            hospitalType"));
38         patient.setHospitalAddr(rs.getString("
            hospitalAddr"));
39
40         patient.setDateSubmitted(rs.getString("
            dateSubmitted"));
41         patient.setDateApproved(rs.getString("

```

Listing 16: RegistrationMapper.java

```

1 package mapper;
2
3 import java.sql.ResultSet;
4 import java.sql.SQLException;
5
6 import org.springframework.jdbc.core.RowMapper;
7
8 import objects.Registrants;
9
10 public class RegistrationMapper implements
    RowMapper {
11
12     public Registrants mapRow(ResultSet rs, int
        rowNum) throws SQLException {
13         Registrants regs = new Registrants();
14         regs.setAccountID(rs.getInt("regID"));
15         regs.setFirstname(rs.getString("firstName"));
16         regs.setMiddlename(rs.getString("middleName")
            );

```

```

36         brgy.setProvinceHW(rs.getString("
provincehealthworker"));
37     }
38     catch(Exception e) {
39     }
40 }
41
42     return brgy;
43 }
44 }

```

```

10 public class DiseaseMapper implements RowMapper<
    Diseases>{
11     public Diseases mapRow(ResultSet rs, int rowNum
        ) throws SQLException {
12         Diseases disease = new Diseases();
13         disease.setDiseaseID(rs.getInt("diseaseID"));
14         disease.setDiseaseName(rs.getString("
            diseaseName"));
15         return disease;
16     }
17 }

```

```

42         dateApproved"));
43     try {
44         patient.setDateApproved2(rs.getString("
            dateApproved2"));
45     }
46     catch(Exception e) {
47     }
48 }
49
50     try {
51         patient.setEditedBy(rs.getString("editedBy")
            );
52     } catch(Exception fas) {}
53     try {
54         patient.setDateEdited(rs.getString("dateEdited
            "));
55     } catch(Exception fas2) {}
56
57         patient.setBarangay(rs.getString("barangay"));
58         patient.setMuncity(rs.getString("muncity"));
59         patient.setProvince(rs.getString("province"));
60         patient.setRegion(rs.getString("region"));
61
62         patient.setRecordID(rs.getInt("recordID"));
63         patient.setHealthworker(rs.getString("
            healthworker"));
64         patient.setMuncityHW(rs.getString("
            muncityhealthworker"));
65
66     try {
67         patient.setProvinceHW(rs.getString("
            provincehealthworker"));
68     }
69     catch(Exception e) {
70     }
71 }
72     try {
73         patient.setStatus(rs.getInt("status"));
74     } catch(Exception fas3) {}
75     try {
76         patient.setEditStatus(rs.getInt("editStatus")
            );
77     } catch(Exception fas4) {}
78
79     return patient;
80 }
81
82
83 }

```

```

17         regs.setSurname(rs.getString("surName"));
18         regs.setUsername(rs.getString("username"));
19         regs.setGender(rs.getString("gender"));
20         regs.setAddress(rs.getString("address"));
21         regs.setUserType(rs.getString("userType"));
22         regs.setRegion(rs.getString("region"));
23         regs.setProvince(rs.getString("province"));
24         regs.setMuncity(rs.getString("munCity"));
25         regs.setBarangay(rs.getString("barangay"));
26         regs.setInstitution(rs.getString("
            institution"));
27         regs.setInstitutionAddr(rs.getString("
            instAddr"));
28         regs.setEmail(rs.getString("email"));
29         regs.setNumber(rs.getString("number"));
30         regs.setStatus(rs.getInt("status"));
31
32     return regs;
33 }
34 }
35 }

```

Listing 17: UserProfileMapper.java

```

1 package mapper;
2
3 import java.sql.ResultSet;
4 import java.sql.SQLException;
5
6 import org.springframework.jdbc.core.RowMapper;
7
8 import objects.UserProfile;
9
10 public class UserProfileMapper implements
    RowMapper {
11
12     public UserProfile mapRow(ResultSet rs, int
        rowNum) throws SQLException {
13         UserProfile user = new UserProfile();
14         try {
15             user.setAccID(rs.getInt("regID"));
16         } catch(Exception e) {}
17         try {
18             user.setAccID(rs.getInt("accountID"));
19         } catch(Exception e) {}
20

```

```

21     user.setFirstname(rs.getString("firstName"));
22     user.setMiddlename(rs.getString("middleName"));
    ;
23     user.setSurname(rs.getString("surName"));
24     user.setUsername(rs.getString("username"));
25     user.setGender(rs.getString("gender"));
26     user.setAddress(rs.getString("address"));
27     user.setUserType(rs.getString("userType"));
28     user.setRegion(rs.getString("region"));
29     user.setProvince(rs.getString("province"));
30     user.setMuncity(rs.getString("munCity"));
31     user.setBarangay(rs.getString("barangay"));
32     user.setInstitution(rs.getString("
        institution"));
33     user.setInstitutionAddr(rs.getString("
        instAddr"));
34     user.setEmail(rs.getString("email"));
35     user.setNumber(rs.getString("number"));
36     user.setStatus(rs.getInt("status"));
37     return user;
38 }
39
40 }

```

Listing 18: Alerts.java

```

1 package objects;
2
3 public class Alerts {
4     private String barangay;
5     private String muncity;
6     private String province;
7     private String region;
8     private String date;
9     private String disease;
10    private int diseaseID;
11    private double outbreak;
12    private int status;
13
14    public String getDisease() {
15        return disease;
16    }
17    public void setDisease(String disease) {
18        this.disease = disease;
19    }
20    public int getDiseaseID() {
21        return diseaseID;
22    }
23    public void setDiseaseID(int diseaseID) {
24        this.diseaseID = diseaseID;
25    }
26    public String getBarangay() {
27        return barangay;
28    }
29    public void setBarangay(String barangay) {
30        this.barangay = barangay;
31    }
32    public String getMuncity() {
33        return muncity;
34    }

```

```

35    public void setMuncity(String muncity) {
36        this.muncity = muncity;
37    }
38    public String getProvince() {
39        return province;
40    }
41    public void setProvince(String province) {
42        this.province = province;
43    }
44    public String getRegion() {
45        return region;
46    }
47    public void setRegion(String region) {
48        this.region = region;
49    }
50    public String getDate() {
51        return date;
52    }
53    public void setDate(String date) {
54        this.date = date;
55    }
56    public double getOutbreak() {
57        return outbreak;
58    }
59    public void setOutbreak(double outbreak) {
60        this.outbreak = outbreak;
61    }
62    public int getStatus() {
63        return status;
64    }
65    public void setStatus(int status) {
66        this.status = status;
67    }
68
69
70 }

```

Listing 19: BarangayRecords.java

```

1 package objects;
2
3 import java.text.DateFormat;
4 import java.text.SimpleDateFormat;
5
6 public class BarangayRecords {
7
8     private String barangay;
9     private String muncity;
10    private String province;
11    private String region;
12    private String name;
13    private String age;
14    private String gender;
15    private String deceased;
16    private String date;
17    private String dateApproved;
18    private String dateAppr;
19    private int accID;
20    private int numEntries;
21    private int diseaseID;
22    private int status;
23    private int recordID;
24    private String healthworker;
25    private String muncityHW;
26    private String provinceHW;
27    private String disease;
28    private String symptoms;
29
30    public int getAccID() {
31        return accID;
32    }
33    public void setAccID(int accID) {
34        this.accID = accID;

```

```

35    }
36    public String getBarangay() {
37        return barangay;
38    }
39    public void setBarangay(String barangay) {
40        this.barangay = barangay;
41    }
42    public String getMuncity() {
43        return muncity;
44    }
45    public void setMuncity(String muncity) {
46        this.muncity = muncity;
47    }
48    public String getProvince() {
49        return province;
50    }
51    public void setProvince(String province) {
52        this.province = province;
53    }
54    public String getRegion() {
55        return region;
56    }
57    public void setRegion(String region) {
58        this.region = region;
59    }
60    public String getName() {
61        return name;
62    }
63    public void setName(String name) {
64        this.name = name;
65    }
66    public String getAge() {
67        return age;
68    }
69    public void setAge(String age) {

```

```

70     this.age = age;
71 }
72 public String getGender() {
73     return gender;
74 }
75 public void setGender(String gender) {
76     this.gender = gender;
77 }
78 public String getDeceased() {
79     return deceased;
80 }
81 public void setDeceased(String deceased) {
82     this.deceased = deceased;
83 }
84 public String getDate() {
85     return date;
86 }
87 public void setDate(String date) {
88     this.date = date;
89 }
90 }
91 public int getNumEntries() {
92     return numEntries;
93 }
94 public void setNumEntries(int numEntries) {
95     this.numEntries = numEntries;
96 }
97 public int getDiseaseID() {
98     return diseaseID;
99 }
100 public void setDiseaseID(int diseaseID) {
101     this.diseaseID = diseaseID;
102 }
103 public String getSymptoms() {
104     return symptoms;
105 }
106 public void setSymptoms(String symptoms) {
107     this.symptoms = symptoms;
108 }
109 public String getDisease() {
110     return disease;
111 }
112 public void setDisease(int disease) {
113     if(diseaseID == 0)
114         this.disease = "Chikungunya";
115     else if(diseaseID == 1)
116         this.disease = "Dengue";
117     else if(diseaseID == 2)
118         this.disease = "Malaria";
119 }
120 public String getDateApproved() {
121     return dateApproved;
122 }
123 public void setDateApproved(String dateApproved) {
124     this.dateApproved = dateApproved;
125 }
126 public String getDateAppr() {
127     return dateAppr;
128 }
129 public void setDateAppr(String dateAppr) {
130     this.dateAppr = dateAppr;
131 }
132 public int getStatus() {
133     return status;
134 }
135 public void setStatus(int status) {
136     this.status = status;
137 }
138 public String getHealthworker() {
139     return healthworker;
140 }
141 public void setHealthworker(String healthworker) {
142     this.healthworker = healthworker;
143 }
144 public int getRecordID() {
145     return recordID;
146 }
147 public void setRecordID(int recordID) {
148     this.recordID = recordID;
149 }
150 public String getMuncityHW() {
151     return muncityHW;
152 }
153 public void setMuncityHW(String muncityHW) {
154     this.muncityHW = muncityHW;
155 }
156 public String getProvinceHW() {
157     return provinceHW;
158 }
159 public void setProvinceHW(String provinceHW) {
160     this.provinceHW = provinceHW;
161 }
162 }
163 }

```

Listing 20: Diseases.java

```

1 package objects;
2
3 public class Diseases {
4
5     private String diseaseName;
6     private int diseaseID;
7
8     public void setDiseaseName(String name){
9         this.diseaseName = name;
10    }
11

```

```

12     public String getDiseaseName(){
13         return diseaseName;
14     }
15
16     public void setDiseaseID(int id){
17         this.diseaseID = id;
18     }
19
20     public int getDiseaseID(){
21         return diseaseID;
22     }
23
24 }

```

Listing 21: PatientsForm.java

```

1 package objects;
2
3 import java.text.DateFormat;
4 import java.text.SimpleDateFormat;
5 import java.util.Date;
6 import java.util.List;
7
8 import javax.validation.constraints.Pattern;
9
10 import org.hibernate.validator.constraints.
11     NotBlank;
11 import org.hibernate.validator.constraints.
12     NotEmpty;
12 import org.hibernate.validator.constraints.Range
13     ;
14 public class PatientsForm {
15
16     private String name;
17
18     @NotEmpty
19     private String address;
20
21     private int age;
22
23     @NotEmpty
24     private String dateofbirth;
25
26     @NotEmpty
27     private String gender;
28
29     @NotEmpty

```

```

30     private String admitted;
31
32     @NotEmpty
33     private String dateOnset;
34
35     private String dateAdmitted;
36
37     @NotEmpty
38     private String disease;
39
40     @NotEmpty
41     private String symptoms;
42
43     @NotEmpty
44     private String outcome;
45
46     private String dateDied;
47
48     private String caseType;
49     private String hospital;
50     private String hospitalType;
51     private String hospitalAddr;
52
53     private String region;
54     private String province;
55
56     @NotEmpty
57     private String muncity;
58
59     @NotEmpty
60     private String barangay;
61
62     private String dateSubmitted;
63     private String dateApproved;

```

```

64 private String dateApproved2;
65 private String dateEdited;
66 private String editedBy;
67
68 private String healthworker;
69 private String muncityHW;
70 private String provinceHW;
71 private int recordID;
72 private int status;
73 private int editStatus;
74
75 public PatientsForm() {
76
77 }
78 public String getName() {
79 return name;
80 }
81 public void setName(String name) {
82 this.name = name;
83 }
84 public String getAddress() {
85 return address;
86 }
87 public void setAddress(String address) {
88 this.address = address;
89 }
90 public int getAge() {
91 return age;
92 }
93 public String getDateofbirth() {
94 return dateofbirth;
95 }
96 public void setDateofbirth(String dateofbirth)
97 {
98 this.dateofbirth = dateofbirth;
99 try {
100 String birthday[] = dateofbirth.split("-");
101 DateFormat d = new SimpleDateFormat("yyyy-MM-
102 dd");
103 Date da = new Date();
104 String toDay[] = d.format(da).split("-");
105 if(Integer.parseInt(birthday[1]) < Integer.
106 parseInt(toDay[1]))
107 this.age = Integer.parseInt(toDay[0]) -
108 Integer.parseInt(birthday[0]) - 1;
109 else if(Integer.parseInt(birthday[1]) >
110 Integer.parseInt(toDay[1]))
111 this.age = Integer.parseInt(toDay[0]) -
112 Integer.parseInt(birthday[0]);
113 else {
114 if(Integer.parseInt(birthday[2]) < Integer.
115 parseInt(toDay[2]))
116 this.age = Integer.parseInt(toDay[0]) -
117 Integer.parseInt(birthday[0]) - 1;
118 else
119 this.age = Integer.parseInt(toDay[0]) -
120 Integer.parseInt(birthday[0]);
121 }
122 }
123 catch(Exception e) {
124 }
125 }
126 public String getGender() {
127 return gender;
128 }
129 public void setGender(String gender) {
130 this.gender = gender;
131 }
132 public String getAdmitted() {
133 return admitted;
134 }
135 public void setAdmitted(String admitted) {
136 this.admitted = admitted;
137 }
138 public String getDateOnset() {
139 return dateOnset;
140 }
141 public void setDateOnset(String dateOnset) {
142 this.dateOnset = dateOnset;
143 }
144 public String getDateAdmitted() {
145 return dateAdmitted;
146 }
147 public void setDateAdmitted(String dateAdmitted)
148 {
149 this.dateAdmitted = dateAdmitted;
150 }
151 public String getDateDied() {
152 return dateDied;
153 }
154 public void setDateDied(String dateDied) {
155 this.dateDied = dateDied;
156 }
157 public String getDisease() {
158 return disease;
159 }
160 public void setDisease(String disease) {
161 this.disease = disease;
162 }
163 }
164
165 public String getCaseType() {
166 return caseType;
167 }
168 public void setCaseType(String caseType) {
169 this.caseType = caseType;
170 }
171 public String getSymptoms() {
172 return symptoms;
173 }
174 public void setSymptoms(String symptoms) {
175 this.symptoms = symptoms;
176 }
177 public String getOutcome() {
178 return outcome;
179 }
180 public void setOutcome(String outcome) {
181 this.outcome = outcome;
182 }
183 public String getHospital() {
184 return hospital;
185 }
186 public void setHospital(String hospital) {
187 this.hospital = hospital;
188 }
189 public String getHospitalType() {
190 return hospitalType;
191 }
192 public void setHospitalType(String hospitalType)
193 {
194 this.hospitalType = hospitalType;
195 }
196 public String getHospitalAddr() {
197 return hospitalAddr;
198 }
199 public void setHospitalAddr(String hospitalAddr)
200 {
201 this.hospitalAddr = hospitalAddr;
202 }
203 public String getRegion() {
204 return region;
205 }
206 public void setRegion(String region) {
207 this.region = region;
208 }
209 public String getProvince() {
210 return province;
211 }
212 public void setProvince(String province) {
213 this.province = province;
214 }
215 public String getMuncity() {
216 return muncity;
217 }
218 public void setMuncity(String muncity) {
219 this.muncity = muncity;
220 }
221 public String getBarangay() {
222 return barangay;
223 }
224 public void setBarangay(String barangay) {
225 this.barangay = barangay;
226 }
227 public String getDateSubmitted() {
228 return dateSubmitted;
229 }
230 public void setDateSubmitted(String
231 dateSubmitted) {
232 this.dateSubmitted = dateSubmitted;
233 }
234 public String getHealthworker() {
235 return healthworker;
236 }
237 public void setHealthworker(String healthworker)
238 {
239 this.healthworker = healthworker;
240 }
241 public String getMuncityHW() {
242 return muncityHW;
243 }
244 public void setMuncityHW(String muncityHW) {
245 this.muncityHW = muncityHW;
246 }
247 public int getRecordID() {
248 return recordID;
249 }
250 public void setRecordID(int recordID) {
251 this.recordID = recordID;
252 }
253 public String getDateApproved() {
254 return dateApproved;
255 }
256 public void setDateApproved(String dateApproved)
257 {
258 this.dateApproved = dateApproved;
259 }
260 public String getDateEdited() {
261 return dateEdited;
262 }
263 }

```

```

247 public void setDateEdited(String dateEdited) {
248     this.dateEdited = dateEdited;
249 }
250 public String getEdittedBy() {
251     return edittedBy;
252 }
253 public void setEdittedBy(String edittedBy) {
254     this.edittedBy = edittedBy;
255 }
256 public String getDateApproved2() {
257     return dateApproved2;
258 }
259 public void setDateApproved2(String
    dateApproved2) {
260     this.dateApproved2 = dateApproved2;
261 }
262 public String getProvinceHW() {
263     return provinceHW;
264 }
265 public void setProvinceHW(String provinceHW) {
266     this.provinceHW = provinceHW;
267 }
268 public int getStatus() {
269     return status;
270 }
271 public void setStatus(int status) {
272     this.status = status;
273 }
274 public int getEditStatus() {
275     return editStatus;
276 }
277 public void setEditStatus(int editStatus) {
278     this.editStatus = editStatus;
279 }
280 }

```

Listing 22: Registrants.java

```

1 package objects;
2
3 public class Registrants {
4
5     private int accountID;
6     private String firstname;
7     private String middlename;
8     private String surname;
9     private String gender;
10    private String address;
11    private String userType;
12    private int userTypeInt;
13    private String username;
14    private String region;
15    private String province;
16    private String muncity;
17    private String barangay;
18    private String institution;
19    private String institutionAddr;
20    private String email;
21    private String number;
22    private int status;
23
24    public int getAccountID() {
25        return accountID;
26    }
27    public void setAccountID(int accountID) {
28        this.accountID = accountID;
29    }
30    public String getFirstname() {
31        return firstname;
32    }
33    public void setFirstname(String firstname) {
34        this.firstname = firstname;
35    }
36    public String getMiddlename() {
37        return middlename;
38    }
39    public void setMiddlename(String middlename) {
40        this.middlename = middlename;
41    }
42    public String getSurname() {
43        return surname;
44    }
45    public void setSurname(String surname) {
46        this.surname = surname;
47    }
48    public String getUsername() {
49        return username;
50    }
51    public void setUsername(String username) {
52        this.username = username;
53    }
54    public String getGender() {
55        return gender;
56    }
57    public void setGender(String gender) {
58        this.gender = gender;
59    }
60    public String getAddress() {
61        return address;
62    }
63    public void setAddress(String address) {
64        this.address = address;
65    }
66    public String getUserType() {
67        return userType;
68    }
69    public int getUserTypeInt() {
70        return userTypeInt;
71    }
72    public void setUserType(String userType) {
73        this.userType = userType;
74        if(userType.equalsIgnoreCase("Barangay Health
    Worker"))
75            this.userTypeInt = 0;
76        else if(userType.equalsIgnoreCase("Municipal/
    City Health Worker"))
77            this.userTypeInt = 1;
78        else if(userType.equalsIgnoreCase("Municipal/
    City Health Official"))
79            this.userTypeInt = 2;
80        else if(userType.equalsIgnoreCase("Provincial
    Health Worker"))
81            this.userTypeInt = 3;
82        else if(userType.equalsIgnoreCase("Provincial
    Health Official"))
83            this.userTypeInt = 4;
84        else if(userType.equalsIgnoreCase("National
    Health Official"))
85            this.userTypeInt = 5;
86        else if(userType.equalsIgnoreCase("Researcher
    "))
87            this.userTypeInt = 6;
88        else if(userType.equalsIgnoreCase("
    Administrator"))
89            this.userTypeInt = 7;
90        else
91            this.userTypeInt = 8;
92    }
93    public String getRegion() {
94        return region;
95    }
96    public void setRegion(String region) {
97        this.region = region;
98    }
99    public String getProvince() {
100        return province;
101    }
102    public void setProvince(String province) {
103        this.province = province;
104    }
105    public String getMuncity() {
106        return muncity;
107    }
108    public void setMuncity(String muncity) {
109        this.muncity = muncity;
110    }
111    public String getBarangay() {
112        return barangay;
113    }
114    public void setBarangay(String barangay) {
115        this.barangay = barangay;
116    }
117    public String getInstitution() {
118        return institution;
119    }
120    public void setInstitution(String institution)
    {
121        this.institution = institution;
122    }
123    public String getInstitutionAddr() {
124        return institutionAddr;
125    }
126    public void setInstitutionAddr(String
    institutionAddr) {
127        this.institutionAddr = institutionAddr;
128    }
129    public String getEmail() {
130        return email;
131    }
132    public void setEmail(String email) {
133        this.email = email;
134    }
135    public String getNumber() {
136        return number;
137    }
138    public void setNumber(String number) {
139        this.number = number;
140    }
141    public int getStatus() {
142        return status;
143    }
144    public void setStatus(int status) {
145        this.status = status;
146    }
147 }

```

Listing 23: UserChange.java

```

1 package objects;
2
3 import org.hibernate.validator.constraints.
    Length;
4 import org.hibernate.validator.constraints.
    NotBlank;
5 import org.hibernate.validator.constraints.
    NotEmpty;
6
7 public class UserChange {
8
9     private int accountID;
10
11     private String accountName;
12
13     @NotEmpty
14     private String accountPassword;
15
16     @NotBlank
17     @Length(min = 8)
18     private String accountPassword2;
19
20     @NotBlank
21     @Length(min = 8)
22     private String accountPassword3;
23     /*
24     @NotEmpty
25     private String oldAccPassWord;
26     */
27     public int getAccountID() {
28         return accountID;
29     }
30     public void setAccountID(int accountID) {
31         this.accountID = accountID;
32     }
33
34     public String getAccountName() {
35         return accountName;
36     }
37     public void setAccountName(String accountName)
38     {
39         this.accountName = accountName;
40     }
41     public String getAccountPassword() {
42         return accountPassword;
43     }
44     public void setAccountPassword(String
45         accountPassword) {
46         this.accountPassword = accountPassword;
47     }
48     public String getAccountPassword2() {
49         return accountPassword2;
50     }
51     public void setAccountPassword2(String
52         accountPassword2) {
53         this.accountPassword2 = accountPassword2;
54     }
55     public String getAccountPassword3() {
56         return accountPassword3;
57     }
58     public void setAccountPassword3(String
59         accountPassword3) {
60         this.accountPassword3 = accountPassword3;
61     }

```

Listing 24: UserProfile.java

```

1 package objects;
2
3 public class UserProfile {
4
5     private int accID;
6     private int status;
7     private String username;
8     private String firstname;
9     private String middlename;
10    private String surname;
11    private String gender;
12    private String address;
13    private String userType;
14    private int userTypeInt;
15    private String region;
16    private String province;
17    private String muncity;
18    private String barangay;
19    private String institution;
20    private String institutionAddr;
21    private String email;
22    private String number;
23
24
25    public int getAccID() {
26        return accID;
27    }
28    public void setAccID(int accID) {
29        this.accID = accID;
30    }
31
32    public String getFirstname() {
33        return firstname;
34    }
35    public void setFirstname(String firstname) {
36        this.firstname = firstname;
37    }
38    public String getMiddlename() {
39        return middlename;
40    }
41    public void setMiddlename(String middlename) {
42        this.middlename = middlename;
43    }
44    public String getSurname() {
45        return surname;
46    }
47    public void setSurname(String surname) {
48        this.surname = surname;
49    }
50    public String getUsername() {
51        return username;
52    }
53    public void setUsername(String username) {
54        this.username = username;
55    }
56    public String getGender() {
57        return gender;
58    }
59    public void setGender(String gender) {
60        this.gender = gender;
61    }
62    public String getAddress() {
63        return address;
64    }
65    public void setAddress(String address) {
66        this.address = address;
67    }
68    public String getUserType() {
69        return userType;
70    }
71    public int getUserTypeInt() {
72        return userTypeInt;
73    }
74    public int getStatus() {
75        return status;
76    }
77    public void setStatus(int status) {
78        this.status = status;
79    }
80    public void setUserType(String userType) {
81        this.userType = userType;
82        if(userType.equalsIgnoreCase("Barangay Health
83            Worker"))
84            this.userTypeInt = 0;
85        else if(userType.equalsIgnoreCase("Municipal/
86            City Health Worker"))
87            this.userTypeInt = 1;
88        else if(userType.equalsIgnoreCase("Municipal/
89            City Health Official"))
90            this.userTypeInt = 2;
91        else if(userType.equalsIgnoreCase("Provincial
92            Health Worker"))
93            this.userTypeInt = 3;
94        else if(userType.equalsIgnoreCase("Provincial
95            Health Official"))
96            this.userTypeInt = 4;
97        else if(userType.equalsIgnoreCase("National
98            Health Official"))
99            this.userTypeInt = 5;
100       else if(userType.equalsIgnoreCase("Researcher
101           "))
102           this.userTypeInt = 6;
103       else if(userType.equalsIgnoreCase("
104           Administrator"))
105           this.userTypeInt = 7;
106       else
107           this.userTypeInt = 8;
108   }
109   public String getRegion() {
110       return region;
111   }
112   public void setRegion(String region) {
113       this.region = region;
114   }
115   public String getProvince() {
116       return province;
117   }
118   public void setProvince(String province) {
119       this.province = province;
120   }
121   public String getMuncity() {
122       return muncity;
123   }

```



```

115 }
116 public void setMuncity(String muncity) {
117     this.muncity = muncity;
118 }
119 public String getBarangay() {
120     return barangay;
121 }
122 public void setBarangay(String barangay) {
123     this.barangay = barangay;
124 }
125 public String getInstitution() {
126     return institution;
127 }
128 public void setInstitution(String institution)
129     {
130     this.institution = institution;
131 }
132 public String getInstitutionAddr() {
133     return institutionAddr;
134 }
135 public void setInstitutionAddr(String
136     institutionAddr) {
137     this.institutionAddr = institutionAddr;
138 }
139 public String getEmail() {
140     return email;
141 }
142 public void setEmail(String email) {
143     this.email = email;
144 }
145 public String getNumber() {
146     return number;
147 }
148 public void setNumber(String number) {
149     this.number = number;
150 }

```

4. Utilities

Listing 25: SendEmail.java

```

1 package misc;
2
3 import java.util.Properties;
4
5 import javax.mail.Message;
6 import javax.mail.MessagingException;
7 import javax.mail.PasswordAuthentication;
8 import javax.mail.Session;
9 import javax.mail.Transport;
10 import javax.mail.internet.InternetAddress;
11 import javax.mail.internet.MimeMessage;
12
13 public class SendEmail {
14
15     public void sendmail(String recipient, String
16         username, String password, String userType,
17         boolean success) {
18         final String user = "vbdmsdev@gmail.com";
19         final String pass = "adminpass2014";
20
21         Properties props = new Properties();
22         props.put("mail.smtp.host", "smtp.gmail.com");
23         props.put("mail.smtp.socketFactory.port",
24             "465");
25         props.put("mail.smtp.socketFactory.class",
26             "javax.net.ssl.SSLSocketFactory");
27         props.put("mail.smtp.auth", "true");
28         props.put("mail.smtp.port", "465");
29
30         Session session = Session.getDefaultInstance(
31             props,
32             new javax.mail.Authenticator() {
33                 protected PasswordAuthentication
34                     getPasswordAuthentication() {
35                     return new PasswordAuthentication(user,
36                         pass);
37                 }
38             });
39         System.out.println("Username: " + user + "
40             Password: " + pass);
41         try {
42             Message message = new MimeMessage(
43                 session);
44             message.setFrom(new InternetAddress(
45                 user));
46             message.setRecipients(Message.
47                 RecipientType.TO,
48                 InternetAddress.parse(recipient)
49                 );
50             message.setSubject("Login
51                 Credentials!");
52             String body = "";
53             if(success) {
54                 body = "You are now successfully
55                     registered to Vector"
56                 + "--Borne Diseases Monitoring System as " +
57                     userType + "!\n\n"
58                 + "Here are your login credentials:\n"
59                 + "Username: " + username + "\n"
60                 + "Password: " + password + "\n"
61                 + "\n\nHave a nice day! :) ";
62             }
63             else {
64                 body = "We are to sorry to inform
65                     you that your registration has been turned
66                     down for some reasons."
67                 + " You are welcome to re-apply
68                     again some time. Thank you.";
69             }
70             message.setText(body);
71             Transport.send(message);
72             System.out.println("Done");
73         } catch (MessagingException e) {
74             throw new RuntimeException(e);
75         }
76     }
77 }

```

Listing 26: SendSMS.java

```

1 package misc;
2
3 import java.io.BufferedReader;
4 import java.io.InputStreamReader;
5 import java.io.OutputStreamWriter;
6 import java.net.URL;
7 import java.net.URLConnection;
8 import java.net.URLEncoder;
9 import java.util.Date;
10 import java.util.List;
11 import java.util.Properties;
12
13 import javax.mail.Message;
14 import javax.mail.Session;
15 import javax.mail.Transport;
16 import javax.mail.internet.InternetAddress;
17 import javax.mail.internet.MimeMessage;
18
19 import objects.UserProfile;
20
21 public class SendSMS {
22
23     public void msgsend(String disease, String
24         barangay, String muncity, String province,
25         List<? extends UserProfile> up) {
26         try {
27             String data = "";
28             String numb = "";
29             int count = 0;
30             for(UserProfile u : up) {
31                 if(count == up.size() - 1)
32                     numb += ("+63" + u.getNumber().
33                         substring(1));
34                 else
35                     numb += ("+63" + u.getNumber().
36                         substring(1) + ", ");
37                 count++;
38             }
39             System.out.println(numb);
40             //String cont = u.getNumber().
41             substring(1, u.getNumber().length() + 1);
42             data += "username=" + URLEncoder.
43                 encode("vbdmsdev", "ISO-8859-1");
44             data += "&password=" + URLEncoder.
45                 encode("adminpass2014", "ISO-8859-1");
46             data += "&message=" + URLEncoder.
47                 encode("VBDM Alert!\n Urgent health
48                     concern; \"\" +
49                     disease + "\" outbreak in Barangay
50                     \" + barangay + \", \" + muncity + \", \" +
51                     province + \". Please verify and
52                     report to other health authorities.", "ISO
53                     -8859-1");
54             data += "&want_report=1";
55         }
56     }
57 }

```

```

45         data += "&msisdn=" + numb;
46
47         URL url = new URL("http://bulksms.
vmsms.net:5667/eapi/submission/send_sms
/2/2.0");
48         URLConnection conn = url.
openConnection();
49         conn.setDoOutput(true);
50         OutputStreamWriter wr = new
OutputStreamWriter(conn.getOutputStream());
51         wr.write(data);
52         wr.flush();
53
54         BufferedReader rd = new
BufferedReader(new InputStreamReader(conn.

```

```

getInputStream());
String line;
55         while ((line = rd.readLine()) !=
56         null)
57             System.out.println(line);
58         wr.close();
59         rd.close();
60
61     } catch (Exception e) {
62         e.printStackTrace();
63     }
64 }
65 }
66 }

```

Listing 27: GenerateExcel.java

```

1 package pdf;
2
3 import java.io.ByteArrayOutputStream;
4 import java.util.List;
5
6 import objects.Patients;
7 import objects.PatientsForm;
8 import objects.Reports;
9
10 import org.apache.poi.hssf.usermodel.HSSFCell;
11 import org.apache.poi.hssf.usermodel.
HSSFCellStyle;
12 import org.apache.poi.hssf.usermodel.HSSFRow;
13 import org.apache.poi.hssf.usermodel.HSSFSheet;
14 import org.apache.poi.hssf.usermodel.
HSSFWorkbook;
15 import org.apache.poi.ss.usermodel.CellStyle;
16
17 import com.itextpdf.text.DocumentException;
18
19 public class GenerateExcel {
20     public ByteArrayOutputStream generateExcel(List
<? extends PatientsForm> patients, Reports
report) {
21
22         ByteArrayOutputStream baosEXC = new
ByteArrayOutputStream();
23
24         HSSFWorkbook workbook = new HSSFWorkbook();
25         HSSFSheet sheet = workbook.createSheet("
FirstSheet");
26         HSSFRow row = sheet.createRow((short) 0);
27         HSSFCell cell2;
28         HSSFCell cell = row.createCell((short) 0);
29         HSSFCellStyle style = workbook.createCellStyle
();
30         String symp = "";
31         for(PatientsForm pat : patients) {
32             symp = pat.getSymptoms();
33         }
34
35         if(report.isSpecRegion()) {
36             style = workbook.createCellStyle();
37             style.setAlignment(CellStyle.ALIGN_CENTER);
38             cell.setCellValue("Datasets for " + report.
getRegion());
39             cell.setCellStyle(style);
40         }
41
42         if(report.isSpecRegion())
43             row = sheet.createRow((short) 1);
44
45         style = workbook.createCellStyle();
46         style.setAlignment(CellStyle.ALIGN_LEFT);
47         cell.setCellValue("Disease:");
48         cell = row.createCell((short) 1);
49         cell.setCellValue(report.getDisease());
50         sheet.autoSizeColumn(1);
51         cell = row.createCell((short) 3);
52         cell.setCellValue("From");
53         cell = row.createCell((short) 4);
54         cell.setCellValue(report.getDateFrom());
55         cell = row.createCell((short) 5);
56         cell.setCellValue("To");
57         cell = row.createCell((short) 6);
58         cell.setCellValue(report.getDateTo());
59
60
61         if(report.isSpecRegion())
62             row = sheet.createRow((short) 2);
63         else
64             row = sheet.createRow((short) 1);
65
66         style = workbook.createCellStyle();
67         style.setAlignment(CellStyle.ALIGN_LEFT);
68
69         try {
70
71             style = workbook.createCellStyle();
72             style.setAlignment(CellStyle.ALIGN_CENTER);

```

```

73
74         cell = row.createCell((short) 0);
75         cell.setCellValue("Record No");
76         cell.setCellStyle(style);
77         sheet.autoSizeColumn(0);
78
79         cell = row.createCell((short) 1);
80         cell.setCellValue("Age");
81         cell.setCellStyle(style);
82         sheet.autoSizeColumn(1);
83
84         cell = row.createCell((short) 2);
85         cell.setCellValue("Gender");
86         cell.setCellStyle(style);
87         sheet.autoSizeColumn(2);
88
89         cell = row.createCell((short) 3);
90         cell.setCellValue("Symptoms");
91         cell.setCellStyle(style);
92         sheet.autoSizeColumn(3);
93
94         cell = row.createCell((short) 4);
95         cell.setCellValue("Date Onset");
96         cell.setCellStyle(style);
97         sheet.autoSizeColumn(4);
98
99         cell = row.createCell((short) 5);
100        cell.setCellValue("Outcome");
101        cell.setCellStyle(style);
102        sheet.autoSizeColumn(5);
103
104        cell = row.createCell((short) 6);
105        cell.setCellValue("Date Died");
106        cell.setCellStyle(style);
107        sheet.autoSizeColumn(6);
108
109        cell = row.createCell((short) 7);
110        cell.setCellValue("Barangay");
111        cell.setCellStyle(style);
112        sheet.autoSizeColumn(6);
113
114        cell = row.createCell((short) 8);
115        cell.setCellValue("Muncity");
116        cell.setCellStyle(style);
117        sheet.autoSizeColumn(7);
118
119        cell = row.createCell((short) 9);
120        cell.setCellValue("Province");
121        cell.setCellStyle(style);
122        sheet.autoSizeColumn(8);
123
124        if(!report.isSpecRegion()) {
125            cell = row.createCell((short) 10);
126            cell.setCellValue("Region");
127            cell.setCellStyle(style);
128        }
129        int i = 2;
130        if(report.isSpecRegion())
131            i = 3;
132
133        for(PatientsForm pat : patients) {
134
135            sheet.autoSizeColumn(i);
136
137            style = workbook.createCellStyle();
138            style.setAlignment(CellStyle.ALIGN_CENTER);
139
140            if(report.isSpecRegion())
141                row = sheet.createRow(i+1);
142            else
143                row = sheet.createRow(i);
144
145            cell = row.createCell((short) 0);
146            cell.setCellValue(i-1);
147            cell.setCellStyle(style);
148
149            cell = row.createCell((short) 1);
150            cell.setCellValue(pat.getAge());
151            cell.setCellStyle(style);
152
153            cell = row.createCell((short) 2);
154            cell.setCellValue(pat.getGender());

```

```

155 cell.setStyle(style);
156
157 cell = row.createCell((short) 3);
158 cell.setCellValue(pat.getSymptoms());
159 cell.setStyle(style);
160
161 cell = row.createCell((short) 4);
162 cell.setCellValue(pat.getDateOnset());
163 cell.setStyle(style);
164
165 cell = row.createCell((short) 5);
166 cell.setCellValue(pat.getOutcome());
167 cell.setStyle(style);
168
169 cell = row.createCell((short) 6);
170 cell.setCellValue(pat.getDateDied());
171 cell.setStyle(style);
172
173 cell = row.createCell((short) 7);
174 cell.setCellValue(pat.getBarangay());
175 cell.setStyle(style);
176 sheet.autoSizeColumn(7);
177
178 cell = row.createCell((short) 8);
179 cell.setCellValue(pat.getMuncity());
180 cell.setStyle(style);
181 sheet.autoSizeColumn(8);
182
183 cell = row.createCell((short) 9);

```

```

184 cell.setCellValue(pat.getProvince());
185 cell.setStyle(style);
186 sheet.autoSizeColumn(9);
187
188 if(!report.isSpecRegion() {
189 cell = row.createCell((short) 10);
190 cell.setCellValue(pat.getRegion());
191 cell.setStyle(style);
192 sheet.autoSizeColumn(10);
193 }
194
195 i++;
196 }
197
198 row = sheet.createRow(i+1);
199 cell = row.createCell((short) 0);
200 cell.setCellValue("Vector-Borne Disease
Monitoring System");
201
202 workbook.write(baosEXC);
203
204 } catch (Exception dex) {
205 dex.printStackTrace();
206 }
207
208 return baosEXC;
209
210 }
211 }

```

Listing 28: GenerateReports.java

```

1 package pdf;
2
3 import java.awt.BasicStroke;
4 import java.awt.Color;
5 import java.awt.Graphics2D;
6 import java.awt.geom.Rectangle2D;
7 import java.awt.image.BufferedImage;
8 import java.io.ByteArrayOutputStream;
9 import java.io.File;
10 import java.io.IOException;
11 import java.net.MalformedURLException;
12 import java.net.URL;
13 import java.nio.file.Files;
14 import java.nio.file.Path;
15 import java.nio.file.Paths;
16 import java.nio.file.attribute.FileTime;
17 import java.text.DateFormat;
18 import java.text.SimpleDateFormat;
19 import java.util.ArrayList;
20 import java.util.Collections;
21 import java.util.Comparator;
22 import java.util.Date;
23 import java.util.HashMap;
24 import java.util.LinkedList;
25 import java.util.List;
26 import java.util.Map;
27 import java.util.Map.Entry;
28 import java.util.Stack;
29
30 import javax.imageio.ImageIO;
31
32 import objects.Patients;
33 import objects.PatientsForm;
34 import objects.Place;
35 import objects.Reports;
36
37 import org.jfree.chart.ChartFactory;
38 import org.jfree.chart.JFreeChart;
39 import org.jfree.chart.axis.CategoryAxis;
40 import org.jfree.chart.axis.
    CategoryLabelPositions;
41 import org.jfree.chart.axis.DateAxis;
42 import org.jfree.chart.plot.CategoryPlot;
43 import org.jfree.chart.plot.PlotOrientation;
44 import org.jfree.chart.plot.XYPlot;
45 import org.jfree.chart.renderer.category.
    CategoryItemRenderer;
46 import org.jfree.chart.renderer.category.
    LineAndShapeRenderer;
47 import org.jfree.chart.renderer.xy.
    XYItemRenderer;
48 import org.jfree.chart.renderer.xy.
    XYLineAndShapeRenderer;
49 import org.jfree.chart.title.TextTitle;
50 import org.jfree.data.category.CategoryDataset;
51 import org.jfree.data.category.
    DefaultCategoryDataset;
52 import org.jfree.data.time.Month;
53 import org.jfree.data.time.TimeSeries;
54 import org.jfree.data.time.TimeSeriesCollection;
55 import org.jfree.data.xy.XYSeries;
56 import org.jfree.data.xy.XYSeriesCollection;
57 import org.jfree.ui.RectangleInsets;
58 import org.springframework.core.io.
    FileSystemResource;
59 import org.springframework.core.io.Resource;
60 import org.springframework.core.io.
    ResourceLoader;
61
62 import com.itextpdf.awt.DefaultFontMapper;
63 import com.itextpdf.text.Document;
64 import com.itextpdf.text.DocumentException;
65 import com.itextpdf.text.Element;
66 import com.itextpdf.text.Font;
67 import com.itextpdf.text.Image;
68 import com.itextpdf.text.PageSize;
69 import com.itextpdf.text.Paragraph;
70 import com.itextpdf.text.Phrase;
71 import com.itextpdf.text.pdf.PdfContentByte;
72 import com.itextpdf.text.pdf.PdfPCell;
73 import com.itextpdf.text.pdf.PdfPTable;
74 import com.itextpdf.text.pdf.PdfTemplate;
75 import com.itextpdf.text.pdf.PdfWriter;
76
77 public class GenerateReports {
78
79     private ResourceLoader reLoad;
80
81     public ByteArrayOutputStream generateReports(
82         List<? extends PatientsForm> newPatients,
83         Reports report,
84         List<? extends Place> place, List<? extends
85             PatientsForm> pd1, List<? extends
86             PatientsForm> pd2) throws DocumentException
87         , MalformedURLException, IOException {
88
89         Document doc = new Document();
90
91         ByteArrayOutputStream baosPDF = new
92             ByteArrayOutputStream();
93         PdfWriter docWriter = null;
94
95         try {
96
97             docWriter = PdfWriter.getInstance(doc,
98                 baosPDF);
99
100            doc.addProducer();
101            doc.addCreator(this.getClass().getName());
102            doc.addTitle("jfreechart pdf");
103
104            doc.setPageSize(PageSize.A4);
105
106            doc.open();
107
108            Font font1 = new Font(Font.FontFamily.
109                TIMES_ROMAN, 20, Font.BOLD);
110
111            URL imageUrl = getClass().getClassLoader().
112                getResource("/img/logo2.jpg");
113            Image image = Image.getInstance(imageUrl);
114            image.scaleToFit(100, 100);
115            doc.add(image);
116
117            DateFormat dateFormat = new
118                SimpleDateFormat("dd-MM-yyyy");
119            Date dateNow = new Date();
120            DateFormat timeFormat = new
121                SimpleDateFormat("HH:mm:ss");
122            Date timeNow = new Date();
123
124            Paragraph p = new Paragraph("Vector-
125                Borne Diseases Monitoring System", new Font
126                (Font.FontFamily.TIMES_ROMAN, 10, Font.
127                ITALIC));

```

```

113 doc.add(p);
114 p = new Paragraph(dateFormat.format(dateNow)
115 + ", " + timeFormat.format(timeNow), new
116 Font(Font.FontFamily.TIMES_ROMAN, 10, Font.
117 ITALIC));
118 doc.add(p);
119
120 URL imageUrl2 = getClass().
121 getClassLoader().getResource("/img/doh-logo
122 .jpg");
123 Image image2 = Image.getInstance(imageUrl2);
124 image2.scaleToFit(100, 100);
125 image2.setAbsolutePosition(450, 720);
126 doc.add(image2);
127 Paragraph paragraph = new Paragraph(
128 report.getDisease() + " Cases", font1);
129 paragraph.setAlignment(Element.ALIGN_CENTER);
130 paragraph.setSpacingAfter(50);
131 paragraph.setSpacingBefore(20);
132 doc.add(paragraph);
133
134 Font font2 = new Font(Font.FontFamily.
135 TIMES_ROMAN, 15, Font.BOLD);
136 Paragraph ph = new Paragraph("Trend\n", font2
137 );
138 ph.setIndentationLeft(40);
139 doc.add(ph);
140
141 String parts[] = splitTheString(report.
142 getDateFrom());
143 String parts2[] = splitTheString(report
144 .getDateTo());
145
146 String d1[] = report.getDateFrom().
147 split("-");
148 String d2[] = report.getDateTo().split
149 ("-");
150 System.out.println("Year: " + d1[0] +
151 "\tMonth: " + d1[1] + "\tDay: " + d1[2]);
152 System.out.println("Year: " + d2[0] +
153 "\tMonth: " + d2[1] + "\tDay: " + d2[2]);
154 //trend
155 String phrase = "A total of " + newPatients.
156 size() + " " + report.getDisease() + "
157 cases was reported nationwide from "
158 + parts[1] + " " + parts[2] + " ", " + parts
159 [0] + " to " + parts2[1] + " " + parts2[2]
160 + " ", " + parts2[0] + " .";
161
162 Font font3 = new Font(Font.FontFamily.
163 TIMES_ROMAN, 15, Font.NORMAL);
164
165 ph = new Paragraph(phrase, font3);
166 ph.setIndentationLeft(40);
167 ph.setSpacingAfter(15);
168 doc.add(ph);
169
170 int countReg[] = new int[17];
171 for(PatientsForm pat : newPatients)
172 {
173     if(pat.getRegion().
174 equalsIgnoreCase("region I"))
175         countReg[0]++;
176     else if(pat.getRegion().
177 equalsIgnoreCase("region II"))
178         countReg[1]++;
179     else if(pat.getRegion().
180 equalsIgnoreCase("region III"))
181         countReg[2]++;
182     else if(pat.getRegion().
183 equalsIgnoreCase("region IV-A"))
184         countReg[3]++;
185     else if(pat.getRegion().
186 equalsIgnoreCase("region IV-B"))
187         countReg[4]++;
188     else if(pat.getRegion().
189 equalsIgnoreCase("region V"))
190         countReg[5]++;
191     else if(pat.getRegion().
192 equalsIgnoreCase("region VI"))
193         countReg[6]++;
194     else if(pat.getRegion().
195 equalsIgnoreCase("region VII"))
196         countReg[7]++;
197     else if(pat.getRegion().
198 equalsIgnoreCase("region VIII"))
199         countReg[8]++;
200     else if(pat.getRegion().
201 equalsIgnoreCase("region IX"))
202         countReg[9]++;
203     else if(pat.getRegion().
204 equalsIgnoreCase("region X"))
205         countReg[10]++;
206     else if(pat.getRegion().
207 equalsIgnoreCase("region XI"))
208         countReg[11]++;
209     else if(pat.getRegion().
210 equalsIgnoreCase("region XII"))
211         countReg[12]++;
212     else if(pat.getRegion().
213 equalsIgnoreCase("ARMM"))
214         countReg[13]++;
215     else if(pat.getRegion().
216 equalsIgnoreCase("CAR"))
217         countReg[14]++;
218     else if(pat.getRegion().
219 equalsIgnoreCase("Region XIII"))
220         countReg[15]++;
221     else if(pat.getRegion().
222 equalsIgnoreCase("NCR"))
223         countReg[16]++;
224 }
225
226 Map <Integer, Integer> map = new
227 HashMap<Integer, Integer>();
228 for(int k = 0; k < countReg.length;
229 k++)
230     map.put(k, countReg[k]);
231
232 List<Map.Entry<Integer, Integer>>
233 list =
234     new LinkedList<Map.Entry<
235 Integer,Integer>>(map.entrySet());
236
237 Collections.sort(list, new
238 Comparator<Map.Entry<Integer,Integer>>() {
239     @Override
240     public int compare(
241         Entry<Integer, Integer>
242 e1,
243         Entry<Integer, Integer>
244 e2) {
245         return e2.getValue().
246             compareTo(e1.getValue());
247     }
248 });
249
250 int indices[] = new int[17];
251 int l = 0;
252 int total = 0;
253 for(Entry<Integer, Integer> lValue:
254 list) {
255     indices[l] = lValue.getKey();
256     total += countReg[l];
257     l++;
258 }
259
260 String regn[] = {"Region I", "Region
261 II", "Region III", "Region IV-A",
262 "Region IV-B", "Region V", "Region
263 VI", "Region VII", "Region VIII",
264 "Region IX", "Region X", "Region
265 XI", "Region XII", "Autonomous Region of
266 Muslim Mindanao",
267 "Cordillera Administrative Region
268 ", "CARAGA/Region XIII", "National Capital
269 Region"};
270
271 ph = new Paragraph("Geographic
272 Distribution\n", font2);
273 ph.setIndentationLeft(40);
274 doc.add(ph);
275
276 phrase = "Most of the " + report.getDisease()
277 .toLowerCase() + " cases were from the
278 following regions: "
279 + regn[indices[0]] + " (" + (countReg[
280 indices[0]]/total)*100 + "%)";
281 if(countReg[indices[1]] > 0) {
282     phrase += ", " + regn[indices[1]] + " (" + (
283 countReg[indices[1]]/total)*100 + "%), "
284 + regn[indices[0]] + " (" + (countReg[
285 indices[0]]/total)*100 + "%). ";
286 }
287
288 ph = new Paragraph(phrase, font3);
289 ph.setIndentationLeft(40);
290 doc.add(ph);
291
292 String[] month = {"", "January", "February",
293 "March", "April", "May", "June", "July", "
294 August",
295 "September", "October", "November", "
296 December"};
297 String month2[] = {"", "Jan", "Feb", "Mar",
298 "Apr", "May", "Jun", "Jul", "Aug", "Sep", "
299 Oct", "Nov", "Dec"};
300 Stack months = new Stack();
301 int count[];
302 String dateCopy = "";
303 int ci = 1;
304 months.add(0, "");
305 if(!d1[0].equalsIgnoreCase(d2[0])) {

```

```

248     for(int i = Integer.parseInt(d1[1]); i <= 329
12; i++) { 330
249     System.out.println("Month: " + month2[i] + 331
"\ti is: " + i); 332
250     months.add(ci, month2[i]);
251     ci++;
252 }
253 int diff = Integer.parseInt(d2[0]) - Integer
.parseInt(d1[0]);
254 while(diff > 0) {
255     for(int i = 1; i <= 12; i++) {
256         if(diff == 1) {
257             if(i <= Integer.parseInt(d2[1])) {
258                 System.out.println("Month: " + month2[i
]);
259                 months.add(ci, month2[i]);
260                 ci++;
261             }
262         }
263     }
264     else {
265         months.add(ci, month2[i]);
266         ci++;
267     }
268     diff--;
269 }
270 count = new int[months.size()];
271 System.out.println("Num Months: " + months.
size());
272 for(PatientsForm pat : newPatients) {
273     String date[] = splitTheString(pat.
getDateSubmitted());
274     dateCopy = date[0];
275     String da [] = pat.getDateSubmitted().split
("-");
276     if(Integer.parseInt(d2[0]) - Integer.
parseInt(d1[0]) > 1) {
277         int nus = 0;
278         int diff2 = Integer.parseInt(d2[0]) -
Integer.parseInt(d1[0]);
279         for(int k = 0; k < diff; k++) {
280             if(k == 0) {
281                 for(int i = Integer.parseInt(d1[1]); i
<= 12; i++) {
282                     if(Integer.parseInt(da[0]) == Integer.
parseInt(d1[0])) {
283                         if(Integer.parseInt(da[1]) == i)
284                             count[nus]++;
285                     }
286                     nus++;
287                 }
288             }
289             else if(k == diff-1) {
290                 for(int i = 1; i <= Integer.parseInt(d2
[1]); i++) {
291                     if(Integer.parseInt(da[0]) == (Integer.
parseInt(d1[0]) + k)) {
292                         if(Integer.parseInt(da[1]) == i)
293                             count[nus]++;
294                     }
295                     nus++;
296                 }
297             }
298             else {
299                 for(int i = 1; i <= 12; i++) {
300                     if(Integer.parseInt(da[0]) == (Integer.
parseInt(d1[0]) + k)) {
301                         if(Integer.parseInt(da[1]) == i)
302                             count[nus]++;
303                     }
304                     nus++;
305                 }
306             }
307         }
308     }
309     else {
310         int nu = 0;
311         for(int i = Integer.parseInt(d1[1]); i <=
312 12; i++) {
313             if(Integer.parseInt(da[1]) == i) {
314                 count[nu]++;
315                 System.out.println("Count index: " + nu)
;
316             }
317             nu++;
318         }
319         for(int i = 1; i <= Integer.parseInt(d2
[1]); i++) {
320             if(Integer.parseInt(da[1]) == i) {
321                 count[nu]++;
322                 System.out.println("Count index: " + nu)
;
323             }
324             nu++;
325         }
326     }
327 }
328 }
else {
329     count = new int[12];
330     for(PatientsForm pat : newPatients) {
331         String date[] = splitTheString(pat.
getDateSubmitted());
332         dateCopy = date[0];
333         for(int i = 1; i <= Integer.parseInt(d2[1])
; i++) {
334             if(date[i].equals(month[i])) {
335                 count[i]++;
336                 System.out.println("Count here: " + count
[i]);
337             }
338         }
339     }
340 }
341 }
342 }
343 }
344
345 DefaultCategoryDataset dataset = new
DefaultCategoryDataset();
346 String tit = "";
347 if(parts[0].equalsIgnoreCase(parts2[0])) {
348     tit = report.getDisease() + " Cases by Month
, Philippines, " + dateCopy;
349     for(int i = 1; i <= Integer.parseInt(d2[1]);
i++) {
350         dataset.addValue(count[i], report.
getDisease() + " cases", month2[i] + " (" +
count[i] + ")");
351     }
352 }
353 else {
354     tit = report.getDisease() + " Cases by Month
, Philippines, " + d1[0] + " - " + d2[0];
355     for(int i = 1; i < months.size(); i++) {
356         dataset.addValue(count[i+1], report.
getDisease() + " cases", months.get(i+1) +
" (" + count[i+1] + ")");
357     }
358 }
359
360 JFreeChart chart = ChartFactory.
createLineChart(tit, // chart title
null, // domain axis label
"Number of Cases", // range axis
label
dataset, // data
PlotOrientation.VERTICAL, //
orientation
true, // include legend
true, // tooltips
false // urls
);
361
362 // trick to change the default font of the
chart
363 chart.setTitle(new TextTitle(tit, new java.
awt.Font("Serif", Font.BOLD, 12));
364 chart.setBackgroundPaint(Color.white);
365 chart.setBorderPaint(Color.black);
366 chart.setBorderStroke(new BasicStroke(1));
367 chart.setBorderVisible(true);
368
369 int width = 570;
370 int height = 450;
371
372 CategoryPlot plot = chart.getCategoryPlot();
373 plot.setBackgroundPaint(Color.white);
374 plot.getDomainAxis().setLabelFont(new java.
awt.Font("Serif", Font.NORMAL, 13));
375 plot.getDomainAxis().
setMaximumCategoryLabelWidthRatio(0.5f);
376 plot.getDomainAxis().
setCategoryLabelPositions(
CategoryLabelPositions.UP_45);
377 CategoryItemRenderer catRen = plot.
getRenderer();
378 if(catRen instanceof LineAndShapeRenderer) {
379     LineAndShapeRenderer renderer = (
LineAndShapeRenderer) catRen;
380     renderer.setBaseShapesFilled(true);
381     renderer.setBaseShapesVisible(true)
;
382     ;
383     renderer.setShapesVisible( true );
384     renderer.setDrawOutlines( true );
385 }
386
387 PdfContentByte dc = docWriter.
getDirectContent();
388 PdfTemplate tp = dc.createTemplate(width,
height);
389 Graphics2D g2 = tp.createGraphics(width-100,
height-50,
new DefaultFontMapper());
390 Rectangle2D r2D = new Rectangle2D.Double(0,
0, width-100, height-50);
391 chart.draw(g2, r2D, null);
392 g2.dispose();

```

```

401 dc.addTemplate(tp, 38, docWriter.
    getVerticalPosition(true) - height);
402
403 if(report.isAgeGroup()) {
404     String category1 = "Unspecified";
405     String category2 = "< 1";
406     String category3 = "1-10";
407     String category4 = "11-20";
408     String category5 = "21-30";
409     String category6 = "31-40";
410     String category7 = "> 40";
411
412     DefaultCategoryDataset barData = new
        DefaultCategoryDataset();
413     int barCount[] = new int[7];
414     for(PatientsForm pat : newPatients) {
415         if(pat.getAge() < 1)
416             barCount[1]++;
417         else if(pat.getAge() >= 1 && pat.getAge()
418             <= 10)
419             barCount[2]++;
420         else if(pat.getAge() >= 11 && pat.getAge()
421             <= 20)
422             barCount[3]++;
423         else if(pat.getAge() >= 21 && pat.getAge()
424             <= 30)
425             barCount[4]++;
426         else if(pat.getAge() >= 31 && pat.getAge()
427             <= 40)
428             barCount[5]++;
429         else if(pat.getAge() > 40)
430             barCount[6]++;
431         else
432             barCount[0]++;
433     }
434     barData.addValue(barCount[0], "Age",
435         category1);
436     barData.addValue(barCount[1], "Age",
437         category2);
438     barData.addValue(barCount[2], "Age",
439         category3);
440     barData.addValue(barCount[3], "Age",
441         category4);
442     barData.addValue(barCount[4], "Age",
443         category5);
444     barData.addValue(barCount[5], "Age",
445         category6);
446     barData.addValue(barCount[6], "Age",
447         category7);
448
449     final JFreeChart barChart = ChartFactory.
450         createBarChart(
451             report.getDisease() + " Cases by
452             Age Group" + report.getDateFrom() + " - "
453             + report.getDateTo(), // chart
454             title
455             null, // domain
456             axis label
457             "Number of Cases",
458             // range axis label
459             barData, //
460             data
461             PlotOrientation.VERTICAL, //
462             orientation
463             true, //
464             include legend
465             true, //
466             tooltips?
467             false //
468             URLs?
469             );
470     barChart.setTitle(new TextTitle(report.
471         getDisease() + " Cases by Age Group", new
472         java.awt.Font("Serif", Font.BOLD, 12));
473     barChart.setBackgroundPaint(Color.white);
474     barChart.setBorderPaint(Color.black);
475     barChart.setBorderStroke(new BasicStroke(1)
476     );
477     barChart.setBorderVisible(true);
478
479     width = 550;
480     height = 350;
481
482     doc.newPage();
483     PdfContentByte dcB = docWriter.
484         getDirectContent();
485     PdfTemplate tpB = dcB.createTemplate(width,
486         height);
487     Graphics2D g2B = tpB.createGraphics(width
488         -100, height-50,
489         new DefaultFontMapper());
490     Rectangle2D r2DB = new Rectangle2D.Double
491         (0, 0, width-100, height-50);
492     barChart.draw(g2B, r2DB, null);
493     g2B.dispose();
494
495     dcB.addTemplate(tpB, 38, docWriter.
496         getVerticalPosition(true) - height);
497 }
498
499 if(report.isFatalAgeGroup()) {
500     String category1 = "Unspecified";
501     String category2 = "< 1";
502     String category3 = "1-10";
503     String category4 = "11-20";
504     String category5 = "21-30";
505     String category6 = "31-40";
506     String category7 = "> 40";
507
508     DefaultCategoryDataset barData = new
509         DefaultCategoryDataset();
510     int barCount[] = new int[7];
511     int barCountD[] = new int[7];
512     for(PatientsForm pat : newPatients) {
513         if(pat.getAge() < 1) {
514             barCount[1]++;
515             if(pat.getOutcome().equalsIgnoreCase("
516                 Died"))
517                 barCountD[1]++;
518         }
519         else if(pat.getAge() >= 1 && pat.getAge()
520             <= 10) {
521             barCount[2]++;
522             if(pat.getOutcome().equalsIgnoreCase("
523                 Died"))
524                 barCountD[2]++;
525         }
526         else if(pat.getAge() >= 11 && pat.getAge()
527             <= 20) {
528             barCount[3]++;
529             if(pat.getOutcome().equalsIgnoreCase("
530                 Died"))
531                 barCountD[3]++;
532         }
533         else if(pat.getAge() >= 21 && pat.getAge()
534             <= 30) {
535             barCount[4]++;
536             if(pat.getOutcome().equalsIgnoreCase("
537                 Died"))
538                 barCountD[4]++;
539         }
540         else if(pat.getAge() >= 31 && pat.getAge()
541             <= 40) {
542             barCount[5]++;
543             if(pat.getOutcome().equalsIgnoreCase("
544                 Died"))
545                 barCountD[5]++;
546         }
547         else if(pat.getAge() > 40) {
548             barCount[6]++;
549             if(pat.getOutcome().equalsIgnoreCase("
550                 Died"))
551                 barCountD[6]++;
552         }
553         else {
554             barCount[0]++;
555             if(pat.getOutcome().equalsIgnoreCase("
556                 Died"))
557                 barCountD[0]++;
558         }
559     }
560     try {
561         barData.addValue((barCountD[0]/barCount
562             [0])*100, "Age", category1);
563     } catch(Exception ds1) {
564         barData.addValue(0, "Age", category1);
565     }
566     try {
567         barData.addValue((barCountD[1]/barCount
568             [1])*100, "Age", category2);
569     } catch(Exception ds2) {
570         barData.addValue(0, "Age", category2);
571     }
572     try {
573         barData.addValue((barCountD[2]/barCount
574             [2])*100, "Age", category3);
575     } catch(Exception ds3) {
576         barData.addValue(0, "Age", category3);
577     }
578     try {
579         barData.addValue((barCountD[3]/barCount
580             [3])*100, "Age", category4);
581     } catch(Exception ds4) {
582         barData.addValue(0, "Age", category4);
583     }
584     try {
585         barData.addValue((barCountD[4]/barCount
586             [4])*100, "Age", category5);
587     } catch(Exception ds5) {
588         barData.addValue(0, "Age", category5);
589     }
590 }

```

```

550     }
551
552     try {
553         barData.addValue((barCountD[5]/barCount
554 [5])*100, "Age", category6);
555     } catch(Exception ds6) {
556         barData.addValue(0, "Age", category6);
557     }
558
559     try {
560         barData.addValue((barCountD[6]/barCount
561 [6])*100, "Age", category7);
562     } catch(Exception ds7) {
563         barData.addValue(0, "Age", category7);
564     }
565
566     final JFreeChart barChart = ChartFactory.
567     createBarChart(
568         report.getDisease() + " Cases by
569         Age Group" + report.getDateFrom() + " - "
570         + report.getDateTo(), // chart
571         title
572         null, // domain
573         axis label
574         "Number of Cases",
575         // range axis label
576         barData, //
577         data
578         PlotOrientation.VERTICAL, //
579         orientation
580         true, //
581         include legend
582         true, //
583         tooltips?
584         false //
585         URLs?
586         );
587
588     barChart.setTitle(new TextTitle(report.
589     getDisease() + " Case Fatality Ration by
590     Age Group", new java.awt.Font("Serif", Font
591     .BOLD, 12)));
592     barChart.setBackgroundPaint(Color.white);
593     barChart.setBorderPaint(Color.black);
594     barChart.setBorderStroke(new BasicStroke(1)
595     );
596     barChart.setBorderVisible(true);
597
598     width = 550;
599     height = 350;
600
601     doc.newPage();
602     PdfContentByte dcB2 = docWriter.
603     getDirectContent();
604     PdfTemplate tpB2 = dcB2.createTemplate(
605     width, height);
606     Graphics2D g2B2 = tpB2.createGraphics(width
607     -100, height-50,
608     new DefaultFontMapper());
609     Rectangle2D r2DB2 = new Rectangle2D.Double
610     (0, 0, width-100, height-50);
611     barChart.draw(g2B2, r2DB2, null);
612     g2B2.dispose();
613
614     dcB2.addTemplate(tpB2, 38, docWriter.
615     getVerticalPosition(true) - height);
616 }
617
618 if(report.isGenderGroup()) {
619     String categoryS1 = "Unspecified";
620     String categoryS2 = "Female";
621     String categoryS3 = "Male";
622
623     DefaultCategoryDataset barDatas = new
624     DefaultCategoryDataset();
625     int barCount2[] = new int[3];
626     for(PatientsForm pat : newPatients) {
627         if(pat.getGender().equalsIgnoreCase("
628         Female"))
629             barCount2[1]++;
630         else if(pat.getGender().equalsIgnoreCase("
631         Male"))
632             barCount2[2]++;
633         else
634             barCount2[0]++;
635     }
636
637     barDatas.addValue(barCount2[0], "Gender",
638     categoryS1);
639     barDatas.addValue(barCount2[1], "Gender",
640     categoryS2);
641     barDatas.addValue(barCount2[2], "Gender",
642     categoryS3);
643
644     final JFreeChart barChart = ChartFactory.
645     createBarChart(
646         report.getDisease() + " Cases by
647         Gender Group", // chart title
648         null, // domain
649         axis label
650         "Number of Cases",
651         // range axis label
652         barDatas, //
653         data
654         PlotOrientation.VERTICAL, //
655         orientation
656         true, //
657         include legend
658         true, //
659         tooltips?
660         false //
661         URLs?
662         );
663
664     barChart.setBackgroundPaint(Color.white);
665     barChart.setBorderPaint(Color.black);
666     barChart.setBorderStroke(new BasicStroke(1)
667     );
668     barChart.setBorderVisible(true);
669
670     width = 550;
671     height = 350;
672
673     doc.newPage();
674     PdfContentByte dcB2 = docWriter.
675     getDirectContent();
676     PdfTemplate tpB2 = dcB2.createTemplate(
677     width, height);
678     Graphics2D g2B2 = tpB2.createGraphics(width
679     -100, height-50,
680     new DefaultFontMapper());
681     Rectangle2D r2DB2 = new Rectangle2D.Double
682     (0, 0, width-100, height-50);
683     barChart.draw(g2B2, r2DB2, null);
684     g2B2.dispose();
685
686     dcB2.addTemplate(tpB2, 38, docWriter.
687     getVerticalPosition(true) - height);
688 }
689
690 if(report.isByRegion()) {
691     doc.newPage();
692     Font font4 = new Font(Font.FontFamily.
693     TIMES_ROMAN, 13, Font.BOLD);
694     ph = new Paragraph(report.getDisease() + "
695     Cases and Deaths by Region", font4);
696     ph.setAlignment(Element.ALIGN_CENTER);
697     doc.add(ph);
698     ph = new Paragraph("Philippines, " + report.
699     getDateFrom() + " - " + report.getDateTo(),
700     font2);
701     ph.setAlignment(Element.ALIGN_CENTER);
702     ph.setSpacingAfter(30);
703     doc.add(ph);
704
705     PdfPTable table = new PdfPTable(4); // 3
706     columns.
707     table.setHorizontalAlignment(Element.
708     ALIGN_CENTER);
709     PdfPCell cell11 = new PdfPCell(new Paragraph
710     ("Region"));
711     PdfPCell cell12 = new PdfPCell(new
712     Paragraph("Cases"));
713     PdfPCell cell13 = new PdfPCell(new
714     Paragraph("Deaths"));
715     PdfPCell cell14 = new PdfPCell(new
716     Paragraph("CFR(%)"));
717     float[] columnWidths = new float[]
718     {4f, 4f, 4f, 4f};
719
720     table.addCell(cell11);
721     table.addCell(cell12);
722     table.addCell(cell13);
723     table.addCell(cell14);
724
725     String reg[] = {"I", "II", "III", "
726     IV-A", "IV-B", "V", "VI", "VII", "VIII", "
727     IX", "X",
728     "XI", "XII", "ARMM", "CAR", "
729     CARAGA", "NCR"};
730
731     int countReg1[] = new int[17];
732     int countD1[] = new int[17];
733
734     for(PatientsForm pat : pd1) {
735         if(pat.getRegion().
736         equalsIgnoreCase("region I")) {
737             countReg1[0]++;
738             if(pat.getOutcome().
739             equalsIgnoreCase("Died"))

```

```

689         countD1[0]++;
690     } else if(pat.getRegion().
equalsIgnoreCase("region II")) {
691         countReg1[1]++;
692         if(pat.getOutcome().
equalsIgnoreCase("Died"))
693             countD1[1]++;
694     } else if(pat.getRegion().
equalsIgnoreCase("region III")) {
695         countReg1[2]++;
696         if(pat.getOutcome().
equalsIgnoreCase("Died"))
697             countD1[2]++;
698     } else if(pat.getRegion().
equalsIgnoreCase("region IV-A")) {
699         countReg1[3]++;
700         if(pat.getOutcome().
equalsIgnoreCase("Died"))
701             countD1[3]++;
702     } else if(pat.getRegion().
equalsIgnoreCase("region IV-B")) {
703         countReg1[4]++;
704         if(pat.getOutcome().
equalsIgnoreCase("Died"))
705             countD1[4]++;
706     } else if(pat.getRegion().
equalsIgnoreCase("region V")) {
707         countReg1[5]++;
708         if(pat.getOutcome().
equalsIgnoreCase("Died"))
709             countD1[5]++;
710     } else if(pat.getRegion().
equalsIgnoreCase("region VI")) {
711         countReg1[6]++;
712         if(pat.getOutcome().
equalsIgnoreCase("Died"))
713             countD1[6]++;
714     } else if(pat.getRegion().
equalsIgnoreCase("region VII")) {
715         countReg1[7]++;
716         if(pat.getOutcome().
equalsIgnoreCase("Died"))
717             countD1[7]++;
718     } else if(pat.getRegion().
equalsIgnoreCase("region VIII")) {
719         countReg1[8]++;
720         if(pat.getOutcome().
equalsIgnoreCase("Died"))
721             countD1[8]++;
722     } else if(pat.getRegion().
equalsIgnoreCase("region IX")) {
723         countReg1[9]++;
724         if(pat.getOutcome().
equalsIgnoreCase("Died"))
725             countD1[9]++;
726     } else if(pat.getRegion().
equalsIgnoreCase("region X")) {
727         countReg1[10]++;
728         if(pat.getOutcome().
equalsIgnoreCase("Died"))
729             countD1[10]++;
730     } else if(pat.getRegion().
equalsIgnoreCase("region XI")) {
731         countReg1[11]++;
732         if(pat.getOutcome().
equalsIgnoreCase("Died"))
733             countD1[11]++;
734     } else if(pat.getRegion().
equalsIgnoreCase("region XII")) {
735         countReg1[12]++;
736         if(pat.getOutcome().
equalsIgnoreCase("Died"))
737             countD1[12]++;
738     } else if(pat.getRegion().
equalsIgnoreCase("ARMM")) {
739         countReg1[13]++;
740         if(pat.getOutcome().
equalsIgnoreCase("Died"))
741             countD1[13]++;
742     } else if(pat.getRegion().
equalsIgnoreCase("CAR")) {
743         countReg1[14]++;
744         if(pat.getOutcome().
equalsIgnoreCase("Died"))
745             countD1[14]++;
746     } else if(pat.getRegion().
equalsIgnoreCase("Region XIII")) {
747         countReg1[15]++;
748         if(pat.getOutcome().
equalsIgnoreCase("Died"))
749             countD1[15]++;
750     } else if(pat.getRegion().
equalsIgnoreCase("NCR")) {
751         countReg1[16]++;
752         if(pat.getOutcome().
equalsIgnoreCase("Died"))
753             countD1[16]++;
754     }
755 }

```

```

756
757
758
759         int countN = 0;
760         int countM = 0;
761
762
763
764         for(int i = 0; i < 17; i++) {
765             table.addCell(reg[i]);
766             table.addCell(countReg1[i] + "");
767             table.addCell(countD1[i] + "");
768             try {
769                 table.addCell((countD1[i]/
countReg1[i])*100 + "");
770             } catch(Exception asna) {
771                 table.addCell("0");
772             }
773             countN += countReg1[i];
774             countM += countD1[i];
775         }
776
777         table.addCell("Total");
778         table.addCell(countN + "");
779         table.addCell(countM + "");
780         try {
781             table.addCell((countM/countN) *
100 + "");
782         } catch(Exception asna) {
783             table.addCell("0");
784         }
785         doc.add(table);
786     }
787 }
788 if(report.isCaseAndDeath()) {
789
790     Stack stackMun = new Stack();
791     Stack stackCou = new Stack(); //Count of
Cases
792     Stack stackDea = new Stack(); //Count of
Deaths
793     int index = 0;
794
795     for(PatientsForm pat : pd1) {
796         if(pat.getRegion().equalsIgnoreCase(report.
getRegion())) {
797             if(!stackMun.contains(pat.getMuncity())) {
798                 stackMun.add(index, pat.getMuncity());
799                 stackCou.add(index, 1);
800                 if(pat.getOutcome().equalsIgnoreCase("
Died"))
801                     stackDea.add(index, 1);
802             } else
803                 stackDea.add(index, 0);
804             index++;
805         } else {
806             int num = (int) stackCou.get(stackMun.
indexOf(pat.getMuncity()));
807             stackCou.remove(stackMun.indexOf(pat.
getMuncity()));
808             stackCou.add(stackMun.indexOf(pat.
getMuncity()), (num+1));
809             int num2 = (int) stackDea.get(stackMun.
indexOf(pat.getMuncity()));
810             if(pat.getOutcome().equalsIgnoreCase("
Died")) {
811                 stackDea.remove(stackMun.indexOf(pat.
getMuncity()));
812                 stackDea.add(stackMun.indexOf(pat.
getMuncity()), (num2+1));
813             }
814         }
815     }
816 }
817
818     Stack stackMun2 = new Stack();
819     Stack stackCou2 = new Stack(); //Count of
Cases
820     Stack stackDea2 = new Stack(); //Count of
Deaths
821     int index2 = 0;
822
823     for(PatientsForm pat : pd2) {
824         if(pat.getRegion().equalsIgnoreCase(report.
getRegion())) {
825             if(!stackMun2.contains(pat.getMuncity()))
826             {
827                 stackMun2.add(index2, pat.getMuncity());
828                 stackCou2.add(index2, 1);
829                 if(pat.getOutcome().equalsIgnoreCase("
Died"))
830                     stackDea2.add(index2, 1);
831             } else
832                 stackDea2.add(index2, 0);
833             index2++;
834         }
835     }
836     else {

```



```

837     int num = (int) stackCou2.get(stackMun2. 903         table2.addCell("0");
      indexOf(pat.getMuncity()));           904     }
838     stackCou.remove(stackMun2.indexOf(pat. 905     try {
      getMuncity()));                       906         countNN += (double) stackCou.get(
839         stackCou.add(stackMun2.indexOf(pat. 907         result));
      getMuncity()), (num+1));             908         result = -1;
840     int num2 = (int) stackDea2.get(stackMun2. 909     }
      indexOf(pat.getMuncity()));           910     catch(Exception e) {
841     if(pat.getOutcome().equalsIgnoreCase(" 911     }
      Died")) {                             912     try {
842         stackDea2.remove(stackMun2.indexOf(pat. 913         countNN2 += (double) stackDea.get
      getMuncity()));                       914         (result);
843         stackDea2.add(stackMun2.indexOf(pat. 915         result = -1;
      getMuncity()), (num2+1));             916         catch(Exception e) {
844     }                                       917     }
845     }                                       918     }
846     }                                       919     }
847     }                                       920
848     doc.newPage();                               921     table2.addCell("Total");
849
850     Font font4 = new Font(Font.FontFamily. 922     table2.addCell(countNN + "");
      TIMES_ROMAN, 13, Font.BOLD);         923     table2.addCell(countNN2 + "");
851     ph = new Paragraph(report.getDisease() + " 924     try {
      Cases and Deaths by Muncity", font4); 925         table2.addCell((countNN2/countNN)
852     ph.setAlignment(Element.ALIGN_CENTER);    *100 + "");
853     doc.add(ph);                               }
854     ph = new Paragraph(report.getRegion() + ", " 926     catch(Exception e) {
      + (Integer.parseInt(d1[0]) - 1) + " & " + 927     table2.addCell("0");
      d1[0], font2);                         928     }
855     ph.setAlignment(Element.ALIGN_CENTER);    929
856     ph.setSpacingAfter(30);                   930
857     doc.add(ph);                               931
858                                               932     doc.add(table2);
859                                               933
860     PdfPTable table2 = new PdfPTable(4); // 3 934     }
      columns.                               935
861     table2.setHorizontalAlignment(Element. 936 } catch (DocumentException dex) {
      ALIGN_CENTER);                         937     baosPDF.reset();
862     PdfPCell cell11 = new PdfPCell(new Paragraph 938     dex.printStackTrace();
      ("Muncity"));                           939     throw new DocumentException(dex);
863         PdfPCell cell12 = new PdfPCell(new 940 } catch (Exception e) {
      Paragraph("Cases"));                   941     e.printStackTrace();
864         PdfPCell cell13 = new PdfPCell(new 942 } finally {
      Paragraph("Deaths"));                 943     if (doc != null && doc.isOpen()) {
865         PdfPCell cell14 = new PdfPCell(new 944     doc.close();
      Paragraph("CFR(%)"));                 945     }
866         float[] columnWidths = new float[] 946     if (docWriter != null) {
      {4f, 4f, 4f, 4f};                     947     docWriter.close();
867         table2.setWidths(columnWidths);     948     }
868                                               949     }
869                                               950
870         table2.addCell(cell11);             951     if (baosPDF.size() < 1) {
871         table2.addCell(cell12);             952     throw new DocumentException("document has " +
872         table2.addCell(cell13);             baosPDF.size()
873         table2.addCell(cell14);             + " bytes");
874                                               954     }
875     double countNN = 0;                       955
876     double countNN2 = 0;                     956
877     int result = -1;                          957
878     for(Place pl : place) {                   958
879         table2.addCell(pl.getMuncity() + 959     public String[] splitTheString(String dates) {
      "");                                   960     String parts[] = dates.split("-");
880     for(int i = 0; i < stackMun.size() 961     String parts[] = dates.split("-");
      ; i++) {                               962
881         if(!((String) stackMun.get(i)). 963     if(parts[1].equals("01")
      equalsIgnoreCase(pl.getMuncity()))    964     parts[1] = "January";
882         result = i;                           965     else if(parts[1].equals("02")
883         else                                    966     parts[1] = "February";
884         result = -1;                           967     else if(parts[1].equals("03")
885         }                                       968     parts[1] = "March";
886         try {                                   969     else if(parts[1].equals("04")
887             table2.addCell(stackCou.get( 970     parts[1] = "April";
      result) + "");                         971     else if(parts[1].equals("05")
888         }                                       972     parts[1] = "May";
889         catch(Exception e) {                   973     else if(parts[1].equals("06")
890             table2.addCell("0");             974     parts[1] = "June";
891         }                                       975     else if(parts[1].equals("07")
892         try {                                   976     parts[1] = "July";
893             table2.addCell(stackDea.get( 977     else if(parts[1].equals("08")
      result) + "");                         978     parts[1] = "August";
894         }                                       979     else if(parts[1].equals("09")
895         }                                       980     parts[1] = "September";
896         catch(Exception e) {                   981     else if(parts[1].equals("10")
897             table2.addCell("0");             982     parts[1] = "October";
898         }                                       983     else if(parts[1].equals("11")
899         try {                                   984     parts[1] = "November";
900             table2.addCell(Double.parseDouble 985     else if(parts[1].equals("12")
      (stackDea.get(result).toString())/Double. 986     parts[1] = "December";
      parseDouble(stackCou.get(result).toString() 987     return parts;
      )*100 + "");                           988
901         }                                       989     }
902     catch(Exception e) {                       990     }

```

5. Views

Listing 29: header.jsp

```

1  <%@ taglib prefix="c" uri="http://java.sun.com/
   jsp/jstl/core" %>
2  <%@ taglib uri="http://java.sun.com/jsp/jstl/
   functions" prefix="fn"%>
3  <%@ taglib prefix="sec" uri="http://www.
   springframework.org/security/tags" %>
4  <!DOCTYPE html>
5  <html>
6  <head>
7  <title>Vector-Borne Diseases Monitoring System
   </title>
8  <meta name="viewport" content="width=device-
   width, initial-scale=1.0">
9  <jsp:include page="css.jsp"/>
10 <script type="text/javascript">
11 // When the document is ready
12 $(document).ready(function () {
13     $('#dp1').datepicker({
14         format: "yyyy-mm-dd"
15     });
16     $('#dp2').datepicker({
17         format: "yyyy-mm-dd"
18     });
19     $('#dp3').datepicker({
20         format: "yyyy-mm-dd"
21     });
22     $('#dp4').datepicker({
23         format: "yyyy-mm-dd"
24     });
25 });
26 </script>
27 <script type="text/javascript">
28     function changeSelect5() {
29         //var sel = document.getElementById("checks")
30         .checked;
31         if(document.getElementById("checks").checked)
32             document.getElementById("select2").disabled
33             = false;
34         else
35             document.getElementById("select2").disabled
36             = true;
37     }
38 </script>
39 <script type="text/javascript">
40 /* Table initialisation */
41 jQuery(document).ready(function($) {
42     $('#alertTable').dataTable( {
43         "bSort": false, // Disable sorting
44         "iDisplayLength": 10, // records per
45         page
46         "sDom": "t<'row'<'col-md-6'i><'col-md-6'p
47         >>",
48         "sPaginationType": "bootstrap"
49     });
50 </script>
51 </head>
52 <body>
53 <div class = "container">
54 <div class = "navbar navbar-default" role = "
55     navigation">
56 <div class = "container-fluid">
57 <div class = "navbar-header">
58 <button type = "button" class = "navbar-
59     toggle" data-toggle = "collapse" data-
60     target = ".navbar-collapse">
61 <span class = "sr-only">Toggle Navigation
62 </span>
63 <span class = "icon-bar"></span>
64 <span class = "icon-bar"></span>
65 <span class = "icon-bar"></span>
66 </button>
67 <a href = "<%= request.getContextPath()
68     %>/home"><img src = "<%= request.
69     getContextPath() %>/resources/dist/img/
70     logo2.jpg" alt = "VBDMS Logo" height = "30
71     px;" width = "93px;" class = "img-
72     responsive"/></a>
73 <!-- <a href = "<%= request.getContextPath
74     () %>/home" class = "navbar-brand">VBDMS<a
75     > -->
76 </div>
77 <div class = "collapse navbar-collapse">
78 <ul class = "nav navbar-nav">
79 <li <c:if test = "${pages == 'home'}">
80     class = "active"</c:if>>
81 <a href = "<%= request.getContextPath()
82     %>/home">Home</a></li>
83
84 <li class = "dropdown">
85 <a href = "" data-toggle = "dropdown"
86     class = "dropdown-toggle">
87     Diseases <b class = "caret"></b>
88 </a>
89 <ul class="dropdown-menu">
90 <c:forEach var="dis" items="${disease
91     }">
92     <li><a href = "<%= request.
93     getContextPath() %>/disease/${dis.
94     diseaseName}">${dis.diseaseName}</a></li>
95 </c:forEach>
96 </ul>
97 </li>
98 <li class = "dropdown">
99 <a href = "" data-toggle = "dropdown"
100     class = "dropdown-toggle">
101     Disease Statistics <b class = "caret
102     "></b>
103 </a>
104 <ul class="dropdown-menu">
105 <c:forEach var="dis" items="${disease
106     }">
107     <li><a href = "<%= request.
108     getContextPath() %>/statistics/${dis.
109     diseaseName}">${dis.diseaseName}</a></li>
110 </c:forEach>
111 </ul>
112 </li>
113 <li><a href = "<%= request.getContextPath
114     () %>/campaigns">Awareness Campaigns</a></li>
115 </ul>
116 </div>
117 <ul class = "nav navbar-nav navbar-right">
118 <sec:authorize access="isAnonymous()">
119 <c:if test = "${pages != 'register'}">
120 <li><a data-toggle = "modal" href = "#
121     loginForm">Login</a></li>
122 <li><a href = "<%= request.getContextPath
123     () %>/register">Register</a></li>
124 </c:if>
125 </sec:authorize>
126 <sec:authorize access="isAuthenticated()">
127 <div class = "btn-group">
128     Logged in as:
129 <sec:authorize access="hasAnyRole('
130     ROLE_BHWorker')">
131 <a href = "<%= request.getContextPath() %>/
132     pages/profile">Barangay Health Worker</a>
133 </sec:authorize>
134 <sec:authorize access="hasAnyRole('
135     ROLE_MHWorker')">
136 <a href = "<%= request.getContextPath() %>/
137     pages/profile">Municipal/City Health Worker
138 </a>
139 </sec:authorize>
140 <sec:authorize access="hasAnyRole('
141     ROLE_MCOfficial')">
142 <a href = "<%= request.getContextPath() %>/
143     pages/profile">Municipal/City Health
144     Official</a>
145 </sec:authorize>
146 <sec:authorize access="hasAnyRole('
147     ROLE_PHWorker')">
148 <a href = "<%= request.getContextPath() %>/
149     pages/profile">Provincial Health Worker</a>
150 </sec:authorize>
151 <sec:authorize access="hasAnyRole('
152     ROLE_PHOfficial')">
153 <a href = "<%= request.getContextPath() %>/
154     pages/profile">Provincial Health Official</
155     a>
156 </sec:authorize>
157 <sec:authorize access="hasAnyRole('
158     ROLE_NHOfficial')">
159 <a href = "<%= request.getContextPath() %>/pages
160     /profile">National Health Official</a>
161 </sec:authorize>
162 <sec:authorize access="hasAnyRole('ROLE_USER')">
163 <a href = "<%= request.getContextPath() %>/pages
164     /profile">Administrator</a>
165 </sec:authorize>
166 <sec:authorize access="hasAnyRole('
167     ROLE_Researcher')">
168 <a href = "<%= request.getContextPath() %>/pages
169     /profile">Researcher</a>
170 </sec:authorize>
171 </div>
172 <div class = "btn-group">
173 <button href="#" data-toggle="dropdown" class="
174     btn dropdown-toggle">
175 <span class = "glyphicon glyphicon-user"></
176     span> <span class = "caret"></span>
177 </button>
178 <ul class="dropdown-menu">
179 <li><a href = "<%= request.getContextPath()
180     %>/pages/profile"><span>Profile</span></a
181     ></li>
182 <sec:authorize access="hasAnyRole('
183     ROLE_MCOfficial', 'ROLE_PHOfficial', '
184     ROLE_NHOfficial', 'ROLE_USER')">
185 <li><a href = "<%= request.
186     getContextPath() %>/pages/notifications">
187     Notifications</a></li>
188 </sec:authorize>
189 <sec:authorize access="hasAnyRole('

```

```

135     ROLE_BHWorker', 'ROLE_PHWorker', '
        ROLE_MHWorker', 'ROLE_USER')">
136     <li><a href = "%= request.
        getContextPath() %>/pages/system">Manage
        Records</a></li>
137     </sec:authorize>
138     <sec:authorize access="hasAnyRole('
        ROLE_MCOfficial', 'ROLE_PHOfficial', '
        ROLE_NHOfficial', 'ROLE_USER')">
139     <li><a href = "%= request.getContextPath
        () %>/pages/workers">Workers</a></li>
140     </sec:authorize>
141     <sec:authorize access="isAuthenticated()">
142     <li><a href = "%= request.getContextPath
        () %>/pages/reports">Reports</a></li>
143     </sec:authorize>
144     <li><a href = "%= request.getContextPath()
        %>/j_spring_security_logout">Log Out</a></
        li>
145     </ul>
146     </div>
147     </sec:authorize>
148     </ul>
149     </div>
150     </div>
151     <div id = "loginForm" class = "modal fade"
        tabindex = "-1">
152     <div class="modal-header">
153     <button type="button" class="close" data-
        dismiss="modal" aria-hidden="true"> </
        button>
154     <div class = "panel-header">
155     <h4 class = "form-signin-head">Vector Borne
        Diseases Monitoring System</h4>
156     </div>
157     </div>
158     <div class = "modal-body">
159     <form method="post" class = "form-signin"
        action = "j_spring_security_check">
        <table class = "table" >
        <tr>
        <td style = "border: none;">
        <label for="login_username">Username</
        label>
        </td>
        <td style = "border: none;">
        <input type="text" name="j_username" id="
        login_username" class="form-control field
        required" title="Please provide your
        username" />
        </td>
        </tr>
        <tr>
        <td style = "border: none;">
        <label for="login_password">Password</
        label>
        </td>
        <td style = "border: none;">
        <input type="password" name="j_password"
        id="login_password" class="form-control
        field required" title="Password is required
        " />
        </td>
        </tr>
        <tr>
        <td style = "border: none;"><button class =
        "btn btn-primary" type="submit" value = "
        submit">Log in</button></td>
        <td style = "border: none;"><a href =
        "%= request.getContextPath() %>/
        forgotPassword" class = "pull-right">Forgot
        Password?</a></td> -->
        </tr>
        </table>
        </form>
        </div>
        </div>

```

Listing 30: css.jsp

```

1 <link href = "%= request.getContextPath() %>/
    resources/dist/css/jquery-ui.css" rel = "
    stylesheet" media = "screen">
2 <link href = "%= request.getContextPath() %>/
    resources/dist/css/bootstrap.css" rel = "
    stylesheet" media = "screen">
3 <link href = "%= request.getContextPath() %>/
    resources/dist/css/bootstrap.min.css" rel =
    "stylesheet" media = "screen">
4 <link href = "%= request.getContextPath() %>/
    resources/datepicker/css/datepicker.css"
    rel = "stylesheet" media = "screen">
5 <link href = "%= request.getContextPath() %>/
    resources/bootstrap-modal-master/css/
    bootstrap-modal-bs3patch.css" rel = "
    stylesheet" media = "screen">
6 <link href = "%= request.getContextPath() %>/
    resources/bootstrap-modal-master/css/
    bootstrap-modal.css" rel = "stylesheet"
    media = "screen">
7 <link href = "%= request.getContextPath() %>/
    resources/datepicker/css/jquery.datepick.
    css" rel = "stylesheet" media = "screen">
8 <script src = "%= request.getContextPath() %>/
    resources/bootstrap/js/jquery-1.10.2.min.js
    " type = "text/javascript"></script>
9 <script src = "%= request.getContextPath() %>/
    resources/dist/js/clickAdd.js" type = "text
    /javascript"></script>
10 <script src = "%= request.getContextPath() %>/
    resources/dist/js/bootstrap.min.js" type =
    "text/javascript"></script>
11 <script src = "%= request.getContextPath() %>/
    resources/dist/js/moment.min.js" type = "
    text/javascript"></script>
12 <script src = "%= request.getContextPath() %>/
    resources/dist/js/combodate.js" type = "
    text/javascript"></script>
13 <script src = "%= request.getContextPath() %>/
    resources/datepicker/js/bootstrap-
    datepicker.js" type = "text/javascript"></
    script>
14 <script src = "%= request.getContextPath() %>/
    resources/datepicker/js/jquery.plugin.js"
    type = "text/javascript"></script>
15 <script src = "%= request.getContextPath() %>/
    resources/datepicker/js/jquery.datepick.js"
    type = "text/javascript"></script>
16 <script src = "%= request.getContextPath() %>/
    resources/bootstrap-modal-master/js/
    bootstrap-modalmanager.js" type = "text/
    javascript"></script>
17 <script src = "%= request.getContextPath() %>/
    resources/bootstrap-modal-master/js/
    bootstrap-modal.js" type = "text/javascript
    "></script>
18 <script src = "%= request.getContextPath() %>/
    resources/bootstrap/js/jquery.dataTables.
    min.js" type = "text/javascript"></script>
19 <script src = "%= request.getContextPath() %>/
    resources/bootstrap/js/jquery-DT-pagination
    .js" type = "text/javascript"></script>
20 <script src = "%= request.getContextPath() %>/
    resources/maps/locas.js" type = "text/
    javascript"></script>
21 <script src = "%= request.getContextPath() %>/
    resources/maps/maps.js" type = "text/
    javascript"></script>
22 <script src = "%= request.getContextPath() %>/
    resources/dist/js/checkPass.js" type = "
    text/javascript"></script>
23 <script src = "%= request.getContextPath() %>/
    resources/charts/highcharts.js" type = "
    text/javascript"></script>
24 <script src = "%= request.getContextPath() %>/
    resources/charts/modules/exporting.js" type
    = "text/javascript"></script>
25 <script src = "%= request.getContextPath() %>/
    resources/maps/dataTab.js" type = "text/
    javascript"></script>
26 <script src="//maps.google.com/maps/api/js?
    sensor=false&libraries=geometry" type="text
    /javascript"></script>
27 <script type="text/javascript" src="%= request.
    getContextPath() %>/resources/dist/js/
    snapshotcontrol.js"></script>

```

Listing 31: home.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
    jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
    functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
    springframework.org/security/tags" %>
4 <jsp:include page="header.jsp"/>
5 <div class = "container-fluid well-lg" style = "
    border: 1px solid #ddd; background-color: #
    eee;">
6 <jsp:include page = "vbd.jsp" />
7 </div>
8 </div>
9 <jsp:include page="footer.jsp"/>

```

Listing 32: campaigns.jsp

```

1  <%@ taglib prefix="c" uri="http://java.sun.com/
   jsp/jstl/core" %>
2  <%@ taglib uri="http://java.sun.com/jsp/jstl/
   functions" prefix="fn"%>
3  <%@ taglib prefix="sec" uri="http://www.
   springframework.org/security/tags" %>
4  <!DOCTYPE html>
5  <html>
6  <head>
7  <title>Vector-Borne Diseases Monitoring System
   </title>
8  <meta name="viewport" content="width=device-
   width, initial-scale=1.0">
9  <jsp:include page="css.jsp"/>
10 </head>
11 <body>
12 <div class="container">
13 <div class="nav navbar-default" role="
   navigation">
14 <div class="container-fluid">
15 <div class="navbar-header">
16 <button type="button" class="navbar-
   toggle" data-toggle="collapse" data-
   target=".navbar-collapse">
17 <span class="sr-only">Toggle Navigation
   </span>
18 <span class="icon-bar"></span>
19 <span class="icon-bar"></span>
20 <span class="icon-bar"></span>
21 </button>
22 <a href="#"><%= request.getContextPath() %>/
   home" class="navbar-brand">VBDMMS</a>
23 </div>
24 <div class="collapse navbar-collapse">
25 <ul class="nav navbar-nav">
26 <li><a href="#"><%= request.getContextPath
   () %>/home">Home</a></li>
27 <li class="dropdown" style="width: 150
   px; text-align: center;">
28 <a href="#" data-toggle="dropdown"
   class="dropdown-toggle">
29 Disease <span class="caret"></span>
30 </a>
31 <ul class="dropdown-menu" style="text-
   align: left;">
32 <c:forEach var="dis" items="${disease}">
33 <li><a href="#"><%= request.
   getContextPath() %>/disease/${dis.
   diseaseName}>${dis.diseaseName}</a></li>
34 </c:forEach>
35 </ul>
36 </li>
37 <li class="dropdown">
38 <a href="#" data-toggle="dropdown"
   class="dropdown-toggle">
39 Disease Statistics <b class="caret"></b>
40 </a>
41 <ul class="dropdown-menu">
42 <c:forEach var="dis" items="${disease}">
43 <li><a href="#"><%= request.getContextPath
   () %>/statistics/${dis.diseaseName}>${dis.
   diseaseName}</a></li>
44 </c:forEach>
45 </ul>
46 </li>
47 <li class="active"><a href="#"><%= request.
   getContextPath() %>/campaigns">Awareness
   Campaigns</a></li>
48 </ul>
49 <ul class="nav navbar-nav navbar-right">
50 <sec:authorize access="isAnonymous()">
51 <c:if test="${pages != 'register'}">
52 <li><a data-toggle="modal" href="#"
   loginForm">Login</a></li>
53 <li><a href="#"><%= request.getContextPath()
   %>/register">Register</a></li>
54 </c:if>
55 </sec:authorize>
56 <sec:authorize access="isAuthenticated()">
57 <div class="btn-group">
58 <button href="#" data-toggle="dropdown" class
   ="btn dropdown-toggle">
59 <span class="glyphicon glyphicon-user"></
   span> <span class="caret"></span>
60 </button>
61 <ul class="dropdown-menu">
62 <li><a href="#"><%= request.getContextPath()
   %>/pages/profile"><span>Profile</span></a
   ></li>
63 <!--
64 <li><a href="#"><%= request.
   getContextPath() %>/pages/messages">
   Messages</a></li>
65 -->
66 <sec:authorize access="hasAnyRole('
   ROLE_MCOfficial', 'ROLE_PHOfficial', '
   ROLE_NHOfficial', 'ROLE_USER')">
67 <li><a href="#"><%= request.
   getContextPath() %>/pages/notifications">
   Notifications</a></li>
68 </sec:authorize>
69 <sec:authorize access="hasAnyRole('
   ROLE_BHWorker', 'ROLE_PHWorker', '
   ROLE_MHWorker', 'ROLE_USER')">
70 <li><a href="#"><%= request.
   getContextPath() %>/pages/system">Manage
   Records</a></li>
71 </sec:authorize>
72 <sec:authorize access="hasAnyRole('
   ROLE_MCOfficial', 'ROLE_PHOfficial', '
   ROLE_NHOfficial', 'ROLE_USER')">
73 <li><a href="#"><%= request.getContextPath
   () %>/pages/workers" data-toggle="tab">
   Workers</a></li>
74 </sec:authorize>
75 <li><a href="#"><%= request.getContextPath
   () %>/pages/reports">Reports</a></li>
76 <li><a href="#"><%= request.getContextPath
   () %>/j_spring_security_logout">Log Out</a
   ></li>
77 </ul>
78 </div>
79 </sec:authorize>
80 </ul>
81 </div>
82 </div>
83 </div>
84 <div class="modal fade" id="loginForm">
85 <div class="modal-dialog" style="background
   -color: #fff;">
86 <div class="modal-body panel panel-info">
87 <div class="panel panel-heading">
88 <h4 class="form-signin-head">Vector Borne
   Diseases Monitoring System</h4>
89 </div>
90 <form method="post" class="form-signin"
   action="j_spring_security_check">
91 <table class="table">
92 <tr>
93 <td style="border: none;">
94 <label for="login_username">Username</
   label>
95 </td>
96 <td style="border: none;">
97 <input type="text" name="j_username" id="
   login_username" class="form-control field
   required" title="Please provide your
   username" />
98 </td>
99 </tr>
100 <tr>
101 <td style="border: none;">
102 <label for="login_password">Password</
   label>
103 </td>
104 <td style="border: none;">
105 <input type="password" name="j_password"
   id="login_password" class="form-control
   field required" title="Password is required
   " />
106 </td>
107 </tr>
108 <tr>
109 <td style="border: none;"><button class
   ="btn btn-primary" type="submit" value="
   submit">Log in</button></td>
110 </tr>
111 </table>
112 </form>
113 </div>
114 </div>
115 </div>
116 <div class="tabbable">
117 <ul class="nav nav-tabs">
118 <li class="active">
119 <a href="#tabChik" data-toggle="tab">
   Chikungunya</a>
120 </li>
121 <li>
122 <a href="#tabDeng" data-toggle="tab">
   Dengue</a>
123 </li>
124 <li>
125 <a href="#tabMal" data-toggle="tab">
   Malaria</a>
126 </li>
127 </ul>
128 </div>
129
130 <div class="tab-content">
131 <div class="tab-pane active" id="tabChik">
132 <br />
133 <div class="container-fluid well well-lg"
   style="border: 1px solid #ddd; background
   -color: #eee;">

```

```

134 <div class = "panel panel-info">
135 <div class = "panel-body">
136 <div class="thumbnail with-caption pull-
right img-responsive" style = "margin-left:
20px;">
137 <img src = "<%= request.getContextPath()
%>/resources/dist/img/chik.jpg"
138 width="350" height = "250" class = "img-
responsive" />
139 <p>
140 <small>dilg.gov.ph</small>
141 </p>
142 </div>
143 <p class = "text-justify">
144 Chikungunya is a viral disease that is
transmitted to people by mosquitoes.
Symptoms
145 usually appear 3-7 days after the
infection and these may include
146 sudden high fever, severe joint and muscle
pain. Often headaches,
147 fatigue, joint swelling and skin rash
could also be experienced.
148 </p>
149 <p class = "text-justify">
150 There is no vaccine or drugs to prevent
this infection. However,
151 medication is used to reduce the symptoms.
Even though chikungunya
152 usually doesn't cause death, some infected
people may develop long-
153 term joint pain.
154 </p>
155 <p class = "text-justify"><strong>
156 The best way to prevent Chikungunya viral
infection is to avoid
157 mosquito bites along with getting rid of
all potential mosquito
158 breeding sites from your surroundings.
159 </strong></p>
160 <p>
161 Additionally, you can:
162 </p>
163 <ul>
164 <li>Use insect repellent on exposed skin.
Always follow the directions on the package
.</li>
165 <li>Wear long sleeves and long pants
particularly when going out doors.</li>
166 <li>Sleep under a mosquito net.</li>
167 <li>Have secure screens on windows and
doors to keep mosquitoes out.</li>
168 <li>Cooperate with local vector control
initiatives.</li>
169 </ul>
170 <p class = "text-justify"><strong>
171 A person with chikungunya fever should
limit their exposure to
172 mosquito bites to avoid further spreading
the infection.
173 </strong></p>
174 <p class = "text-justify">
175 Stored water for agriculture and gardening
purposes are potential breeding
176 sites for mosquitoes. Therefore, all water
holding receptacles such
177 as buckets, drums, black PVC tanks should
be covered with mesh or
178 cloth-nets. Fishes also can be placed in
these reservoirs as a
179 mosquito control measure, since they feed
on mosquito larvae.
180 </p>
181 <p class = "text-justify">
182 Likewise swimming pools can pose a hazard
if they are not properly
183 maintained or neglected. If you are unable
to maintain your swimming
184 pool, ask the Vector Control Unit about
using fish as a mosquito
185 control measure.
186 </p>
187 <p class = "text-left">
188 <small><em><strong>Source: </strong> <a
href = "http://www.gov.ms/2014/01/31/
189 health-bulletin-chikungunya-awareness-and-
prevention/">http://www.gov.ms/2014/01/31/
health-bulletin-chikungunya-awareness-and-
prevention/</a></em></small>
190 </p>
191 </div>
192 </div>
193 </div>
194 </div>
195 <div class = "tab-pane" id = "tabDeng">
196 <br />
197 <div class = "container-fluid well well-lg"
style = "border: 1px solid #ddd; background
-color: #eee;">
198 <div class = "panel panel-info">
199 <div class = "panel-body">
200 <div class="thumbnail with-caption pull-
right img-responsive" style = "margin-left:
20px;">
201 <img src = "<%= request.getContextPath()
%>/resources/dist/img/doh-dengue.jpg"
202 width="450" height = "350" class = "img-
responsive" />
203 <p>
204 <small>pinoykidsmd.com</small>
205 </p>
206 </div>
207 <p class = "text-justify">
208 To prevent the spread of dengue fever,
you must first prevent the breeding of its
vector, the Aedes mosquitoes. The Aedes
mosquito is easily identifiable by its
distinctive black and white stripes on
their body. It prefers to breed in clean,
stagnant water easily found in our homes.
You can get rid of the Aedes mosquito by
frequently checking and removing stagnant
water in your premises.
The guidelines below will give you an
overview of how you can prevent the Aedes
mosquito from breeding.
</p>
<ul>
<li>Keep your homes and offices protected
against mosquitoes.</li>
<li>Keep homes and offices airy, bright
and safe from moisture.</li>
<li>Fix nets on doors and windows.</li>
<li>Wear full sleeves clothes.</li>
<li>Use mosquito nets while sleeping.</li>
<li>Don't leave the overhead tanks open
.</li>
<li>Don't keep water in containers for
more than a week. Instead, empty them every
week, let them dry and then fill again.</li>
<li>Don't let the water falling from the
overhead tanks to accumulate permanently.
Instead dry it.</li>
<li>Don't let the water accumulate in any
case both inside or outside the home.</li>
<li>Be mindful of your home's cleanliness
.</li>
<li>Be mindful of your home's cleanliness
.</li>
<li>Keep the fence and hedge boundaries
duly cut both inside and outside the home,
and spray over them with insecticides,
particularly in the evening.</li>
<li>Don't let the water stay all the time
in the flower pots, gamias of plants.
Instead water them only in the morning
every alternate day.</li>
<li>The best way to fight dengue is
through sustained cleanliness and
sanitation.</li>
</ul>
<p class = "text-left">
<small><em><strong>Source: </strong> <a
href = "http://www.taclobanhoteles.com/
CAMPAIGN-AGAINST-DENGUE.htm">http://www.
taclobanhoteles.com/CAMPAIGN-AGAINST-DENGUE.
htm</a></em></small>
</p>
</div>
</div>
</div>
<div class = "tab-pane" id = "tabMal">
<br />
<div class = "container-fluid well well-lg"
style = "border: 1px solid #ddd; background
-color: #eee;">
<div class = "panel panel-info">
<div class = "panel-body">
<div class="thumbnail with-caption pull-
right img-responsive" style = "margin-left:
20px;">
<img src = "<%= request.getContextPath()
%>/resources/dist/img/malariacycle.jpg"
width="400" height = "300" class = "img-
responsive" />
<p>
<small>www.mangalorian.com</small>
</p>
</div>
</div>
<p class = "text-justify">
<strong>Malaria in the Philippines</strong>
</p>
<p class = "text-justify">
Philippines is a country known for its

```

```

252 rich ecosystem. The biological
environment in this country is
253 prosperous enough that more
tourists and researchers are flocking in
254 this Pacific country.
Philippines boasts of biodiversities and
conservations like
255 forests, mountains, hills, coral reefs,
caves, waterfalls and
256 more. Because of the rich ecosystem,
there are a lot of species
257 too. But sometimes, some of these
species can be really
258 dangerous especially the insects.
Are you familiar with
259 Malaria?
260 </p>
261 <p class = "text-justify">
262 In a statistics by World Health
Organization, over 12
263 million people in the Philippines (
almost 13% of the population)
264 are at high-risk of malaria causing
death. Most malaria cases in
265 the country occur in swampy, hilly,
mountainous regions and
266 forested area that is infested with
Plasmodium and any other
267 deadly species. Plasmodium species can
transfer virus by
268 puncturing the human skin where the
virus will then flow on the
269 human's bloodstream, spreading in the
whole body. Malaria can
270 kill an individual in just an hour, if
the one being infected is
271 not aware that he's been bitten. So in
order to stop the
272 increasing cases of malaria, the World
Health Organization,
273 together with the Department of Health
created various programs
274 that tackled about treatments and
awareness about the disease.
275 </p>
276 <p class = "text-justify">
277 In the year 1990, 86,200+ cases of
malaria in the Philippines
278 were recorded. It's a very alarming
report that made WHO act
279 immediately. Nurses and doctors were
deployed on community areas
280 with malaria outbreak. Equipment were
also transferred on these
281 places to treat the affected patients
more effectively. The
282 operation went well and successful.
Because of the the
283 consistent advocacy of the the
international organization,
284 malaria cases in this country was said
to have plummeted every
285 year. WHO and the government provided
millions of dollars to put
286 this project up. Every effort made by
the joining parties paid
287 off when Cebu, Catanduanes, and Leyte
were declared of being a
288 malaria-free city. Currently, the World
Health Organization
289 keeps on eliminating this virus in
various parts of the country
and other countries as well.
290 </p>
291 <p class = "text-justify"><strong>
292 People must remember these simple
293 prevention steps from the World Health
294 Organization:
</strong></p>
295 <p class = "text-justify">
296 Sleep in a
297 room that has screens that can protect
298 you from mosquitos. If
299 there are a lot of mosquitos in your
300 place, be sure to have
mosquito repellent sprays or lotion for
301 protection. Practice
hygiene and sanitation. If you see some
302 signs and symptoms of
malaria, rush the individual to the
303 nearest clinic or hospital.
Make sure that the authorities in your
304 place will know about the
malaria outbreak (if there is such) as
305 soon as possible. They
say that prevention is better than cure
306 so make sure that you're
taking precautionary measures to avoid
being infected with
307 malaria.
308 </p>
309 <p class = "text-left">
310 <small><em><strong>Source: </strong> <a
311 href = "http://www.who.org.ph/malaria.
html">http://www.who.org.ph/malaria.html</a
></em></small>
312 </p>
313 </div>
314 </div>
315 </div>
316 </div>
317 </div>
318 </div>
319 </div>
320 </div>
321 <jsp:include page = "footer.jsp" />

```

Listing 33: diseases.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
springframework.org/security/tags" %>
4 <!DOCTYPE html>
5 <html>
6 <head>
7 <title>Vector-Borne Diseases Monitoring System
</title>
8 <meta name="viewport" content="width=device-
width, initial-scale=1.0">
9 <jsp:include page="css.jsp"/>
10 <style type="text/css">
11 .style1 {background-color:#ffffff;font-weight:
bold;border:2px #006699 solid;}
12 </style>
13 <script src="<%= request.getContextPath() %>/
resources/maps/maps3.js" type="text/
javascript"></script>
14 <script type = "text/javascript">
15 function initialize() {
16 var cityCircle;
17 var latlng = new google.maps.LatLng(9.1732164,
124.7298765);
18 var num = '<c:out value = "${fn:length(maps)}'
/>';
19 var i = 0;
20 var arrayX = new Array(num);
21 var arrayY = new Array(num);
22 var gry = new Array(num);
23 var countCases = new Array(num);
24 var countDeaths = new Array(num);
25 var pops = new Array(num);
26 var coords = new Array(num);
27 <c:forEach var = "map" items = "${maps}">
28 arrayX[i] = '<c:out value = "${map.x}"/>';
29 arrayY[i] = '<c:out value = "${map.y}"/>';
30 gry[i] = '<c:out value = "${map.barangay}"/>';
31 countCases[i] = '<c:out value = "${map.
countCases}"/>';
32 countDeaths[i] = '<c:out value = "${map.
countDeaths}"/>';
33 pops[i] = '<c:out value = "${map.population
}"/>';
34 coords[i] = new google.maps.LatLng(arrayY[i],
arrayX[i]);
35 i++;
36 </c:forEach>
37
38
39 var myOptions = {
40 zoom: 5,
41 center: latlng,
42 mapTypeId: google.maps.MapTypeId.ROADMAP
};
43
44
45 var map = new google.maps.Map(document.
getElementById("map_canvas"),
myOptions);
46
47 // add control
48 var snapOpts = {
49 map: map
50 };
51
52 var snapShotControl = new SnapShotControl(
snapOpts);
53
54 for(var j = 0; j < num; j++) {
55 marker = new google.maps.Marker({
56 map: map,
57 title: "Barangay " + gry[j] + "\nNo. of
Cases: " + countCases[j] + "\nNo. of Deaths
: " + countDeaths[j] +
58 "\nPopulation: " + pops[j],

```

```

59     draggable: false,
60     animation: google.maps.Animation.DROP,
61     position: coords[j],
62     icon: new google.maps.MarkerImage(
63         "http://chart.googleapis.com/chart?chst=
        d_bubble_text_small&chld=bb|" + gry[j]+ "|
        FF8080|000000",
64         null, null, new google.maps.Point(0, 42)),
65     shadow: new google.maps.MarkerImage(
66         "http://chart.googleapis.com/chart?chst=
        d_bubble_text_small_shadow&chld=bb|" + gry[
        j]+ " ",
67         null, null, new google.maps.Point(0, 45))
68     });
69 }
70
71 for(var j = 0; j < num; j++) {
72     var stroke = "";
73     var fill = "";
74     // #FF0000
75     // #FF0000
76
77     if(countCases < pops[j]/5){
78         stroke = "#91D8EE";
79         fill = "#91D8EE";
80     }
81     else if(countCases < pops[j]/4){
82         stroke = "#C1F3F3";
83         fill = "#C1F3F3";
84     }
85     else if(countCases < pops[j]/3){
86         stroke = "#FFC398";
87         fill = "#FFC398";
88     }
89     else if(countCases < pops[j]/2){
90         stroke = "#FAA99A";
91         fill = "#FAA99A";
92     }
93     else {
94         stroke = "#E68C98";
95         fill = "#E68C98";
96     }
97
98     var populationOptions = {
99         strokeColor: "#FAA99A",
100        strokeOpacity: 0.8,
101        strokeWeight: 2,
102        fillColor: "#FAA99A",
103        fillOpacity: 0.35,
104        map: map,
105        center: coords[j],
106        radius: 350
107    };
108    cityCircle = new google.maps.Circle(
109        populationOptions);
110 }
111 }
112 </script>
113 </head>
114 <body onload = "initialize()">
115 <div class = "container">
116 <div class = "navbar navbar-default" role = "
    navigation">
117 <div class = "container-fluid">
118 <div class = "navbar-header">
119 <button type = "button" class = "navbar-
    toggle" data-toggle = "collapse" data-
    target = ".navbar-collapse">
120 <span class = "sr-only">Toggle Navigation
    </span>
121 <span class = "icon-bar"></span>
122 <span class = "icon-bar"></span>
123 <span class = "icon-bar"></span>
124 </button>
125 <a href = "<%= request.getContextPath()
    %>/home"><img src = "<%= request.
    getContextPath() %>/resources/dist/img/
    logo2.jpg" alt = "VBDM Logo" height = "30
    px;" width = "93px;" class = "img-
    responsive"/></a>
126 </div>
127 <div class = "collapse navbar-collapse">
128 <ul class = "nav navbar-nav">
129 <li><a href = "<%= request.getContextPath
    () %>/home">Home</a></li>
130 <li class = "active dropdown" style = "
    width: 150px; text-align: center;">
131 <a href = "" data-toggle = "dropdown"
    class = "dropdown-toggle">
132 <span class = "caret"></
    span>
133 </a>
134 <ul class="dropdown-menu" style = "text-
    align: left;">
135 <c:forEach var="dis" items="{dis
    }">
136 <c:if test = "${dis.diseaseName !=
    diseaseName}">
137 <li><a href = "<%= request.
    getContextPath() %>/disease/${dis.
    diseaseName}">${dis.diseaseName}</a></li>
138 </c:if>
139 </c:forEach>
140 </ul>
141 </li>
142 <li class = "dropdown">
143 <a href = "" data-toggle = "dropdown"
    class = "dropdown-toggle">
144 Disease Statistics <b class = "caret"></b>
145 </a>
146 <ul class="dropdown-menu">
147 <c:forEach var="dis" items="{dis}">
148 <li><a href = "<%= request.getContextPath
    () %>/statistics/${dis.diseaseName}">${dis.
    diseaseName}</a></li>
149 </c:forEach>
150 </ul>
151 </li>
152 <li><a href = "<%= request.getContextPath()
    %>/campaigns">Awareness Campaigns</a></li>
153 </ul>
154 <ul class = "nav navbar-nav navbar-right">
155 <sec:authorize access="isAnonymous()">
156 <c:if test = "${pages != 'register'}">
157 <li><a data-toggle = "modal" href = "#
    loginForm">Login</a></li>
158 <li><a href = "<%= request.getContextPath()
    %>/register">Register</a></li>
159 </c:if>
160 </sec:authorize>
161 <sec:authorize access="isAuthenticated()">
162 <div class = "btn-group">
163 Logged in as:
164 <sec:authorize access="hasAnyRole('
    ROLE_BHWorker')">
165 <a href = "<%= request.getContextPath() %>/
    pages/profile">Barangay Health Worker</a>
166 </sec:authorize>
167 <sec:authorize access="hasAnyRole('
    ROLE_MHWorker')">
168 <a href = "<%= request.getContextPath() %>/
    pages/profile">Municipal/City Health Worker
    </a>
169 </sec:authorize>
170 <sec:authorize access="hasAnyRole('
    ROLE_MCOfficial')">
171 <a href = "<%= request.getContextPath() %>/
    pages/profile">Municipal/City Health
    Official</a>
172 </sec:authorize>
173 <sec:authorize access="hasAnyRole('ROLE_PHWorker
    ')">
174 <a href = "<%= request.getContextPath() %>/pages
    /profile">Provincial Health Worker</a>
175 </sec:authorize>
176 <sec:authorize access="hasAnyRole('
    ROLE_PHOfficial')">
177 <a href = "<%= request.getContextPath() %>/pages
    /profile">Provincial Health Official</a>
178 </sec:authorize>
179 <sec:authorize access="hasAnyRole('
    ROLE_NHOfficial')">
180 <a href = "<%= request.getContextPath() %>/pages
    /profile">National Health Official</a>
181 </sec:authorize>
182 <sec:authorize access="hasAnyRole('ROLE_USER')">
183 <a href = "<%= request.getContextPath() %>/pages
    /profile">Administrator</a>
184 </sec:authorize>
185 <sec:authorize access="hasAnyRole('
    ROLE_Researcher')">
186 <a href = "<%= request.getContextPath() %>/pages
    /profile">Researcher</a>
187 </sec:authorize>
188 </div>
189 <div class = "btn-group">
190 <button href="#" data-toggle="dropdown" class="
    btn dropdown-toggle" >
191 <span class = "glyphicon glyphicon-user"></
    span> <span class = "caret"></span>
192 </button>
193 <ul class="dropdown-menu">
194 <li><a href = "<%= request.getContextPath()
    %>/pages/profile"><span>Profile</span></a>
195 </li>
196 <!--
197 <li><a href = "<%= request.
    getContextPath() %>/pages/messages">
    Messages</a></li>
198 -->
199 <sec:authorize access="hasAnyRole('
    ROLE_MCOfficial', 'ROLE_PHOfficial', '
    ROLE_NHOfficial', 'ROLE_USER')">
200 <li><a href = "<%= request.
    getContextPath() %>/pages/notifications">

```

```

201     Notifications</a></li>
202     </sec:authorize>
203     <sec:authorize access="hasAnyRole('
204     ROLE_BHWorker', 'ROLE_PHWorker', '
205     ROLE_MHWorker', 'ROLE_USER')">
206     <li><a href = "<%= request.
207     getContextPath() %>/pages/system">Manage
208     Records</a></li>
209     </sec:authorize>
210     <sec:authorize access="hasAnyRole('
211     ROLE_MCOfficial', 'ROLE_PHOfficial', '
212     ROLE_NHOfficial', 'ROLE_USER')">
213     <li><a href = "<%= request.getContextPath
214     () %>/pages/workers">Workers</a></li>
215     </sec:authorize>
216     <sec:authorize access="isAuthenticated()">
217     <li><a href = "<%= request.getContextPath
218     () %>/pages/reports">Reports</a></li>
219     </sec:authorize>
220     <li><a href = "<%= request.getContextPath()
221     %>/j_spring_security_logout">Log Out</a></li>
222     </ul>
223     </div>
224     </sec:authorize>
225     </ul>
226     </div>
227     </div>
228     <div id = "loginForm" class = "modal fade"
229     tabindex = "-1">
230     <div class="modal-header">
231     <button type="button" class="close" data-
232     dismiss="modal" aria-hidden="true"> </
233     button>
234     <div class = "panel-header">
235     <h4 class = "form-signin-head">Vector Borne
236     Diseases Monitoring System</h4>
237     </div>
238     </div>
239     <div class = "modal-body">
240     <form method="post" class = "form-signin"
241     action = "/j_spring_security_check">
242     <table class = "table" >
243     <tr>
244     <td style = "border: none;">
245     <label for="login_username">Username</
246     label>
247     </td>
248     <td style = "border: none;">
249     <input type="text" name="j_username" id="
250     login_username" class="form-control field
251     required" title="Please provide your
252     username" />
253     </td>
254     </tr>
255     <tr>
256     <td style = "border: none;">
257     <label for="login_password">Password</
258     label>
259     </td>
260     <td style = "border: none;">
261     <input type="password" name="j_password"
262     id="login_password" class="form-control
263     field required" title="Password is required
264     " />
265     </td>
266     </tr>
267     <tr>
268     <td style = "border: none;">
269     <input type="password" name="j_password"
270     id="login_password" class="form-control
271     field required" title="Password is required
272     " />
273     </td>
274     <td style = "border: none;">
275     <button class =
276     "btn btn-primary" type="submit" value = "
277     submit">Log in</button></td>
278     <td style = "border: none;"><a href =
279     "<%= request.getContextPath() %>/
280     forgotPassword" class = "pull-right">Forgot
281     Password?</a></td> -->
282     </tr>
283     </table>
284     </form>
285     </div>
286     </div>
287     <div class = "container-fluid well well-lg"
288     style = "border: 1px solid #ddd; background
289     -color: #eee;">
290     <div id="map_canvas" style="margin-left: 20px;
291     width:500px; height:500px" class = "pull-
292     right thumbnail">
293     </div>
294     <c:if test = "${diseaseName == 'Chikungunya'}">
295     <jsp:include page = "chikungunya.jsp" />
296     </c:if>
297     <c:if test = "${diseaseName == 'Dengue'}">
298     <jsp:include page = "dengue.jsp" />
299     </c:if>
300     <c:if test = "${diseaseName == 'Malaria'}">
301     <jsp:include page = "malaria.jsp" />
302     </c:if>
303     </div>
304     </div>
305     <jsp:include page = "footer.jsp" />

```

Listing 34: chikungunya.jsp

```

1 <div class="center-block well-lg" style = "
2 background-color: #fff;">
3 <h2>Chikungunya</h2>
4 <p class = "text-justify">
5 <strong><em>Chikungunya</em></strong> is a
6 viral disease that
7 causes fever and severe joint paint. Sharing
8 clinical signs with
9 dengue, it is usually misdiagnosed in areas
10 where dengue is common. It
11 is transmitted by infected mosquitoes, usually
12 the genus Aedes.
13 </p>
14 <br />
15 <ol type = "a">
16 <li><strong>Signs and Symptoms</strong></li>
17 <br />
18 <p class = "text-justify">
19 Aside from abrupt onset of fever frequently
20 accompanied by joint pains, other signs and
21 symptoms of chikungunya
22 include muscle pain, headache, nausea,
23 fatigue and rash. Joint paint
24 usually lasts within a few days or weeks and
25 while most patients
26 recover fully from this, some cases may
27 persist for several months, or
28 even years. Occasionally, cases of eye,
29 neurological and heart
30 complications as well as gastrointestinal
31 complaints have been
32 reported. This disease not often cause
33 serious complications but can
34 contribute to death among older population.
35 </p>
36 <br />
37 <li><strong>Mode Of Transmission</strong></li>
38 </ol>
39 <p class = "text-justify">Chikungunya virus is
40 transmitted to humans from infected
41 non-human primates and other humans by the
42 bite of Aedes
43 mosquitoesFootnote 1. Evidence exists that
44 CHIKV can also be passed
45 from an infected mother to a developing
46 foetusFootnote 2 Footnote 13.
47 Furthermore, inhalation of aerosolised
48 chikungunya virus in a laboratory setting
49 may lead to chikungunya virus infection.</p>
50 <br />
51 <li><strong>Incubation Period</strong></li>
52 <p class="text-justify">Usually 2 to 3
53 daysFootnote 1 Footnote 2,
54 with a range of 1 to 10 days</p>
55 <br />
56 <li><strong>Communicability</strong></li>
57 <p class="text-justify">Usually 2 to 3
58 daysFootnote 1 Footnote 2,
59 with a range of 1 to 10 days</p>
60 <br />
61 <li><strong>Treatment</strong></li>
62 <p class = "text-justify">
63 <br />
64 Since there are no specific drugs to cure
65 the disease, relieving the symptoms
66 including the joint pains are directed
67 primarily. There is no
68 commercial chikungunya vaccine.
69 </p>
70 <p class = "text-left">
71 <small><em><strong>Source: </strong><a
72 href = "http://www.phac-aspc.gc.ca/lab-
73 bio/res/psds-ftss/chikungunya-eng.php">http
74 ://www.phac-aspc.gc.ca/lab-bio/res/psds-
75 ftss/chikungunya-eng.php</a></em></small>
76 </p>
77 </ol>

```

Listing 35: dengue.jsp

```

1 <div class="center-block well-lg" style = "
2 background-color: #fff;">
3 <h2>Dengue</h2>
4 <p class = "text-justify"><strong><em>Dengue</
5 em></strong> is a mosquito-borne viral
6 infection
7 that causes u-like illness, and occa-sionally
8 develops into a
9 potentially lethal complication called severe
10 dengue. It is 12 found

```



```

6   in tropical and sub-tropical climates and
7   mostly in urban and
8   semi-urban areas. It is transmitted by an
9   infected Aedes mosquito
10  (Aedes aegypti or Aedes albopictus).
11  </p>
12  <br />
13  <ol type = "a">
14  <li><strong>Signs and Symptoms</strong></li>
15  <p class = "text-justify">
16  <br />
17  When running a high fever (40&ordmC/100&ordmF
18  ) and is accompanied by two symptoms,
19  namely, severe headache, pain behind the eyes
20  , muscle and joint pains,
21  nausea, vomiting, swollen glands or rash,
22  dengue should be suspected.
23  These symptoms usually last for 2 to 7 days;
24  that is after 4-10 days
25  of incubation period after the bite from an
26  infected mosquito. A
27  potentially deadly complication from dengue,
28  severe dengue, is due to
29  plasma leaking, uid accommodation,
30  respiratory diseases, severe
31  bleeding, or organ impairment. After 3-7 days
32  when the rst symptoms in
33  conjunction with a decrease in temperature of
34  below 38&ordmC/100&ordmF, warning
35  sign occurs that includes severe abdominal
36  pain, persistent vomiting,
37  rapid breathing, bleeding gums, fatigue,
38  restlessness and blood in
39  vomit. Proper medical care is needed to avoid
40  complications and risk
41  of death especially in the next 24-48 hours
42  of the critical stage.
43  </p>
44  <br />
45  <li><strong>Mode of Transmission</strong></li>
46  <p class = "text-justify">
47  <br />
48  Dengue is transmitted by the bite of an
49  infected mosquito,
50  particularly Aedes aegypti. This was first
51  recognised by workers in
52  Queensland early in the 20th century. Aedes
53  aegypti breeds in fresh
54  water and particularly in man made containers
55  such as old tyres, pot
56  plant holders, buckets and tree hollows in
57  urban areas. Aedes
58  albopictus is a mosquito common in South East
59  Asia and Papua New
60  Guinea and can also be an important vector.
61  Other Aedes species are
62  involved in the enzootic monkey cycle.
63  </p>
64  <br />
65  <li><strong>Period of Communicability</strong>
66  </li>
67  <p class = "text-justify">
68  <br />
69  There is no evidence of person to person
70  transmission.
71  </p>
72  <br />
73  <li><strong>Treatment</strong></li>
74  <p class = "text-justify">
75  <br />
76  There is no speci c treatment for dengue
77  fever. For severe
78  dengue, medical care by physicians and nurses
79  experienced with the
80  effects and progression of the disease can
81  save lives. Maintenance of
82  the patients body fluid volume is critical to
83  severe dengue care.
84  </p>
85  <p class = "text-left">
86  <small><em><strong>Source: </strong><a
87  href = "http://ideas.health.vic.gov.au/
88  bluebook/dengue.asp">http://ideas.health.
89  vic.gov.au/bluebook/dengue.asp</a></em></
90  small>
91  </p>
92  </ol>
93  </div>

```

Listing 36: malaria.jsp

```

1  <div class = "center-block well-lg" style = "
2  background-color: #fff;">
3  <h2>Malaria</h2>
4  <p class = "text-justify">
5  <strong><em>Malaria</em></strong>, though
6  preventable and curable, is
7  a life-threatening disease caused by parasites
8  that are transmitted to
9  people through the bites of infected
10 mosquitoes. The burden of malaria
11 in many places has been dramatically reduced
12 by increased malaria
13 prevention and control measures. The
14 Plasmodiumparasite causes malaria
15 to humans and is transmitted by Anopheles
16 mosquitoes.
17 </p>
18 <br />
19 <ol type = "a">
20 <li><strong>Signs and Symptoms</strong></li>
21 <p class = "text-justify">
22 <br />
23 Malaria is an illness characterized with
24 acute fever. Symptoms appear 7
25 days or more (10-15 days) for non-immune
26 individuals after being bitten
27 by an infected mosquito. The first symptoms
28 include fever, headache,
29 chills and vomiting which may be mild and
30 difficult to be recognized as
31 malaria. Residents in malaria endemic areas
32 may develop partial
33 immunity which allows asymptomatic infections
34 to occur.
35 </p>
36 <li><strong>Treatment</strong></li>
37 <p class = "text-justify">
38 <br />
39 Early diagnosis and treatment of malaria
40 reduces disease and prevents
41 deaths. It also contributes to reducing
42 malaria transmission. The best
43 available treatment, particularly for P.
44 falciparum malaria, is
45 artemisinin-based combina-tion therapy (ACT)
46 </p>
47 </ol>
48 </div>

```

Listing 37: statistics.jsp

```

1  <%@ taglib prefix="c" uri="http://java.sun.com/
2  jsp/jstl/core" %>
3  <%@ taglib uri="http://java.sun.com/jsp/jstl/
4  functions" prefix="fn"%>
5  <%@ taglib prefix="sec" uri="http://www.
6  springframework.org/security/tags" %>
7  <%@ taglib prefix="fmt" uri="http://java.sun.com
8  /jsp/jstl/fmt" %>
9  <!DOCTYPE html>
10 <html>
11 <head>
12 <title>Vector-Borne Diseases Monitoring System
13 </title>
14 <meta name="viewport" content="width=device-
15 width, initial-scale=1.0">
16 <jsp:include page="css.jsp"/>
17 <script type="text/javascript">
18 $(function () {
19     var d=new Date();
20     var month=new Array();
21     month[0]="January";
22     month[1]="February";
23     month[2]="March";
24     month[3]="April";
25     month[4]="May";
26     month[5]="June";
27     month[6]="July";
28     month[7]="August";
29     month[8]="September";
30     month[9]="October";
31     month[10]="November";
32     month[11]="December";
33     var n = month[d.getMonth()];
34     var years = d.getFullYear();
35     $('#containsLine').highcharts({
36     title: {
37         text: '<c:out value = "${titles}" />',
38         x: -20 //center
39     },
40     subtitle: {
41         text: 'Philippines, ' + n + ", " +
42         years,
43         x: -20
44     },
45     xAxis: {
46         categories: ['Jan', 'Feb', 'Mar',
47         'Apr', 'May', 'Jun',
48         'Jul', 'Aug', 'Sep', 'Oct', 'Nov',

```

```

41     'Dec']
42     },
43     yAxis: {
44         title: {
45             text: 'Number of Person'
46         },
47         plotLines: [{
48             value: 0,
49             width: 1,
50             color: '#808080'
51         }],
52         tooltip: {
53             valueSuffix: ''
54         },
55         legend: {
56             layout: 'vertical',
57             align: 'right',
58             verticalAlign: 'middle',
59             borderWidth: 0
60         },
61         series: [{
62             name: 'Cases',
63             data: <:out value = "${data1}" />
64         }, {
65             name: 'Deaths',
66             data: <:out value = "${data2}" />
67         }]
68     });
69 });
70 </script>
71 </head>
72 <body>
73     <div class = "container">
74         <div class = "navbar navbar-default" role = "
75         navigation">
76             <div class = "container-fluid">
77                 <div class = "navbar-header">
78                     <button type = "button" class = "navbar-
79                     toggle" data-toggle = "collapse" data-
80                     target = ".navbar-collapse">
81                         <span class = "sr-only">Toggle Navigation
82                     </span>
83                     <span class = "icon-bar"></span>
84                     <span class = "icon-bar"></span>
85                     <span class = "icon-bar"></span>
86                 </div>
87                 <a href = "<%= request.getContextPath() %>/
88                 home"><img src = "<%= request.
89                 getContextPath() %>/resources/dist/img/
90                 logo2.jpg" alt = "VBDM Logo" height = "30
91                 px;" width = "93px;" class = "img-
92                 responsive"/></a>
93             </div>
94             <div class = "collapse navbar-collapse">
95                 <ul class = "nav navbar-nav">
96                     <li <c:if test = "${pages == 'home'}">
97                         class = "active"</c:if>
98                     <a href = "<%= request.getContextPath()
99                     %>/home">Home</a></li>
100                     <li class = "dropdown">
101                         <a href = "" data-toggle = "dropdown"
102                         class = "dropdown-toggle">
103                             Diseases <b class = "caret"></b>
104                         </a>
105                         <ul class="dropdown-menu">
106                             <c:forEach var="dis" items="${disease
107                             }">
108                                 <li><a href = "<%= request.
109                                 getContextPath() %>/disease/${dis.
110                                 diseaseName}">${dis.diseaseName}</a></li>
111                             </c:forEach>
112                         </ul>
113                     </li>
114                     <li class = "active dropdown" style = "
115                     width: 150px; text-align: center;">
116                         <a href = "" data-toggle = "dropdown"
117                         class = "dropdown-toggle">
118                             ${diseaseName} <span class = "caret"></
119                             span>
120                         </a>
121                         <ul class="dropdown-menu" style = "text-
122                         align: left;">
123                             <c:forEach var="dis" items="${disease}">
124                                 <c:if test = "${dis.diseaseName !=
125                                 diseaseName}">
126                                     <li><a href = "<%= request.
127                                     getContextPath() %>/statistics/${dis.
128                                     diseaseName}">${dis.diseaseName}</a></li>
129                                 </c:if>
130                                 <c:forEach>
131                                     </ul>
132                             </li>
133                             <li><a href = "<%= request.getContextPath()
134                             %>/campaigns">Awareness Campaigns</a></li>
135                         </ul>
136                     </li>
137                 <ul class = "nav navbar-nav navbar-right">
138                     <sec:authorize access="isAnonymous()">
139                         <c:if test = "${pages != 'register'}">
140                             <li><a data-toggle = "modal" href = "#
141                             loginForm">Login</a></li>
142                             <li><a href = "<%= request.getContextPath()
143                             %>/register">Register</a></li>
144                         </c:if>
145                     </sec:authorize>
146                     <sec:authorize access="isAuthenticated()">
147                         <div class = "btn-group">
148                             Logged in as:
149                             <sec:authorize access="hasAnyRole('
150                             ROLE_BHWorker')">
151                                 <a href = "<%= request.getContextPath() %>/
152                                 pages/profile">Barangay Health Worker</a>
153                             </sec:authorize>
154                             <sec:authorize access="hasAnyRole('
155                             ROLE_MHWorker')">
156                                 <a href = "<%= request.getContextPath() %>/
157                                 pages/profile">Municipal/City Health Worker
158                                 </a>
159                             </sec:authorize>
160                             <sec:authorize access="hasAnyRole('
161                             ROLE_MCOfficial')">
162                                 <a href = "<%= request.getContextPath() %>/
163                                 pages/profile">Municipal/City Health
164                                 Official</a>
165                             </sec:authorize>
166                             <sec:authorize access="hasAnyRole('ROLE_PHWorker
167                             ')">
168                                 <a href = "<%= request.getContextPath() %>/pages
169                                 /profile">Provincial Health Worker</a>
170                             </sec:authorize>
171                             <sec:authorize access="hasAnyRole('
172                             ROLE_PHOfficial')">
173                                 <a href = "<%= request.getContextPath() %>/pages
174                                 /profile">Provincial Health Official</a>
175                             </sec:authorize>
176                             <sec:authorize access="hasAnyRole('
177                             ROLE_NHOfficial')">
178                                 <a href = "<%= request.getContextPath() %>/pages
179                                 /profile">National Health Official</a>
180                             </sec:authorize>
181                             <sec:authorize access="hasAnyRole('ROLE_USER')">
182                                 <a href = "<%= request.getContextPath() %>/pages
183                                 /profile">Administrator</a>
184                             </sec:authorize>
185                             <sec:authorize access="hasAnyRole('
186                             ROLE_Researcher')">
187                                 <a href = "<%= request.getContextPath() %>/pages
188                                 /profile">Researcher</a>
189                             </sec:authorize>
190                         </div>
191                     <div class = "btn-group">
192                         <button href="#" data-toggle="dropdown" class="
193                         btn dropdown-toggle">
194                             <span class = "glyphicon glyphicon-user"></
195                             span> <span class = "caret"></span>
196                         </button>
197                         <ul class="dropdown-menu">
198                             <li><a href = "<%= request.getContextPath()
199                             %>/pages/profile"><span>Profile</span></a>
200                             </li>
201                             <sec:authorize access="hasAnyRole('
202                             ROLE_MCOfficial', 'ROLE_PHOfficial', '
203                             ROLE_NHOfficial', 'ROLE_USER')">
204                                 <li><a href = "<%= request.
205                                 getContextPath() %>/pages/notifications">
206                                 Notifications</a></li>
207                             </sec:authorize>
208                             <sec:authorize access="hasAnyRole('
209                             ROLE_BHWorker', 'ROLE_PHWorker', '
210                             ROLE_MHWorker', 'ROLE_USER')">
211                                 <li><a href = "<%= request.
212                                 getContextPath() %>/pages/system">Manage
213                                 Records</a></li>
214                             </sec:authorize>
215                             <sec:authorize access="hasAnyRole('
216                             ROLE_MCOfficial', 'ROLE_PHOfficial', '
217                             ROLE_NHOfficial', 'ROLE_USER')">
218                                 <li><a href = "<%= request.getContextPath()
219                                 %>/pages/workers">Workers</a></li>
220                             </sec:authorize>
221                             <sec:authorize access="isAuthenticated()">
222                                 <li><a href = "<%= request.getContextPath()
223                                 %>/pages/reports">Reports</a></li>
224                             </sec:authorize>
225                             <li><a href = "<%= request.getContextPath()
226                                 %>/j_spring_security_logout">Log Out</a></
227                             li>
228                         </ul>
229                     </div>
230                 </ul>
231             </div>
232         </div>
233         <div class = "modal fade" id = "loginForm">
234             <div class = "modal-dialog" style = "background

```



```

/2010-census-population-and-housing-reveals 341 </div>
-philippine-population-9234-million">http 342 </div>
://www.census.gov.ph/content/2010-census- 343 </div>
population-and-housing-reveals-philippine- 344 </div>
population-9234-million</a></em></small> 345 <jsp:include page="footer.jsp"/>
340 </p>

```

Listing 38: vbd.jsp

```

1 <div class="center-block well-lg" style = "
  background-color: #fff;">
2 <h2>Vector-Borne Diseases</h2>
3 <div class="thumbnail with-caption pull-right
  img-responsive" style = "margin-left: 20px
  ;">
4 <img src = "<%= request.getContextPath()%>/
  resources/dist/img/images.jpg"
5 width="320" class = "img-responsive" />
6 <p>
7 <small>www.philstar.com</small>
8 </p>
9 </div>
10 <p class="text-justify">
11 <br/>
12 <em><strong>"Vector-borne disease"</strong></
  em> is the term commonly used to
13 describe an illness caused by an infectious
  microbe that is
14 transmitted to people by blood-sucking
  arthropods. The arthropods
15 (insects or arachnids) that most commonly
  serve as vectors include:
16 <ul>
17 <li>blood sucking insects such as mosquitoes,
  fleas, lice, biting
18 flies and bugs, and</li>
19 <li>blood sucking arachnids such as mites
  and ticks.</li>
20 </ul>
21 The term "vector" refers to any arthropod
  that transmits a disease
  through feeding activity.
22 </p>
23 <p class="text-justify">
24 <br/>
25 Vectors typically become infected by a
26 disease agent while
27 feeding on infected vertebrates (e.g., birds,
  rodents, other larger
28 animals, or humans), and then pass on the
29 microbe to a susceptible
30 person or other animal. In almost all cases,
  an infectious microbe
31 must infect and multiply inside the arthropod
  before the arthropod is
32 able to transmit the disease through its
  salivary glands. The most
33 common vector-borne diseases in Virginia are
  carried by mosquitoes and
34 ticks.
35 </p>
36 <p class = "text-left">
37 <small><em><strong>Source: </strong> <a
  href = "http://www.vdh.state.va.us/
  epidemiology/dee/vectorborne/">http://www.
  vdh.state.va.us/epidemiology/dee/
  vectorborne/</a></em></small>
38 </p>
39 </div>

```

Listing 39: register.jsp

```

1 <%@ taglib uri="http://www.springframework.org/
  tags/form" prefix="form"%>
2 <%@ taglib prefix="c" uri="http://java.sun.com/
  jsp/jstl/core" %>
3 <%@ taglib uri="http://java.sun.com/jsp/jstl/
  functions" prefix="fn"%>
4 <jsp:include page="header.jsp"/>
5 <div class="panel panel-info">
6 <div class="panel-heading">Registration Form</
  div>
7 <div class = "panel-body">
8 <div class = "alert alert-info">
9 <font color = "red">* </font><i>Required
  fields</i>
10 </div>
11 <form:form method = "POST" action = "
  registerUser" modelAttribute = "registers"
  id = "regist">
12 <table class = "table table-striped">
13 <tbody>
14 <tr>
15 <td>
16 <label for="firstname" >First Name<font
  color = "red"> *</font></label>
17 </td>
18 <td>
19 <form:input path="firstname" type = "text"
  class = "form-control" id = "firstname"
  required = "true" maxlength = "50"/>
20 <c:set var="eFirstname"><form:errors path
  ="firstname"/></c:set>
21 <c:if test="{not empty eFirstname}">
22 <div class = "alert alert-danger alert-
  dismissable">
23 <button type="button" class="close" data-
  dismiss="alert" aria-hidden="true">&times
  ;</button>
24 <strong><form:errors path = "firstname
  "></form:errors></strong>
25 </div>
26 </c:if>
27 </td>
28 <td>
29 <label for = "middlename">Middle Name</
  label>
30 </td>
31 <td>
32 <form:input path = "middlename" type = "
  text" class = "form-control"/>
33 <c:set var="eMiddlename"><form:errors path
  ="middlename"/></c:set>
34 <c:if test="{not empty eMiddlename}">
35 <div class = "alert alert-danger alert-
  dismissable">
36 <button type="button" class="close" data-
  dismiss="alert" aria-hidden="true">&times
  ;</button>
  dismiss="alert" aria-hidden="true">&times
  ;</button>
  <strong><form:errors path = "middlename
  "></form:errors></strong>
  </div>
  </c:if>
  <td>
  <label for="gender">Gender</label>
  <td>
  <form:select path = "gender" class = "form-
  control">
  <option></option>
  <option>Male</option>
  <option>Female</option>
  </form:select>
  <c:set var="eGender"><form:errors path="gender
  "/></c:set>
  <c:if test="{not empty eGender}">
  <div class = "alert alert-danger alert-
  dismissable">
  <button type="button" class="close" data-
  dismiss="alert" aria-hidden="true">&times
  ;</button>
  <strong><form:errors path = "gender"></form:
  errors></strong>
  </div>
  </c:if>
  </td>
  <tr>
  <td>
  <label for="address">Home Address</label>

```

```

77 </td>
78 <td colspan = '6'>
79 <form:textarea path = "address" type = "text"
      class = "form-control" rows = "3"/>
80 <c:set var="eAddress"><form:errors path="
      address"/></c:set>
81 <c:if test="${not empty eAddress}">
82 <div class = "alert alert-danger alert-
      dismissable">
83 <button type="button" class="close" data-
      dismiss="alert" aria-hidden="true">&times
      ;</button>
84 <strong><form:errors path = "address"></form:
      errors></strong>
85 </div>
86 </c:if>
87 </td>
88 </tr>
89 <tr>
90 <td>
91 <label for="usertype">Register as<font color =
      "red"> *</font></label>
92 </td>
93 <td>
94 <form:select path = "userType" class = "form-
      control" id = "usertype" required = "true"
      >
95 <option></option>
96 <option <c:if test = "${registers.userType ==
      'Barangay Health Worker'}">selected = "
      selected"</c:if>>Barangay Health Worker</
      option>
97 <option <c:if test = "${registers.userType ==
      'Municipal/City Health Worker'}">selected =
      "selected"</c:if>>Municipal/City Health
      Worker</option>
98 <option <c:if test = "${registers.userType ==
      'Municipal/City Health Official'}">selected
      = "selected"</c:if>>Municipal/City Health
      Official</option>
99 <option <c:if test = "${registers.userType ==
      'Provincial Health Worker'}">selected = "
      selected"</c:if>>Provincial Health Worker</
      option>
100 <option <c:if test = "${registers.userType ==
      'Provincial Health Official'}">selected = "
      selected"</c:if>>Provincial Health Official
      </option>
101 <option <c:if test = "${registers.userType ==
      'National Health Official'}">selected = "
      selected"</c:if>>National Health Official</
      option>
102 <option <c:if test = "${registers.userType ==
      'Researcher'}">selected = "selected"</c:if
      >>Researcher</option>
103 </form:select>
104 <c:set var="eUserType"><form:errors path="
      userType"/></c:set>
105 <c:if test="${not empty eUserType}">
106 <div class = "alert alert-danger alert-
      dismissable">
107 <button type="button" class="close" data-
      dismiss="alert" aria-hidden="true">&times
      ;</button>
108 <strong><form:errors path = "userType"></form:
      errors></strong>
109 </div>
110 </c:if>
111 </td>
112 <td>
113 <label for="username" >Username<font color = "
      red"> *</font></label>
114 </td>
115 <td>
116 <form:input path="username" type = "text" class
      = "form-control" id = "username" required
      = "true" maxlength = "20"/>
117 <c:set var="eUsername"><form:errors path="
      username"/></c:set>
118 <c:if test="${not empty eUsername}">
119 <div class = "alert alert-danger alert-
      dismissable">
120 <button type="button" class="close" data-
      dismiss="alert" aria-hidden="true">&times
      ;</button>
121 <strong><form:errors path = "username"></form:
      errors><br />${nameTaken}</strong>
122 </div>
123 </c:if>
124 <c:if test="${not empty nameTaken}">
125 <div class = "alert alert-danger alert-
      dismissable">
126 <button type="button" class="close" data-
      dismiss="alert" aria-hidden="true">&times
      ;</button>
127 <strong>${nameTaken}</strong>
128 </div>
129 </c:if>
130 </td>
131 </tr>
132 <tr>
133 <td>
134 <label for = "region">Region<font color = "red
      "> *</font></label>
135 </td>
136 <td>
137 <form:select path = "region" class = "form-
      control" id = "region" onChange = "
      regionSelect()" required = "true">
138 <option></option>
139 <option value = "ARMM" <c:if test = "${
      registers.region == 'ARMM'}">selected = "
      selected"</c:if>>ARMM</option>
140 <option value = "CAR" <c:if test = "${
      registers.region == 'CAR'}">selected = "
      selected"</c:if>>CAR</option>
141 <option value = "NCR" <c:if test = "${
      registers.region == 'NCR'}">selected = "
      selected"</c:if>>NCR</option>
142 <option value = "Region I" <c:if test = "${
      registers.region == 'Region I'}">selected =
      "selected"</c:if>>Region I</option>
143 <option value = "Region II" <c:if test = "${
      registers.region == 'Region II'}">selected
      = "selected"</c:if>>Region II</option>
144 <option value = "Region III" <c:if test = "${
      registers.region == 'Region III'}">selected
      = "selected"</c:if>>Region III</option>
145 <option value = "Region IV-A" <c:if test = "${
      registers.region == 'Region IV-A'}">
      selected = "selected"</c:if>>Region IV-A</
      option>
146 <option value = "Region IV-B" <c:if test = "${
      registers.region == 'Region IV-B'}">
      selected = "selected"</c:if>>Region IV-B</
      option>
147 <option value = "Region V" <c:if test = "${
      registers.region == 'Region V'}">selected =
      "selected"</c:if>>Region V</option>
148 <option value = "Region VI" <c:if test = "${
      registers.region == 'Region VI'}">selected
      = "selected"</c:if>>Region VI</option>
149 <option value = "Region VII" <c:if test = "${
      registers.region == 'Region VII'}">selected
      = "selected"</c:if>>Region VII</option>
150 <option value = "Region VIII" <c:if test = "${
      registers.region == 'Region VIII'}">
      selected = "selected"</c:if>>Region VIII</
      option>
151 <option value = "Region IX" <c:if test = "${
      registers.region == 'Region IX'}">selected
      = "selected"</c:if>>Region IX</option>
152 <option value = "Region X" <c:if test = "${
      registers.region == 'Region X'}">selected =
      "selected"</c:if>>Region X</option>
153 <option value = "Region XI" <c:if test = "${
      registers.region == 'Region XI'}">selected
      = "selected"</c:if>>Region XI</option>
154 <option value = "Region XII" <c:if test = "${
      registers.region == 'Region XII'}">selected
      = "selected"</c:if>>Region XII</option>
155 <option value = "Region XIII" <c:if test = "${
      registers.region == 'Region XIII'}">
      selected = "selected"</c:if>>Region XIII</
      option>
156 </form:select>
157 <c:set var="eRegion"><form:errors path="region
      "/></c:set>
158 <c:if test="${not empty eRegion}">
159 <div class = "alert alert-danger alert-
      dismissable">
160 <button type="button" class="close" data-
      dismiss="alert" aria-hidden="true">&times
      ;</button>
161 <strong><form:errors path = "region"></form:
      errors></strong>
162 </div>
163 </c:if>
164 </td>
165 <td>
166 <label for = "province">Province<font color = "
      red"> *</font></label>
167 </td>
168 <td>
169 <form:select path = "province" type = "text"
      class = "form-control" id = "province"
      onChange = "provSelect()" required = "true"
      disabled = "${empty registers.province}">
170 <c:if test = "${not empty registers.province}">
171 <option>${registers.province}</option>
172 </c:if>
173 </form:select>
174 <c:set var="eProvince"><form:errors path="
      province"/></c:set>
175 <c:if test="${not empty eProvince}">
176 <div class = "alert alert-danger alert-
      dismissable">
177 <button type="button" class="close" data-

```

```

                dismiss="alert" aria-hidden="true">&times
                ;</button>
178 <strong><form:errors path = "province"></form:
                errors></strong>
179 </div>
180 </c:if>
181 </td>
182 </tr>
183 <tr>
184 <td>
185 <label for = "muncity">City/Municipality<font
                color = "red"> *</font></label>
186 </td>
187 <td>
188 <form:select path = "muncity" type = "text"
                class = "form-control" id = "muncity"
                onChange = "muncityChange()" required = "
                true" disabled = "${empty registers.muncity
                }">
189 <c:if test = "${not empty registers.muncity}">
190 <option>${registers.muncity}</option>
191 </c:if>
192 </form:select>
193 <c:set var="eMuncity"><form:errors path="muncity
                "/></c:set>
194 <c:if test="${not empty eMuncity}">
195 <div class = "alert alert-danger alert-
                dismissable">
196 <button type="button" class="close" data-
                dismiss="alert" aria-hidden="true">&times
                ;</button>
197 <strong><form:errors path = "muncity"></form:
                errors></strong>
198 </div>
199 </c:if>
200 </td>
201 <td>
202 <label for = "barangay">Barangay<font color = "
                red"> *</font></label>
203 </td>
204 <td>
205 <form:select path = "barangay" type = "text"
                class = "form-control" id = "barangay"
                required = "true" disabled = "${empty
                registers.barangay}">
206 <c:if test = "${not empty registers.barangay}">
207 <option>${registers.barangay}</option>
208 </c:if>
209 </form:select>
210 <c:set var="eBarangay"><form:errors path="
                barangay"/></c:set>
211 <c:if test="${not empty eBarangay}">
212 <div class = "alert alert-danger alert-
                dismissable">
213 <button type="button" class="close" data-
                dismiss="alert" aria-hidden="true">&times
                ;</button>
214 <strong><form:errors path = "barangay"></form:
                errors></strong>
215 </div>
216 </c:if>
217 </td>
218 </tr>
219 <tr>
220 <td>
221 <label for="institution">Institution<font
                color = "red"> *</font></label>
222 </td>
223 <td colspan = '6'>
224 <form:textarea path = "institution" class = "
                form-control" rows = "2" type = "text" id =
                "institution" required = "true" maxlength
                = "50" />
225 <c:set var="eInstitution"><form:errors path="
                institution"/></c:set>
226 <c:if test="${not empty eInstitution}">
227 <div class = "alert alert-danger alert-
                dismissable">
228 <button type="button" class="close" data-
                dismiss="alert" aria-hidden="true">&times
                ;</button>
229 <strong><form:errors path = "institution"></
                form:errors></strong>
230 </div>
231 </c:if>
232 </td>
233 </tr>
234 <tr>
235 <td>
236 <label for="institutionAddr">Institution's
                Address<font color = "red"> *</font></label
                >
                </td>
237 </td>
238 <td colspan = '6'>
239 <form:textarea path = "institutionAddr" type =
                "text" class = "form-control" rows = "3"
                id = "institutionAddr" required = "true"
                maxlength = "100"/>
240 <c:set var="eInstitutionAddr"><form:errors
                path="institutionAddr"/></c:set>
241 <c:if test="${not empty eInstitutionAddr}">
242 <div class = "alert alert-danger alert-
                dismissable">
243 <button type="button" class="close" data-
                dismiss="alert" aria-hidden="true">&times
                ;</button>
244 <strong><form:errors path = "institutionAddr
                "></form:errors></strong>
245 </div>
246 </c:if>
247 </td>
248 </tr>
249 <tr>
250 <td>
251 <label for="email">Email<font color = "red">
                *</font></label>
252 </td>
253 <td>
254 <form:input path = "email" type = "email"
                class = "form-control" id = "email"
                required = "true" maxlength = "30"/>
255 <c:set var="eEmail"><form:errors path="email
                "/></c:set>
256 <c:if test="${not empty eEmail}">
257 <div class = "alert alert-danger alert-
                dismissable">
258 <button type="button" class="close" data-
                dismiss="alert" aria-hidden="true">&times
                ;</button>
259 <strong><form:errors path = "email"></form:
                errors></strong>
260 </div>
261 </c:if>
262 </td>
263 <td>
264 <label for="number">Contact Number<font color =
                "red"> *</font></label>
265 </td>
266 <td>
267 <form:input path = "number" type = "text" class
                = "form-control" maxlength = "11" id = "
                number" required = "true"/>
268 <c:set var="eNumber"><form:errors path="number
                "/></c:set>
269 <c:if test="${not empty eNumber}">
270 <div class = "alert alert-danger alert-
                dismissable">
271 <button type="button" class="close" data-
                dismiss="alert" aria-hidden="true">&times
                ;</button>
272 <strong><form:errors path = "number"></form:
                errors><br />${numTaken}</strong>
273 </div>
274 </c:if>
275 <c:if test="${not empty numTaken}">
276 <div class = "alert alert-danger alert-
                dismissable">
277 <button type="button" class="close" data-
                dismiss="alert" aria-hidden="true">&times
                ;</button>
278 <strong>${numTaken}</strong>
279 </div>
280 </c:if>
281 </td>
282 </tr>
283 </tbody>
284 </table>
285 <br />
286 <br />
287 <center><button type="submit" class = "btn btn-
                primary" value="Registration" id = "sBut">
                Register</button></center>
288 <br />
289 <br />
290 <br />
291 </form:form>
292 </div>
293 </div>
294 <jsp:include page="footer.jsp"/>

```

Listing 40: successRegister.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
  jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
  functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
  springframework.org/security/tags" %>
  <jsp:include page="header.jsp"/>
  <div class = "modal-dialog">
  <div class = "panel panel-info">
  <div class = "panel-heading">Successful
  Registration!</div>

```

```

9     <div class = "panel-body">
10     Thank you for registering! You will receive
        an email for your registration's approval
        together with the
11     system generated username and password which
        you can modify anytime.

```

Listing 41: linkings.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
    jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
    functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
    springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
    springframework.org/tags/form" %>
5
6 <ul class = "nav nav-tabs">
7 <sec:authorize access="hasAnyRole('
    ROLE_BHWorker', 'ROLE_MCHWorker', '
    ROLE_PHWorker')">
8 <li <c:if test = "${thePage == 'system'}">class
    = "active"</c:if>
9 >><a href = "<%= request.getContextPath() %>/
    pages/system">Manage Records</a>
10 </li>
11 </sec:authorize>
12 <sec:authorize access="hasAnyRole('
    ROLE_MCOfficial', 'ROLE_PHOfficial', '
    ROLE_NHOfficial', 'ROLE_USER')">
13 <li <c:if test = "${thePage == 'notifications
    '}">class = "active"</c:if>
14 >><a href = "<%= request.getContextPath() %>/
    pages/notifications">Notifications</a>

```

Listing 42: profile.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
    jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
    functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
    springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
    springframework.org/tags/form" %>
5 <jsp:include page = "../header.jsp" />
6
7 <div class = "modal fade" id = "changeUP">
8 <div class = "modal-header">
9 <button type="button" class="close" data-
    dismiss="modal" aria-hidden="true"> </
    button>
10 <h4 class = "form-signin-head">Vector Borne
    Diseases Monitoring System</h4>
11 </div>
12 <form:form method="post" class = "form-signin"
    action = "${pageContext.request.
    contextPath}/pages/changeUserDetails"
    modelAttribute = "changeUser">
13 <div class = "modal-body">
14 <table class = "table" >
15 <tr>
16 <td style = "border: none;">
17 <label for="login_password">Old Password:</
    label>
18 </td>
19 <td style = "border: none;">
20 <form:input path = "accountPassword" type="
    password" class="form-control" title="
    Password is required" />
21 </td>
22 </tr>
23 <tr>
24 <td style = "border: none;">
25 <label for="login_password">New Password:</
    label>
26 </td>
27 <td style = "border: none;">
28 <form:input path = "accountPassword2" type
    ="password" id = "oldPassword" class="form-
    control" title="Password is required" />
29 </td>
30 </tr>
31 <tr>
32 <td style = "border: none;">
33 <label for="login_password">New Password:</
    label>
34 </td>
35 <td style = "border: none;">
36 <form:input path = "accountPassword3" type
    ="password" id = "newPassword" class="form-
    control" title="Password is required" />
37 </td>
38 </tr>
39 <tr colspan = '6'>
40 <td style = "border: none;" colspan = '2'>
41 <div id = "confirmPass" class = "alert

```

```

12 </div>
13 </div>
14 </div>
15 </div>
16 <jsp:include page="footer.jsp"/>
17
18 </li>
19 </sec:authorize>
20 <sec:authorize access="hasAnyRole('
    ROLE_MCOfficial', 'ROLE_PHOfficial', '
    ROLE_NHOfficial', 'ROLE_USER')">
21 <li <c:if test = "${thePage == 'workers'}">class
    = "active"</c:if>
22 >><a href = "<%= request.getContextPath() %>/
    pages/workers">Health Workers and Officials
    </a>
23 </li>
24 </sec:authorize>
25 <sec:authorize access="isAuthenticated()">
26 <li <c:if test = "${thePage == 'reports'}">class
    = "active"</c:if>
27 >><a href = "<%= request.getContextPath() %>/
    pages/reports">Reports</a>
28 </li>
29 </sec:authorize>
30 <sec:authorize access="isAuthenticated()">
31 <li <c:if test = "${thePage == 'profile'}">class
    = "active"</c:if>
32 >><a href = "<%= request.getContextPath() %>/
    pages/profile">Profile</a>
33 </li>
34 </sec:authorize>
35 </ul>

```

```

42 <div class = "alert-danger">
43 </div>
44 </td>
45 </tr>
46 </table>
47 </div>
48 <div class = "modal-footer">
49 <button class = "btn btn-primary" type="submit"
    value = "submit">Change</button>
50 </div>
51 </form:form>
52 </div>
53 <div class = "modal fade" id = "changeContacts">
54 <div class = "modal-header">
55 <button type="button" class="close" data-
    dismiss="modal" aria-hidden="true"> </
    button>
56 <h4 class = "form-signin-head">Vector Borne
    Diseases Monitoring System</h4>
57 </div>
58 <form:form method="post" class = "form-signin"
    action = "${pageContext.request.
    contextPath}/pages/changeUserContacts"
    modelAttribute = "changeContacts">
59 <div class = "modal-body">
60 <table class = "table" >
61 <tr>
62 <td style = "border: none;">
63 <label>Email:</label>
64 </td>
65 <td style = "border: none;">
66 <form:input path = "email" type="text"
    class="form-control" value = "${user.email
    }" />
67 </td>
68 </tr>
69 <tr>
70 <td style = "border: none;">
71 <label for="login_password">Contact Number
    </label>
72 </td>
73 <td style = "border: none;">
74 <form:input path = "number" type="text"
    class="form-control" value = "${user.number
    }" />
75 </td>
76 </tr>
77 </table>
78 </div>
79 <div class = "modal-footer">
80 <button class = "btn btn-primary" type="submit"
    value = "submit">Change</button>
81 </div>
82 </form:form>
83 </div>
84 <div class = "tabbable">
85 <jsp:include page = "linkings.jsp" />
86 <div class = "tab-content">
87 <div class = "tab-pane active">
88 <br />
89 <div class = "container-fluid well-lg" style
    = "background-color: #eee;">

```

```

89     <div class = "panel panel-default">
90     <br /><br />
91     <center><img src = "<%= request.
getContextPath() %>/resources/dist/img/doh-
logo.jpg" alt = "Profile Picture" height =
"150" width = "150" class = "img-responsive
"/></center>
92     <br />
93
94     <div class = "modal-dialog">
95     <div class = "container-fluid well-lg"
style = "background-color: #eee;">
96     <div class = "panel panel-info">
97     <div class = "panel-heading">
98     Account Information
99     </div>
100    <div class = "panel-body">
101    <div class = "row">
102    <div class = "col-lg-6">
103    <label>Username: </label>
104    </div>
105    <div class = "col-lg-6">
106    ${userDetails.accountName}
107    </div>
108    </div>
109    <br />
110    <div class = "row">
111    <button type = "button" data-toggle =
"modal" href = "#changeUP" class = "btn btn
-primary pull-right">Change Password</
button>
112    </div>
113    </div>
114    </div>
115    <br />
116    <div class = "panel panel-info">
117    <div class = "panel-heading">
118    Institutional Information
119    </div>
120    <div class = "panel-body">
121    <div class = "row">
122    <div class = "col-lg-6">
123    <label>User Type: </label>
124    </div>
125    <div class = "col-lg-6">
126    ${user.userType}
127    </div>
128    </div>
129    <br />
130    <div class = "row">
131    <div class = "col-lg-6">
132    <label>Region: </label>
133    </div>
134    <div class = "col-lg-6">
135    ${user.region}
136    </div>
137    </div>
138    <br />
139    <div class = "row">
140    <div class = "col-lg-6">
141    <label>Province: </label>
142    </div>
143    <div class = "col-lg-6">
144    ${user.province}
145    </div>
146    </div>
147    <br />
148    <div class = "row">
149    <div class = "col-lg-6">
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Listing 43: reports.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
springframework.org/tags/form" %>
5 <jsp:include page = "../header.jsp" />
6 <div class = "tabbable">
7 <jsp:include page = "linkings.jsp" />
8 <div class = "tab-content">
9 <div class = "tab-pane active">
10 <div class = "tabbable">
11 <ul class = "nav nav-pills">
12 <li <c:if test = "${empty reportsPage}">
class = "active" </c:if>>
13 <a href = "#taba" data-toggle = "tab">
Export Reports</a>
14 </li <c:if test = "${not empty reportsPage
}">class = "active" </c:if>>
15 </li>
16 <a href = "#tabb" data-toggle = "tab">
Export Data</a>
17 </li>
18 </ul>
19 </div>
20 <br />
21 <div class = "tab-content">
22 <div class = 'tab-pane <c:if test = "${empty
reportsPage}">active</c:if>' id = "taba">
23 <jsp:include page = "reports/pdf.jsp" />
24 </div>
25 <div class = 'tab-pane <c:if test = "${not
empty reportsPage}">active</c:if>' id = "
tabb">
26 <jsp:include page = "reports/excel.jsp" />
27 </div>
28 </div>
29 </div>
30 </div>
31 </div>
32 </div>
33 <jsp:include page = "../footer.jsp" />

```


Listing 44: excel.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
  jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
  functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
  springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
  springframework.org/tags/form" %>
5
6 <div class = "container-fluid well-lg" style = "
  background-color: #eee;">
7 <div class = "panel panel-default">
8 <div class = "panel-heading">
9 Generate reports
10 </div>
11 <div class = "panel-body">
12 <form:form action = "${pageContext.request.
  contextPath}/pages/generateExcel" id = "
  form1" modelAttribute = "forExcel" method =
  "POST">
13 <div class = "row">
14 <div class = "col-lg-6">
15 <label>INCLUDE DATES</label>
16 </div>
17 </div>
18 <div class = "row">
19 <div class = "col-lg-3">
20 <label>From:</label>
21 </div>
22 <div class = "col-lg-3">
23 <div class="input-group date" id="dp3" data-
  -date="12-2012" data-date-format="yyyy-mm-
  dd">
24 <form:input path = "dateFrom" class="form-
  control" type="text" value="2014-01-01" />
25 <span class="input-group-addon"><i class="
  glyphicon glyphicon-calendar"></i></span>
26 </div>
27 <c:set var="eDateFrom"><form:errors path="
  dateFrom"/></c:set>
28 <c:if test="${not empty eDateFrom}">
29 <div class = "alert alert-danger alert-
  dismissable">
30 <button type="button" class="close" data-
  dismiss="alert" aria-hidden="true">&times
  ;</button>
31 <strong><form:errors path = "dateFrom"></
  form:errors></strong>
32 </div>
33 </c:if>
34 </div>
35 <div class = "col-lg-3">
36 <label>To:</label>
37 </div>
38 <div class = "col-lg-3">
39 <div class="input-group date" id="dp4" data-
  -date="12-2012" data-date-format="yyyy-mm-dd
  ">
40 <form:input path = "dateTo" class="form-
  control" type="text" value="2014-01-01" />
41 <span class="input-group-addon"><i class="
  glyphicon glyphicon-calendar"></i></span>
42 </div>
43 <c:set var="eDateTo"><form:errors path="
  dateTo"/></c:set>
44 <c:if test="${not empty eDateTo}">
45 <div class = "alert alert-danger alert-
  dismissable">
46 <button type="button" class="close" data-
  dismiss="alert" aria-hidden="true">&times
  ;</button>
47 <strong><form:errors path = "dateTo"></form
  :errors></strong>
48 </div>
49 </c:if>
50 </div>
51 </div>
52 <br />
53 <div class = "row">
54 <div class = "col-lg-3">
55 <label>Disease</label>
56 </div>
57 <div class = "col-lg-3">
58 <form:select path = "disease" class = "form-
  control" >
59 <option></option>
60 <option>Chikungunya</option>
61 <option>Dengue</option>
62 <option>Malaria</option>
63 </form:select>
64 <c:set var="eDisease"><form:errors path="
  disease"/></c:set>
65 <c:if test="${not empty eDisease}">
66 <div class = "alert alert-danger alert-
  dismissable">
67 <button type="button" class="close" data-
  dismiss="alert" aria-hidden="true">&times
  ;</button>
68 <strong><form:errors path = "disease"></form:
  errors></strong>
69 </div>
70 </c:if>
71 </div>
72 </div>
73 <br />
74 <sec:authorize access="hasAnyRole('
  ROLE_Researcher')">
75 <div class = "row">
76 <div class = "col-lg-6">
77 <div class = "input-group">
78 <span class = "input-group-addon">
79 <form:checkbox path = "specRegion" id = "
  checksNew"/>
80 </span>
81 <input type = "text" value = "For Specific
  Region: " class = "form-control" style = "
  background-color: #fff;" disabled />
82 </div>
83 </div>
84 <div class = "col-lg-6">
85 <form:select path = "region" class = "form-
  control" disabled = "disabled" id = "
  select4">
86 <option></option>
87 <option>ARMM</option>
88 <option>CAR</option>
89 <option>NCR</option>
90 <option>Region I</option>
91 <option>Region II</option>
92 <option>Region III</option>
93 <option>Region IV-A</option>
94 <option>Region IV-B</option>
95 <option>Region V</option>
96 <option>Region VI</option>
97 <option>Region VII</option>
98 <option>Region VIII</option>
99 <option>Region IX</option>
100 <option>Region X</option>
101 <option>Region XI</option>
102 <option>Region XII</option>
103 <option>Region XIII</option>
104 </form:select>
105 <c:set var="eRegion"><form:errors path="region
  "/></c:set>
106 <c:if test="${not empty eRegion}">
107 <div class = "alert alert-danger alert-
  dismissable">
108 <button type="button" class="close" data-
  dismiss="alert" aria-hidden="true">&times
  ;</button>
109 <strong><form:errors path = "region"></form:
  errors></strong>
110 </div>
111 </c:if>
112 <c:if test="${not empty error}">
113 <div class = "alert alert-danger alert-
  dismissable">
114 <button type="button" class="close" data-
  dismiss="alert" aria-hidden="true">&times
  ;</button>
115 <strong>${error}</strong>
116 </div>
117 </c:if>
118 </div>
119 </div>
120 </div>
121 </sec:authorize>
122 <br />
123 <div class = "row">
124 <div class = "col-lg-4">
125 </div>
126 <div class = "col-lg-4">
127 <button class = "btn btn-primary" type = "
  submit">Generate Excel</button>
128 </div>
129 </div>
130 </div>
131 </form:form>
132 </div>
133 </div>

```

Listing 45: pdf.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
  jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
  functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
  springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
  springframework.org/tags/form" %>
5
6 <div class = "container-fluid well-lg" style = "

```

```

7         background-color: #eee;">
8 <div class = "panel panel-default">
9 <div class = "panel-heading">
10 Generate reports
11 </div>
12 <div class = "panel-body">
13 <form:form action = "${pageContext.request.
14 contextPath}/pages/generateReports" id = "
15 form1" modelAttribute = "forReports" method
16 = "POST">
17 <div class = "row">
18 <div class = "col-lg-6">
19 <label>INCLUDE DATES</label>
20 </div>
21 </div>
22 <div class = "row">
23 <div class = "col-lg-3">
24 <label>From:</label>
25 </div>
26 <div class = "col-lg-3">
27 <div class="input-group date" id="dp1" data-
28 date="12-2012" data-date-format="yyyy-mm-
29 dd">
30 <form:input path = "dateFrom" class="form-
31 control" type="text" value="2014-01-01" />
32 <span class="input-group-addon"><i class="
33 glyphicon glyphicon-calendar"></i></span>
34 </div>
35 <c:set var="eDateFrom"><form:errors path="
36 dateFrom"/></c:set>
37 <c:if test="${not empty eDateFrom}">
38 <div class = "alert alert-danger alert-
39 dismissable">
40 <button type="button" class="close" data-
41 dismiss="alert" aria-hidden="true">&times
42 ;</button>
43 <strong><form:errors path = "dateFrom"></
44 form:errors></strong>
45 </div>
46 </c:if>
47 </div>
48 <div class = "col-lg-3">
49 <label>To:</label>
50 </div>
51 <div class = "col-lg-3">
52 <div class="input-group date" id="dp2" data-
53 date="12-2012" data-date-format="yyyy-mm-dd
54 ">
55 <form:input path = "dateTo" class="form-
56 control" type="text" value="2014-01-01" />
57 <span class="input-group-addon"><i class="
58 glyphicon glyphicon-calendar"></i></span>
59 </div>
60 <c:set var="eDateTo"><form:errors path="
61 dateTo"/></c:set>
62 <c:if test="${not empty eDateTo}">
63 <div class = "alert alert-danger alert-
64 dismissable">
65 <button type="button" class="close" data-
66 dismiss="alert" aria-hidden="true">&times
67 ;</button>
68 <strong><form:errors path = "dateTo"></form
69 :errors></strong>
70 </div>
71 </c:if>
72 </div>
73 </div>
74 <br />
75 <div class = "row">
76 <div class = "col-lg-6">
77 <label>Include following charts:</label>
78 </div>
79 <br />
80 <div class = "row">
81 <div class = "col-lg-6">
82 <div class = "input-group">
83 <span class = "input-group-addon">
84 <form:checkbox path = "ageGroup" />
85 </span>
86 <input type = "text" value = "Disease Case by
87 Age Group" class = "form-control" style =
88 "background-color: #fff;" disabled />
89 </div>
90 </div>
91 <div class = "col-lg-6">
92 <div class = "input-group">
93 <span class = "input-group-addon">
94 <form:checkbox path = "fatalAgeGroup" />
95 </span>
96 <input type = "text" value = "Cases Fatality
97 Ratio by Age Group" class = "form-control"
98 style = "background-color: #fff;" disabled
99 />
100 </div>
101 </div>
102 </div>
103 <br />
104 <div class = "row">
105 <div class = "col-lg-6">
106 <div class = "input-group">
107 <span class = "input-group-addon">
108 <form:checkbox path = "genderGroup" />
109 </span>
110 <input type = "text" value = "Disease Case by
111 Gender Group" class = "form-control" style
112 = "background-color: #fff;" disabled />
113 </div>
114 </div>
115 </div>
116 <div class = "col-lg-6">
117 <div class = "input-group">
118 <span class = "input-group-addon">
119 <form:checkbox path = "byRegion" />
120 </span>
121 <input type = "text" value = "Cases and Death
122 by Region" class = "form-control" style =
123 "background-color: #fff;" disabled />
124 </div>
125 </div>
126 </div>
127 <div class = "col-lg-6">
128 <form:select path = "region" class = "form-
129 control" disabled = "${!forReports.
130 caseAndDeath}" id = "select2">
131 <option></option>
132 <option>ARMM</option>
133 <option>CAR</option>
134 <option>NCR</option>
135 <option>Region I</option>
136 <option>Region II</option>
137 <option>Region III</option>
138 <option>Region IV-A</option>
139 <option>Region IV-B</option>
140 <option>Region V</option>
141 <option>Region VI</option>
142 <option>Region VII</option>
143 <option>Region VIII</option>
144 <option>Region IX</option>
145 <option>Region X</option>
146 <option>Region XI</option>
147 <option>Region XII</option>
148 <option>Region XIII</option>
149 </form:select>
150 <c:set var="eRegion"><form:errors path="region
151 "/></c:set>
152 <c:if test="${not empty eRegion}">
153 <div class = "alert alert-danger alert-
154 dismissable">
155 <button type="button" class="close" data-
156 dismiss="alert" aria-hidden="true">&times
157 ;</button>
158 <strong><form:errors path = "region"></form:
159 errors></strong>
160 </div>
161 </c:if>
162 <c:if test="${not empty error}">
163 <div class = "alert alert-danger alert-

```

```

        dismissable">
157 <button type="button" class="close" data-
        dismiss="alert" aria-hidden="true">&times
        ;</button>
158 <strong>${error}</strong>
159 </div>
160
161 </c:if>
162 </div>
163 </div>
164 <br />
165 <div class = "row">

```

```

166 <div class = "col-lg-4">
167 </div>
168 <div class = "col-lg-4">
169 <button class = "btn btn-primary" type = "
        submit">Generate Report</button>
170 </div>
171 </div>
172 </div>
173 </form:form>
174 </div>
175 </div>

```

Listing 46: system.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
    jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
    functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
    springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
    springframework.org/tags/form" %>
5 <!DOCTYPE html>
6 <html>
7 <head>
8 <title>Vector-Borne Diseases Monitoring System
    </title>
9 <meta name="viewport" content="width=device-
    width, initial-scale=1.0">
10 <jsp:include page="./css.jsp"/>
11 <script type="text/javascript">
12 function changeSelect() {
13     var sel = document.getElementById("admitted").
        value;
14     if(sel == "Yes") {
15         document.getElementById("datE3").disabled =
            false;
16         document.getElementById("hospital").disabled
            = false;
17         document.getElementById("hospitalType").
            disabled = false;
18         document.getElementById("hospitalAddr").
            disabled = false;
19         $('#datE3').combodate();
20     }
21     else {
22         $('#datE3').combodate('destroy');
23         document.getElementById("datE3").disabled =
            true;
24         document.getElementById("hospital").disabled
            = true;
25         document.getElementById("hospitalType").
            disabled = true;
26         document.getElementById("hospitalAddr").
            disabled = true;
27     }
28 }
29 </script>
30 <script type="text/javascript">
31 function changeSelect2() {
32     var sel = document.getElementById("theDisease
        ").value;
33     if(sel == "Dengue") {
34         document.getElementById("caseType").options.
            length = 0;
35         var element2 = document.getElementById("
            caseType");
36         var option2 = document.createElement("option
            ");
37         option2.appendChild(document.createTextNode
            (""));
38         element2.add(option2);
39         if(sel == "Dengue") {
40             document.getElementById("caseType").disabled
                = false;
41             var option = document.createElement("option
                ");
42             option.appendChild(document.createTextNode("
                Dengue Without Warning Signs"));
43             var option3 = document.createElement("option
                ");
44             option3.appendChild(document.createTextNode
                ("Dengue With Warning Signs"));
45             var option4 = document.createElement("option
                ");
46             option4.appendChild(document.createTextNode
                ("Severe Dengue"));
47             element2.appendChild(option);
48             element2.appendChild(option3);
49             element2.appendChild(option4);
50         }
51     }
52 }
53 else {
54     document.getElementById("caseType").disabled
        = true;
55     document.getElementById("caseType").options.

```

```

        length = 0;
56     }
57 }
58 </script>
59 <script type="text/javascript">
60 function changeSelect4() {
61     var sel = document.getElementById("outcome").
        value;
62     if(sel == "Died") {
63         document.getElementById("datE4").disabled =
            false;
64         $('#datE4').combodate();
65     }
66     else {
67         $('#datE4').combodate('destroy');
68         document.getElementById("datE4").value = "";
69         document.getElementById("datE4").disabled =
            true;
70     }
71 </script>
72 <script>
73 $(function(){
74     $('#datE1').combodate();
75     $('#datE2').combodate();
76 });
77 </script>
78 <script type = "text/javascript">
79 function changeSelect3() {
80     var sel = document.getElementById("
        theSelect").value;
81     if(sel != "") {
82         document.getElementById("
        theSelected").disabled = false;
83         document.getElementById("
        theSelected").options.length = 0;
84         var element2 = document.
            getElementById("theSelected");
85         var option2 = document.
            createElement("option");
86         option2.appendChild(document.
            createTextNode(""));
87         element2.add(option2);
88         <c:forEach var = "nic" items = "${
            munci}" varStatus = "stu">
89             <c:forEach var = "br" items = "${
            brgys[stu.index]}">
90                 if(sel == "<c:out value = '${br.
                    muncity}' />") {
91                     var option = document.
                        createElement("option");
92                     option.appendChild(document.
                        createTextNode("<c:out value = '${br.
                            barangay}' />"));
93                     element2.appendChild(option);
94                 }
95             </c:forEach>
96         </c:forEach>
97     }
98     else {
99         document.getElementById("
        theSelected").disabled = true;
100     }
101 }
102 </script>
103 <script type="text/javascript">
104 /* Table initialisation */
105 jQuery(document).ready(function($) {
106     $('#example1').dataTable( {
107         "bSort": false, // Disable sorting
108         "iDisplayLength": 10, // records per
109         page
110         "sDom": "t<'row'<'col-md-6'i><'col-md-6'p
        >>",
111         "sPaginationType": "bootstrap"
112     });
113     <c:forEach var = "mc" items = "${
        munci}" varStatus = "stat">
114         <c:if test = "${fn:toLowerCase(mc.
            muncity)} = fn:toLowerCase(user.muncity)}">
115             <c:set var = "chosenMun" value = "${
            stat.index}" />

```

```

119         <c:out value = '#{chosenMun}' />
120     </c:if>
121     </c:forEach>
122     <c:out value = '#{chosenMun}' />
123     var example = new Array(<c:out
value = '#{fn:length(brgys[chosenMun])}'
/>);
124     for(var i = 0; i < <c:out value = '
#{fn:length(brgys[chosenMun])}' />; i++) {
125         example[i] = "#example2_" + i;
126         $(example[i]).dataTable( {
127             "bSort": false, // Disable sorting
128             "iDisplayLength": 10, // records per
page
129             "sDom": "t<'row'<'col-md-6'i><'col-md-6'p
>>",
130             "sPaginationType": "bootstrap"
131         });
132     }
133     $('#example3').dataTable( {
134         "bSort": false, // Disable sorting
135         "iDisplayLength": 10, // records per
page
136         "sDom": "t<'row'<'col-md-6'i><'col-md-6'p
>>",
137         "sPaginationType": "bootstrap"
138     });
139     for(var i = 0; i < <c:out value = '
#{fn:length(brgys[chosenMun])}' />; i++) {
140         example[i] = "#example4_" + i;
141         $(example[i]).dataTable( {
142             "bSort": false, // Disable sorting
143             "iDisplayLength": 10, // records per
page
144             "sDom": "t<'row'<'col-md-6'i><'col-md-6'p
>>",
145             "sPaginationType": "bootstrap"
146         });
147     }
148     $('#example5').dataTable( {
149         "bSort": false, // Disable sorting
150         "iDisplayLength": 10, // records per
page
151         "sDom": "t<'row'<'col-md-6'i><'col-md-6'p
>>",
152         "sPaginationType": "bootstrap"
153     });
154     for(var i = 0; i < <c:out value = '
#{fn:length(brgys[chosenMun])}' />; i++) {
155         example[i] = "#example6_" + i;
156         $(example[i]).dataTable( {
157             "bSort": false, // Disable sorting
158             "iDisplayLength": 10, // records per
page
159             "sDom": "t<'row'<'col-md-6'i><'col-md-6'p
>>",
160             "sPaginationType": "bootstrap"
161         });
162     }
163     $('#example7').dataTable( {
164         "bSort": false, // Disable sorting
165         "iDisplayLength": 10, // records per
page
166         "sDom": "t<'row'<'col-md-6'i><'col-md-6'p
>>",
167         "sPaginationType": "bootstrap"
168     });
169     for(var i = 0; i < <c:out value = '
#{fn:length(brgys[chosenMun])}' />; i++) {
170         example[i] = "#example8_" + i;
171         $(example[i]).dataTable( {
172             "bSort": false, // Disable sorting
173             "iDisplayLength": 10, // records per
page
174             "sDom": "t<'row'<'col-md-6'i><'col-md-6'p
>>",
175             "sPaginationType": "bootstrap"
176         });
177     }
178     });
179
180 </script>
181 <script type="text/javascript">
182 /* Table initialisation */
183 jQuery(document).ready(function($) {
184
185     $('#examplePV1').dataTable( {
186         "bSort": false, // Disable sorting
187         "iDisplayLength": 10, // records per
page
188         "sDom": "t<'row'<'col-md-6'i><'col-md-6'p
>>",
189         "sPaginationType": "bootstrap"
190     });
191
192     var example = new Array(<c:out value = '#{fn:
length(munci)}' />);
193     <c:set var = "countsAll" value = "0" />
194     for(var i = 0; i < <c:out value = '#{fn:length(
munci)}' />; i++) {
195         example[i] = "#examplePV2_" + i;
196         $(example[i]).dataTable( {
197             "bSort": false, // Disable sorting
198             "iDisplayLength": 5, // records per
page
199             "sDom": "t<'row'<'col-md-6'i><'col-md-6'p
>>",
200             "sPaginationType": "bootstrap"
201         });
202     }
203     var exampleP = new Array(<c:out value = '#{fn:
length(brgys[countsAll])}' />);
204     for(var j = 0; j < <c:out value = '#{fn:length(
brgys[countsAll])}' />; j++) {
205         exampleP[j] = "#exampleBP1_" + i + "_" + j;
206         $(exampleP[j]).dataTable( {
207             "bSort": false, // Disable
sorting
208             "iDisplayLength": 10, // records per
page
209             "sDom": "t<'row'<'col-md-6'i><'col-md-6'
p>>",
210             "sPaginationType": "bootstrap"
211         });
212     }
213     <c:set var = "countsAll" value = "${countsAll
+ 1}" />
214 }
215
216     $('#examplePV3').dataTable( {
217         "bSort": false, // Disable sorting
218         "iDisplayLength": 10, // records per
page
219         "sDom": "t<'row'<'col-md-6'i><'col-md-6'p
>>",
220         "sPaginationType": "bootstrap"
221     });
222
223     <c:set var = "countsAll" value = "0" />
224     for(var i = 0; i < <c:out value = '#{fn:length(
munci)}' />; i++) {
225         example[i] = "#examplePV4_" + i;
226         $(example[i]).dataTable( {
227             "bSort": false, // Disable sorting
228             "iDisplayLength": 10, // records per
page
229             "sDom": "t<'row'<'col-md-6'i><'col-md-6'p
>>",
230             "sPaginationType": "bootstrap"
231         });
232     }
233     var exampleP = new Array(<c:out value = '#{fn:
length(brgys[countsAll])}' />);
234     for(var j = 0; j < <c:out value = '#{fn:length(
brgys[countsAll])}' />; j++) {
235         exampleP[j] = "#exampleBP2_" + i + "_" + j;
236         $(exampleP[j]).dataTable( {
237             "bSort": false, // Disable
sorting
238             "iDisplayLength": 10, // records per
page
239             "sDom": "t<'row'<'col-md-6'i><'col-md-6'
p>>",
240             "sPaginationType": "bootstrap"
241         });
242     }
243     <c:set var = "countsAll" value = "${countsAll
+ 1}" />
244 }
245
246     $('#examplePV5').dataTable( {
247         "bSort": false, // Disable sorting
248         "iDisplayLength": 10, // records per
page
249         "sDom": "t<'row'<'col-md-6'i><'col-md-6'p
>>",
250         "sPaginationType": "bootstrap"
251     });
252
253     <c:set var = "countsAll" value = "0" />
254     for(var i = 0; i < <c:out value = '#{fn:length(
munci)}' />; i++) {
255         example[i] = "#examplePV6_" + i;
256         $(example[i]).dataTable( {
257             "bSort": false, // Disable sorting
258             "iDisplayLength": 10, // records per
page
259             "sDom": "t<'row'<'col-md-6'i><'col-md-6'p
>>",
260             "sPaginationType": "bootstrap"
261         });
262     }
263     var exampleP = new Array(<c:out value = '#{fn:
length(brgys[countsAll])}' />);
264     for(var j = 0; j < <c:out value = '#{fn:length(
brgys[countsAll])}' />; j++) {
265         exampleP[j] = "#exampleBP3_" + i + "_" + j;
266         $(exampleP[j]).dataTable( {
267             "bSort": false, // Disable

```

```

267     sorting
268     "iDisplayLength": 10, // records per
269     page
270     "sDom": "t<'row'<'col-md-6'i><'col-md-6'
271     p>>",
272     "sPaginationType": "bootstrap"
273     });
274 }
275 <c:set var = "countsAll" value = "${countsAll
276 + 1}" />
277 }
278 $( '#examplePV7' ).dataTable( {
279     "bSort": false, // Disable sorting
280     "iDisplayLength": 10, // records per
281     page
282     "sDom": "t<'row'<'col-md-6'i><'col-md-6'p
283     >>",
284     "sPaginationType": "bootstrap"
285     });
286 <c:set var = "countsAll" value = "0" />
287 for( var i = 0; i < <c:out value = '${fn:length(
288 munci)}/>; i++) {
289     example[i] = "#examplePV8_" + i;
290     $(example[i]).dataTable( {
291         "bSort": false, // Disable sorting
292         "iDisplayLength": 10, // records per
293         page
294         "sDom": "t<'row'<'col-md-6'i><'col-md-6'p
295         >>",
296         "sPaginationType": "bootstrap"
297         });
298     var exampleP = new Array(<c:out value = '${fn:
299 length(brgys[countsAll])}/>');
300     for( var j = 0; j < <c:out value = '${fn:length
301 (brgys[countsAll])}/>; j++) {
302         exampleP[j] = "#exampleBP4_" + i + "_" + j;
303         $(exampleP[j]).dataTable( {
304             "bSort": false, // Disable
305             sorting
306             "iDisplayLength": 10, // records per
307             page
308             "sDom": "t<'row'<'col-md-6'i><'col-md-6'
309             p>>",
310             "sPaginationType": "bootstrap"
311             });
312         <c:set var = "countsAll" value = "${countsAll
313 + 1}" />
314     }
315 }
316 </script>
317 </head>
318 <body>
319 <div class = "container">
320 <div class = "navbar navbar-default" role = "
321 navigation">
322 <div class = "container-fluid">
323 <div class = "navbar-header">
324 <button type = "button" class = "navbar-
325 toggle" data-toggle = "collapse" data-
326 target = ".navbar-collapse">
327 <span class = "sr-only">Toggle Navigation
328 </span>
329 <span class = "icon-bar"></span>
330 <span class = "icon-bar"></span>
331 <span class = "icon-bar"></span>
332 </button>
333 <a href = "<%= request.getContextPath() %>/
334 home"><img src = "<%= request.
335 getContextPath() %>/resources/dist/img/
336 logo2.jpg" alt = "VBDM Logo" height = "30
337 px;" width = "93px;" class = "img-
338 responsive"/></a>
339 </div>
340 <div class = "collapse navbar-collapse">
341 <ul class = "nav navbar-nav">
342 <li <c:if test = "${pages == 'home'}">
343 class = "active"</c:if>
344 <a href = "<%= request.getContextPath()
345 %>/home">Home</a></li>
346 <li class = "dropdown">
347 <a href = "" data-toggle = "dropdown"
348 class = "dropdown-toggle">
349 Diseases <b class = "caret"></b>
350 </a>
351 <ul class="dropdown-menu">
352 <c:forEach var="dis" items="${disease
353 }">
354 <li><a href = "<%= request.
355 getContextPath() %>/disease/${dis.
356 diseaseName}">${dis.diseaseName}</a></li>
357 </c:forEach>
358 </ul>
359 </li>
360 <li class = "dropdown">
361 <a href = "" data-toggle = "dropdown"
362 class = "dropdown-toggle">
363 Disease Statistics <b class = "caret"></b>
364 </a>
365 <ul class="dropdown-menu">
366 <c:forEach var="dis" items="${disease}">
367 <li><a href = "<%= request.
368 getContextPath() %>/statistics/${dis.
369 diseaseName}">${dis.diseaseName}</a></li>
370 </c:forEach>
371 </ul>
372 </li>
373 <li><a href = "<%= request.getContextPath()
374 %>/campaigns">Awareness Campaigns</a></li>
375 </ul>
376 <ul class = "nav navbar-nav navbar-right">
377 <sec:authorize access="isAnonymous()">
378 <c:if test = "${pages != 'register'}">
379 <li><a data-toggle = "modal" href = "#
380 loginForm">Login</a></li>
381 <li><a href = "<%= request.getContextPath()
382 %>/register">Register</a></li>
383 </c:if>
384 </sec:authorize>
385 <sec:authorize access="isAuthenticated()">
386 <div class = "btn-group">
387 Logged in as:
388 <sec:authorize access="hasAnyRole('
389 ROLE_BHWorker')">
390 <a href = "<%= request.getContextPath() %>/
391 pages/profile">Barangay Health Worker</a>
392 </sec:authorize>
393 <sec:authorize access="hasAnyRole('
394 ROLE_MCHWorker')">
395 <a href = "<%= request.getContextPath() %>/
396 pages/profile">Municipal/City Health Worker
397 </a>
398 </sec:authorize>
399 <sec:authorize access="hasAnyRole('
400 ROLE_MCOfficial')">
401 <a href = "<%= request.getContextPath() %>/
402 pages/profile">Municipal/City Health
403 Official</a>
404 </sec:authorize>
405 <sec:authorize access="hasAnyRole('
406 ROLE_PHWorker')">
407 <a href = "<%= request.getContextPath() %>/
408 pages/profile">Provincial Health Worker</a>
409 </sec:authorize>
410 <sec:authorize access="hasAnyRole('
411 ROLE_PHOfficial')">
412 <a href = "<%= request.getContextPath() %>/pages
413 /profile">Provincial Health Official</a>
414 </sec:authorize>
415 <sec:authorize access="hasAnyRole('
416 ROLE_NHOfficial')">
417 <a href = "<%= request.getContextPath() %>/pages
418 /profile">National Health Official</a>
419 </sec:authorize>
420 <sec:authorize access="hasAnyRole('ROLE_USER')">
421 <a href = "<%= request.getContextPath() %>/pages
422 /profile">Administrator</a>
423 </sec:authorize>
424 <sec:authorize access="hasAnyRole('
425 ROLE_Researcher')">
426 <a href = "<%= request.getContextPath() %>/pages
427 /profile">Researcher</a>
428 </sec:authorize>
429 </div>
430 <div class = "btn-group">
431 <button href="#" data-toggle="dropdown" class="
432 btn dropdown-toggle" >
433 <span class = "glyphicon glyphicon-user"></
434 span> <span class = "caret"></span>
435 </button>
436 <ul class="dropdown-menu">
437 <li><a href = "<%= request.getContextPath()
438 %>/pages/profile"><span>Profile</span></a>
439 </li>
440 <sec:authorize access="hasAnyRole('
441 ROLE_MCOfficial', 'ROLE_PHOfficial', '
442 ROLE_NHOfficial', 'ROLE_USER')">
443 <li><a href = "<%= request.getContextPath()
444 %>/pages/notifications">Notifications</a></li>
445 </sec:authorize>
446 <sec:authorize access="hasAnyRole('
447 ROLE_BHWorker', 'ROLE_PHWorker', '
448 ROLE_MCHWorker', 'ROLE_USER')">
449 <li><a href = "<%= request.getContextPath() %>/
450 pages/system">Manage Records</a></li>
451 </sec:authorize>
452 <sec:authorize access="hasAnyRole('
453 ROLE_MCOfficial', 'ROLE_PHOfficial', '
454 ROLE_NHOfficial', 'ROLE_USER')">
455 <li><a href = "<%= request.getContextPath() %>/

```

```

        pages/workers" data-toggle = "tab">Workers
    </a></li>
398 </sec:authorize>
399 <li><a href = "%= request.getContextPath() %>/
    pages/reports">Reports</a></li>
400 <li><a href = "%= request.getContextPath() %>/
    j_spring_security_logout">Log Out</a></li>
401 </ul>
402 </div>
403 </sec:authorize>
404 </ul>
405 </div>
406 </div>
407 </div>
408 <div class = "modal fade" id = "loginForm">
409 <div class = "modal-dialog" style = "background
    -color: #fff;">
410 <div class = "modal-body panel panel-info">
411 <div class = "panel panel-heading">
412 <h4 class = "form-signin-head">Vector Borne
    Diseases Monitoring System</h4>
413 </div>
414 <form method="post" class = "form-signin"
    action = "j_spring_security_check">
415 <table class = "table" >
416 <tr>
417 <td style = "border: none;">
418 <label for="login_username">Username</
    label>
419 </td>
420 <td style = "border: none;">
421 <input type="text" name="j_username" id="
    login_username" class="form-control field
    required" title="Please provide your
    username" />
422 </td>
423 </tr>
424 <tr>
425 <td style = "border: none;">
426 <label for="login_password">Password</
    label>
427 </td>
428 <td style = "border: none;">
429 <input type="password" name="j_password"
    id="login_password" class="form-control
    field required" title="Password is required
    " />
</td>
</tr>
</table>
</form>
</div>
</div>
<div class = "tabbable">
<jsp:include page = "linkings.jsp" />
<div class = "tab-content">
<br />
<sec:authorize access="hasAnyRole('
    ROLE_BHWorker')">
<c:if test = "${empty success}">
<jsp:include page = "systems/barangay.jsp" />
</c:if>
<c:if test = "${not empty success}">
<jsp:include page = "systems/
    addRecordSuccessful.jsp" />
</c:if>
</sec:authorize>
<sec:authorize access="hasAnyRole('
    ROLE_MCHWorker')">
<jsp:include page = "systems/muncity.jsp" />
</sec:authorize>
<sec:authorize access="hasAnyRole('ROLE_PHWorker
    ')">
<jsp:include page = "systems/province.jsp" />
</sec:authorize>
</div>
</div>
</div>
<jsp:include page = "../footer.jsp" />

```

Listing 47: barangay.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
    jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
    functions" prefix="fn" %>
3 <%@ taglib prefix="sec" uri="http://www.
    springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
    springframework.org/tags/form" %>
5
6 <div class = "tab-pane active">
7 <div class = "container-fluid well-lg" style =
    "background-color: #eee;">
8 <div class="panel panel-info">
9 <div class = "panel-heading">Inserting
    Barangay Patient Record</div>
10 <c:if test = "${not empty success}">
11 <div class = "alert alert-success alert-
    dismissable">
12 <button type="button" class="close" data-
    dismiss="alert" aria-hidden="true">&times
    ;</button>
13 <strong>${success}</strong>
14 </div>
15 </c:if>
16 <form:form method = "POST" action = "${
    pageContext.request.contextPath}/pages/
    system/toInsertRecord" modelAttribute = "
    patientForm" id = "barangay">
17 <div class = "panel-body">
18 <div class = "panel panel-success">
19 <div class = "panel-heading">
20 <label>Patient's Record Information</label>
21 </div>
22 <div class = "panel-body">
23 <div class = "row">
24 <div class = "col-lg-2">
25 <label>Default Region:</label>
26 </div>
27 <div class = "col-lg-4">
28 <form:input path = "region" class = "form
    -control" name = "region" value = "${user.
    region}" readonly = "true" style = "
    background-color: #fff;" />
29 </div>
30 <div class = "col-lg-2">
31 <label>Default Province:</label>
32 </div>
33 <div class = "col-lg-4">
34 <form:input path = "province" class = "
    form-control" name = "province" value = "${
    user.province}" readonly = "true" style = "
    background-color: #fff;" />
35 </div>
36 </div>
37 <br />
38 <div class = "row">
39 <div class = "col-lg-2">
40 <strong>Municipality/City:</strong>
41 </div>
42 <div class = "col-lg-4">
43 <form:select path = "muncity" id = "
    theSelect" onChange = "changeSelect3()"
    class = "form-control">
44 <option></option>
45 <c:forEach var = "mc" items = "${munci}">
46 <option>${mc.muncity}</option>
47 </c:forEach>
48 </form:select>
49 </div>
50 <div class = "col-lg-2">
51 <strong>Barangay:</strong>
52 </div>
53 <div class = "col-lg-4">
54 <form:select path = "barangay" id = "
    theSelected" class = "form-control"
    disabled = "true">
55 <option></option>
56 </form:select>
57 </div>
58 </div>
59 </div>
60 <br />
61 <div class = "panel panel-success">
62 <div class = "panel-heading">
63 <label>Patient's General Information</label>
64 </div>
65 <div class = "panel-body">
66 <div class = "row">
67 <div class = "col-lg-1">
68 <label>Name:</label>
69 </div>
70 <div class = "col-lg-5">
71 <form:input path = "name" class = "form-
    control"/>
72 </div>
73 </div>
74 <div class = "col-lg-2">
75 <label>Complete Address:</label>
76 </div>
77 <div class = "col-lg-4">
78 <form:textarea path = "address" class = "
    form-control"></form:textarea>
79 </div>
80 </div>

```

```

81 <br />
82 <div class = "row">
83 <div class = "col-lg-2">
84 <label>Date of Birth:</label>
85 </div>
86 <div class = "col-lg-4">
87 <form:input path = "dateofbirth" type="text"
id="datE1" data-format="YYYY-MM-DD" data-
template="D MMM YYYY" value="" />
88 </div>
89 <div class = "col-lg-2">
90 <label>Gender:</label>
91 </div>
92 <div class = "col-lg-4">
93 <form:select path = "gender" class = "form-
control">
94 <option></option>
95 <option>Male</option>
96 <option>Female</option>
97 </form:select>
98 </div>
99 </div>
100 </div>
101 </div>
102 <br />
103 <div class = "panel panel-success">
104 <div class = "panel-heading">
105 <strong>Patient's Clinical Information</strong>
106 </div>
107 <div class = "panel-body">
108 <div class = "row">
109 <div class = "col-lg-2">
110 <label>Select disease:</label>
111 </div>
112 <div class = "col-lg-4">
113 <form:select path = "disease" id = "
theDisease" onChange = "changeSelect2()"
class = "form-control">
114 <option></option>
115 <option
116 <c:if test = "${cBrgyForm.disease == '
Chikungunya'}">
selected = "selected"
117 </c:if>
118 >Chikungunya</option>
119 <option<c:if test = "${cBrgyForm.disease == '
Dengue'}">
selected = "selected"
120 </c:if>
121 >Dengue</option>
122 <option<c:if test = "${cBrgyForm.disease == '
Malaria'}">
selected = "selected"
123 </c:if>
124 >Malaria</option>
125 </form:select>
126 </div>
127 >Malaria</option>
128 </form:select>
129 </div>
130 <div class = "col-lg-2">
131 <label>Date Onset:</label>
132 </div>
133 <div class = "col-lg-4">
134 <form:input path = "dateOnset" type="text" id="
datE2" data-format="YYYY-MM-DD" data-
template="D MMM YYYY" value="" />
135 </div>
136 </div>
137 <br />
138 <div class = "row">
139 <div class = "col-lg-2">
140 <label>Symptoms</label>
141 </div>
142 <div class = "col-lg-4">
143 <form:textarea path = "symptoms" class = "form
-control"></form:textarea>
144 </div>
145 <div class = "col-lg-2">
146 <label>Case Classification</label>
147 </div>
148 <div class = "col-lg-4">
149 <form:select path = "caseType" id = "caseType"
class = "form-control" disabled = "${
patientForm.disease != 'Dengue'}">
150 <option></option>
151 </form:select>
152 </div>
153 </div>
154 <br />
155 <div class = "row">
156 <div class = "col-lg-2">
157 <label>Patient Admitted?:</label>
158 </div>
159 <div class = "col-lg-4">
160 <form:select path = "admitted" id = "admitted
" onChange = "changeSelect()" class = "form
-control">
161 <option></option>
162 <option>Yes</option>
163 <option>No</option>
164 </form:select>
165 </div>
166 <div class = "col-lg-2">
167 <label>Date Admitted:</label>
168 </div>
169 <div class = "col-lg-4">
170 <form:input path = "dateAdmitted" type="text"
id="datE3" data-format="YYYY-MM-DD" data-
template="D MMM YYYY" value="" disabled =
"${patientForm.admitted != 'Yes'}"/>
171 </div>
172 </div>
173 <br />
174 <div class = "row">
175 <div class = "col-lg-2">
176 <label>Hospital/Clinic:</label>
177 </div>
178 <div class = "col-lg-5">
179 <form:textarea path = "hospital" id = "
hospital" class = "form-control" disabled =
"${patientForm.admitted != 'Yes'}"></form:
textarea>
180 </div>
181 <div class = "col-lg-2">
182 <label>Hospital/Clinic Type:</label>
183 </div>
184 <div class = "col-lg-3">
185 <form:select path = "hospitalType" id = "
hospitalType" class = "form-control"
disabled = "${patientForm.admitted != 'Yes
'}">
186 <option></option>
187 <option>Level 1</option>
188 <option>Level 2</option>
189 <option>Level 3</option>
190 <option>Level 4</option>
191 <option>Barangay Health Center</option>
192 </form:select>
193 </div>
194 </div>
195 <br />
196 <div class = "row">
197 <div class = "col-lg-3">
198 <label>Hospital/Clinic Address:</label>
199 </div>
200 <div class = "col-lg-9">
201 <form:textarea path = "hospitalAddr" id = "
hospitalAddr" class = "form-control"
disabled = "${patientForm.admitted != 'Yes
'}"></form:textarea>
202 </div>
203 </div>
204 <br />
205 <div class = "row">
206 <div class = "col-lg-2">
207 <label>Outcome:</label>
208 </div>
209 <div class = "col-lg-4">
210 <form:select path = "outcome" id = "outcome"
onChange = "changeSelect4()" class = "form-
control">
211 <option></option>
212 <option>Recovered</option>
213 <option>Died</option>
214 </form:select>
215 </div>
216 <div class = "col-lg-2">
217 <label>Date Died:</label>
218 </div>
219 <div class = "col-lg-4">
220 <form:input path = "dateDied" type="text" id="
datE4" data-format="YYYY-MM-DD" data-
template="D MMM YYYY" value="" disabled =
"${patientForm.outcome != 'Died'}" />
221 </div>
222 </div>
223 <br />
224 </div>
225 </div>
226 <br />
227 </div>
228 <div class = "panel-footer">
229 <center>
230 <button type = "submit" class = "btn btn-
primary" value = "Create Record">Submit</
button>
231 </center>
232 </div>
233 <!--
234 <div id = "yesNo" class = "modal
fade">
235 <div class = "modal-header">
236 <button type="button" class="
close" data-dismiss="modal" aria-hidden="
true"> </button>
237 </div>
238 <div class = "modal-body">
239 Are you sure you want to add this
patient record?

```



```

144     </div>
145 </div>
146 </c:if>
147 <c:if test = "${bR1.editStatus == 1}">
148 <br />
149 <div class = "row">
150 <div class = "col-lg-2">
151 <label>Edited by:</label>
152 </div>
153 <div class = "col-lg-4">
154 <input type = "text" class = "form-control"
    readonly = "true" value = "${bR1.editedBy}" />
155 </div>
156 <div class = "col-lg-2">
157 <label>Date Edited:</label>
158 </div>
159 <div class = "col-lg-4">
160 <input type = "text" class = "form-control"
    readonly = "true" value = "${bR1.dateEdited}" />
161 </div>
162 </div>
163 </c:if>
164 </div>
165 </div>
166 <div class = "panel panel-info">
167 <div class = "panel-heading">
168 <label>Patient's Information</label>
169 </div>
170 <div class = "panel-body">
171 <c:if test = "${not empty bR1.name}">
172 <div class = "row">
173 <div class = "col-lg-1">
174 <label>Name:</label>
175 </div>
176 <div class = "col-lg-5">
177 <input type = "text" class = "form-control"
    readonly = "true" value = "${bR1.name}" />
178 </div>
179 </div>
180 <br />
181 </c:if>
182 <c:if test = "${not empty bR1.address}">
183 <div class = "row">
184 <div class = "col-lg-1">
185 <label>Complete Address:</label>
186 </div>
187 <div class = "col-lg-5">
188 <textarea class = "form-control" readonly = "
    true" value = "${bR1.address}">${bR1.
    address}</textarea>
189 </div>
190 </div>
191 <br />
192 </c:if>
193 <div class = "row">
194 <div class = "col-lg-1">
195 <label>Age:</label>
196 </div>
197 <div class = "col-lg-2">
198 <input type = "text" class = "form-control"
    readonly = "true" value = "${bR1.age}" />
199 </div>
200 <div class = "col-lg-1">
201 <label>Date of Birth:</label>
202 </div>
203 <div class = "col-lg-3">
204 <input class="form-control" type="text"
    readonly = "true" value="${bR1.dateofbirth}" />
205 </div>
206 <div class = "col-lg-1">
207 <label>Gender:</label>
208 </div>
209 <div class = "col-lg-3">
210 <input class = "form-control" type = "text"
    readonly = "true" value = "${bR1.gender}" />
211 </div>
212 </div>
213 </div>
214 </div>
215 <div class = "panel panel-info">
216 <div class = "panel-heading">
217 <label>Patient's Clinical Information</label>
218 </div>
219 <div class = "panel-body">
220 <div class = "row">
221 <div class = "col-lg-2">
222 <label>Disease:</label>
223 </div>
224 <div class = "col-lg-4">
225 <input type = "text" class = "form-control"
    readonly = "true" value = "${bR1.disease}" />
226 </div>
227 <div class = "col-lg-2">
228 <label>Date Onset:</label>
229 </div>
230 <div class = "col-lg-4">
231 <input class="form-control" type="text"
    readonly = "true" value="${bR1.dateOnset}" />
232 </div>
233 </div>
234 <br />
235 <div class = "row">
236 <div class = "col-lg-2">
237 <label>Symptoms:</label>
238 </div>
239 <div class = "col-lg-4">
240 <input type = "text" class = "form-control"
    readonly = "true" value = "${bR1.symptoms}" />
241 </div>
242 <div class = "col-lg-2">
243 <label>Case Classification:</label>
244 </div>
245 <div class = "col-lg-4">
246 <input class="form-control" type="text"
    readonly = "true" value="${bR1.caseType}" />
247 </div>
248 </div>
249 <br />
250 <c:if test = "${bR1.admitted == 'No'}">
251 <div class = "row">
252 <div class = "col-lg-2">
253 <label>Patient Admitted?:</label>
254 </div>
255 <div class = "col-lg-4">
256 <input type = "text" class = "form-control"
    readonly = "true" value = "${bR1.admitted}" />
257 </div>
258 </div>
259 </c:if>
260 <c:if test = "${bR1.admitted == 'Yes'}">
261 <div class = "row">
262 <div class = "col-lg-2">
263 <label>Patient Admitted?:</label>
264 </div>
265 <div class = "col-lg-4">
266 <input type = "text" class = "form-control"
    readonly = "true" value = "${bR1.admitted}" />
267 </div>
268 <div class = "col-lg-2">
269 <label>Date Admitted:</label>
270 </div>
271 <div class = "col-lg-4">
272 <input class="form-control" type="text"
    readonly = "true" value="${bR1.dateAdmitted}" />
273 </div>
274 </div>
275 <br />
276 <div class = "row">
277 <div class = "col-lg-2">
278 <label>Hospital/Clinic:</label>
279 </div>
280 <div class = "col-lg-5">
281 <input type = "text" class = "form-control"
    readonly = "true" value = "${bR1.hospital}" />
282 </div>
283 <div class = "col-lg-2">
284 <label>Hospital/Clinic Type:</label>
285 </div>
286 <div class = "col-lg-3">
287 <input class="form-control" type="text"
    readonly = "true" value="${bR1.hospitalType}" />
288 </div>
289 </div>
290 <br />
291 <div class = "row">
292 <div class = "col-lg-3">
293 <label>Hospital/Clinic Address:</label>
294 </div>
295 <div class = "col-lg-9">
296 <input type = "text" class = "form-control"
    readonly = "true" value = "${bR1.
    hospitalAddr}" />
297 </div>
298 </div>
299 </c:if>
300 <br />
301 <div class = "row">
302 <div class = "col-lg-2">
303 <label>Outcome</label>
304 </div>
305 <div class = "col-lg-4">
306 <input type = "text" class = "form-control"
    readonly = "true" value = "${bR1.outcome}" />
307 </div>
308 <c:if test = "${bR1.outcome == 'Died'}">

```

```

309 <div class = "col-lg-2">
310 <label>Date Died:</label>
311 </div>
312 <div class = "col-lg-4">
313 <input type = "text" class = "form-control"
    readonly = "true" value = "${bR1.dateDied
    }"/>
314 </div>
315 </c:if>
316 </div>
317 <br />
318 </div>
319 </div>
320 </div>
321 <c:if test = "${whatStat == 0 || whatStat == 3}
    && (bR1.editStatus == 0 || bR1.editStatus
    == 1)}">
322 <div class="modal-footer">
323 <button type = "button" data-toggle = "modal"
    href = "#editIt_${status.index}" class = "
    btn btn-primary">Edit</button>
324 <button type = "button" data-toggle = "modal"
    href = "#rejectIt_${status.index}" class =
    "btn btn-primary">Reject</button>
325 <button type = "button" data-toggle = "modal"
    href = "#submitIt_${status.index}" class =
    "btn btn-primary">Submit</button>
326 </div>
327 </c:if>
328 </div>
329 <div id = "editIt_${status.index}" class = "
    modal fade" tabindex = "-1" data-focus-on =
    "input:first">
330 <div class = "modal-header">
331 <button type="button" class="close" data-
    dismiss="modal" aria-hidden="true"> </
    button>
332 </div>
333 <div class = "modal-body">
334 Are you sure you want to edit this patient
    record?
335 </div>
336 <div class = "modal-footer">
337 <a href = "${pageContext.request.contextPath}/
    pages/system/editRecord/${bR1.recordID}"
    class = "btn btn-primary">Yes</a>
338 <a href = "" class = "btn btn-primary" data-
    dismiss="modal">No</a>
339 </div>
340 </div>
341 <div id = "rejectIt_${status.index}" class = "
    modal fade" tabindex = "-1" data-focus-on =
    "input:first">
342 <div class = "modal-header">
343 <button type="button" class="close" data-
    dismiss="modal" aria-hidden="true"> </
    button>
344 </div>
345 <div class = "modal-body">
346 Are you sure you want to reject this patient
    record?
347 </div>
348 <div class = "modal-footer">
349 <a href = "${pageContext.request.contextPath}/
    pages/system/rejectMunCityRecords/${bR1.
    recordID}" class = "btn btn-primary">Yes</a>
350 <a href = "" class = "btn btn-primary" data-
    dismiss="modal">No</a>
351 </div>
352 </div>
353 <div id = "submitIt_${status.index}" class = "
    modal fade" tabindex = "-1" data-focus-on =
    "input:first">
354 <div class = "modal-header">
355 <button type="button" class="close" data-
    dismiss="modal" aria-hidden="true"> </
    button>
356 </div>
357 <div class = "modal-body">
358 Are you sure you want to submit this patient
    record?
359 </div>
360 <div class = "modal-footer">
361 <a href = "${pageContext.request.contextPath}/
    pages/system/addToMunCityRecords/${bR1.
    recordID}" class = "btn btn-primary">Yes</a>
362 <a href = "" class = "btn btn-primary" data-
    dismiss="modal">No</a>
363 </div>
364 </div>
365 </c:if>
366 </c:forEach>
367 </tbody>

```

Listing 49: muncity.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
    jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
    functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
    springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
    springframework.org/tags/form" %>
5
6 <div class = "tab-pane active">
7 <div class = "tabbable">
8 <ul class = "nav nav-pills">
9 <li <c:if test = "${empty rejected}">class =
    "active"</c:if>>
10 <a href = "#tab4a" data-toggle = "tab">
    Pending Records <span class="badge pull-
    right"></span></a>
11 </li>
12 <li>
13 <a href = "#tab4b" data-toggle = "tab">
    Approved Records <span class="badge pull-
    right"></span></a>
14 </li>
15 <li <c:if test = "${not empty rejected}">
    class = "active"</c:if>>
16 <a href = "#tab4c" data-toggle = "tab">
    Rejected Records <span class="badge pull-
    right"></span></a>
17 </li>
18 <li>
19 <a href = "#tab4d" data-toggle = "tab">
    Edited Records <span class="badge pull-
    right"></span></a>
20 </li>
21 </ul>
22 </div>
23 <div class = "tab-content">
24 <div class = "tab-pane <c:if test = "${empty
    rejected}">active</c:if>" id = "tab4a">
25 <br />
26 <div class = "container-fluid well-lg" style
    = "background-color: #eee;">
27 <div class = "panel panel-info">
28 <div class = "panel-heading">
29 Pending Barangay Records
30 </div>
31 <br />
32 <div class = "panel-body">
33 <c:if test = "${not empty success}">
34 <div class = "alert alert-success">
35 ${success}
36 </div>
37 </c:if>
38 <table class="table table-striped" id="
    example1">
39 <thead>
40 <tr>
41 <th>Barangay</th>
42 <th>Pending Records</th>
43 </tr>
44 </thead>
45 <tbody>
46 <c:forEach var = "mc" items = "${muncity}
    varStatus = "stat">
47 <c:if test = "${fn:toLowerCase(mc.muncity
    ) == fn:toLowerCase(user.muncity)}">
48 <c:set var = "chosenMun" value = "${stat.
    index}" />
49 </c:if>
50 </c:forEach>
51 <c:forEach var = "br" items = "${brgys[
    chosenMun]}" varStatus = "sts">
52 <tr>
53 <td>${br.barangay}</td>
54 <td>
55 <button type = "button" class = "btn btn-
    primary" data-toggle = "modal" href = "#
    allStack1_${sts.index}">See All</button>
56 </td>
57 </tr>
58 </c:forEach>
59 </tbody>
60 </table>
61 </div>
62 </div>
63 </div>
64 <c:forEach var = "br" items = "${brgys[chosenMun
    ]}" varStatus = "sts">
65 <div id = "allStack1_${sts.index}" class = "
    modal fade" data-width = "600">
66 <c:set var = "whatStat" value = "0" scope = "
    request" />
67 <c:set var = "barangay" value = "${br.barangay
    }" scope = "request" />
68 <div class = "modal-body">
69 <table class="table table-striped" id="

```

```

        example2_${sts.index}">
70     <jsp:include page = "records.jsp" />
71 </table>
72 </div>
73 </div>
74 </c:forEach>
75 </div>
76 <div class = "tab-pane" id = "tab4b">
77 <br />
78 <div class = "container-fluid well-lg" style =
    "background-color: #eee;">
79 <div class = "panel panel-info">
80 <div class = "panel-heading">
81     Approved Barangay Records
82 </div>
83 <br />
84 <div class = "panel-body">
85 <c:if test = "${not empty success}">
86 <div class = "alert alert-success">
87     ${success}
88 </div>
89 </c:if>
90 <table class="table table-striped" id="
    example3">
91 <thead>
92 <tr>
93 <th>Barangay</th>
94 <th>Approved Records</th>
95 </tr>
96 </thead>
97 <tbody>
98 <c:forEach var = "mc" items = "${munci}"
    varStatus = "stat">
99 <c:if test = "${fn:toLowerCase(mc.muncity)}
    == fn:toLowerCase(user.muncity)}">
100 <c:set var = "chosenMun" value = "${stat.
    index}" />
101 </c:if>
102 </c:forEach>
103 <c:forEach var = "br" items = "${brgys[
    chosenMun]}" varStatus = "sts">
104 <tr>
105 <td>${br.barangay}</td>
106 <td>
107 <button type = "button" class = "btn btn-
    primary" data-toggle = "modal" href = "#
    allStack2_${sts.index}">See All</button>
108 </td>
109 </tr>
110 </c:forEach>
111 </tbody>
112 </table>
113 </div>
114 </div>
115 <c:forEach var = "br" items = "${brgys[chosenMun
    ]}" varStatus = "sts">
116 <div id = "allStack2_${sts.index}" class = "
    modal fade" data-width = "700">
117 <c:set var = "whatStat" value = "1" scope = "
    request" />
118 <c:set var = "barangay" value = "${br.barangay
    }" scope = "request" />
119 <div class = "modal-body">
120 <table class="table table-striped" id="
    example4_${sts.index}">
121 <jsp:include page = "records.jsp" />
122 </table>
123 </div>
124 </div>
125 </c:forEach>
126 </div>
127 </div>
128 <div class = "tab-pane" id = "tab4b">
129 <br />
130 <div class = "container-fluid well-lg" style =
    "background-color: #eee;">
131 <div class = "panel panel-info">
132 <div class = "panel-heading">
133     Rejected Barangay Records
134 </div>
135 <br />
136 <div class = "panel-body">
137 <c:if test = "${not empty success}">
138 <div class = "alert alert-success">
139     ${success}
140 </div>
141 </c:if>
142 <table class="table table-striped" id="
    example5">
143 <thead>
144 <tr>
145 <th>Barangay</th>
146 <th>Rejected Records</th>
147 </tr>
148 </thead>
149 <tbody>
150 <c:forEach var = "mc" items = "${munci}"
    varStatus = "stat">
151 <c:if test = "${fn:toLowerCase(mc.muncity)}
    == fn:toLowerCase(user.muncity)}">
152 <c:set var = "chosenMun" value = "${stat.
    index}" />
153 </c:if>
154 </c:forEach>
155 <c:forEach var = "br" items = "${brgys[
    chosenMun]}" varStatus = "sts">
156 <tr>
157 <td>${br.barangay}</td>
158 <td>
159 <button type = "button" class = "btn btn-
    primary" data-toggle = "modal" href = "#
    allStack3_${sts.index}">See All</button>
160 </td>
161 </tr>
162 </c:forEach>
163 </tbody>
164 </table>
165 </div>
166 </div>
167 <c:forEach var = "br" items = "${brgys[chosenMun
    ]}" varStatus = "sts">
168 <div id = "allStack3_${sts.index}" class = "
    modal fade" data-width = "700">
169 <c:set var = "whatStat" value = "2" scope = "
    request" />
170 <c:set var = "barangay" value = "${br.barangay
    }" scope = "request" />
171 <div class = "modal-body">
172 <table class="table table-striped" id="
    example6_${sts.index}">
173 <jsp:include page = "records.jsp" />
174 </table>
175 </div>
176 </div>
177 </c:forEach>
178 </div>
179 </div>
180 <div class = "tab-pane <c:if test = "${not empty
    rejected}">active</c:if"> id = "tab4c">
181 <br />
182 <div class = "container-fluid well-lg" style =
    "background-color: #eee;">
183 <div class = "panel panel-info">
184 <div class = "panel-heading">
185     Rejected Barangay Records
186 </div>
187 <br />
188 <div class = "panel-body">
189 <c:if test = "${not empty rej}">
190 <div class = "alert alert-success">
191     ${rej}
192 </div>
193 </c:if>
194 <table class="table table-striped" id="
    example5">
195 <thead>
196 <tr>
197 <th>Barangay</th>
198 <th>Rejected Records</th>
199 </tr>
200 </thead>
201 <tbody>
202 <c:forEach var = "mc" items = "${munci}"
    varStatus = "stat">
203 <c:if test = "${fn:toLowerCase(mc.muncity)}
    == fn:toLowerCase(user.muncity)}">
204 <c:set var = "chosenMun" value = "${stat.
    index}" />
205 </c:if>
206 </c:forEach>
207 <c:forEach var = "br" items = "${brgys[
    chosenMun]}" varStatus = "sts">
208 <tr>
209 <td>${br.barangay}</td>
210 <td>
211 <button type = "button" class = "btn btn-
    primary" data-toggle = "modal" href = "#
    allStack3_${sts.index}">See All</button>
212 </td>
213 </tr>
214 </c:forEach>
215 </tbody>
216 </table>
217 </div>
218 </div>
219 <c:forEach var = "br" items = "${brgys[chosenMun
    ]}" varStatus = "sts">
220 <div id = "allStack3_${sts.index}" class = "
    modal fade" data-width = "700">
221 <c:set var = "whatStat" value = "2" scope = "
    request" />
222 <c:set var = "barangay" value = "${br.barangay
    }" scope = "request" />
223 <div class = "modal-body">
224 <table class="table table-striped" id="
    example6_${sts.index}">
225 <jsp:include page = "records.jsp" />
226 </table>

```

```

227 </div>
228 </div>
229 </c:forEach>
230 </div>
231 </div>
232 <div class = "tab-pane" id = "tab4d">
233 <br />
234 <div class = "container-fluid well-lg" style =
    "background-color: #eee;">
235 <div class = "panel panel-info">
236 <div class = "panel-heading">
237 Edited Barangay Records
238 </div>
239 <br />
240 <div class = "panel-body">
241 <c:if test = "${not empty success}">
242 <div class = "alert alert-success">
243 ${success}
244 </div>
245 </c:if>
246 <table class="table table-striped" id="
    example7">
247 <thead>
248 <tr>
249 <th>Barangay</th>
250 <th>Edited Records</th>
251 </tr>
252 </thead>
253 <tbody>
254 <c:forEach var = "mc" items = "${muncity}"
    varStatus = "stat">
255 <c:if test = "${fn:toLowerCase(mc.muncity)}
    == fn:toLowerCase(user.muncity)}">
256 <c:set var = "chosenMun" value = "${stat.
    index}" />
257 </c:if>
258 </c:forEach>

```

Listing 50: province.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
    jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
    functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
    springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
    springframework.org/tags/form" %>
5
6 <div class = "tab-pane active">
7 <div class = "tabbable">
8 <ul class = "nav nav-pills">
9 <li <c:if test = "${empty rejected}">class =
    "active"</c:if>
10 <a href = "#tab4a" data-toggle = "tab">
    Pending Records <span class="badge pull-
    right"></span></a>
11 </li>
12 <li>
13 <a href = "#tab4b" data-toggle = "tab">
    Approved Records <span class="badge pull-
    right"></span></a>
14 </li>
15 <li <c:if test = "${not empty rejected}">
    class = "active"</c:if>
16 <a href = "#tab4c" data-toggle = "tab">
    Rejected Records <span class="badge pull-
    right"></span></a>
17 </li>
18 <li>
19 <a href = "#tab4d" data-toggle = "tab">
    Edited Records <span class="badge pull-
    right"></span></a>
20 </li>
21 </ul>
22 </div>
23 <div class = "tab-content">
24 <div class = "tab-pane <c:if test = "${empty
    rejected}">active</c:if"> id = "tab4a">
25 <br />
26 <div class = "container-fluid well-lg" style
    = "background-color: #eee;">
27 <div class = "panel panel-info">
28 <div class = "panel-heading">
29 Pending Municipal/City Records
30 </div>
31 <br />
32 <div class = "panel-body">
33 <c:if test = "${not empty success}">
34 <div class = "alert alert-success">
35 ${success}
36 </div>
37 </c:if>
38 <table class="table table-striped" id="
    examplePV1">
39 <thead>
40 <tr>
41 <th>Municipality/City</th>
42 <th>Pending Records</th>
43 </tr>
44 </thead>
45 <tbody>
46 <c:forEach var = "mc" items = "${muncity}"
    varStatus = "stat">
47 <tr>
48 <td>${mc.muncity}</td>
49 <td>
50 <button type = "button" class = "btn
    btn-primary" data-toggle = "modal" href =
    "#newStack1_${stat.index}">See All</button>
51 </td>
52 </tr>
53 </c:forEach>
54 </tbody>
55 </table>
56 </div>
57 </div>
58 <c:forEach var = "mc" items = "${muncity}"
    varStatus = "stat">
59 <div class = "modal fade" id = "newStack1_{$
    stat.index}" data-width = "600">
60 <div class = "modal-body">
61 <table class="table table-striped" id =
    "examplePV2_${stat.index}">
62 <thead>
63 <tr>
64 <th>Barangay</th>
65 <th>Pending Records</th>
66 </tr>
67 </thead>
68 <tbody>
69 <c:forEach var = "br" items = "${brgys[
    stat.index]}" varStatus = "sts">
70 <tr>
71 <td>${br.barangay}</td>
72 <td>
73 <button type = "button" class = "btn btn
    -primary" data-toggle = "modal" href = "#
    allStack1_${stat.index}_${sts.index}">See
    All</button>
74 </td>
75 </tr>
76 </c:forEach>
77 </tbody>
78 </table>
79 </div>
80 </div>
81 </c:forEach>
82 <c:forEach var = "mc" items = "${muncity}"
    varStatus = "stat">
83 <c:forEach var = "br" items = "${brgys[stat.
    index]}" varStatus = "sts">
84 <div class = "modal fade" id = "allStack1_{$
    stat.index}_${sts.index}" data-width = "850">
85 <div class = "modal-body">
86 <c:set var = "whatStat" value = "0" scope =
    "request" />
87 <c:set var = "muncity" value = "${mc.muncity}"

```

```

scope = "request" />
88 <c:set var = "barangay" value = "${br.barangay
} " scope = "request" />
89 <table class="table table-striped" id="
exampleBP1_${stat.index}_${sts.index}">
90 <jsp:include page = "records2.jsp" />
91 </table>
92 </div>
93 </div>
94 </c:forEach>
95 </c:forEach>
96 </div>
97 </div>
98 <div class = "tab-pane" id = "tab4b">
99 <br />
100 <div class = "container-fluid well-lg" style =
"background-color: #eee;">
101 <div class = "panel panel-info">
102 <div class = "panel-heading">
103 Approved Municipal/City Records
104 </div>
105 <br />
106 <div class = "panel-body">
107 <c:if test = "${not empty success}">
108 <div class = "alert alert-success">
109 ${success}
110 </div>
111 </c:if>
112 <table class="table table-striped" id="
examplePV3">
113 <thead>
114 <tr>
115 <th>Municipality/City</th>
116 <th>Approved Records</th>
117 </tr>
118 </thead>
119 <tbody>
120 <c:forEach var = "mc" items = "${munci}"
varStatus = "stat">
121 <tr>
122 <td>${mc.muncity}</td>
123 <td>
124 <button type = "button" class = "btn btn-
primary" data-toggle = "modal" href = "#
newStack2_${stat.index}">See All</button>
125 </td>
126 </tr>
127 </c:forEach>
128 </tbody>
129 </table>
130 </div>
131 </div>
132 <c:forEach var = "mc" items = "${munci}"
varStatus = "stat">
133 <div class = "modal fade" id = "newStack2_${stat
.index}" data-width = "600">
134 <div class = "modal-body">
135 <table class="table table-striped" id = "
examplePV4_${stat.index}">
136 <thead>
137 <tr>
138 <th>Barangay</th>
139 <th>Approved Records</th>
140 </tr>
141 </thead>
142 <tbody>
143 <c:forEach var = "br" items = "${brgys[stat.
index]}" varStatus = "sts">
144 <tr>
145 <td>${br.barangay}</td>
146 <td>
147 <button type = "button" class = "btn btn-
primary" data-toggle = "modal" href = "#
allStack2_${stat.index}_${sts.index}">See
All</button>
148 </td>
149 </tr>
150 </c:forEach>
151 </tbody>
152 </table>
153 </div>
154 </div>
155 </c:forEach>
156 <c:forEach var = "mc" items = "${munci}"
varStatus = "stat">
157 <c:forEach var = "br" items = "${brgys[stat.
index]}" varStatus = "sts">
158 <div class = "modal fade" id = "allStack2_${stat
.index}_${sts.index}" data-width = "850">
159 <div class = "modal-body">
160 <c:set var = "whatStat" value = "1" scope = "
request" />
161 <c:set var = "muncity" value = "${mc.muncity}"
scope = "request" />
162 <c:set var = "barangay" value = "${br.barangay
}" scope = "request" />
163 <table class="table table-striped" id="
exampleBP2_${stat.index}_${sts.index}">
164 <jsp:include page = "records2.jsp" />
165 </table>
166 </div>
167 </div>
168 </c:forEach>
169 </c:forEach>
170 </div>
171 </div>
172 <div class = "tab-pane <c:if test = "${not empty
rejected}">active</c:if"> id = "tab4c">
173 <br />
174 <div class = "container-fluid well-lg" style =
"background-color: #eee;">
175 <div class = "panel panel-info">
176 <div class = "panel-heading">
177 Rejected Municipal/City Records
178 </div>
179 <br />
180 <div class = "panel-body">
181 <c:if test = "${not empty rej}">
182 <div class = "alert alert-success">
183 ${rej}
184 </div>
185 </c:if>
186 <table class="table table-striped" id="
examplePV5">
187 <thead>
188 <tr>
189 <th>Municipality/City</th>
190 <th>Rejected Records</th>
191 </tr>
192 </thead>
193 <tbody>
194 <c:forEach var = "mc" items = "${munci}"
varStatus = "stat">
195 <tr>
196 <td>${mc.muncity}</td>
197 <td>
198 <button type = "button" class = "btn btn-
primary" data-toggle = "modal" href = "#
newStack3_${stat.index}">See All</button>
199 </td>
200 </tr>
201 </c:forEach>
202 </tbody>
203 </table>
204 </div>
205 </div>
206 <c:forEach var = "mc" items = "${munci}"
varStatus = "stat">
207 <div class = "modal fade" id = "newStack3_${stat
.index}" data-width = "600">
208 <div class = "modal-body">
209 <table class="table table-striped" id = "
examplePV6_${stat.index}">
210 <thead>
211 <tr>
212 <th>Barangay</th>
213 <th>Rejected Records</th>
214 </tr>
215 </thead>
216 <tbody>
217 <c:forEach var = "br" items = "${brgys[stat.
index]}" varStatus = "sts">
218 <tr>
219 <td>${br.barangay}</td>
220 <td>
221 <button type = "button" class = "btn btn-
primary" data-toggle = "modal" href = "#
allStack3_${stat.index}_${sts.index}">See
All</button>
222 </td>
223 </tr>
224 </c:forEach>
225 </tbody>
226 </table>
227 </div>
228 </div>
229 </c:forEach>
230 <c:forEach var = "mc" items = "${munci}"
varStatus = "stat">
231 <c:forEach var = "br" items = "${brgys[stat.
index]}" varStatus = "sts">
232 <div class = "modal fade" id = "allStack3_${stat
.index}_${sts.index}" data-width = "850">
233 <div class = "modal-body">
234 <c:set var = "whatStat" value = "2" scope = "
request" />
235 <c:set var = "muncity" value = "${mc.muncity}"
scope = "request" />
236 <c:set var = "barangay" value = "${br.barangay
}" scope = "request" />
237 <table class="table table-striped" id="
exampleBP3_${stat.index}_${sts.index}">
238 <jsp:include page = "records2.jsp" />
239 </table>
240 </div>
241 </div>
242 </c:forEach>
243 </c:forEach>

```

```

244 </div>
245 </div>
246 <div class = "tab-pane" id = "tab4d">
247 <br />
248 <div class = "container-fluid well-lg" style =
    "background-color: #eee;">
249 <div class = "panel panel-info">
250 <div class = "panel-heading">
251 Edited Municipal/City Records
252 </div>
253 <br />
254 <div class = "panel-body">
255 <c:if test = "${not empty success}">
256 <div class = "alert alert-success">
257 ${success}
258 </div>
259 </c:if>
260 <table class="table table-striped" id="
    examplePV7">
261 <thead>
262 <tr>
263 <th>Municipality/City</th>
264 <th>Edited Records</th>
265 </tr>
266 </thead>
267 <tbody>
268 <c:forEach var = "mc" items = "${munci}"
    varStatus = "stat">
269 <tr>
270 <td>${mc.muncity}</td>
271 <td>
272 <button type = "button" class = "btn btn-
    primary" data-toggle = "modal" href = "#
    newStack4_${stat.index}">See All</button>
273 </td>
274 </tr>
275 </c:forEach>
276 </tbody>
277 </table>
278 </div>
279 </div>
280 <c:forEach var = "mc" items = "${munci}"
    varStatus = "stat">
281 <div class = "modal fade" id = "newStack4_${stat
    .index}" data-width = "600">
282 <div class = "modal-body">
283 <table class="table table-striped" id = "
    examplePV8_${stat.index}">
284 <thead>
285 <tr>
286 <th>Barangay</th>
287 <th>Edited Records</th>
288 </tr>
289 </thead>
290 <tbody>
291 <c:forEach var = "br" items = "${brgys[stat.
    index]}" varStatus = "sts">
292 <tr>
293 <td>${br.barangay}</td>
294 <td>
295 <button type = "button" class = "btn btn-
    primary" data-toggle = "modal" href = "#
    allStack4_${stat.index}_${sts.index}">See
    All</button>
296 </td>
297 </tr>
298 </c:forEach>
299 </tbody>
300 </table>
301 </div>
302 </div>
303 </c:forEach>
304 <c:forEach var = "mc" items = "${munci}"
    varStatus = "stat">
305 <c:forEach var = "br" items = "${brgys[stat.
    index]}" varStatus = "sts">
306 <div class = "modal fade" id = "allStack4_${stat
    .index}_${sts.index}" data-width = "850">
307 <div class = "modal-body">
308 <c:set var = "whatStat" value = "3" scope = "
    request" />
309 <c:set var = "editStat" value = "1" scope = "
    request" />
310 <c:set var = "muncity" value = "${mc.muncity}"
    scope = "request" />
311 <c:set var = "barangay" value = "${br.barangay
    }" scope = "request" />
312 <table class="table table-striped" id="
    exampleBP4_${stat.index}_${sts.index}">
313 <jsp:include page = "records2.jsp" />
314 </table>
315 </div>
316 </div>
317 </c:forEach>
318 </c:forEach>
319 </div>
320 </div>
321 </div>
322 </div>

```

Listing 51: records.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
    jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
    functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
    springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
    springframework.org/tags/form" %>
5
6 <thead>
7 <tr>
8 <th>Barangay</th>
9 <th>Date Submitted</th>
10 <th>Submitted by</th>
11 <c:if test = "${whatStat == 1}">
12 <th>Date Approved</th>
13 <th>Approved by</th>
14 </c:if>
15 <c:if test = "${whatStat == 2}">
16 <th>Date Rejected</th>
17 <th>Rejected by</th>
18 </c:if>
19 <c:if test = "${editStat == 1}">
20 <th>Date Edited</th>
21 <th>Edited by</th>
22 </c:if>
23 <th>View</th>
24 </tr>
25 </thead>
26 <tbody>
27 <c:forEach var="bR1" items="${allBRecords}"
    varStatus = "status">
28 <c:if test = "${barangay == bR1.barangay && (
    bR1.status == whatStat || (whatStat == 3 &&
    bR1.editStatus == editStat)) }">
29 <tr>
30 <td>
31 ${bR1.barangay}
32 </td>
33 <td>
34 ${bR1.dateSubmitted}
35 </td>
36 <td>
37 ${bR1.healthworker}
38 </td>
39 <c:if test = "${whatStat == 1 || whatStat ==
    2}">
40 <td>
41 ${bR1.dateApproved}
42 </td>
43 <td>
44 ${bR1.muncityHW}
45 </td>
46 </c:if>
47 <c:if test = "${editStat == 1}">
48 <td>
49 ${bR1.dateEdited}
50 </td>
51 <td>
52 ${bR1.edittedBy}
53 </td>
54 </c:if>
55 <td>
56 <button type = "button" class = "btn btn-
    primary" data-toggle = "modal" href = "#
    stack1_${status.index}">View</button>
57 </td>
58 </tr>
59 <div id = "stack1_${status.index}" class="modal
    fade" tabIndex="-1" data-width = "860">
60 <div class="modal-header">
61 <button type="button" class="close" data-
    dismiss="modal" aria-hidden="true"> </
    button>
62 </div>
63 <div class="modal-body">
64 <div class = "panel panel-info">
65 <div class = "panel-heading">
66 Record Information
67 </div>
68 <div class = "panel-body">
69 <div class = "row">
70 <div class = "col-lg-2">
71 <label>Region:</label>
72 </div>
73 <div class = "col-lg-4">
74 <input type = "text" class = "form-control
    " readonly = "true" value = "${bR1.region}"
    />
75 </div>
76 <div class = "col-lg-2">
77 <label>Province:</label>

```

```

78     </div>
79     <div class = "col-lg-4">
80     <input type = "text" class = "form-control
      " readonly = "true" value = "${bR1.province
      }" />
81     </div>
82     </div>
83     <br />
84     <div class = "row">
85     <div class = "col-lg-2">
86     <label>Municipality/City:</label>
87     </div>
88     <div class = "col-lg-4">
89     <input type = "text" class = "form-control
      " readonly = "true" value = "${bR1.muncity
      }" />
90     </div>
91     <div class = "col-lg-2">
92     <label>Barangay</label>
93     </div>
94     <div class = "col-lg-4">
95     <input type = "text" class = "form-control
      " readonly = "true" value = "${bR1.barangay
      }" />
96     </div>
97     </div>
98     <br />
99     <div class = "row">
100    <div class = "col-lg-2">
101    <label>Barangay Health Worker:</label>
102    </div>
103    <div class = "col-lg-4">
104    <input type = "text" class = "form-control
      " readonly = "true" value = "${bR1.
      healthworker}" />
105    </div>
106    <div class = "col-lg-2">
107    <label>Date Submitted:</label>
108    </div>
109    <div class = "col-lg-4">
110    <input type = "text" class = "form-control
      " readonly = "true" value = "${bR1.
      dateSubmitted}" />
111    </div>
112    </div>
113    <c:if test = "${whatStat == 1}">
114    <br />
115    <div class = "row">
116    <div class = "col-lg-2">
117    <label>Municipality/City Health Worker:</
      label>
118    </div>
119    <div class = "col-lg-4">
120    <input type = "text" class = "form-control
      " readonly = "true" value = "${bR1.
      muncityHW}" />
121    </div>
122    <div class = "col-lg-2">
123    <label>Date Approved:</label>
124    </div>
125    <div class = "col-lg-4">
126    <input type = "text" class = "form-control
      " readonly = "true" value = "${bR1.
      dateApproved}" />
127    </div>
128    </div>
129    </c:if>
130    <c:if test = "${whatStat == 2}">
131    <br />
132    <div class = "row">
133    <div class = "col-lg-2">
134    <label>Municipality/City Health Worker:</
      label>
135    </div>
136    <div class = "col-lg-4">
137    <input type = "text" class = "form-control"
      readonly = "true" value = "${bR1.muncityHW
      }" />
138    </div>
139    <div class = "col-lg-2">
140    <label>Date Rejected:</label>
141    </div>
142    <div class = "col-lg-4">
143    <input type = "text" class = "form-control"
      readonly = "true" value = "${bR1.
      dateApproved}" />
144    </div>
145    </div>
146    </c:if>
147    <c:if test = "${bR1.editStatus == 1}">
148    <br />
149    <div class = "row">
150    <div class = "col-lg-2">
151    <label>Edited by:</label>
152    </div>
153    <div class = "col-lg-4">
154    <input type = "text" class = "form-control"
      readonly = "true" value = "${bR1.edittedBy
      }" />
155    </div>
156    <div class = "col-lg-2">
157    <label>Date Edited:</label>
158    </div>
159    <div class = "col-lg-4">
160    <input type = "text" class = "form-control"
      readonly = "true" value = "${bR1.dateEdited
      }" />
161    </div>
162    </div>
163    </c:if>
164    </div>
165    </div>
166    <div class = "panel panel-info">
167    <div class = "panel-heading">
168    <label>Patient's Information</label>
169    </div>
170    <div class = "panel-body">
171    <c:if test = "${not empty bR1.name}">
172    <div class = "row">
173    <div class = "col-lg-1">
174    <label>Name:</label>
175    </div>
176    <div class = "col-lg-5">
177    <input type = "text" class = "form-control"
      readonly = "true" value = "${bR1.name}" />
178    </div>
179    </div>
180    <br />
181    </c:if>
182    <c:if test = "${not empty bR1.address}">
183    <div class = "row">
184    <div class = "col-lg-1">
185    <label>Complete Address:</label>
186    </div>
187    <div class = "col-lg-5">
188    <textarea class = "form-control" readonly = "
      true" value = "${bR1.address}">${bR1.
      address}</textarea>
189    </div>
190    </div>
191    <br />
192    </c:if>
193    <div class = "row">
194    <div class = "col-lg-1">
195    <label>Age:</label>
196    </div>
197    <div class = "col-lg-2">
198    <input type = "text" class = "form-control"
      readonly = "true" value = "${bR1.age}" />
199    </div>
200    <div class = "col-lg-1">
201    <label>Date of Birth:</label>
202    </div>
203    <div class = "col-lg-3">
204    <input class="form-control" type="text"
      readonly = "true" value="${bR1.dateofbirth
      }" />
205    </div>
206    <div class = "col-lg-1">
207    <label>Gender:</label>
208    </div>
209    <div class = "col-lg-3">
210    <input class = "form-control" type = "text"
      readonly = "true" value = "${bR1.gender}" />
211    </div>
212    </div>
213    </div>
214    </div>
215    <div class = "panel panel-info">
216    <div class = "panel-heading">
217    <label>Patient's Clinical Information</label>
218    </div>
219    <div class = "panel-body">
220    <div class = "row">
221    <div class = "col-lg-2">
222    <label>Disease:</label>
223    </div>
224    <div class = "col-lg-4">
225    <input type = "text" class = "form-control"
      readonly = "true" value = "${bR1.disease
      }" />
226    </div>
227    <div class = "col-lg-2">
228    <label>Date Onset:</label>
229    </div>
230    <div class = "col-lg-4">
231    <input class="form-control" type="text"
      readonly = "true" value="${bR1.dateOnset}"
      />
232    </div>
233    </div>
234    <br />
235    <div class = "row">
236    <div class = "col-lg-2">
237    <label>Symptoms:</label>
238    </div>
239    <div class = "col-lg-4">
240    <input type = "text" class = "form-control"

```

```

        readonly = "true" value = "${bR1.symptoms
241     }"/>
242 </div>
243 <div class = "col-lg-2">
244     <label>Case Classification:</label>
245 </div>
246 <div class = "col-lg-4">
247     <input class="form-control" type="text"
248         readonly = "true" value="${bR1.caseType}"
249     />
250 </div>
251 <br />
252 <c:if test = "${bR1.admitted == 'No'}">
253 <div class = "row">
254 <div class = "col-lg-2">
255     <label>Patient Admitted?:</label>
256 </div>
257 <div class = "col-lg-4">
258     <input type = "text" class = "form-control"
259         readonly = "true" value = "${bR1.admitted
260     }"/>
261 </div>
262 </div>
263 </c:if>
264 <c:if test = "${bR1.admitted == 'Yes'}">
265 <div class = "row">
266 <div class = "col-lg-2">
267     <label>Patient Admitted?:</label>
268 </div>
269 <div class = "col-lg-4">
270     <input type = "text" class = "form-control"
271         readonly = "true" value = "${bR1.admitted
272     }"/>
273 </div>
274 </div>
275 <br />
276 <div class = "row">
277 <div class = "col-lg-2">
278     <label>Hospital/Clinic:</label>
279 </div>
280 <div class = "col-lg-5">
281     <input type = "text" class = "form-control"
282         readonly = "true" value = "${bR1.hospital
283     }"/>
284 </div>
285 <div class = "col-lg-2">
286     <label>Hospital/Clinic Type:</label>
287 </div>
288 <div class = "col-lg-3">
289     <input class="form-control" type="text"
290         readonly = "true" value="${bR1.hospitalType
291     }" />
292 </div>
293 </div>
294 <br />
295 <div class = "row">
296 <div class = "col-lg-3">
297     <label>Hospital/Clinic Address:</label>
298 </div>
299 <div class = "col-lg-9">
300     <input type = "text" class = "form-control"
301         readonly = "true" value = "${bR1.
302     hospitalAddr}"/>
303 </div>
304 </div>
305 </c:if>
306 <br />
307 <div class = "row">
308 <div class = "col-lg-2">
309     <label>Outcome</label>
310 </div>
311 <div class = "col-lg-4">
312     <input type = "text" class = "form-control"
313         readonly = "true" value = "${bR1.outcome
314     }"/>
315 </div>
316 </div>
317 <br />
318 </div>
319 </div>
320 </div>
321 <c:if test = "${(whatStat == 0 || whatStat == 3)
322     && (bR1.editStatus == 0 || bR1.editStatus
323     == 1)}">
324 <div class="modal-footer">
325     <button type = "button" data-toggle = "modal"
326         href = "#editIt_${status.index}" class = "
327         btn btn-primary">Edit</button>
328     <button type = "button" data-toggle = "modal"
329         href = "#rejectIt_${status.index}" class =
330         "btn btn-primary">Reject</button>
331     <button type = "button" data-toggle = "modal"
332         href = "#submitIt_${status.index}" class =
333         "btn btn-primary">Submit</button>
334 </div>
335 </c:if>
336 </div>
337 <div id = "editIt_${status.index}" class = "
338     modal fade" tabindex = "-1" data-focus-on =
339     "input:first">
340 <div class = "modal-header">
341     <button type="button" class="close" data-
342     dismiss="modal" aria-hidden="true"> </
343     button>
344 </div>
345 <div class = "modal-body">
346     Are you sure you want to edit this patient
347     record?
348 </div>
349 <div class = "modal-footer">
350     <a href = "${pageContext.request.contextPath}/
351     pages/system/editRecord/${bR1.recordID}"
352     class = "btn btn-primary">Yes</a>
353     <a href = "" class = "btn btn-primary" data-
354     dismiss="modal">No</a>
355 </div>
356 </div>
357 <div id = "rejectIt_${status.index}" class = "
358     modal fade" tabindex = "-1" data-focus-on =
359     "input:first">
360 <div class = "modal-header">
361     <button type="button" class="close" data-
362     dismiss="modal" aria-hidden="true"> </
363     button>
364 </div>
365 <div class = "modal-body">
366     Are you sure you want to reject this patient
367     record?
368 </div>
369 <div class = "modal-footer">
370     <a href = "${pageContext.request.contextPath}/
371     pages/system/rejectMunCityRecords/${bR1.
372     recordID}" class = "btn btn-primary">Yes</a>
373     <a href = "" class = "btn btn-primary" data-
374     dismiss="modal">No</a>
375 </div>
376 </div>
377 <div id = "submitIt_${status.index}" class = "
378     modal fade" tabindex = "-1" data-focus-on =
379     "input:first">
380 <div class = "modal-header">
381     <button type="button" class="close" data-
382     dismiss="modal" aria-hidden="true"> </
383     button>
384 </div>
385 <div class = "modal-body">
386     Are you sure you want to submit this patient
387     record?
388 </div>
389 <div class = "modal-footer">
390     <a href = "${pageContext.request.contextPath}/
391     pages/system/addToMunCityRecords/${bR1.
392     recordID}" class = "btn btn-primary">Yes</a>
393     <a href = "" class = "btn btn-primary" data-
394     dismiss="modal">No</a>
395 </div>
396 </div>
397 </c:forEach>
398 </tbody>

```

Listing 52: notifications.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
3 <%@ taglib prefix="sec" uri="http://www.
4 <%@ taglib prefix="form" uri="http://www.
5 <%@ taglib prefix="fmt" uri="http://java.sun.com
    functions" prefix="fn"%>
    springframework.org/security/tags" %>
    springframework.org/tags/form" %>
    /jsp/jstl/fmt" %>

```



```

6
7 <jsp:include page = "../header.jsp" />
8 <div class = "tabbable">
9 <jsp:include page = "linkings.jsp" />
10 <div class = "tab-content">
11 <div class = "tab-pane active">
12 <div class = "tabbable">
13 <ul class = "nav nav-pills">
14 <li <c:if test = "${empty success}">class =
15 <active"</c:if>>
16 <a href = "#taba" data-toggle = "tab">
17 Alerts<span class="badge pull-right">${
18 numAlerts}</span></a>
19 </li>
20 <li class = "dropdown <c:if test = "${not
21 empty success}">active</c:if>>
22 <a href = "" data-toggle = "dropdown"
23 class = "dropdown-toggle">Registrations <
24 span class = "caret"></span></a>
25 <ul class="dropdown-menu" style = "text-
26 align: left;">
27 <li <c:if test = "${not empty success}">
28 class = "active"</c:if>><a href = "#tabb1"
29 data-toggle = "tab">Pending<span class="
30 badge pull-right">${pending}</span></a></li>
31 <li><a href = "#tabb2" data-toggle = "tab
32 ">Approved<span class="badge pull-right">${
33 approved}</span></a></li>
34 <li><a href = "#tabb3" data-toggle = "tab
35 ">Rejected<span class="badge pull-right">${
36 rejected}</span></a></li>
37 </ul>
38 </li>
39 </ul>
40 </div>
41 <div class = "tab-content" style = "height:
42 800px;">
43 <div class = "tab-pane <c:if test = "${empty
44 success}">active</c:if>" id = "taba">
45 <br />
46 <table class = "table table-striped" id = "
47 alertTable">
48 <thead>
49 <tr>
50 <th>Date</th>
51 <th>Disease</th>
52 <th>Outbreak Percentage</th>
53 <th>Barangay</th>
54 <th>Municipality</th>
55 <th>Province</th>
56 <th>Region</th>
57 </tr>
58 </thead>
59 <tbody>
60 <c:forEach var = "alertS" items = "${
61 alerts}">
62 <tr <c:if test = "${alertS.status == 0}">
63 class = "danger"</c:if>>
64 <td>${alertS.date}</td>
65 <td>${alertS.disease}</td>
66 <td><fmt:formatNumber type="number"
67 maxFractionDigits="5" groupingUsed="false"
68 value="${alertS.outbreak * 100}" />%</td>
69 <td>${alertS.barangay}</td>
70 <td>${alertS.muncity}</td>
71 <td>${alertS.province}</td>
72 <td>${alertS.region}</td>
73 </tr>
74 </c:forEach>
75 </tbody>
76 </table>
77 </div>
78 <div class = "tab-pane" id = "tabb1">
79 <div class = "tabbable">
80 <ul class = "nav nav-pills">
81 <sec:authorize access="hasAnyRole('
82 ROLE_PHOfficial', 'ROLE_MCOfficial')">
83 <li <c:if test = "${pageVisit == '
84 pofficial' || pageVisit == 'mofficial'}">
85 class = "active"</c:if>>
86 <a href = "#tabbia" data-toggle = "tab">
87 Barangay<span class="badge pull-right"></
88 span></a>
89 </li>
90 </sec:authorize>
91 <sec:authorize access="hasAnyRole('
92 ROLE_MCOfficial', 'ROLE_PHOfficial', '
93 ROLE_NHOfficial', 'ROLE_USER')">
94 <li <c:if test = "${pageVisit == 'nofficial'
95 || pageVisit == 'admin'}">class = "active
96 "</c:if>>
97 <a href = "#tabb1b" data-toggle = "tab">
98 Municipal/City<span class="badge pull-right
99 "></span></a>
100 </li>
101 </sec:authorize>
102 <sec:authorize access="hasAnyRole('
103 ROLE_PHOfficial', 'ROLE_NHOfficial', '
104 ROLE_USER')">
105 <li>
106 <a href = "#tabbic" data-toggle = "tab">
107 Provincial<span class="badge pull-right"></
108 span></a>
109 </li>
110 </sec:authorize>
111 <sec:authorize access="hasAnyRole('
112 ROLE_NHOfficial', 'ROLE_USER')">
113 <li>
114 <a href = "#tabbie" data-toggle = "tab">
115 Researchers<span class="badge pull-right
116 "></span></a>
117 </li>
118 </sec:authorize>
119 </ul>
120 </div>
121 </div>
122 <div class = "tab-content">
123 <div class = "tab-pane <c:if test = "${
124 pageVisit == 'pofficial' || pageVisit == '
125 mofficial'}">active</c:if>" id = "tabbia">
126 <div class = "container-fluid well-lg" style =
127 "background-color: #eee;">
128 <div class = "panel panel-info">
129 <div class = "panel-heading"><h3>Pending
130 Registrations</h3></div>
131 <c:if test = "${pageVisit == 'pofficial'}">
132 <jsp:include page = "notifs/barangay.jsp" />
133 </c:if>
134 <c:if test = "${pageVisit == 'mofficial'}">
135 <jsp:include page = "notifs/bar.jsp" />
136 </c:if>
137 </div>
138 </div>
139 <div class = "tab-pane <c:if test = "${pageVisit
140 == 'nofficial' || pageVisit == 'admin'}">
141 active</c:if>" id = "tabb1b">
142 <div class = "container-fluid well-lg" style =
143 "background-color: #eee;">
144 <div class = "panel panel-info">
145 <div class = "panel-heading"><h3>Pending
146 Registrations</h3></div>
147 <c:if test = "${pageVisit == 'pofficial' ||
148 pageVisit == 'admin' || pageVisit == '
149 nofficial'}">
150 <jsp:include page = "notifs/muncity.jsp" />
151 </c:if>
152 <c:if test = "${pageVisit == 'mofficial'}">
153 <jsp:include page = "notifs/mun.jsp" />
154 </c:if>
155 </div>
156 </div>
157 <div class = "tab-pane" id = "tabbic">
158 <div class = "container-fluid well-lg" style =
159 "background-color: #eee;">
160 <div class = "panel panel-info">
161 <div class = "panel-heading"><h3>Pending
162 Registrations</h3></div>
163 <jsp:include page = "notifs/province.jsp" />
164 </div>
165 </div>
166 <div class = "tab-pane" id = "tabbie">
167 <div class = "container-fluid well-lg" style =
168 "background-color: #eee;">
169 <div class = "panel panel-info">
170 <div class = "panel-heading"><h3>Pending
171 Registrations</h3></div>
172 <jsp:include page = "notifs/researcher.jsp"
173 />
174 </div>
175 </div>
176 <div class = "tab-pane <c:if test = "${pageVisit
177 == 'nofficial' || pageVisit == 'admin'}">
178 active</c:if>" id = "tabb1b">
179 </div>
180 </div>
181 </div>

```

```

142 <div class = "tab-pane" id = "tabb2">
143 <div class = "tabbable">
144 <ul class = "nav nav-pills">
145 <sec:authorize access="hasAnyRole('
146 ROLE_PHOfficial', 'ROLE_MCOfficial')">
147 <li <c:if test = "${pageVisit == 'pofficial'
148 || pageVisit == 'mofficial'}">class = "
149 active"</c:if>>
150 <a href = "#tabb2a" data-toggle = "tab">
151 Barangay<span class="badge pull-right"></
152 span></a>
153 </li>
154 </sec:authorize>
155 <sec:authorize access="hasAnyRole('
156 ROLE_MCOfficial', 'ROLE_PHOfficial', '
157 ROLE_NHOfficial', 'ROLE_USER')">
158 <li <c:if test = "${pageVisit == 'nofficial'
159 || pageVisit == 'admin'}">class = "active
160 "</c:if>>
161 <a href = "#tabb2b" data-toggle = "tab">
162 Municipal/City<span class="badge pull-right
163 "></span></a>
164 </li>
165 </sec:authorize>
166 <sec:authorize access="hasAnyRole('
167 ROLE_PHOfficial', 'ROLE_NHOfficial', '
168 ROLE_USER')">
169 <li <c:if test = "${pageVisit == 'nofficial'
170 || pageVisit == 'admin'}">class = "active
171 "</c:if>>
172 <a href = "#tabb2c" data-toggle = "tab">
173 Provincial<span class="badge pull-right"></
174 span></a>
175 </li>
176 </sec:authorize>
177 <sec:authorize access="hasAnyRole('
178 ROLE_USER')">
179 <li <c:if test = "${pageVisit == 'nofficial'
180 || pageVisit == 'admin'}">class = "active
181 "</c:if>>
182 <a href = "#tabb2d" data-toggle = "tab">
183 National<span class="badge pull-right"></
184 span></a>
185 </li>
186 </sec:authorize>
187 <sec:authorize access="hasAnyRole('
188 ROLE_PHOfficial', 'ROLE_NHOfficial', '
189 ROLE_USER')">
190 <li <c:if test = "${pageVisit == 'nofficial'
191 || pageVisit == 'admin'}">class = "active
192 "</c:if>>
193 <a href = "#tabb2e" data-toggle = "tab">
194 Researchers<span class="badge pull-right
195 "></span></a>
196 </li>
197 </sec:authorize>
198 </ul>
199 </div>
200 <div class = "tab-content">
201 <div class = "tab-pane <c:if test = "${
202 pageVisit == 'pofficial' || pageVisit == '
203 mofficial'}">active</c:if>" id = "tabb2a">
204 <div class = "container-fluid well-lg" style =
205 "background-color: #eee;">
206 <div class = "panel panel-info">
207 <div class = "panel-heading"><h3>Approved
208 Registrations</h3></div>
209 <c:if test = "${pageVisit == 'pofficial'}">
210 <jsp:include page = "notifs/barangayA.jsp"
211 />
212 </c:if>
213 <c:if test = "${pageVisit == 'mofficial'}">
214 <jsp:include page = "notifs/barA.jsp" />
215 </c:if>
216 </div>
217 </div>
218 <div class = "tab-pane <c:if test = "${
219 pageVisit == 'nofficial' || pageVisit == '
220 admin'}">active</c:if>" id = "tabb2b">
221 <div class = "container-fluid well-lg" style =
222 "background-color: #eee;">
223 <div class = "panel panel-info">
224 <div class = "panel-heading"><h3>Approved
225 Registrations</h3></div>
226 <c:if test = "${pageVisit == 'pofficial' ||
227 pageVisit == 'admin'}">
228 <jsp:include page = "notifs/muncityA.jsp" />
229 </c:if>
230 <c:if test = "${pageVisit == 'nofficial' ||
231 pageVisit == 'admin'}">
232 <jsp:include page = "notifs/munA.jsp" />
233 </c:if>
234 </div>
235 </div>
236 <div class = "tab-pane" id = "tabb2c">
237 <div class = "container-fluid well-lg" style =
238 "background-color: #eee;">
239 <div class = "panel panel-info">
240 <div class = "panel-heading"><h3>Approved
241 Registrations</h3></div>
242 <c:if test = "${pageVisit == 'pofficial'}">
243 <jsp:include page = "notifs/nationalA.jsp" />
244 </c:if>
245 <c:if test = "${pageVisit == 'nofficial' ||
246 pageVisit == 'admin'}">
247 <jsp:include page = "notifs/researcherA.jsp"
248 />
249 </c:if>
250 </div>
251 </div>
252 <div class = "tab-pane" id = "tabb3">
253 <div class = "tabbable">
254 <ul class = "nav nav-pills">
255 <sec:authorize access="hasAnyRole('
256 ROLE_PHOfficial', 'ROLE_MCOfficial')">
257 <li <c:if test = "${pageVisit == 'pofficial'
258 || pageVisit == 'mofficial'}">class = "
259 active"</c:if>>
260 <a href = "#tabb3a" data-toggle = "tab">
261 Barangay<span class="badge pull-right"></
262 span></a>
263 </li>
264 </sec:authorize>
265 <sec:authorize access="hasAnyRole('
266 ROLE_MCOfficial', 'ROLE_PHOfficial', '
267 ROLE_NHOfficial', 'ROLE_USER')">
268 <li <c:if test = "${pageVisit == 'nofficial'
269 || pageVisit == 'admin'}">class = "active
270 "</c:if>>
271 <a href = "#tabb3b" data-toggle = "tab">
272 Municipal/City<span class="badge pull-right
273 "></span></a>
274 </li>
275 </sec:authorize>
276 <sec:authorize access="hasAnyRole('
277 ROLE_PHOfficial', 'ROLE_NHOfficial', '
278 ROLE_USER')">
279 <li <c:if test = "${pageVisit == 'nofficial'
280 || pageVisit == 'admin'}">class = "active
281 "</c:if>>
282 <a href = "#tabb3c" data-toggle = "tab">
283 Provincial<span class="badge pull-right"></
284 span></a>
285 </li>
286 </sec:authorize>
287 <sec:authorize access="hasAnyRole('
288 ROLE_USER')">
289 <li <c:if test = "${pageVisit == 'nofficial'
290 || pageVisit == 'admin'}">class = "active
291 "</c:if>>
292 <a href = "#tabb3d" data-toggle = "tab">
293 National<span class="badge pull-right"></
294 span></a>
295 </li>
296 </sec:authorize>
297 <sec:authorize access="hasAnyRole('
298 ROLE_NHOfficial', 'ROLE_USER')">
299 <li <c:if test = "${pageVisit == 'nofficial'
300 || pageVisit == 'admin'}">class = "active
301 "</c:if>>
302 <a href = "#tabb3e" data-toggle = "tab">
303 Researchers<span class="badge pull-right
304 "></span></a>
305 </li>
306 </sec:authorize>
307 </ul>
308 </div>
309 <div class = "tab-content">
310 <div class = "tab-pane <c:if test = "${
311 pageVisit == 'pofficial' || pageVisit == '
312 mofficial'}">active</c:if>" id = "tabb3a">
313 <div class = "container-fluid well-lg" style =
314 "background-color: #eee;">
315 <div class = "panel panel-info">
316 <div class = "panel-heading"><h3>Rejected
317 Registrations</h3></div>
318 <c:if test = "${pageVisit == 'pofficial'}">
319 <jsp:include page = "notifs/barangayP.jsp"
320 />
321 </c:if>
322 <c:if test = "${pageVisit == 'nofficial'}">
323 <jsp:include page = "notifs/barP.jsp" />
324 </c:if>
325 </div>
326 </div>
327 <div class = "tab-pane <c:if test = "${
328 pageVisit == 'nofficial' || pageVisit == '
329 admin'}">active</c:if>" id = "tabb3b">
330 <div class = "container-fluid well-lg" style =
331 "background-color: #eee;">
332 <div class = "panel panel-info">
333 <div class = "panel-heading"><h3>Rejected
334 Registrations</h3></div>
335 </div>
336 </div>
337 <div class = "tab-pane <c:if test = "${
338 pageVisit == 'nofficial' || pageVisit == '
339 admin'}">active</c:if>" id = "tabb3c">
340 <div class = "container-fluid well-lg" style =
341 "background-color: #eee;">
342 <div class = "panel panel-info">
343 <div class = "panel-heading"><h3>Rejected
344 Registrations</h3></div>
345 </div>
346 </div>
347 <div class = "tab-pane <c:if test = "${
348 pageVisit == 'nofficial' || pageVisit == '
349 admin'}">active</c:if>" id = "tabb3d">
350 <div class = "container-fluid well-lg" style =
351 "background-color: #eee;">
352 <div class = "panel panel-info">
353 <div class = "panel-heading"><h3>Rejected
354 Registrations</h3></div>
355 </div>
356 </div>
357 <div class = "tab-pane <c:if test = "${
358 pageVisit == 'nofficial' || pageVisit == '
359 admin'}">active</c:if>" id = "tabb3e">
360 <div class = "container-fluid well-lg" style =
361 "background-color: #eee;">
362 <div class = "panel panel-info">
363 <div class = "panel-heading"><h3>Rejected
364 Registrations</h3></div>
365 </div>
366 </div>

```

```

273 <c:if test = "${pageVisit == 'pofficial' ||
    pageVisit == 'admin' || pageVisit == '
    nonofficial'}">
274 <jsp:include page = "notifs/muncityP.jsp" />
275 </c:if>
276 <c:if test = "${pageVisit == 'mofficial'}">
277 <jsp:include page = "notifs/munP.jsp" />
278 </c:if>
279 </div>
280 </div>
281 </div>
282 <div class = "tab-pane" id = "tabb3c">
283 <div class = "container-fluid well-lg" style =
    "background-color: #eee;">
284 <div class = "panel panel-info">
285 <div class = "panel-heading"><h3>Rejected
    Registrations</h3></div>
286 <jsp:include page = "notifs/provinceP.jsp" />
287 </div>
288 </div>
289 </div>
290 <div class = "tab-pane" id = "tabb3d">
291 <div class = "container-fluid well-lg" style =
    "background-color: #eee;">
292 <div class = "panel panel-info">
293 <div class = "panel-heading"><h3>Rejected
    Registrations</h3></div>
294 <jsp:include page = "notifs/nationalP.jsp" />
295 </div>
296 </div>
297 </div>
298 <div class = "tab-pane" id = "tabb3e">
299 <div class = "container-fluid well-lg" style =
    "background-color: #eee;">
300 <div class = "panel panel-info">
301 <div class = "panel-heading"><h3>Rejected
    Registrations</h3></div>
302 <jsp:include page = "notifs/researcherP.jsp"
    />
303 </div>
304 </div>
305 </div>
306 </div>
307 </div>
308 </div>
309 </div>
310 </div>
311 </div>
312 </div>
313 <jsp:include page = "../footer.jsp" />

```

Listing 53: barangay.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
    jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
    functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
    springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
    springframework.org/tags/form" %>
5
6
7 <c:forEach var = "i" begin = "0" end = "16">
8 <c:forEach var = "provs" items = "${prov[i]}"
    varStatus = "stat">
9 <c:if test = "${fn:toLowerCase(user.province) ==
    fn:toLowerCase(provs.province)}">
10 <table class="table table-striped" id = "
    example1">
11 <thead>
12 <tr>
13 <th>Municipality/City</th>
14 <th>No. of Barangay Health Workers</th>
15 </tr>
16 </thead>
17 <tbody>
18
19 <c:forEach var = "j" begin = "0" end = "${fn:
    length(mun) - 1}">
20 <c:forEach var = "mun" items = "${mun[j]}"
    varStatus = "munSt">
21 <c:if test = "${fn:toLowerCase(user.province)
    == fn:toLowerCase(mun.province) && mun.
    region == regions[i]}">
22 <c:set var = "counter" value = "0" />
23 <c:forEach var = "reg" items = "${
    registrations}" varStatus = "st">
24 <c:if test = "${fn:toLowerCase(reg.province)
    == fn:toLowerCase(provs.province) && fn:
    toLowerCase(reg.muncity) == fn:toLowerCase(
    mun.muncity)
25 && reg.userType == 'Barangay Health Worker' &&
    reg.status == 0}">
26 <c:set var = "counter" value = "${counter +
    1}" />
27 </c:if>
28 </c:forEach>
29 <tr>
30 <th>
31 <button type = "button" class = "btn btn-
    primary" href = "#muncityModali_${i}_${stat.
    index}_${j}_${munSt.index}" data-toggle =
    "modal">${mun.muncity}</button>
32 </th>
33 <th>${counter}</th>
34 </tr>
35 </c:if>
36 </c:forEach>
37 </c:forEach>
38 </tbody>
39 </table>
40 </c:if>
41 </c:forEach>
42 </c:forEach>
43 <c:forEach var = "i" begin = "0" end = "16">
44 <c:forEach var = "provs" items = "${prov[i]}"
    varStatus = "stat">
45 <c:if test = "${fn:toLowerCase(user.province) ==
    fn:toLowerCase(provs.province)}">
46 <c:forEach var = "j" begin = "0" end = "${fn:
    length(mun) - 1}">
47 <c:forEach var = "mun" items = "${mun[j]}"
    varStatus = "munSt">
48 <c:if test = "${fn:toLowerCase(user.province) ==
    fn:toLowerCase(mun.province) && mun.region
    == regions[i]}">
49 <div class = "modal fade" id = "muncityModali_${
    i}_${stat.index}_${j}_${munSt.index}" data-
    width = "700">
50 <div class = "modal-body">
51 <table class="table table-striped" id = "
    example1">
52 <thead>
53 <tr>
54 <th>Barangay</th>
55 <th>No. of Barangay Health Workers</th>
56 <th>View All Users</th>
57 </tr>
58 </thead>
59 <tbody>
60
61 <c:forEach var = "k" begin = "0" end = "${fn:
    length(brgy) - 1}">
62 <c:forEach var = "brgy" items = "${brgy[k]}"
    varStatus = "brgSt">
63 <c:if test = "${fn:toLowerCase(user.muncity)
    == fn:toLowerCase(brgy.muncity) && fn:
    toLowerCase(user.province) == fn:
    toLowerCase(mun.province) && mun.region ==
    regions[i]}">
64 <c:set var = "counter" value = "0" />
65 <c:forEach var = "reg" items = "${
    registrations}" varStatus = "st">
66 <c:if test = "${fn:toLowerCase(reg.province)
    == fn:toLowerCase(provs.province) && reg.
    region == regions[i]
67 && fn:toLowerCase(reg.muncity) == fn:
    toLowerCase(mun.muncity) && reg.barangay ==
    brgy.barangay
68 && reg.userType == 'Barangay Health Worker'
    && reg.status == 0}">
69 <c:set var = "counter" value = "${counter +
    1}" />
70 </c:if>
71 </c:forEach>
72 <tr>
73 <th>${brgy.barangay}</th>
74 <th>${counter}</th>
75 <th>
76 <button type = "button" class = "btn btn-
    primary" href = "#usersModali_${i}_${stat.
    index}_${j}_${munSt.index}_${k}_${brgSt.
    index}" data-toggle = "modal">See All</
    button>
77 </th>
78 </tr>
79 </c:if>
80 </c:forEach>
81 </c:forEach>
82 </tbody>
83 </table>
84 </div>
85 </div>
86 </c:if>
87 </c:forEach>
88 </c:forEach>
89 </c:if>
90 </c:forEach>
91 </c:forEach>
92 <c:forEach var = "i" begin = "0" end = "16">
93 <c:forEach var = "provs" items = "${prov[i]}"
    varStatus = "stat">
94 <c:if test = "${fn:toLowerCase(user.province) ==
    fn:toLowerCase(provs.province)}">

```

```

95 <c:forEach var = "j" begin = "0" end = "${fn:
length(mun) - 1}">
96 <c:forEach var = "mun" items = "${mun[j]}"
varStatus = "munSt">
97 <c:if test = "${fn:toLowerCase(user.province) ==
fn:toLowerCase(mun.province) && mun.region
== regions[i]}">
98 <c:forEach var = "k" begin = "0" end = "${fn:
length(brgy) - 1}">
99 <c:forEach var = "brgy" items = "${brgy[k]}"
varStatus = "brgSt">
100 <c:if test = "${fn:toLowerCase(user.muncity) ==
fn:toLowerCase(brgy.muncity) && fn:
toLowerCase(user.province) == fn:
toLowerCase(mun.province) && mun.region ==
regions[i]}">
101 <div class = "modal fade" id = "usersModal1_${i}
_${stat.index}_${j}_${munSt.index}_${k}_${
brgSt.index}" data-width = "700">
102 <div class = "modal-body">
103 <table class="table table-striped" id = "
example1">
104 <thead>
105 <tr>
106 <th>Name</th>
107 <th>User Type</th>
108 <th>Profile</th>
109 <th>Approve</th>
110 <th>Reject</th>
111 </tr>
112 </thead>
113 <tbody>
114 <c:forEach var = "reg" items = "${
registrations}" varStatus = "st">
115 <c:if test = "${fn:toLowerCase(reg.province)
== fn:toLowerCase(provs.province) && reg.
region == regions[i]
&& fn:toLowerCase(reg.muncity) == fn:
toLowerCase(mun.muncity) && reg.barangay ==
brgy.barangay
&& reg.userType == 'Barangay Health Worker'
&& reg.status == 0}">
116 <tr>
117 <td>${reg.firstname} ${reg.middlename} ${
reg.surname}</td>
118 <td>${reg.userType}</td>
119 <td><button class = "btn btn-primary" href
= "#profile1_${i}_${stat.index}_${j}_${
munSt.index}_${k}_${brgSt.index}_${st.index
}" data-toggle = "modal">View</button></td>
120 <td><button class = "btn btn-primary" href
= "#approve1_${st.index}" data-toggle = "
modal">Approve</button></td>
121 <td><button class = "btn btn-primary" href
= "#reject1_${st.index}" data-toggle = "
modal">Reject</button></td>
122 </tr>
123 <div class = "modal fade" id = "approve1_${
st.index}">
124 <div class = "modal-body">
125 Are you sure you want to approve this
application?
126 </div>
127 <div class = "modal-footer">
128 <a href = "<%= request.getContextPath()
%>/pages/addUser/${reg.accountID}" class =
"btn btn-primary">Yes</a>
129 <a href = "" class = "btn btn-primary"
data-dismiss="modal">No</a>
130 </div>
131 <div class = "modal fade" id = "reject1_${st
.index}">
132 <div class = "modal-body">
133 Are you sure you want to reject this
application?
134 </div>
135 <div class = "modal-footer">
136 <a href = "<%= request.getContextPath()
%>/pages/rejectUser/${reg.accountID}" class
= "btn btn-primary">Yes</a>
137 <a href = "" class = "btn btn-primary"
data-dismiss="modal">No</a>
138 </div>
139 </c:if>
140 </tbody>
141 </table>
142 </div>
143 </c:if>
144 </tbody>
145 </table>
146 </div>
147 </div>
148 </c:if>
149 </c:forEach>
150 </c:forEach>
151 </c:forEach>
152 </c:if>
153 </c:forEach>
154 </c:forEach>
155 </c:if>
156 </c:forEach>
157 </c:forEach>
158 <c:forEach var = "i" begin = "0" end = "16">
159 <c:forEach var = "provs" items = "${prov[i]}"
varStatus = "stat">
160 <c:if test = "${fn:toLowerCase(user.province) ==
fn:toLowerCase(provs.province)}">
161 <c:forEach var = "j" begin = "0" end = "${fn:
length(mun) - 1}">
162 <c:forEach var = "mun" items = "${mun[j]}"
varStatus = "munSt">
163 <c:if test = "${fn:toLowerCase(user.province) ==
fn:toLowerCase(mun.province) && mun.region
== regions[i]}">
164 <c:forEach var = "k" begin = "0" end = "${fn:
length(brgy) - 1}">
165 <c:forEach var = "brgy" items = "${brgy[k]}"
varStatus = "brgSt">
166 <c:if test = "${fn:toLowerCase(user.muncity) ==
fn:toLowerCase(brgy.muncity) && fn:
toLowerCase(user.province) == fn:
toLowerCase(mun.province) && mun.region ==
regions[i]}">
167 <c:forEach var = "reg" items = "${registrations
}" varStatus = "st">
168 <c:if test = "${fn:toLowerCase(reg.province) ==
fn:toLowerCase(provs.province) && reg.
region == regions[i]
&& fn:toLowerCase(reg.muncity) == fn:toLowerCase
(mun.muncity) && reg.barangay == brgy.
barangay
&& reg.userType == 'Barangay Health Worker' &&
reg.status == 0}">
169 <div class = "modal fade" id = "profile1_${i}_${
stat.index}_${j}_${munSt.index}_${k}_${
brgSt.index}_${st.index}" data-width =
"700">
170 <div class = "modal-body">
171 <div class = "panel panel-info">
172 <div class = "panel-body">
173 <br />
174 <br />
175 <table class = "table table-striped">
176 <tr>
177 <td>First Name:</td>
178 <td>
179 <td>${reg.firstname}
180 </td>
181 </tr>
182 <tr>
183 <td>Middle Name:</td>
184 <td>
185 <td>${reg.middlename}
186 </td>
187 </tr>
188 <tr>
189 <td>Surname:</td>
190 <td>
191 <td>${reg.surname}
192 </td>
193 </tr>
194 <tr>
195 <td>Region:</td>
196 <td>
197 <td>${reg.region}
198 </td>
199 </tr>
200 <tr>
201 <td>Province:</td>
202 <td>
203 <td>${reg.province}
204 </td>
205 </tr>
206 <tr>
207 <td>Institution:</td>
208 <td>
209 <td>${reg.institution}
210 </td>
211 </tr>
212 <tr>
213 <td>Institution's Address:</td>
214 <td>
215 <td>${reg.institutionAddr}
216 </td>
217 </tr>
218 <tr>
219 <td>Email:</td>
220 <td>
221 <td>${reg.email}
222 </td>
223 </tr>
224 <tr>
225 <td>Contact Number:</td>
226 <td>
227 <td>${reg.number}
228 </td>
229 </tr>
230 </table>
231 </div>
232 </div>
233 </div>
234 </div>

```

```

235 </div>
236 </div>
237 </c:if>
238 </c:forEach>
239 </c:if>
240 </c:forEach>
241 </c:forEach>

```

Listing 54: muncity.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
  jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
  functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
  springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
  springframework.org/tags/form" %>
5
6 <div class = "panel-body">
7 <c:if test = "${not empty successD}">
8 <div class = "alert alert-success">
9 <strong>${successD}</strong>
10 </div>
11 </c:if>
12 <c:if test = "${pageVisit == 'admin' ||
  pageVisit == 'nonofficial'}">
13 <table class="table table-striped" id="example1
  ">
14 <thead>
15 <tr>
16 <th>Region</th>
17 <th>No. of Municipal/City Users</th>
18 </tr>
19 </thead>
20 <tbody>
21 <c:forEach var = "i" begin = "0" end = "16">
22 <tr>
23 <th>
24 <button type = "button" class = "btn btn-
  primary" href = "#regionModal2_${i}" data-
  toggle = "modal">${regions[i]}</button>
25 </th>
26 <c:set var = "counter" value = "0" />
27 <c:forEach var = "reg" items = "${
  registrations}">
28 <c:if test = "${reg.region == regions[i] && (
  reg.userType == 'Municipal/City Health
  Worker' || reg.userType == 'Municipal/City
  Health Official') && reg.status == 0}">
29 <c:set var = "counter" value = "${counter +
  1}" />
30 </c:if>
31 </c:forEach>
32 <th>
33 ${counter}
34 </th>
35 </tr>
36 </c:forEach>
37 </tbody>
38 </table>
39 <c:forEach var = "i" begin = "0" end = "16">
40 <div class = "modal fade" id = "regionModal2_${i}
  ">
41 <div class = "modal-body">
42 <table class="table table-striped" id="
  example1">
43 <thead>
44 <tr>
45 <th>Province</th>
46 <th>No. of Municipal/City Users</th>
47 </tr>
48 </thead>
49 <tbody>
50 <c:forEach var = "provs" items = "${prov[i]
  }" varStatus = "stat">
51 <c:set var = "counter" value = "0" />
52 <c:forEach var = "reg" items = "${
  registrations}">
53 <c:if test = "${fn:toLowerCase(reg.province)
  == fn:toLowerCase(provs.province) && reg.
  region == regions[i]
54 && (reg.userType == 'Municipal/City Health
  Worker' || reg.userType == 'Municipal/City
  Health Official') && reg.status == 0}">
55 <c:set var = "counter" value = "${counter +
  1}" />
56 </c:if>
57 </c:forEach>
58 <tr>
59 <th>
60 <button type = "button" class = "btn btn-
  primary" href = "#provinceModal2_${i}_${
  stat.index}" data-toggle = "modal">${provs.
  province}</button>
61 </th>
62 <th>${counter}</th>
63 </tr>
64 </c:forEach>
65 </tbody>
66 </table>
67 </div>
68 </div>
69 </c:forEach>
70 </c:if>
71 <c:if test = "${pageVisit == 'pofficial'}">
72 <c:forEach var = "i" begin = "0" end = "16">
73 <c:forEach var = "provs" items = "${prov[i]}"
  varStatus = "stat">
74 <c:if test = "${fn:toLowerCase(user.province) ==
  fn:toLowerCase(provs.province)}">
75 <table class="table table-striped" id="example1
  ">
76 <thead>
77 <tr>
78 <th>Municipality/City</th>
79 <th>No. of Municipal/City Health Workers/
  Officials</th>
80 <th>View All Users</th>
81 </tr>
82 </thead>
83 <tbody>
84 <c:forEach var = "j" begin = "0" end = "${fn:
  length(mun) - 1}">
85 <c:forEach var = "mun" items = "${mun[j]}"
  varStatus = "munSt">
86 <c:if test = "${fn:toLowerCase(user.province)
  == fn:toLowerCase(mun.province) && mun.
  region == regions[i]}">
87 <c:set var = "counter" value = "0" />
88 <c:forEach var = "reg" items = "${
  registrations}" varStatus = "st">
89 <c:if test = "${fn:toLowerCase(reg.province)
  == fn:toLowerCase(provs.province) && reg.
  region == regions[i]
90 && fn:toLowerCase(reg.muncity) == fn:
  toLowerCase(mun.muncity) && (reg.userType
  == 'Municipal/City Health Worker' ||
91 reg.userType == 'Municipal/City Health
  Official') && reg.status == 0}">
92 <c:set var = "counter" value = "${counter +
  1}" />
93 </c:if>
94 </c:forEach>
95 <tr>
96 <th>${mun.muncity}</th>
97 <th>${counter}</th>
98 <th>
99 <button class = "btn btn-primary" href = "#
  usersModal2_${i}_${stat.index}_${j}_${munSt
  .index}" data-toggle = "modal">See All</
  button>
100 </th>
101 </tr>
102 </tbody>
103 </c:if>
104 </c:forEach>
105 </c:forEach>
106 </tbody>
107 </table>
108 </c:if>
109 </c:forEach>
110 </c:forEach>
111 </c:if>
112 <c:if test = "${pageVisit == 'admin' ||
  pageVisit == 'nonofficial'}">
113 <c:forEach var = "i" begin = "0" end = "16">
114 <c:forEach var = "provs" items = "${prov[i]}"
  varStatus = "stat">
115 <div class = "modal fade" id = "provinceModal2_${
  i}_${stat.index}" data-width = "700">
116 <div class = "modal-body">
117 <table class="table table-striped" id="
  example1">
118 <thead>
119 <tr>
120 <th>Municipality/City</th>
121 <th>No. of Municipal/City Health Workers/
  Officials</th>
122 <th>View All Users</th>
123 </tr>
124 </thead>
125 <tbody>
126 <c:forEach var = "j" begin = "0" end = "${fn
  :length(mun) - 1}">
127 <c:forEach var = "mun" items = "${mun[j]}"
  varStatus = "munSt">
128 <c:if test = "${fn:toLowerCase(provs.

```

```

        province) == fn:toLowerCase(mun.province)
        && mun.region == regions[i]]">
129 <c:set var = "counter" value = "0" />
130 <c:forEach var = "reg" items = "${
registrations}" varStatus = "st">
131 <c:if test = "${fn:toLowerCase(reg.muncity)
== fn:toLowerCase(mun.muncity) && fn:
toLowerCase(reg.province) == fn:toLowerCase
(provs.province) && reg.region == regions[i
]}
132 && (reg.userType == 'Municipal/City Health
Worker' || reg.userType == 'Municipal/City
Health Official') && reg.status == 0}">
133 <c:set var = "counter" value = "${counter +
1}" />
134 </c:if>
135 </c:forEach>
136 <tr>
137 <th>${mun.muncity}</th>
138 <th>${counter}</th>
139 <th>
140 <button class = "btn btn-primary" href = "#
usersModal2_${i}_${stat.index}_${j}_${munSt
.index}" data-toggle = "modal">See All</
button>
141 </th>
142 </tr>
143 </c:if>
144 </c:forEach>
145 </c:forEach>
146 </tbody>
147 </table>
148
149 </div>
150 </div>
151 </c:forEach>
152 </c:forEach>
153 </c:if>
154 <c:forEach var = "i" begin = "0" end = "16">
155 <c:forEach var = "provs" items = "${prov[i]}"
varStatus = "stat">
156 <c:forEach var = "j" begin = "0" end = "${fn:
length(mun) - 1}">
157 <c:forEach var = "mun" items = "${mun[j]}"
varStatus = "munSt">
158 <c:if test = "${fn:toLowerCase(provs.province)
== fn:toLowerCase(mun.province) && mun.
region == regions[i]}">
159 <div class = "modal fade" id = "usersModal2_${i}
_${stat.index}_${j}_${munSt.index}" data-
width = "700">
160 <div class = "modal-body">
161
162 <table class="table table-striped" id="
example1">
163 <thead>
164 <tr>
165 <th>Name</th>
166 <th>Type of User Applied</th>
167 <th>Profile</th>
168 <th>Approve</th>
169 <th>Reject</th>
170 </tr>
171 </thead>
172 <tbody>
173 <c:forEach var = "reg" items = "${
registrations}" varStatus = "st">
174 <c:if test = "${fn:toLowerCase(reg.province)
== fn:toLowerCase(provs.province) && reg.
region == regions[i]
175 && fn:toLowerCase(reg.muncity) == fn:
toLowerCase(mun.muncity) && (reg.userType
== 'Municipal/City Health Worker' ||
176 reg.userType == 'Municipal/City Health
Official') && reg.status == 0}">
177 <tr>
178 <td>${reg.firstname} ${reg.middlename} ${
reg.surname}</td>
179 <td>${reg.userType}</td>
180 <td><button class = "btn btn-primary" href
= "#profile2_${i}_${stat.index}_${j}_${
munSt.index}_${st.index}" data-toggle = "
modal">View</button></td>
181 <td><button class = "btn btn-primary" href
= "#approve2_${st.index}" data-toggle = "
modal">Approve</button></td>
182 <td><button class = "btn btn-primary" href
= "#reject2_${st.index}" data-toggle = "
modal">Reject</button></td>
183 </tr>
184 <div class = "modal fade" id = "approve2_${
st.index}">
185 <div class = "modal-body">
186 Are you sure you want to approve this
application?
187 </div>
188 <div class = "modal-footer">
189 <a href = "<%= request.getContextPath()
%>/pages/addUser/${reg.accountID}" class =
"btn btn-primary">Yes</a>
<a href = "" class = "btn btn-primary"
data-dismiss="modal">No</a>
</div>
</div>
<div class = "modal fade" id = "reject2_${st
.index}">
<div class = "modal-body">
Are you sure you want to reject this
application?
</div>
<div class = "modal-footer">
<a href = "<%= request.getContextPath()
%>/pages/rejectUser/${reg.accountID}" class
= "btn btn-primary">Yes</a>
<a href = "" class = "btn btn-primary"
data-dismiss="modal">No</a>
</div>
</div>
</tbody>
</table>
</div>
</c:if>
</c:forEach>
</c:forEach>
</c:forEach>
</c:forEach>
</c:forEach>
</c:forEach>
<c:forEach var = "i" begin = "0" end = "16">
<c:forEach var = "provs" items = "${prov[i]}"
varStatus = "stat">
<c:forEach var = "j" begin = "0" end = "${fn:
length(mun) - 1}">
<c:forEach var = "mun" items = "${mun[j]}"
varStatus = "munSt">
<c:if test = "${fn:toLowerCase(provs.province)
== fn:toLowerCase(mun.province) && mun.
region == regions[i]}">
<div class = "modal fade" id = "profile2_${i}_${
stat.index}_${j}_${munSt.index}_${st.index
}" data-width = "700">
<div class = "modal-body">
<div class = "panel panel-info">
<div class = "panel-heading">
<h3>User Profile (${reg.firstname} ${reg.
surname})</h3>
</div>
<div class = "panel-body">
<br />
<br />
<table class = "table table-striped">
<tr>
<td>First Name:</td>
<td>
${reg.firstname}
</td>
</tr>
<tr>
<td>Middle Name:</td>
<td>
${reg.middlename}
</td>
</tr>
<tr>
<td>Surname:</td>
<td>
${reg.surname}
</td>
</tr>
<tr>
<td>Region:</td>
<td>
${reg.region}
</td>
</tr>
<tr>
<td>Province:</td>
<td>
${reg.province}
</td>
</tr>
<tr>
<td>Institution:</td>
<td>
${reg.institution}
</td>
</tr>

```

```

268 <tr>
269 <td>Institution's Address:</td>
270 <td>
271 <reg.institutionAddr>
272 </td>
273 </tr>
274 <tr>
275 <td>Email:</td>
276 <td>
277 <reg.email>
278 </td>
279 </tr>
280 <tr>
281 <td>Contact Number:</td>
282 <td>
283 <reg.number>

```

Listing 55: province.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
  jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
  functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
  springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
  springframework.org/tags/form" %>
5 <c:if test = "${not empty successD}">
6 <div class = "alert alert-success">
7 <strong>${successD}</strong>
8 </div>
9 </c:if>
10 <c:if test = "${pageVisit == 'admin' ||
  pageVisit == 'nonofficial'}">
11 <table class="table table-striped" id="example1
  ">
12 <thead>
13 <tr>
14 <th>Region</th>
15 <th>No. of Provincial Users</th>
16 </tr>
17 </thead>
18 <tbody>
19 <c:forEach var = "i" begin = "0" end = "16">
20 <tr>
21 <th>
22 <button type = "button" class = "btn btn-
  primary" href = "#regionModal14_${i}" data-
  toggle = "modal">${regions[i]}</button>
23 </th>
24 <c:set var = "counter" value = "0" />
25 <c:forEach var = "reg" items = "${
  registrations}">
26 <c:if test = "${reg.region == regions[i] && (
  reg.userType == 'Provincial Health Worker'
  || reg.userType == 'Provincial Health
  Official') && reg.status == 0}">
27 <c:set var = "counter" value = "${counter +
  1}" />
28 </c:if>
29 </c:forEach>
30 <th>
31 <reg.institutionAddr>
32 </th>
33 </tr>
34 </c:forEach>
35 </tbody>
36 </table>
37 <c:forEach var = "i" begin = "0" end = "16">
38 <div class = "modal fade" id = "regionModal14_${
  i}">
39 <div class = "modal-body">
40 <table class="table table-striped" id="
  example1">
41 <thead>
42 <tr>
43 <th>Province</th>
44 <th>No. of Provincial Workers/Officials</th>
45 <th>View All Users</th>
46 </tr>
47 </thead>
48 <tbody>
49 <c:forEach var = "provs" items = "${prov[i
  ]}" varStatus = "stat">
50 <c:set var = "counter" value = "0" />
51 <c:forEach var = "reg" items = "${
  registrations}">
52 <c:if test = "${fn:toLowerCase(reg.province)
  == fn:toLowerCase(provs.province) && reg.
  region == regions[i]
53 && (reg.userType == 'Provincial Health
  Worker' || reg.userType == 'Provincial
  Health Official') && reg.status == 0}">
54 <c:set var = "counter" value = "${counter +
  1}" />
55 </c:if>
56 </c:forEach>
57 </tbody>
58 </table>
59 <th>${provs.province}</th>
60 <th>${counter}</th>
61 <th>
62 <button type = "button" class = "btn btn-
  primary" href = "#usersModal14_${i}_${stat.
  index}" data-toggle = "modal">See All</
  button>
63 </th>
64 </tr>
65 </c:forEach>
66 </tbody>
67 </table>
68 </div>
69 </c:forEach>
70 </c:if>
71 <c:if test = "${pageVisit == 'pofficial'}">
72 <table class="table table-striped" id="example1
  ">
73 <thead>
74 <tr>
75 <th>Province</th>
76 <th>No. of Provincial Workers/Officials</th>
77 <th>View All Users</th>
78 </tr>
79 </thead>
80 <tbody>
81 <c:forEach var = "i" begin = "0" end = "${fn:
  length(prov) - 1}">
82 <c:forEach var = "provs" items = "${prov[i]}"
  varStatus = "stat">
83 <c:if test = "${fn:toLowerCase(user.province)
  == fn:toLowerCase(provs.province)}">
84 <c:set var = "counter" value = "0" />
85 <c:forEach var = "reg" items = "${
  registrations}">
86 <c:if test = "${fn:toLowerCase(reg.province)
  == fn:toLowerCase(provs.province) && reg.
  region == regions[i]
87 && (reg.userType == 'Provincial Health Worker'
  || reg.userType == 'Provincial Health
  Official') && reg.status == 0}">
88 <c:set var = "counter" value = "${counter +
  1}" />
89 </c:if>
90 </c:forEach>
91 </tbody>
92 </table>
93 <tr>
94 <th>${provs.province}</th>
95 <th>${counter}</th>
96 <th>
97 <button type = "button" class = "btn btn-
  primary" href = "#usersModal14_${i}_${stat.
  index}" data-toggle = "modal">See All</
  button>
98 </th>
99 </tr>
100 </c:if>
101 </c:forEach>
102 </tbody>
103 </table>
104 </c:if>
105 </c:forEach>
106 <c:forEach var = "i" begin = "0" end = "16">
107 <c:forEach var = "provs" items = "${prov[i]}"
  varStatus = "stat">
108 <div class = "modal fade" id = "usersModal14_${i}
 }_${stat.index}" data-width = "700">
109 <div class = "modal-body">
110 <table class="table table-striped" id="
  example1">
111 <thead>
112 <tr>
113 <th>Name</th>
114 <th>Type of User Applied</th>
115 <th>Profile</th>
116 <th>Approve</th>
117 <th>Reject</th>
118 </tr>

```

```

119 </thead>
120 <tbody>
121 <c:forEach var = "reg" items = "${
registrations}" varStatus = "st">
122 <c:if test = "${fn:toLowerCase(reg.province)
== fn:toLowerCase(provs.province) && reg.
region == regions[i]
123 && (reg.userType == 'Provincial Health
Worker' || reg.userType == 'Provincial
Health Official') && reg.status == 0}">
124 <tr>
125 <td>${reg.firstname} ${reg.middlename} ${
reg.surname}</td>
126 <td>${reg.userType}</td>
127 <td><button class = "btn btn-primary" href
= "#profile14_${i}_${stat.index}_${st.index
}" data-toggle = "modal">View</button></td>
128 <td><button class = "btn btn-primary" href
= "#approve14_${st.index}" data-toggle = "
modal">Approve</button></td>
129 <td><button class = "btn btn-primary" href
= "#reject14_${st.index}" data-toggle = "
modal">Reject</button></td>
130 </tr>
131 <div class = "modal fade" id = "approve14_${
st.index}">
132 <div class = "modal-body">
133 Are you sure you want to approve this
application?
134 </div>
135 <div class = "modal-footer">
136 <a href = "<%= request.getContextPath()
%>/pages/addUser/${reg.accountID}" class =
"btn btn-primary">Yes</a>
137 <a href = "" class = "btn btn-primary"
data-dismiss="modal">No</a>
138 </div>
139 </div>
140 <div class = "modal fade" id = "reject14_${
st.index}">
141 <div class = "modal-body">
142 Are you sure you want to reject this
application?
143 </div>
144 <div class = "modal-footer">
145 <a href = "<%= request.getContextPath()
%>/pages/rejectUser/${reg.accountID}" class
= "btn btn-primary">Yes</a>
146 <a href = "" class = "btn btn-primary"
data-dismiss="modal">No</a>
147 </div>
148 </div>
149 </c:if>
150 </c:forEach>
151 </tbody>
152 </table>
153 </div>
154 </div>
155 </c:forEach>
156 </c:forEach>
157 <c:forEach var = "i" begin = "0" end = "16">
158 <c:forEach var = "provs" items = "${prov[i]}"
varStatus = "stat">
159 <c:forEach var = "reg" items = "${allUsers}"
varStatus = "st">
160 <div class = "modal fade" id = "profile14_${i}_${
stat.index}_${st.index}">
161 <div class = "modal-body">
162 <c:if test = "${fn:toLowerCase(reg.province)
== fn:toLowerCase(provs.province) && reg.
region == provs.region
163 && (reg.userType == 'Provincial Health Worker'
|| reg.userType == 'Provincial Health
Official') && reg.status == 0}">
164 <div class = "panel panel-info">
165 <div class = "panel-heading">
166 <h4>User Profile (${reg.firstname} ${reg.
middlename} ${reg.surname})</h4>
167 </div>
168 <div class = "panel-body">
169 <br />
170 <br />
171 <table class = "table table-striped">
172 <tr>
173 <td>First Name:</td>
174 <td>
175 ${reg.firstname}
176 </td>
177 </tr>
178 <tr>
179 <td>Middle Name:</td>
180 <td>
181 ${reg.middlename}
182 </td>
183 </tr>
184 <tr>
185 <td>Surname:</td>
186 <td>
187 ${reg.surname}
188 </td>
189 </tr>
190 <tr>
191 <td>Region:</td>
192 <td>
193 ${reg.region}
194 </td>
195 </tr>
196 <tr>
197 <td>Province:</td>
198 <td>
199 ${reg.province}
200 </td>
201 </tr>
202 <tr>
203 <td>Institution:</td>
204 <td>
205 ${reg.institution}
206 </td>
207 </tr>
208 <tr>
209 <td>Institution's Address:</td>
210 <td>
211 ${reg.institutionAddr}
212 </td>
213 </tr>
214 <tr>
215 <td>Email:</td>
216 <td>
217 ${reg.email}
218 </td>
219 </tr>
220 <tr>
221 <td>Contact Number:</td>
222 <td>
223 ${reg.number}
224 </td>
225 </tr>
226 </table>
227 </div>
228 </div>
229 </c:if>
230 </div>
231 </div>
232 </c:forEach>
233 </c:forEach>
234 </c:forEach>

```

Listing 56: national.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
springframework.org/tags/form" %>
5
6 <table class="table table-striped" id="example1
">
7 <thead>
8 <tr>
9 <th>Name</th>
10 <th>User Type</th>
11 <th>Profile</th>
12 <th>Approve</th>
13 <th>Reject</th>
14 </tr>
15 </thead>
16 <tbody>
17 <c:forEach var = "reg" items = "${
registrations}" varStatus = "st">
18 <c:if test = "${reg.userType == 'National
Health Official' && reg.status == 0}">
19 <tr>
20 <td>${reg.firstname} ${reg.middlename} ${reg.
surname}</td>
21 <td>${reg.userType}</td>
22 <td><button class = "btn btn-primary" href =
"#profile5_${st.index}" data-toggle = "
modal">View</button></td>
23 <td><button class = "btn btn-primary" href =
"#approve5_${reg.accountID}" data-toggle =
"modal">Approve</button></td>
24 <td><button class = "btn btn-primary" href =
"#reject5_${reg.accountID}" data-toggle = "
modal">Reject</button></td>
25 </tr>
26 <div class = "modal fade" id = "approve5_${reg
.accountID}">
27 <div class = "modal-body">
28 Are you sure you want to approve this
application?
29 </div>
30 <div class = "modal-footer">

```



```

31     <a href = "<%= request.getContextPath() %>/
      pages/addUser/${reg.accountID}" class = "
      btn btn-primary">Yes</a>
32     <a href = "" class = "btn btn-primary" data-
      dismiss="modal">No</a>
33   </div>
34 </div>
35 <div class = "modal fade" id = "reject5_${reg.
      accountID}">
36   <div class = "modal-body">
37     Are you sure you want to reject this
      application?
38   </div>
39   <div class = "modal-footer">
40     <a href = "<%= request.getContextPath() %>/
      pages/rejectUser/${reg.accountID}" class =
      "btn btn-primary">Yes</a>
41     <a href = "" class = "btn btn-primary" data-
      dismiss="modal">No</a>
42   </div>
43 </div>
44 </c:if>
45 </c:forEach>
46 </tbody>
47 </table>
48 <c:forEach var = "reg" items = "${registrations
      }" varStatus = "st">
49 <c:if test = "${reg.userType == 'National Health
      Official' && reg.status == 0}">
50 <div class = "modal fade" id = "profile5_${st.
      index}" data-width = "700">
51   <div class = "modal-body">
52     <div class = "panel panel-info">
53       <div class = "panel-heading">
54         <h4>User Profile (${reg.firstname} ${reg.
          middlename} ${reg.surname})</h4>
55       </div>
56       <div class = "panel-body">
57         <br />
58         <table class = "table table-striped">
59           <tr>
60             <td>First Name:</td>
61             <td>
62               ${reg.firstname}
63             </td>
64           </tr>
65           <tr>
66             <td>Middle Name:</td>
67             <td>
68               ${reg.middlename}
69             </td>
70           </tr>
71         </table>
72         <td>Surname:</td>
73         <td>
74           ${reg.surname}
75         </td>
76       </tr>
77     <tr>
78       <td>Region:</td>
79       <td>
80         ${reg.region}
81       </td>
82     </tr>
83     <tr>
84       <td>Province:</td>
85       <td>
86         ${reg.province}
87       </td>
88     </tr>
89     <tr>
90       <td>Institution:</td>
91       <td>
92         ${reg.institution}
93       </td>
94     </tr>
95     <tr>
96       <td>Institution's Address:</td>
97       <td>
98         ${reg.institutionAddr}
99       </td>
100    </tr>
101    <tr>
102      <td>Email:</td>
103      <td>
104        ${reg.email}
105      </td>
106    </tr>
107    <tr>
108      <td>Contact Number:</td>
109      <td>
110        ${reg.number}
111      </td>
112    </tr>
113  </table>
114 </div>
115 </div>
116 </div>
117 </div>
118 </c:if>
119 </c:forEach>

```

Listing 57: researcher.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
  jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
  functions" prefix="fn" %>
3 <%@ taglib prefix="sec" uri="http://www.
  springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
  springframework.org/tags/form" %>
5
6 <table class="table table-striped" id="example1
  ">
7   <thead>
8     <tr>
9       <th>Name</th>
10      <th>User Type</th>
11      <th>Profile</th>
12      <th>Approve</th>
13      <th>Reject</th>
14    </tr>
15  </thead>
16  <tbody>
17    <c:forEach var = "reg" items = "${
      registrations}" varStatus = "st">
18    <c:if test = "${reg.userType == 'Researcher'
      && reg.status == 0}">
19    <tr>
20      <td>${reg.firstname} ${reg.middlename} ${reg.
        surname}</td>
21      <td>${reg.userType}</td>
22      <td><button class = "btn btn-primary" href =
        "#profile6_${st.index}" data-toggle = "
        modal">View</button></td>
23      <td><button class = "btn btn-primary" href =
        "#approve6_${reg.accountID}" data-toggle =
        "modal">Approve</button></td>
24      <td><button class = "btn btn-primary" href =
        "#reject6_${reg.accountID}" data-toggle = "
        modal">Reject</button></td>
25    </tr>
26    <div class = "modal fade" id = "approve6_${reg
      .accountID}">
27      <div class = "modal-body">
28        Are you sure you want to approve this
        application?
29      </div>
30    <div class = "modal-footer">
31      <a href = "<%= request.getContextPath() %>/
      pages/addUser/${reg.accountID}" class = "
      btn btn-primary">Yes</a>
32      <a href = "" class = "btn btn-primary" data-
      dismiss="modal">No</a>
33    </div>
34  </div>
35  <div class = "modal fade" id = "reject6_${reg.
      accountID}">
36    <div class = "modal-body">
37      Are you sure you want to reject this
      application?
38    </div>
39    <div class = "modal-footer">
40      <a href = "<%= request.getContextPath() %>/
      pages/rejectUser/${reg.accountID}" class =
      "btn btn-primary">Yes</a>
41      <a href = "" class = "btn btn-primary" data-
      dismiss="modal">No</a>
42    </div>
43  </div>
44 </c:if>
45 </c:forEach>
46 </tbody>
47 </table>
48 <c:forEach var = "reg" items = "${registrations
      }" varStatus = "st">
49 <c:if test = "${reg.userType == 'Researcher' &&
      reg.status == 0}">
50 <div class = "modal fade" id = "profile6_${st.
      index}" data-width = "700">
51   <div class = "modal-body">
52     <div class = "panel panel-info">
53       <div class = "panel-heading">
54         <h4>User Profile (${reg.firstname} ${reg.
          middlename} ${reg.surname})</h4>
55       </div>
56       <div class = "panel-body">
57         <br />
58         <table class = "table table-striped">
59           <tr>
60             <td>First Name:</td>
61             <td>
62               ${reg.firstname}

```

```

63     </td>
64 </tr>
65 <tr>
66 <td>Middle Name:</td>
67 <td>
68     ${reg.middlename}
69 </td>
70 </tr>
71 <tr>
72 <td>Surname:</td>
73 <td>
74     ${reg.surname}
75 </td>
76 </tr>
77 <tr>
78 <td>Region:</td>
79 <td>
80     ${reg.region}
81 </td>
82 </tr>
83 <tr>
84 <td>Province:</td>
85 <td>
86     ${reg.province}
87 </td>
88 </tr>
89 <tr>
90 <td>Institution:</td>
91 <td>

```

```

92     ${reg.institution}
93 </td>
94 </tr>
95 <tr>
96 <td>Institution's Address:</td>
97 <td>
98     ${reg.institutionAddr}
99 </td>
100 </tr>
101 <tr>
102 <td>Email:</td>
103 <td>
104     ${reg.email}
105 </td>
106 </tr>
107 <tr>
108 <td>Contact Number:</td>
109 <td>
110     ${reg.number}
111 </td>
112 </tr>
113 </table>
114 </div>
115 </div>
116 </div>
117 </div>
118 </c:if>
119 </c:forEach>

```

Listing 58: workers.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
  jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
  functions" prefix="fn" %>
3 <%@ taglib prefix="sec" uri="http://www.
  springframework.org/security/tags" %>
4 <!DOCTYPE html>
5 <html>
6 <head>
7 <title>Vector-Borne Diseases Monitoring System
  </title>
8 <meta name="viewport" content="width=device-
  width, initial-scale=1.0">
9 <jsp:include page="../css.jsp"/>
10 <script type="text/javascript">
11 // When the document is ready
12 $(document).ready(function () {
13     $('#dp1').datepicker({
14         format: "yyyy-mm-dd"
15     });
16     $('#dp2').datepicker({
17         format: "yyyy-mm-dd"
18     });
19     $('#dp3').datepicker({
20         format: "yyyy-mm-dd"
21     });
22     $('#dp4').datepicker({
23         format: "yyyy-mm-dd"
24     });
25 });
26 </script>
27 <script type="text/javascript">
28 /* Table initialisation */
29 jQuery(document).ready(function($) {
30
31     <c:forEach var = "i" begin = "0"
32     end = "16">
33     <c:forEach var = "provs" items = "$
34     {prov[i]}" varStatus = "stat">
35         var nameTag = "#example1_" + <c:out
36         value = "${i}" /> + "_" + <c:out value = "
37     ${stat.index}" />;
38         $(nameTag).dataTable( {
39             "bSort": false, // Disable
40             sorting
41             "iDisplayLength": 10, // records per
42             page
43             "sDom": "t<'row'<'col-md-6'i<'col-md
44             -6'p>>",
45             "sPaginationType": "bootstrap"
46         });
47     </c:forEach>
48     </c:forEach>
49
50     </script>
51 </head>
52 <body>
53 <div class = "container">
54 <div class = "navbar navbar-default"
55     role = "navigation">
56 <div class = "container-fluid">
57 <div class = "navbar-header">
58 <button type = "button" class = "
59     navbar-toggle" data-toggle = "collapse"
60     data-target = ".navbar-collapse">
61 <span class = "sr-only">Toggle
62     Navigation</span>
63 <span class = "icon-bar"></span>
64 <span class = "icon-bar"></span>
65 <span class = "icon-bar"></span>
66 </button>
67 <a href = "<%= request.
68     getContextPath() %>/home"><img src = "<%=
69     request.getContextPath() %>/resources/dist/
70     img/logo2.jpg" alt = "VBDMS Logo" height =
71     "30px;" width = "93px;" class = "img-
72     responsive"/></a>
73 <!-- <a href = "<%= request.
74     getContextPath() %>/home" class = "navbar-
75     brand">VBDMS</a> -->
76 </div>
77 <div class = "collapse navbar-
78     collapse">
79 <ul class = "nav navbar-nav">
80 <li <c:if test = "${pages == 'home
81     }"> class = "active"</c:if>
82 <a href = "<%= request.
83     getContextPath() %>/home">Home</a></li>
84
85 <li class = "dropdown">
86 <a href = "" data-toggle = "
87     dropdown" class = "dropdown-toggle">
88     Diseases <b class = "caret"></b>
89 </a>
90 <ul class="dropdown-menu">
91 <c:forEach var="dis" items="${
92     disease}">
93 <li><a href = "<%= request.
94     getContextPath() %>/disease/${dis.
95     diseaseName}">${dis.diseaseName}</a></li>
96 </c:forEach>
97 </ul>
98 </li>
99 <li class = "dropdown">
100 <a href = "" data-toggle = "
101     dropdown" class = "dropdown-toggle">
102     Disease Statistics <b class = "
103     caret"></b>
104 </a>
105 <ul class="dropdown-menu">
106 <c:forEach var="dis" items="${
107     disease}">
108 <li><a href = "<%= request.
109     getContextPath() %>/statistics/${dis.
110     diseaseName}">${dis.diseaseName}</a></li>
111 </c:forEach>
112 </ul>
113 </li>
114 <li><a href = "<%= request.
115     getContextPath() %>/campaigns">Awareness
116     Campaigns</a></li>
117 </ul>
118 <ul class = "nav navbar-nav navbar-
119     right">
120 <sec:authorize access="isAnonymous
121     ()">
122 <c:if test = "${pages != 'register
123     }">
124 <li><a data-toggle = "modal" href = "
125     #loginForm">Login</a></li>
126 <li><a href = "<%= request.

```

```

getContextPath() %>/register">Register</a
></li>
93 </c:if>
94 </sec:authorize>
95 <sec:authorize access="
isAuthenticated()">
96 <div class = "btn-group">
97 <div class = "btn-group">
98 <sec:authorize access="hasAnyRole('
ROLE_BHWorker')">
99 <a href = "<%= request.
getContextPath() %>/pages/profile">Barangay
Health Worker</a>
100 </sec:authorize>
101 <sec:authorize access="hasAnyRole('
ROLE_MHWorker')">
102 <a href = "<%= request.getContextPath
() %>/pages/profile">Municipal/City Health
Worker</a>
103 </sec:authorize>
104 <sec:authorize access="hasAnyRole('
ROLE_MCOfficial')">
105 <a href = "<%= request.getContextPath
() %>/pages/profile">Municipal/City Health
Official</a>
106 </sec:authorize>
107 <sec:authorize access="hasAnyRole('
ROLE_PHWorker')">
108 <a href = "<%= request.getContextPath()
%>/pages/profile">Provincial Health Worker
</a>
109 </sec:authorize>
110 <sec:authorize access="hasAnyRole('
ROLE_PHOfficial')">
111 <a href = "<%= request.getContextPath()
%>/pages/profile">Provincial Health
Official</a>
112 </sec:authorize>
113 <sec:authorize access="hasAnyRole('
ROLE_NHOfficial')">
114 <a href = "<%= request.getContextPath() %>/
pages/profile">National Health Official</a>
115 </sec:authorize>
116 <sec:authorize access="hasAnyRole('ROLE_USER')">
117 <a href = "<%= request.getContextPath() %>/pages
/profile">Administrator</a>
118 </sec:authorize>
119 <sec:authorize access="hasAnyRole('
ROLE_Researcher')">
120 <a href = "<%= request.getContextPath() %>/pages
/profile">Researcher</a>
121 </sec:authorize>
122 </div>
123 </div>
124 <div class = "btn-group">
125 <button href="#" data-toggle="dropdown" class="
btn dropdown-toggle" >
126 <span class = "glyphicon glyphicon-user"></
span> <span class = "caret"></span>
127 </button>
128 <ul class="dropdown-menu">
129 <li><a href = "<%= request.getContextPath()
%>/pages/profile">Profile</a></li>
130 <sec:authorize access="hasAnyRole('
ROLE_MCOfficial', 'ROLE_PHOfficial', '
ROLE_NHOfficial', 'ROLE_USER')">
131 <li><a href = "<%= request.
getContextPath() %>/pages/notifications">
Notifications</a></li>
132 </sec:authorize>
133 <sec:authorize access="hasAnyRole('
ROLE_BHWorker', 'ROLE_PHWorker', '
ROLE_MHWorker', 'ROLE_USER')">
134 <li><a href = "<%= request.
getContextPath() %>/pages/system">Manage
Records</a></li>
135 </sec:authorize>
136 <sec:authorize access="hasAnyRole('
ROLE_MCOfficial', 'ROLE_PHOfficial', '
ROLE_NHOfficial', 'ROLE_USER')">
137 <li><a href = "<%= request.getContextPath
() %>/pages/workers" data-toggle = "tab">
Workers</a></li>
138 </sec:authorize>
139 <li><a href = "<%= request.getContextPath
() %>/pages/reports">Reports</a></li>
140 <li><a href = "<%= request.getContextPath
() %>/j_spring_security_logout">Log Out</a
></li>
141 </ul>
142 </div>
143 </sec:authorize>
144 </ul>
145 </div>
146 </div>
147 </div>
148 <div id = "loginForm" class = "modal fade"
tabindex = "-1">
149 <div class="modal-header">
150 <button type="button" class="close" data-
dismiss="modal" aria-hidden="true"> </
button>
151 <div class = "panel-header">
152 <h4 class = "form-signin-head">Vector Borne
Diseases Monitoring System</h4>
153 </div>
154 </div>
155 <div class = "modal-body">
156 <form method="post" class = "form-signin"
action = "j_spring_security_check">
157 <table class = "table" >
158 <tr>
159 <td style = "border: none;">
160 <label for="login_username">Username</
label>
161 </td>
162 <td style = "border: none;">
163 <input type="text" name="j_username" id="
login_username" class="form-control field
required" title="Please provide your
username" />
164 </td>
165 </tr>
166 <tr>
167 <td style = "border: none;">
168 <label for="login_password">Password</
label>
169 </td>
170 <td style = "border: none;">
171 <input type="password" name="j_password"
id="login_password" class="form-control
field required" title="Password is required
" />
172 </td>
173 </tr>
174 <tr>
175 <td style = "border: none;"><button class =
"btn btn-primary" type="submit" value = "
submit">Log in</button></td>
176 </tr>
177 </table>
178 </form>
179 </div>
180 </div>
181 <div class = "tabbable">
182 <jsp:include page = "linkings.jsp" />
183 <div class = "tab-content">
184 <div class = "tab-pane active">
185 <div class = "tabbable">
186 <ul class = "nav nav-pills">
187 <sec:authorize access="hasAnyRole('
ROLE_PHOfficial', 'ROLE_MCOfficial')">
188 <li <c:if test = "${pageVisit == 'mofficial'
|| pageVisit == 'pofficial'}">class = "
active"</c:if>>
189 <a href = "#taba" data-toggle = "tab">
Barangay<span class="badge pull-right"></
span></a>
190 </li>
191 </sec:authorize>
192 <sec:authorize access="hasAnyRole('
ROLE_MCOfficial', 'ROLE_PHOfficial', '
ROLE_NHOfficial', 'ROLE_USER')">
193 <li <c:if test = "${pageVisit == 'admin' ||
pageVisit == 'nofficial'}">class = "active
"</c:if>>
194 <a href = "#tabb" data-toggle = "tab">
Municipality/City<span class="badge pull-
right"></span></a>
195 </li>
196 </sec:authorize>
197 <sec:authorize access="hasAnyRole('
ROLE_PHOfficial', 'ROLE_NHOfficial', '
ROLE_USER')">
198 <li>
199 <a href = "#tabc" data-toggle = "tab">
Provincial<span class="badge pull-right"></
span></a>
200 </li>
201 </sec:authorize>
202 <sec:authorize access="hasAnyRole('ROLE_USER')
">
203 <li>
204 <a href = "#tabd" data-toggle = "tab">
National<span class="badge pull-right"></
span></a>
205 </li>
206 </sec:authorize>
207 <sec:authorize access="hasAnyRole('
ROLE_NHOfficial', 'ROLE_USER')">
208 <li>
209 <a href = "#tabe" data-toggle = "tab">
Researchers<span class="badge pull-right
"></span></a>
210 </li>
211 </sec:authorize>
212 </ul>

```

```

213 </div>
214 <br />
215 <div class = "tab-content" style = "height: 800
    px;">
216 <div class = "tab-pane <c:if test = "${
    pageVisit == 'mofficial' || pageVisit == '
    pofficial'}">active</c:if" id = "taba">
217 <c:if test = "${pageVisit == 'mofficial'}">
218 <jsp:include page = "allworkers/bar.jsp" />
219 </c:if>
220 <c:if test = "${pageVisit == 'pofficial'}">
221 <jsp:include page = "allworkers/barangay.jsp"
    />
222 </c:if>
223 </div>
224 <div class = "tab-pane <c:if test = "${pageVisit
    == 'admin' || pageVisit == 'nofficial'}">
    active</c:if" id = "tabb">
225 <c:if test = "${pageVisit != 'mofficial'}">
226 <jsp:include page = "allworkers/muncity.jsp" />
227 </c:if>
228 <c:if test = "${pageVisit == 'mofficial'}">
229 <jsp:include page = "allworkers/mun.jsp" />
230 </c:if>
231 </div>
232 <div class = "tab-pane" id = "tabc">
233 <c:if test = "${pageVisit != 'pofficial'}">
234 <jsp:include page = "allworkers/province.jsp"
    />
235 </c:if>
236 <c:if test = "${pageVisit == 'pofficial'}">
237 <jsp:include page = "allworkers/prov.jsp" />
238 </c:if>
239 </div>
240 <div class = "tab-pane" id = "tabd">
241 <jsp:include page = "allworkers/national.jsp"
    />
242 </div>
243 <div class = "tab-pane" id = "tabe">
244 <jsp:include page = "allworkers/researcher.jsp"
    />
245 </div>
246 </div>
247 </div>
248 </div>
249 </div>
250 </div>
251 <jsp:include page = "../footer.jsp" />

```

Listing 59: barangay.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
    jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
    functions" prefix="fn" %>
3 <%@ taglib prefix="sec" uri="http://www.
    springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
    springframework.org/tags/form" %>
5
6 <div class = "container-fluid well-lg" style = "
    background-color: #eee;">
7 <div class = "panel panel-info">
8 <c:if test = "${not empty successD}">
9 <div class = "alert alert-success">
10 <strong>${successD}</strong>
11 </div>
12 </c:if>
13 <c:forEach var = "i" begin = "0" end = "16">
14 <c:forEach var = "provs" items = "${prov[i]}"
    varStatus = "stat">
15 <c:if test = "${fn:toLowerCase(user.province)
    == fn:toLowerCase(provs.province)}">
16 <table class="table table-striped" id = "
    example1">
17 <thead>
18 <tr>
19 <th>Municipality/City</th>
20 <th>No. of Barangay Health Workers</th>
21 </tr>
22 </thead>
23 <tbody>
24
25 <c:forEach var = "j" begin = "0" end = "${fn:
    length(mun) - 1}">
26 <c:forEach var = "mun" items = "${mun[j]}"
    varStatus = "munSt">
27 <c:if test = "${fn:toLowerCase(user.province)
    == fn:toLowerCase(mun.province) && mun.
    region == regions[i]}">
28 <c:set var = "counter" value = "0" />
29 <c:forEach var = "user" items = "${allUsers}"
    varStatus = "st">
30 <c:if test = "${fn:toLowerCase(user.province)
    == fn:toLowerCase(provs.province) && user.
    region == provs.region
31 && fn:toLowerCase(user.muncity) == fn:
    toLowerCase(mun.muncity) && user.userType
    == 'Barangay Health Worker'}">
32 <c:set var = "counter" value = "${counter +
    1}" />
33 </c:if>
34 </c:forEach>
35 <tr>
36 <th>
37 <button type = "button" class = "btn btn-
    primary" href = "#muncityModal1_${i}_${stat
    .index}_${j}_${munSt.index}" data-toggle =
    "modal">${mun.muncity}</button>
38 </th>
39 <th>${counter}</th>
40 </tr>
41
42 </c:if>
43 </c:forEach>
44 </c:forEach>
45 </tbody>
46 </table>
47 </c:if>
48 </c:forEach>
49 </c:forEach>
50 <c:forEach var = "i" begin = "0" end = "16">
51 <c:forEach var = "provs" items = "${prov[i]}"
    varStatus = "stat">
52 <c:forEach var = "j" begin = "0" end = "${fn:
    length(mun) - 1}">
53 <c:forEach var = "mun" items = "${mun[j]}"
    varStatus = "munSt">
54 <c:if test = "${fn:toLowerCase(user.province)
    == fn:toLowerCase(mun.province) && mun.region
    == regions[i]}">
55 <div class = "modal fade" id = "muncityModal1_${
    i}_${stat.index}_${j}_${munSt.index}" data-
    width = "700">
56 <div class = "modal-body">
57 <table class="table table-striped" id = "
    example1">
58 <thead>
59 <tr>
60 <th>Barangay</th>
61 <th>No. of Barangay Health Workers</th>
62 <th>View All Users</th>
63 </tr>
64 </thead>
65 <tbody>
66
67 <c:forEach var = "k" begin = "0" end = "${fn
    :length(brgy) - 1}">
68 <c:forEach var = "brgy" items = "${brgy[k]}"
    varStatus = "brgSt">
69 <c:if test = "${fn:toLowerCase(user.muncity)
    == fn:toLowerCase(brgy.muncity) && fn:
    toLowerCase(user.province) == fn:
    toLowerCase(mun.province) && mun.region ==
    regions[i]}">
70 <c:set var = "counter" value = "0" />
71 <c:forEach var = "user" items = "${allUsers
    }" varStatus = "st">
72 <c:if test = "${fn:toLowerCase(user.province)
    == fn:toLowerCase(provs.province) && user
    .region == provs.region
73 && fn:toLowerCase(user.muncity) == fn:
    toLowerCase(mun.muncity) && user.barangay
    == brgy.barangay
74 && user.userType == 'Barangay Health Worker
    '}">
75 <c:set var = "counter" value = "${counter +
    1}" />
76 </c:if>
77 </c:forEach>
78 <tr>
79 <th>${brgy.barangay}</th>
80 <th>${counter}</th>
81 <th>
82 <button type = "button" class = "btn btn-
    primary" href = "#usersModal1_${i}_${stat.
    index}_${j}_${munSt.index}_${k}_${brgSt.
    index}" data-toggle = "modal">See All</
    button>
83 </th>
84 </tr>
85 </c:if>
86 </c:forEach>
87 </c:forEach>
88 </tbody>
89 </table>
90 </div>
91 </div>
92 </c:if>
93 </c:forEach>
94 </c:forEach>
95 </c:forEach>
96 </c:forEach>
97 <c:forEach var = "i" begin = "0" end = "16">
98 <c:forEach var = "provs" items = "${prov[i]}"

```

```

    varStatus = "stat">
99 <c:forEach var = "j" begin = "0" end = "${fn:
    length(mun) - 1}">
100 <c:forEach var = "mun" items = "${mun[j]}"
    varStatus = "munSt">
101 <c:if test = "${fn:toLowerCase(user.province) ==
    fn:toLowerCase(mun.province) && mun.region
    == regions[i]}">
102 <c:forEach var = "k" begin = "0" end = "${fn:
    length(brgy) - 1}">
103 <c:forEach var = "brgy" items = "${brgy[k]}"
    varStatus = "brgSt">
104 <c:if test = "${fn:toLowerCase(user.muncity) ==
    fn:toLowerCase(brgy.muncity) && fn:
    toLowerCase(user.province) == fn:
    toLowerCase(mun.province) && mun.region ==
    regions[i]}">
105 <div class = "modal fade" id = "usersModal1_${i}
    _${stat.index}_${j}_${munSt.index}_${k}_${
    brgSt.index}" data-width = "700">
106 <div class = "modal-body">
107 <table class="table table-striped" id = "
    example1">
108 <thead>
109 <tr>
110 <th>Name</th>
111 <th>User Type</th>
112 <th>Profile</th>
113 <th>User Logs</th>
114 <th>Account Status</th>
115 <th>Activate or Deactivate</th>
116 </tr>
117 </thead>
118 <tbody>
119 <c:forEach var = "user" items = "${allUsers
    }" varStatus = "st">
120 <c:if test = "${fn:toLowerCase(user.province
    ) == fn:toLowerCase(provs.province) && user
    .region == provs.region
121 && fn:toLowerCase(user.muncity) == fn:
    toLowerCase(mun.muncity) && user.barangay
    == brgy.barangay
122 && user.userType == 'Barangay Health Worker
    '}">
123 <tr>
124 <td>${user.firstname} ${user.middlename} ${
    user.surname}</td>
125 <td>${user.userType}</td>
126 <td><button class = "btn btn-primary" href
    = "#profile1_${i}_${stat.index}_${j}_${
    munSt.index}_${k}_${brgSt.index}_${st.index
    }" data-toggle = "modal">View</button></td>
127 <td><button class = "btn btn-primary" href
    = "#logs1_${i}_${stat.index}_${j}_${munSt.
    index}_${k}_${brgSt.index}_${st.index}"
    data-toggle = "modal">View</button></td>
128 <td><c:if test = "${user.status == 1}">
    Active</c:if><c:if test = "${user.status ==
    0}">Inactive</c:if></td>
129 <td><c:if test = "${user.status == 0}"><
    button class = "btn btn-primary" data-
    toggle = "modal" href = "#activ_${user.
    accID}">Activate</button></c:if><c:if test
    = "${user.status == 1}"><button class = "
    btn btn-primary" data-toggle = "modal" href
    = "#deact_${user.accID}">Deactivate</
    button></c:if></td>
130 </tr>
131 <div class = "modal fade" id = "deact_${user
    .accID}">
132 <div class = "modal-body">
133 Are you sure you want to deactivate this
    user?
134 </div>
135 <div class = "modal-footer">
136 <a href = "<%= request.getContextPath()
    %>/pages/deactivateUser/${user.accID}"><
    button class = "btn btn-primary">Yes</
    button></a>
137 <button class = "btn btn-primary" data-
    dismiss = "modal">No</button>
138 </div>
139 </div>
140 <div class = "modal fade" id = "activ_${user
    .accID}">
141 <div class = "modal-body">
142 Are you sure you want to activate this
    user?
143 </div>
144 <div class = "modal-footer">
145 <a href = "<%= request.getContextPath()
    %>/pages/activateUser/${user.accID}"><
    button class = "btn btn-primary">Yes</
    button></a>
146 <button class = "btn btn-primary" data-
    dismiss = "modal">No</button>
147 </div>
148 </div>
149 </c:if>
150 </c:forEach>
151 </tbody>
152 </table>
153 </div>
154 </div>
155 </c:if>
156 </c:forEach>
157 </c:forEach>
158 </c:if>
159 </c:forEach>
160 </c:forEach>
161 </c:forEach>
162 </c:forEach>
163 <c:forEach var = "i" begin = "0" end = "16">
164 <c:forEach var = "provs" items = "${prov[i]}"
    varStatus = "stat">
165 <c:forEach var = "j" begin = "0" end = "${fn:
    length(mun) - 1}">
166 <c:forEach var = "mun" items = "${mun[j]}"
    varStatus = "munSt">
167 <c:if test = "${fn:toLowerCase(user.province) ==
    fn:toLowerCase(mun.province) && mun.region
    == regions[i]}">
168 <c:forEach var = "k" begin = "0" end = "${fn:
    length(brgy) - 1}">
169 <c:forEach var = "brgy" items = "${brgy[k]}"
    varStatus = "brgSt">
170 <c:if test = "${fn:toLowerCase(user.muncity) ==
    fn:toLowerCase(brgy.muncity) && fn:
    toLowerCase(user.province) == fn:
    toLowerCase(mun.province) && mun.region ==
    regions[i]}">
171 <c:forEach var = "user" items = "${allUsers}"
    varStatus = "st">
172 <c:if test = "${fn:toLowerCase(user.province) ==
    fn:toLowerCase(provs.province) && user.
    region == provs.region
173 && fn:toLowerCase(user.muncity) == fn:
    toLowerCase(mun.muncity) && user.barangay
    == brgy.barangay
174 && user.userType == 'Barangay Health Worker'}">
175 <div class = "modal fade" id = "profile1_${i}
    _${stat.index}_${j}_${munSt.index}_${k}_${
    brgSt.index}_${st.index}" data-width =
    "700">
176 <div class = "modal-body">
177 <div class = "panel panel-info">
178 <div class = "panel-body">
179 <br />
180 <table class = "table table-striped">
181 <tr>
182 <td>First Name:</td>
183 <td>
184 ${user.firstname}
185 </td>
186 </tr>
187 <tr>
188 <td>Middle Name:</td>
189 <td>
190 ${user.middlename}
191 </td>
192 </tr>
193 <tr>
194 <td>Surname:</td>
195 <td>
196 ${user.surname}
197 </td>
198 </tr>
199 <tr>
200 <td>Region:</td>
201 <td>
202 ${user.region}
203 </td>
204 </tr>
205 <tr>
206 <td>Province:</td>
207 <td>
208 ${user.province}
209 </td>
210 </tr>
211 <tr>
212 <td>Institution:</td>
213 <td>
214 ${user.institution}
215 </td>
216 </tr>
217 <tr>
218 <td>Institution's Address:</td>
219 <td>
220 ${user.institutionAddr}
221 </td>
222 </tr>
223 <tr>
224 <td>Email:</td>
225 <td>
226 ${user.email}
227 </td>
228 </tr>
229 <tr>

```

```

230     <td>Contact Number:</td>
231     <td>
232     {user.number}
233     </td>
234     </tr>
235     </table>
236     </div>
237     </div>
238     </div>
239     </div>
240     </c:if>
241     </c:forEach>
242     </c:if>
243     </c:forEach>
244     </c:forEach>
245     </c:if>
246     </c:forEach>
247     </c:forEach>
248     </c:forEach>
249     </c:forEach>
250     <c:forEach var = "i" begin = "0" end = "16">
251     <c:forEach var = "provs" items = "${prov[i]}"
252     varStatus = "stat">
253     <c:forEach var = "j" begin = "0" end = "${fn:
254     length(mun) - 1}">
255     <c:forEach var = "mun" items = "${mun[j]}"
256     varStatus = "munSt">
257     <c:if test = "${fn:toLowerCase(user.province) ==
258     fn:toLowerCase(mun.province) && mun.region
259     == regions[i]}">
260     <c:forEach var = "k" begin = "0" end = "${fn:
261     length(brgy) - 1}">
262     <c:forEach var = "brgy" items = "${brgy[k]}"
263     varStatus = "brgSt">
264     <c:if test = "${fn:toLowerCase(user.muncity) ==
265     fn:toLowerCase(brgy.muncity) && fn:
266     toLowerCase(user.province) == fn:
267     toLowerCase(mun.province) && mun.region ==
268     regions[i]}">
269     <c:forEach var = "user" items = "${allUsers}"
270     varStatus = "st">
271     <c:if test = "${fn:toLowerCase(user.province) ==
272     fn:toLowerCase(provs.province) && user.
273     region == provs.region
274     && fn:toLowerCase(user.muncity) == fn:
275     toLowerCase(mun.muncity) && user.barangay
276     == brgy.barangay
277     && user.userType == 'Barangay Health Worker'}">
278     <div class = "modal fade" id = "logs1_{i}_{
279     stat.index}_{j}_{munSt.index}_{k}_{
280     brgSt.index}_{st.index}" data-width =
281     "700">
282     <div class = "modal-body">
283     <div class = "panel panel-info">
284     <div class = "panel-body">
285     <br />
286     <br />
287     <table class = "table table-striped">
288     <thead>
289     <tr>
290     <th>Date</th>
291     <th>Time</th>
292     <th>Action</th>
293     <th>View Record</th>
294     </tr>
295     </thead>
296     <tbody>
297     <c:forEach var = "logs" items = "${allLogs
298     }" varStatus = "lgStat">
299     <c:if test = "${logs.recordtable == '
300     barangayrecords' && logs.accountID == user.
301     accID}">
302     <tr>
303     <td>${logs.date}</td>
304     <td>${logs.time}</td>
305     <td>${logs.action}</td>
306     <c:forEach var = "1" begin = "0" end = "${
307     fn:length(brgR) - 1}">
308     <c:forEach var = "brgR" items = "${brgR[1
309     ]}" varStatus = "brgR">
310     <c:if test = "${logs.accountID == user.
311     accID && logs.recordID == brgR.recordID}">
312     <td><button class = "btn btn-primary"
313     href = "#logsRecord1_{i}_{stat.index}_{j
314     }_{munSt.index}_{k}_{brgSt.index}_{st.
315     index}_{1}_{brgR.index}" data-toggle = "
316     modal">View</button></td>
317     </c:if>
318     </c:forEach>
319     </tr>
320     </tbody>
321     </table>
322     </div>
323     </div>
324     </div>
325     </div>
326     </c:if>
327     </c:forEach>
328     </tbody>
329     </table>
330     </div>
331     </div>
332     </div>
333     </div>
334     </div>
335     </div>
336     </div>
337     </div>
338     </div>
339     </div>
340     </div>
341     </div>
342     </div>
343     </div>
344     </div>
345     </div>
346     </div>
347     </div>
348     </div>
349     </div>
350     </div>
351     </div>
352     </div>
353     </div>
354     </div>
355     </div>
356     </div>
357     </div>
358     </div>
359     </div>
360     </div>
361     </div>
362     </div>
363     </div>
364     </div>
365     </div>
366     </div>
367     </div>
368     </div>
369     </div>

```

```

370 <c:forEach var = "i" begin = "0" end = "16"> 435
371 <c:forEach var = "provs" items = "${prov[i]}" 436
    varStatus = "stat"> 437
372 <c:forEach var = "j" begin = "0" end = "${fn: 438
    length(mun) - 1}"> 439
373 <c:forEach var = "mun" items = "${mun[j]}" 440
    varStatus = "munSt"> 441
374 <c:if test = "${fn:toLowerCase(user.province) == 442
    fn:toLowerCase(mun.province) && mun.region 443
    == regions[i]}"> 444
375 <c:forEach var = "k" begin = "0" end = "${fn: 445
    length(brgy) - 1}"> 446
376 <c:forEach var = "brgy" items = "${brgy[k]}" 447
    varStatus = "brgSt"> 448
377 <c:if test = "${fn:toLowerCase(user.muncity) == 449
    fn:toLowerCase(brgy.muncity) && fn: 450
    toLowerCase(user.province) == fn: 451
    toLowerCase(mun.province) && mun.region == 452
    regions[i]}"> 453
378 <c:forEach var = "user" items = "${allUsers}" 454
    varStatus = "st"> 455
379 <c:if test = "${fn:toLowerCase(user.province) == 456
    fn:toLowerCase(provs.province) && user. 457
    region == provs.region 458
    && fn:toLowerCase(user.muncity) == fn: 459
    toLowerCase(mun.muncity) && user.barangay 460
    == brgy.barangay 461
    && user.userType == 'Barangay Health Worker'}"> 462
381 <c:forEach var = "logs" items = "${allLogs}" 463
    varStatus = "lgStat"> 464
382 <c:if test = "${logs.recordtable == ' 465
    barangayrecords'}"> 466
383 <c:forEach var = "l" begin = "0" end = "${fn: 467
    length(brgR) - 1}"> 468
384 <c:forEach var = "brgR" items = "${brgR[l]}" 469
    varStatus = "bgR"> 470
385 <c:if test = "${logs.accountID == user.accID && 471
    logs.recordID == brgR.recordID}"> 472
386 <div class = "modal fade" id = "logsRecord1_${i} 473
    _${stat.index}_${j}_${munSt.index}_${k}_${ 474
    brgSt.index}_${st.index}_${l}_${bgR.index}" 475
    data-width = "780"> 476
387 <div class = "modal-body"> 477
388 <div class = "panel panel-info"> 478
389 <div class = "panel-body"> 479
390 <div class = "panel panel-info"> 480
391 <div class = "panel-heading"> 481
392 Record Information 482
393 </div> 483
394 <div class = "panel-body"> 484
395 <div class = "row"> 485
396 <div class = "col-lg-2"> 486
397 <label>Region:</label> 487
398 </div> 488
399 <div class = "col-lg-4"> 489
400 <input type = "text" class = "form- 490
    control" readonly = "true" value = "${brgR. 491
    region}" /> 492
401 </div> 493
402 <div class = "col-lg-2"> 494
403 <label>Province:</label> 495
404 </div> 496
405 <div class = "col-lg-4"> 497
406 <input type = "text" class = "form- 498
    control" readonly = "true" value = "${brgR. 499
    province}" /> 500
407 </div> 501
408 </div> 502
409 <br /> 503
410 <div class = "row"> 504
411 <div class = "col-lg-2"> 505
412 <label>Municipality/City:</label> 506
413 </div> 507
414 <div class = "col-lg-4"> 508
415 <input type = "text" class = "form- 509
    control" readonly = "true" value = "${brgR. 510
    muncity}" /> 511
416 </div> 512
417 <div class = "col-lg-2"> 513
418 <label>Barangay</label> 514
419 </div> 515
420 <div class = "col-lg-4">
421 <input type = "text" class = "form-
422 control" readonly = "true" value = "${brgR.
    barangay}" />
423 </div>
424 </div>
425 <br />
426 <div class = "row">
427 <div class = "col-lg-2">
428 <label>Barangay Health Worker:</label>
429 </div>
430 <div class = "col-lg-4">
431 <input type = "text" class = "form-
432 control" readonly = "true" value = "${brgR.
    healthworker}" />
433 </div>
434 <div class = "col-lg-2">
    <label>Date Submitted:</label>
435 </div>
436 <div class = "col-lg-4">
    <input type = "text" class = "form-
437 control" readonly = "true" value = "${brgR.
    dateSubmitted}" />
438 </div>
439 </div>
440 <c:if test = "${brgR.status == 1}">
441 <br />
442 <div class = "row">
443 <div class = "col-lg-2">
    <label>Municipality/City Health Worker
444 :</label>
445 </div>
446 <div class = "col-lg-4">
    <input type = "text" class = "form-
447 control" readonly = "true" value = "${brgR.
    muncityHW}" />
448 </div>
449 <div class = "col-lg-2">
    <label>Date Approved:</label>
450 </div>
451 <div class = "col-lg-4">
    <input type = "text" class = "form-
452 control" readonly = "true" value = "${brgR.
    dateApproved}" />
453 </div>
454 </div>
455 </c:if>
456 <c:if test = "${brgR.status == 2}">
457 <br />
458 <div class = "row">
459 <div class = "col-lg-2">
    <label>Municipality/City Health Worker:</
460 label>
461 </div>
462 <div class = "col-lg-4">
    <input type = "text" class = "form-
463 control" readonly = "true" value = "${brgR.
    muncityHW}" />
464 </div>
465 <div class = "col-lg-2">
    <label>Date Rejected:</label>
466 </div>
467 <div class = "col-lg-4">
    <input type = "text" class = "form-
468 control" readonly = "true" value = "${brgR.
    dateApproved}" />
469 </div>
470 </div>
471 </c:if>
472 <c:if test = "${brgR.editStatus == 1}">
473 <br />
474 <div class = "row">
475 <div class = "col-lg-2">
    <label>Edited by:</label>
476 </div>
477 <div class = "col-lg-4">
    <input type = "text" class = "form-control
478 " readonly = "true" value = "${brgR.
    editedBy}" />
479 </div>
480 <div class = "col-lg-2">
    <label>Date Edited:</label>
481 </div>
482 <div class = "col-lg-3">
    <input type = "text" class = "form-control
483 " readonly = "true" value = "${brgR.
    dateEdited}" />
484 </div>
485 <div class = "col-lg-1">
    <button class = "btn btn-primary">View
486 Original</button>
487 </div>
488 </div>
489 </c:if>
490 </div>
491 <div class = "panel panel-info">
492 <div class = "panel-heading">
493 <label>Patient's Information</label>
494 </div>
495 <div class = "panel-body">
496 <c:if test = "${not empty brgR.name}">
497 <div class = "row">
498 <div class = "col-lg-1">
    <label>Name:</label>
499 </div>
500 <div class = "col-lg-5">
    <input type = "text" class = "form-control"
501 readonly = "true" value = "${brgR.name}" />
502 </div>
503 </div>
504 </c:if>
505 <c:if test = "${not empty br1.address}">
506 <div class = "row">
507 <div class = "col-lg-1">
    <label>Complete Address:</label>
508 </div>
509 </div>
510 <br />
511 </c:if>
512 <c:if test = "${not empty br1.address}">
513 <div class = "row">
514 <div class = "col-lg-1">
    <label>Complete Address:</label>
515 </div>

```

```

516     </div>
517     <div class = "col-lg-5">
518         <textarea class = "form-control" readonly =
          "true" value = "${brgR.address}">${brgR.
            address}</textarea>
519     </div>
520 </div>
521 <br />
522 </c:if>
523 <div class = "row">
524     <div class = "col-lg-1">
525         <label>Age:</label>
526     </div>
527     <div class = "col-lg-2">
528         <input type = "text" class = "form-control"
          readonly = "true" value = "${brgR.age}" />
529     </div>
530     <div class = "col-lg-1">
531         <label>Date of Birth:</label>
532     </div>
533     <div class = "col-lg-3">
534         <input class="form-control" type="text"
          readonly = "true" value="${brgR.dateofbirth
            }" />
535     </div>
536     <div class = "col-lg-1">
537         <label>Gender:</label>
538     </div>
539     <div class = "col-lg-3">
540         <input class = "form-control" type = "text"
          readonly = "true" value = "${brgR.gender
            }" />
541     </div>
542 </div>
543 </div>
544 </div>
545 <div class = "panel panel-info">
546     <div class = "panel-heading">
547         <label>Patient's Clinical Information</label>
548     </div>
549     <div class = "panel-body">
550         <div class = "row">
551             <div class = "col-lg-2">
552                 <label>Disease:</label>
553             </div>
554             <div class = "col-lg-4">
555                 <input type = "text" class = "form-control"
                  readonly = "true" value = "${brgR.disease
                    }" />
556             </div>
557             <div class = "col-lg-2">
558                 <label>Date Onset:</label>
559             </div>
560             <div class = "col-lg-4">
561                 <input class="form-control" type="text"
                  readonly = "true" value="${brgR.dateOnset}"
                    />
562             </div>
563 </div>
564 <br />
565 <div class = "row">
566     <div class = "col-lg-2">
567         <label>Symptoms:</label>
568     </div>
569     <div class = "col-lg-4">
570         <input type = "text" class = "form-control"
          readonly = "true" value = "${brgR.symptoms
            }" />
571     </div>
572 </div class = "col-lg-2">
573     <label>Case Classification:</label>
574 </div>
575     <div class = "col-lg-4">
576         <input class="form-control" type="text"
          readonly = "true" value="${brgR.caseType}"
            />
577     </div>
578 </div>
579 <br />
580 <c:if test = "${brgR.admitted == 'No'}">
581 <div class = "row">
582     <div class = "col-lg-2">
583         <label>Patient Admitted?:</label>
584     </div>
585     <div class = "col-lg-4">
586         <input type = "text" class = "form-control"
          readonly = "true" value = "${brgR.admitted
            }" />
587     </div>
588 </div>
589 </c:if>
590 <c:if test = "${brgR.admitted == 'Yes'}">
591 <div class = "row">
592     <div class = "col-lg-2">
593         <label>Patient Admitted?:</label>
594     </div>
595     <div class = "col-lg-4">
596         <input type = "text" class = "form-control"
          readonly = "true" value = "${brgR.admitted
            }" />
597     </div>
598 </div class = "col-lg-2">
599     <label>Date Admitted:</label>
600 </div>
601     <div class = "col-lg-4">
602         <input class="form-control" type="text"
          readonly = "true" value="${brgR.
            dateAdmitted}" />
603     </div>
604 </div>
605 <br />
606 <div class = "row">
607     <div class = "col-lg-2">
608         <label>Hospital/Clinic:</label>
609     </div>
610     <div class = "col-lg-5">
611         <input type = "text" class = "form-control"
          readonly = "true" value = "${brgR.hospital
            }" />
612     </div>
613     <div class = "col-lg-2">
614         <label>Hospital/Clinic Type:</label>
615     </div>
616     <div class = "col-lg-3">
617         <input class="form-control" type="text"
          readonly = "true" value="${brgR.
            hospitalType}" />
618     </div>
619 </div>
620 <br />
621 <div class = "row">
622     <div class = "col-lg-3">
623         <label>Hospital/Clinic Address:</label>
624     </div>
625     <div class = "col-lg-9">
626         <input type = "text" class = "form-control"
          readonly = "true" value = "${brgR.
            hospitalAddr}" />
627     </div>
628 </div>
629 </c:if>
630 <br />
631 <div class = "row">
632     <div class = "col-lg-2">
633         <label>Outcome</label>
634     </div>
635     <div class = "col-lg-4">
636         <input type = "text" class = "form-control"
          readonly = "true" value = "${brgR.outcome
            }" />
637     </div>
638 <c:if test = "${brgR.outcome == 'Died'}">
639     <div class = "col-lg-2">
640         <label>Date Died:</label>
641     </div>
642     <div class = "col-lg-4">
643         <input type = "text" class = "form-control"
          readonly = "true" value = "${brgR.dateDied
            }" />
644     </div>
645 </c:if>
646 </div>
647 <br />
648 </div>
649 </div>
650 </div>
651 </div>
652 </div>
653 </div>
654 </c:if>
655 </c:forEach>
656 </c:forEach>
657 </c:if>
658 </c:forEach>
659 </c:if>
660 </c:forEach>
661 </c:if>
662 </c:forEach>
663 </c:forEach>
664 </c:if>
665 </c:forEach>
666 </c:forEach>
667 </c:forEach>
668 </c:forEach>
669 </div>
670 </div>

```


Listing 60: muncity.jsp

```

1  <% taglib prefix="c" uri="http://java.sun.com/
2     jsp/jstl/core" %>
3  <% taglib uri="http://java.sun.com/jsp/jstl/
4     functions" prefix="fn"%>
5  <% taglib prefix="sec" uri="http://www.
6     springframework.org/security/tags" %>
7  <% taglib prefix="form" uri="http://www.
8     springframework.org/tags/form" %>
9
10 <div class = "container-fluid well-lg" style = "
11     background-color: #eee;">
12 <div class = "panel panel-info">
13 <c:if test = "${not empty successD}">
14 <div class = "alert alert-success">
15 <strong>${successD}</strong>
16 </div>
17 </c:if>
18 <c:if test = "${pageVisit == 'admin' ||
19     pageVisit == 'nonofficial'}">
20 <table class="table table-striped" id="example1
21     ">
22 <thead>
23 <tr>
24 <th>Region</th>
25 <th>No. of Municipal/City Users</th>
26 </tr>
27 </thead>
28 <tbody>
29 <c:forEach var = "i" begin = "0" end = "16">
30 <tr>
31 <th>
32 <button type = "button" class = "btn btn-
33     primary" href = "#regionModal2_${i}" data-
34     toggle = "modal">${regions[i]}</button>
35 </th>
36 <td>
37 <c:set var = "counter" value = "0" />
38 <c:forEach var = "user" items = "${allUsers
39     }">
40 <c:if test = "${user.region == regions[i] &&
41     (user.userType == 'Municipal/City Health
42     Worker' || user.userType == 'Municipal/City
43     Health Official')}">
44 <c:set var = "counter" value = "${counter +
45     1}" />
46 </c:if>
47 </c:forEach>
48 <th>
49 <td>
50 <td>
51 <td>
52 <td>
53 <td>
54 <td>
55 <td>
56 <td>
57 <td>
58 <td>
59 <td>
60 <td>
61 <td>
62 <td>
63 <td>
64 <td>
65 <td>
66 <td>
67 <td>
68 <td>
69 <td>
70 <td>
71 <td>
72 <td>
73 <td>
74 <td>
75 <td>
76 <td>
77 <td>
78 <td>
79 <td>
80 <td>
81 <td>
82 <td>
83 <td>
84 <td>
85 <td>
86 <td>
87 <td>
88 <td>
89 <td>
90 <td>
91 <td>
92 <td>
93 <td>
94 <td>
95 <td>
96 <td>
97 <td>
98 <td>
99 <td>
100 <td>
101 <td>
102 <td>
103 <td>
104 <td>
105 <td>
106 <td>
107 <td>
108 <td>
109 <td>
110 <td>
111 <td>
112 <td>
113 <td>
114 <td>
115 <td>
116 <td>
117 <td>
118 <td>
119 <td>
120 <td>
121 <td>
122 <td>
123 <td>
124 <td>
125 <td>
126 <td>
127 <td>
128 <td>
129 <td>
130 <td>
131 <td>
132 <td>
133 <td>
134 <td>
135 <td>
136 <td>

```

```

137 </c:forEach>
138 <tr>
139 <th>${mun.muncity}</th>
140 <th>${counter}</th>
141 <th>
142 <button class = "btn btn-primary" href = "#
usersModal2_${i}_${stat.index}_${j}_${munSt
.index}" data-toggle = "modal">See All</
button>
143 </th>
144 </tr>
145
146 </c:if>
147 </c:forEach>
148 </c:forEach>
149 </tbody>
150 </table>
151
152 </div>
153 </div>
154 </c:forEach>
155 </c:forEach>
156 </c:if>
157 <c:forEach var = "i" begin = "0" end = "16">
158 <c:forEach var = "provs" items = "${prov[i]}"
varStatus = "stat">
159 <c:forEach var = "j" begin = "0" end = "${fn:
length(mun) - 1}">
160 <c:forEach var = "mun" items = "${mun[j]}"
varStatus = "munSt">
161 <c:if test = "${fn:toLowerCase(user.province) ==
fn:toLowerCase(mun.province) && mun.region
== regions[i]}">
162 <div class = "modal fade" id = "usersModal2_${i}
_${stat.index}_${j}_${munSt.index}" data-
width = "700">
163 <div class = "modal-body">
164
165 <table class="table table-striped" id="
example1">
166 <thead>
167 <tr>
168 <th>Name</th>
169 <th>User Type</th>
170 <th>Profile</th>
171 <th>User Logs</th>
172 <th>Account Status</th>
173 <th>Activate or Deactivate</th>
174 </tr>
175 </thead>
176 <tbody>
177 <c:forEach var = "user" items = "${allUsers
}" varStatus = "st">
178 <c:if test = "${fn:toLowerCase(user.province
)} == fn:toLowerCase(provs.province) && user
.region == provs.region
179 && fn:toLowerCase(user.muncity) == fn:
toLowerCase(mun.muncity) && (user.userType
== 'Municipal/City Health Worker' ||
180 user.userType == 'Municipal/City Health
Official')}">
181 <tr>
182 <td>${user.firstname} ${user.middlename} ${
user.surname}</td>
183 <td>${user.userType}</td>
184 <td><button class = "btn btn-primary" href
= "#profile2_${i}_${stat.index}_${j}_${
munSt.index}_${st.index}" data-toggle = "
modal">View</button></td>
185 <td><button class = "btn btn-primary" href
= "#logs2_${i}_${stat.index}_${j}_${munSt.
index}_${st.index}" data-toggle = "modal">
View</button></td>
186 <td><c:if test = "${user.status == 1}">
Active</c:if><c:if test = "${user.status ==
0}">Inactive</c:if></td>
187 <td><c:if test = "${user.status == 0}"><
button class = "btn btn-primary" data-
toggle = "modal" href = "#activ_${user.
accID}">Activate</button></c:if><c:if test
= "${user.status == 1}"><button class = "
btn btn-primary" data-toggle = "modal" href
= "#deact_${user.accID}">Deactivate</
button></c:if></td>
188 </tr>
189 <div class = "modal fade" id = "deact_${user
.accID}">
190 <div class = "modal-body">
191 Are you sure you want to deactivate this
user?
192 </div>
193 <div class = "modal-footer">
194 <a href = "%= request.getContextPath()
%>/pages/activateUser/${user.accID}"><
button class = "btn btn-primary">Yes</
button></a>
195 <button class = "btn btn-primary" data-
dismiss = "modal">No</button>
196 </div>
197 </div>
198 <div class = "modal fade" id = "activ_${user
.accID}">
199 <div class = "modal-body">
200 Are you sure you want to activate this
user?
201 </div>
202 <div class = "modal-footer">
203 <a href = "%= request.getContextPath()
%>/pages/activateUser/${user.accID}"><
button class = "btn btn-primary">Yes</
button></a>
204 <button class = "btn btn-primary" data-
dismiss = "modal">No</button>
205 </div>
206 </div>
207 </c:if>
208 </c:forEach>
209 </tbody>
210 </table>
211 </div>
212 </div>
213 </c:if>
214 </c:forEach>
215 </c:forEach>
216 </c:forEach>
217 </c:forEach>
218 <c:forEach var = "i" begin = "0" end = "16">
219 <c:forEach var = "provs" items = "${prov[i]}"
varStatus = "stat">
220 <c:forEach var = "j" begin = "0" end = "${fn:
length(mun) - 1}">
221 <c:forEach var = "mun" items = "${mun[j]}"
varStatus = "munSt">
222 <c:if test = "${fn:toLowerCase(user.province) ==
fn:toLowerCase(mun.province) && mun.region
== regions[i]}">
223 <c:forEach var = "user" items = "${allUsers}"
varStatus = "st">
224 <c:if test = "${fn:toLowerCase(user.province) ==
fn:toLowerCase(provs.province) && user.
region == provs.region
225 && fn:toLowerCase(user.muncity) == fn:
toLowerCase(mun.muncity) && (user.userType
== 'Municipal/City Health Worker' ||
226 user.userType == 'Municipal/City Health
Official')}">
227 <div class = "modal fade" id = "profile2_${i}_${
stat.index}_${j}_${munSt.index}_${st.index
}" data-width = "700">
228 <div class = "modal-body">
229 <div class = "panel panel-info">
230 <div class = "panel-body">
231 <br />
232 <br />
233 <table class = "table table-striped">
234 <tr>
235 <td>First Name:</td>
236 <td>
237 ${user.firstname}
238 </td>
239 </tr>
240 <tr>
241 <td>Middle Name:</td>
242 <td>
243 ${user.middlename}
244 </td>
245 </tr>
246 <tr>
247 <td>Surname:</td>
248 <td>
249 ${user.surname}
250 </td>
251 </tr>
252 <tr>
253 <td>Region:</td>
254 <td>
255 ${user.region}
256 </td>
257 </tr>
258 <tr>
259 <td>Province:</td>
260 <td>
261 ${user.province}
262 </td>
263 </tr>
264 <tr>
265 <td>Institution:</td>
266 <td>
267 ${user.institution}
268 </td>
269 </tr>
270 <tr>
271 <td>Institution's Address:</td>
272 <td>
273 ${user.institutionAddr}
274 </td>
275 </tr>
276 <tr>

```

```

277 <td>Email:</td>
278 <td>
279     ${user.email}
280 </td>
281 </tr>
282 <tr>
283 <td>Contact Number:</td>
284 <td>
285     ${user.number}
286 </td>
287 </tr>
288 </table>
289 </div>
290 </div>
291 </div>
292 </div>
293 </c:if>
294 </c:forEach>
295 </c:if>
296 </c:forEach>
297 </c:forEach>
298 </c:forEach>
299 </c:forEach>
300 <c:forEach var = "i" begin = "0" end = "16">
301 <c:forEach var = "provs" items = "${prov[i]}"
    varStatus = "stat">
302 <c:forEach var = "j" begin = "0" end = "${fn:
    length(mun) - 1}">
303 <c:forEach var = "mun" items = "${mun[j]}"
    varStatus = "munSt">
304 <c:if test = "${fn:toLowerCase(user.province) ==
    fn:toLowerCase(mun.province) && mun.region
    == regions[i]}">
305 <c:forEach var = "user" items = "${allUsers}"
    varStatus = "st">
306 <c:if test = "${fn:toLowerCase(user.province) ==
    fn:toLowerCase(provs.province) && user.
    region == provs.region
307 && fn:toLowerCase(user.muncity) == fn:
    toLowerCase(mun.muncity) && (user.userType
    == 'Municipal/City Health Worker' ||
308 user.userType == 'Municipal/City Health Official
    ')}">
309 <div class = "modal fade" id = "logs2_${i}_${
    stat.index}_${j}_${munSt.index}_${st.index
    }" data-width = "700">
310 <div class = "modal-body">
311 <div class = "panel panel-info">
312 <div class = "panel-body">
313 <br />
314 <br />
315 <table class = "table table-striped">
316 <thead>
317 <tr>
318 <th>Date</th>
319 <th>Time</th>
320 <th>Action</th>
321 <th>view Record</th>
322 </tr>
323 </thead>
324 <tbody>
325 <c:forEach var = "logs" items = "${allLogs
    }" varStatus = "lgStat">
326 <c:if test = "${logs.accountID == user.
    accID}">
327 <c:if test = "${(logs.recordtable == '
    muncityrecords') && logs.recordtype == 1}">
328 <tr>
329 <td>${logs.date}</td>
330 <td>${logs.time}</td>
331 <td>${logs.action}</td>
332 <c:forEach var = "k" begin = "0" end = "${
    fn:length(munR) - 1}">
333 <c:forEach var = "munR" items = "${munR[k
    ]}" varStatus = "mnR">
334 <c:if test = "${logs.recordID == munR.
    recordID}">
335 <td><button class = "btn btn-primary"
    href = "#logsRecord2_${i}_${stat.index}_${j
    }_${munSt.index}_${st.index}_${k}_${mnR.
    index}" data-toggle = "modal">View</button
    ></td>
336 </c:if>
337 </c:forEach>
338 </c:forEach>
339 </tr>
340 </c:if>
341 <c:if test = "${(logs.recordtable == '
    editbarangayrecords' && logs.recordtype ==
    0) || (logs.recordtable == 'barangayrecords
    ' && logs.recordtype == 1)}">
342 <tr>
343 <td>${logs.date}</td>
344 <td>${logs.time}</td>
345 <td>${logs.action}</td>
346 <c:forEach var = "k" begin = "${fn:length(
    munR)}" end = "${fn:length(munR) + fn:
    length(editB) - 1}">
347 <c:forEach var = "munR" items = "${editB[k -
    fn:length(munR)]}" varStatus = "mnR">
348 <c:if test = "${logs.recordID == munR.
    recordID}">
349 <td><button class = "btn btn-primary" href =
    "#logsRecord2_${i}_${stat.index}_${j}_${
    munSt.index}_${st.index}_${k}_${mnR.index}"
    data-toggle = "modal">View</button></td>
350 </c:if>
351 </c:forEach>
352 </c:forEach>
353 </tr>
354 </c:if>
355 </c:if>
356 </c:forEach>
357 </tbody>
358 </table>
359 </div>
360 </div>
361 </div>
362 </div>
363 </c:if>
364 </c:forEach>
365 </c:if>
366 </c:forEach>
367 </c:forEach>
368 </c:forEach>
369 </c:forEach>
370 <c:forEach var = "i" begin = "0" end = "16">
371 <c:forEach var = "provs" items = "${prov[i]}"
    varStatus = "stat">
372 <c:forEach var = "j" begin = "0" end = "${fn:
    length(mun) - 1}">
373 <c:forEach var = "mun" items = "${mun[j]}"
    varStatus = "munSt">
374 <c:if test = "${fn:toLowerCase(user.province) ==
    fn:toLowerCase(mun.province) && mun.region
    == regions[i]}">
375 <c:forEach var = "user" items = "${allUsers}"
    varStatus = "st">
376 <c:if test = "${fn:toLowerCase(user.province) ==
    fn:toLowerCase(provs.province) && user.
    region == provs.region
377 && fn:toLowerCase(user.muncity) == fn:
    toLowerCase(mun.muncity) && (user.userType
    == 'Municipal/City Health Worker' ||
378 user.userType == 'Municipal/City Health Official
    ')}">
379 <c:forEach var = "logs" items = "${allLogs}"
    varStatus = "lgStat">
380 <c:forEach var = "k" begin = "0" end = "${fn:
    length(munR) - 1}">
381 <c:forEach var = "munR" items = "${munR[k]}"
    varStatus = "mnR">
382 <c:if test = "${logs.accountID == user.accID &&
    logs.recordID == munR.recordID}">
383 <div class = "modal fade" id = "logsRecord2_${i
    }_${stat.index}_${j}_${munSt.index}_${st.
    index}_${k}_${mnR.index}" data-width =
    "780">
384 <div class = "modal-body">
385 <div class = "panel panel-info">
386 <div class = "panel-body">
387 <div class = "panel panel-info">
388 <div class = "panel-heading">
389 Record Information
390 </div>
391 <div class = "panel-body">
392 <div class = "row">
393 <div class = "col-lg-2">
394 <label>Region:</label>
395 </div>
396 <div class = "col-lg-4">
397 <input type = "text" class = "form-
    control" readonly = "true" value = "${munR.
    region}" />
398 </div>
399 <div class = "col-lg-2">
400 <label>Province:</label>
401 </div>
402 <div class = "col-lg-4">
403 <input type = "text" class = "form-
    control" readonly = "true" value = "${munR.
    province}" />
404 </div>
405 </div>
406 <br />
407 <div class = "row">
408 <div class = "col-lg-2">
409 <label>Municipality/City:</label>
410 </div>
411 <div class = "col-lg-4">
412 <input type = "text" class = "form-
    control" readonly = "true" value = "${munR.
    muncity}" />
413 </div>
414 <div class = "col-lg-2">
415 <label>Barangay</label>
416 </div>
417 <div class = "col-lg-4">

```

```

418     <input type = "text" class = "form-      497     <label>Name:</label>
control" readonly = "true" value = "${munR.      498     </div>
barangay}" />      499     <div class = "col-lg-5">
419     </div>      500     <input type = "text" class = "form-control"
420     </div>      501     <input type = "text" class = "form-control"
421     <br />      502     </div>
422     <div class = "row">      503     <br />
423     <div class = "col-lg-2">      504     </c:if>
424     <label>Barangay Health Worker:</label>      505     <c:if test = "${not empty munR.address}">
425     </div>      506     <div class = "row">
426     <div class = "col-lg-4">      507     <div class = "col-lg-1">
427     <input type = "text" class = "form-      508     <label>Complete Address:</label>
control" readonly = "true" value = "${munR.      509     </div>
healthworker}" />      510     <div class = "col-lg-5">
428     </div>      511     <textarea class = "form-control" readonly =
429     <div class = "col-lg-2">      512     <input type = "text" class = "form-control"
430     <label>Date Submitted:</label>      513     </div>
431     </div>      514     <br />
432     <div class = "col-lg-4">      515     </c:if>
433     <input type = "text" class = "form-      516     <div class = "row">
control" readonly = "true" value = "${munR.      517     <div class = "col-lg-1">
dateSubmitted}" />      518     <label>Age:</label>
434     </div>      519     </div>
435     </div>      520     <div class = "col-lg-2">
436     <c:if test = "${munR.status == 1}">      521     <input type = "text" class = "form-control"
437     <br />      522     <input type = "text" class = "form-control"
438     <div class = "row">      523     </div>
439     <div class = "col-lg-2">      524     <div class = "col-lg-1">
440     <label>Municipality/City Health Worker      525     <label>Date of Birth:</label>
:</label>      526     </div>
441     </div>      527     <div class = "col-lg-3">
442     <div class = "col-lg-4">      528     <input class="form-control" type="text"
443     <input type = "text" class = "form-      529     <input type = "text" class = "form-control"
control" readonly = "true" value = "${munR.      530     <input type = "text" class = "form-control"
muncityHW}" />      531     <input type = "text" class = "form-control"
444     </div>      532     <input type = "text" class = "form-control"
445     <div class = "col-lg-2">      533     <input type = "text" class = "form-control"
446     <label>Date Approved:</label>      534     </div>
447     </div>      535     </div>
448     <div class = "col-lg-4">      536     </div>
449     <input type = "text" class = "form-      537     </div>
control" readonly = "true" value = "${munR.      538     <div class = "panel panel-info">
dateApproved}" />      539     <div class = "panel-heading">
450     </div>      540     <label>Patient's Clinical Information</label>
451     </div>      541     </div>
452     </c:if>      542     <div class = "panel-body">
453     <c:if test = "${munR.status == 2}">      543     <div class = "row">
454     <br />      544     <div class = "col-lg-2">
455     <div class = "row">      545     <label>Disease:</label>
456     <div class = "col-lg-2">      546     </div>
457     <label>Municipality/City Health Worker:</      547     <div class = "col-lg-4">
label>      548     <input type = "text" class = "form-control"
458     </div>      549     <input type = "text" class = "form-control"
459     <div class = "col-lg-4">      550     </div>
460     <input type = "text" class = "form-      551     <div class = "col-lg-2">
control" readonly = "true" value = "${munR.      552     <label>Date Onset:</label>
muncityHW}" />      553     </div>
461     </div>      554     <div class = "col-lg-4">
462     <div class = "col-lg-2">      555     <input class="form-control" type="text"
463     <label>Date Rejected:</label>      556     <input type = "text" class = "form-control"
464     </div>      557     </div>
465     <div class = "col-lg-4">      558     </div>
466     <input type = "text" class = "form-      559     <div class = "row">
control" readonly = "true" value = "${munR.      560     <div class = "col-lg-2">
dateApproved}" />      561     <label>Symptoms:</label>
467     </div>      562     </div>
468     </div>      563     <div class = "col-lg-4">
469     </c:if>      564     <input type = "text" class = "form-control"
470     <c:if test = "${munR.editStatus == 1}">      565     <input type = "text" class = "form-control"
471     <br />      566     <input type = "text" class = "form-control"
472     <div class = "row">      567     </div>
473     <div class = "col-lg-2">      568     <div class = "col-lg-4">
474     <label>Edited by:</label>      569     <input class="form-control" type="text"
475     </div>      570     <input type = "text" class = "form-control"
476     <div class = "col-lg-4">      571     </div>
477     <input type = "text" class = "form-control      572     </div>
" readonly = "true" value = "${munR.      573     <c:if test = "${munR.admitted == 'No'}">
editedBy}" />      574     <div class = "row">
478     </div>      575     <div class = "col-lg-2">
479     <div class = "col-lg-2">      576     <label>Patient Admitted?:</label>
480     <label>Date Edited:</label>      577     </div>
481     </div>      578     <div class = "col-lg-4">
482     <div class = "col-lg-3">      579     <input type = "text" class = "form-control"
483     <input type = "text" class = "form-control      580     </div>
" readonly = "true" value = "${munR.
dateEdited}" />
484     </div>
485     </div>
486     </c:if>
487     </div>
488     </div>
489     <div class = "panel panel-info">
490     <div class = "panel-heading">
491     <label>Patient's Information</label>
492     </div>
493     <div class = "panel-body">
494     <c:if test = "${not empty munR.name}">
495     <div class = "row">
496     <div class = "col-lg-1">

```

```

        readonly = "true" value = "${munR.admitted
580     }"/>
581   </div>
582 </c:if>
583 <c:if test = "${munR.admitted == 'Yes'}">
584 <div class = "row">
585   <div class = "col-lg-2">
586     <label>Patient Admitted?:</label>
587   </div>
588   <div class = "col-lg-4">
589     <input type = "text" class = "form-control"
        readonly = "true" value = "${munR.admitted
590     }"/>
591   </div>
592   <div class = "col-lg-2">
593     <label>Date Admitted:</label>
594   </div>
595   <div class = "col-lg-4">
596     <input class="form-control" type="text"
        readonly = "true" value="${munR.
        dateAdmitted}" />
597   </div>
598 <br />
599 <div class = "row">
600   <div class = "col-lg-2">
601     <label>Hospital/Clinic:</label>
602   </div>
603   <div class = "col-lg-5">
604     <input type = "text" class = "form-control"
        readonly = "true" value = "${munR.hospital
605     }"/>
606   </div>
607   <div class = "col-lg-2">
608     <label>Hospital/Clinic Type:</label>
609   </div>
610   <div class = "col-lg-3">
611     <input class="form-control" type="text"
        readonly = "true" value="${munR.
        hospitalType}" />
612   </div>
613 <br />
614 <div class = "row">
615   <div class = "col-lg-3">
616     <label>Hospital/Clinic Address:</label>
617   </div>
618   <div class = "col-lg-9">
619     <input type = "text" class = "form-control"
        readonly = "true" value = "${munR.
        hospitalAddr}"/>
620   </div>
621 </div>
622 </c:if>
623 <br />
624 <div class = "row">
625   <div class = "col-lg-2">
626     <label>Outcome</label>
627   </div>
628   <div class = "col-lg-4">
629     <input type = "text" class = "form-control"
        readonly = "true" value = "${munR.outcome
630     }"/>
631   </div>
632 <c:if test = "${munR.outcome == 'Died'}">
633   <div class = "col-lg-2">
634     <label>Date Died:</label>
635   </div>
636   <div class = "col-lg-4">
637     <input type = "text" class = "form-control"
        readonly = "true" value = "${munR.dateDied
638     }"/>
639   </div>
640 <br />
641 </div>
642 </div>
643 </div>
644 </div>
645 </div>
646 </div>
647 </c:if>
648 </c:forEach>
649 </c:forEach>
650 <c:forEach var = "k" begin = "${fn:length(munR)
        }" end = "${fn:length(munR) + fn:length(
        editB) - 1}">
651 <c:forEach var = "munR" items = "${editB[k - fn:
        length(munR)]}" varStatus = "mnR">
652 <c:if test = "${logs.accountID == user.accountID &&
        logs.recordID == munR.recordID}">
653 <div class = "modal fade" id = "logsRecord2_${i}
        _${stat.index}_${j}_${munSt.index}_${st.
        index}_${k}_${mnR.index}" data-width =
        "780">
654   <div class = "modal-body">
655     <div class = "panel panel-info">

```

```

<div class = "panel-body">
<div class = "panel panel-info">
<div class = "panel-heading">
  Record Information
</div>
<div class = "panel-body">
<div class = "row">
<div class = "col-lg-2">
  <label>Region:</label>
</div>
<div class = "col-lg-4">
  <input type = "text" class = "form-
control" readonly = "true" value = "${munR.
region}" />
</div>
<div class = "col-lg-2">
  <label>Province:</label>
</div>
<div class = "col-lg-4">
  <input type = "text" class = "form-
control" readonly = "true" value = "${munR.
province}" />
</div>
</div>
<br />
<div class = "row">
<div class = "col-lg-2">
  <label>Municipality/City:</label>
</div>
<div class = "col-lg-4">
  <input type = "text" class = "form-
control" readonly = "true" value = "${munR.
muncity}" />
</div>
<div class = "col-lg-2">
  <label>Barangay</label>
</div>
<div class = "col-lg-4">
  <input type = "text" class = "form-
control" readonly = "true" value = "${munR.
barangay}" />
</div>
</div>
<br />
<div class = "row">
<div class = "col-lg-2">
  <label>Barangay Health Worker:</label>
</div>
<div class = "col-lg-4">
  <input type = "text" class = "form-
control" readonly = "true" value = "${munR.
healthworker}" />
</div>
<div class = "col-lg-2">
  <label>Date Submitted:</label>
</div>
<div class = "col-lg-4">
  <input type = "text" class = "form-
control" readonly = "true" value = "${munR.
dateSubmitted}" />
</div>
</div>
<c:if test = "${munR.status == 1}">
<br />
<div class = "row">
<div class = "col-lg-2">
  <label>Municipality/City Health Worker
:</label>
</div>
<div class = "col-lg-4">
  <input type = "text" class = "form-
control" readonly = "true" value = "${munR.
muncityHW}" />
</div>
<div class = "col-lg-2">
  <label>Date Approved:</label>
</div>
<div class = "col-lg-4">
  <input type = "text" class = "form-
control" readonly = "true" value = "${munR.
dateApproved}" />
</div>
</div>
</c:if>
<c:if test = "${munR.status == 2}">
<br />
<div class = "row">
<div class = "col-lg-2">
  <label>Municipality/City Health Worker:</
label>
</div>
<div class = "col-lg-4">
  <input type = "text" class = "form-
control" readonly = "true" value = "${munR.
muncityHW}" />
</div>
<div class = "col-lg-2">
  <label>Date Rejected:</label>
</div>

```

```

735     <div class = "col-lg-4">
736     <input type = "text" class = "form-
control" readonly = "true" value = "${munR.
dateApproved}" />
737     </div>
738     </div>
739     </c:if>
740     <c:if test = "${munR.editStatus == 1}">
741     <br />
742     <div class = "row">
743     <div class = "col-lg-2">
744     <label>Edited by:</label>
745     </div>
746     <div class = "col-lg-4">
747     <input type = "text" class = "form-control
" readonly = "true" value = "${munR.
editedBy}" />
748     </div>
749     <div class = "col-lg-2">
750     <label>Date Edited:</label>
751     </div>
752     <div class = "col-lg-3">
753     <input type = "text" class = "form-control
" readonly = "true" value = "${munR.
dateEdited}" />
754     </div>
755     </div>
756     </c:if>
757     </div>
758     </div>
759     <div class = "panel panel-info">
760     <div class = "panel-heading">
761     <label>Patient's Information</label>
762     </div>
763     <div class = "panel-body">
764     <c:if test = "${not empty munR.name}">
765     <div class = "row">
766     <div class = "col-lg-1">
767     <label>Name:</label>
768     </div>
769     <div class = "col-lg-5">
770     <input type = "text" class = "form-control"
readonly = "true" value = "${munR.name}" />
771     </div>
772     </div>
773     <br />
774     </c:if>
775     <c:if test = "${not empty bR1.address}">
776     <div class = "row">
777     <div class = "col-lg-1">
778     <label>Complete Address:</label>
779     </div>
780     <div class = "col-lg-5">
781     <textarea class = "form-control" readonly =
"true" value = "${munR.address}">${munR.
address}</textarea>
782     </div>
783     </div>
784     <br />
785     </c:if>
786     <div class = "row">
787     <div class = "col-lg-1">
788     <label>Age:</label>
789     </div>
790     <div class = "col-lg-2">
791     <input type = "text" class = "form-control"
readonly = "true" value = "${munR.age}" />
792     </div>
793     <div class = "col-lg-1">
794     <label>Date of Birth:</label>
795     </div>
796     <div class = "col-lg-3">
797     <input class="form-control" type="text"
readonly = "true" value="${munR.dateofbirth
}" />
798     </div>
799     <div class = "col-lg-1">
800     <label>Gender:</label>
801     </div>
802     <div class = "col-lg-3">
803     <input class = "form-control" type = "text"
readonly = "true" value = "${munR.gender
}" />
804     </div>
805     </div>
806     </div>
807     </div>
808     <div class = "panel panel-info">
809     <div class = "panel-heading">
810     <label>Patient's Clinical Information</label>
811     </div>
812     <div class = "panel-body">
813     <div class = "row">
814     <div class = "col-lg-2">
815     <label>Disease:</label>
816     </div>
817     <div class = "col-lg-4">
818     <input type = "text" class = "form-control"
readonly = "true" value = "${munR.disease
}"/>
819     </div>
820     <div class = "col-lg-2">
821     <label>Date Onset:</label>
822     </div>
823     <div class = "col-lg-4">
824     <input class="form-control" type="text"
readonly = "true" value="${munR.dateOnset}"
/>
825     </div>
826     </div>
827     <br />
828     <div class = "row">
829     <div class = "col-lg-2">
830     <label>Symptoms:</label>
831     </div>
832     <div class = "col-lg-4">
833     <input type = "text" class = "form-control"
readonly = "true" value = "${munR.symptoms
}"/>
834     </div>
835     <div class = "col-lg-2">
836     <label>Case Classification:</label>
837     </div>
838     <div class = "col-lg-4">
839     <input class="form-control" type="text"
readonly = "true" value="${munR.caseType}"
/>
840     </div>
841     </div>
842     <br />
843     <c:if test = "${munR.admitted == 'No'}">
844     <div class = "row">
845     <div class = "col-lg-2">
846     <label>Patient Admitted?:</label>
847     </div>
848     <div class = "col-lg-4">
849     <input type = "text" class = "form-control"
readonly = "true" value = "${munR.admitted
}"/>
850     </div>
851     </div>
852     </c:if>
853     <c:if test = "${munR.admitted == 'Yes'}">
854     <div class = "row">
855     <div class = "col-lg-2">
856     <label>Patient Admitted?:</label>
857     </div>
858     <div class = "col-lg-4">
859     <input type = "text" class = "form-control"
readonly = "true" value = "${munR.admitted
}"/>
860     </div>
861     <div class = "col-lg-2">
862     <label>Date Admitted:</label>
863     </div>
864     <div class = "col-lg-4">
865     <input class="form-control" type="text"
readonly = "true" value="${munR.
dateAdmitted}" />
866     </div>
867     </div>
868     <br />
869     <div class = "row">
870     <div class = "col-lg-2">
871     <label>Hospital/Clinic:</label>
872     </div>
873     <div class = "col-lg-5">
874     <input type = "text" class = "form-control"
readonly = "true" value = "${munR.hospital
}"/>
875     </div>
876     <div class = "col-lg-2">
877     <label>Hospital/Clinic Type:</label>
878     </div>
879     <div class = "col-lg-3">
880     <input class="form-control" type="text"
readonly = "true" value="${munR.
hospitalType}" />
881     </div>
882     </div>
883     <br />
884     <div class = "row">
885     <div class = "col-lg-3">
886     <label>Hospital/Clinic Address:</label>
887     </div>
888     <div class = "col-lg-9">
889     <input type = "text" class = "form-control"
readonly = "true" value = "${munR.
hospitalAddr}" />
890     </div>
891     </div>
892     </c:if>
893     <br />
894     <div class = "row">
895     <div class = "col-lg-2">
896     <label>Outcome</label>
897     </div>
898     <div class = "col-lg-4">

```

```

899     <input type = "text" class = "form-control"
        readonly = "true" value = "${munR.outcome
        }"/>
900 </div>
901 <c:if test = "${munR.outcome == 'Died'}">
902 <div class = "col-lg-2">
903   <label>Date Died:</label>
904 </div>
905 <div class = "col-lg-4">
906   <input type = "text" class = "form-control"
        readonly = "true" value = "${munR.dateDied
        }"/>
907 </div>
908 </c:if>
909 </div>
910 <br />
911 </div>
912 </div>

```

```

913 </div>
914 </div>
915 </div>
916 </div>
917 </c:if>
918 </c:forEach>
919 </c:forEach>
920 </c:forEach>
921 </c:if>
922 </c:forEach>
923 </c:if>
924 </c:forEach>
925 </c:forEach>
926 </c:forEach>
927 </c:forEach>
928 </div>
929 </div>

```

Listing 61: province.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
  jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
  functions" prefix="fn" %>
3 <%@ taglib prefix="sec" uri="http://www.
  springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
  springframework.org/tags/form" %>
5 <div class = "container-fluid well-lg" style = "
  background-color: #eee;">
6 <div class = "panel panel-info">
7 <c:if test = "${not empty successD}">
8 <div class = "alert alert-success">
9 <strong>${successD}</strong>
10 </div>
11 </c:if>
12 <table class="table table-striped" id="example1
  ">
13 <thead>
14 <tr>
15 <th>Region</th>
16 <th>No. of Provincial Users</th>
17 </tr>
18 </thead>
19 <tbody>
20 <c:forEach var = "i" begin = "0" end = "16">
21 <tr>
22 <th>
23 <button type = "button" class = "btn btn-
  primary" href = "#regionModal3_${i}" data-
  toggle = "modal">${regions[i]}</button>
24 </th>
25 <c:set var = "counter" value = "0" />
26 <c:forEach var = "user" items = "${allUsers
  }">
27 <c:if test = "${user.region == regions[i] &&
  (user.userType == 'Provincial Health
  Worker' || user.userType == 'Provincial
  Health Official')}">
28 <c:set var = "counter" value = "${counter +
  1}" />
29 </c:if>
30 </c:forEach>
31 <th>
32 <strong>${counter}</strong>
33 </th>
34 </tr>
35 </c:forEach>
36 </tbody>
37 </table>
38 <c:forEach var = "i" begin = "0" end = "16">
39 <div class = "modal fade" id = "regionModal3_${i}
  ">
40 <div class = "modal-body">
41 <table class="table table-striped" id="
  example1">
42 <thead>
43 <tr>
44 <th>Province</th>
45 <th>No. of Provincial Workers/Officials</th>
46 <th>View All Users</th>
47 </tr>
48 </thead>
49 <tbody>
50 <c:forEach var = "provs" items = "${prov[i]
  }" varStatus = "stat">
51 <c:set var = "counter" value = "0" />
52 <c:forEach var = "user" items = "${allUsers
  }">
53 <c:if test = "${fn:toLowerCase(user.province
  ) == fn:toLowerCase(provs.province) && user
  .region == provs.region
54 && (user.userType == 'Provincial Health
  Worker' || user.userType == 'Provincial
  Health Official')}">
55 <c:set var = "counter" value = "${counter +
  1}" />
56 </c:if>
57 </c:forEach>
58 <tr>
59 <th>${provs.province}</th>
60 <th>${counter}</th>
61 <th>
62 <button type = "button" class = "btn btn-
  primary" href = "#usersModal3_${i}_${stat.
  index}" data-toggle = "modal">See All</
  button>
63 </th>
64 </tr>
65 </tbody>
66 </table>
67 </tbody>
68 </table>
69 </div>
70 </div>
71 </c:forEach>
72 <c:forEach var = "i" begin = "0" end = "16">
73 <c:forEach var = "provs" items = "${prov[i]}"
  varStatus = "stat">
74 <div class = "modal fade" id = "usersModal3_${i}
  _${stat.index}" data-width = "700">
75 <div class = "modal-body">
76 <table class="table table-striped" id="
  example1">
77 <thead>
78 <tr>
79 <th>Name</th>
80 <th>User Type</th>
81 <th>Profile</th>
82 <th>User Logs</th>
83 <th>Account Status</th>
84 <th>Activate or Deactivate</th>
85 </tr>
86 </thead>
87 <tbody>
88 <c:forEach var = "user" items = "${allUsers
  }" varStatus = "st">
89 <c:if test = "${fn:toLowerCase(user.province
  ) == fn:toLowerCase(provs.province) && user
  .region == provs.region
90 && (user.userType == 'Provincial Health
  Worker' || user.userType == 'Provincial
  Health Official')}">
91 <tr>
92 <td>${user.firstname} ${user.middlename} ${
  user.surname}</td>
93 <td>${user.userType}</td>
94 <td><button class = "btn btn-primary" href
  = "#profile3_${i}_${stat.index}_${st.index}
  " data-toggle = "modal">View</button></td>
95 <td><button class = "btn btn-primary" href
  = "#logs3_${i}_${stat.index}_${st.index}"
  data-toggle = "modal">View</button></td>
96 <td><c:if test = "${user.status == 1}">
  Active</c:if><c:if test = "${user.status ==
  0}">Inactive</c:if></td>
97 <td><c:if test = "${user.status == 0}"><
  button class = "btn btn-primary" data-
  toggle = "modal" href = "#activ_${user.
  accID}">Activate</button></c:if><c:if test
  = "${user.status == 1}"><button class = "
  btn btn-primary" data-toggle = "modal" href
  = "#deact_${user.accID}">Deactivate</
  button></c:if></td>
98 </tr>
99 </div>
100 <div class = "modal-body">
101 Are you sure you want to deactivate this
  user?
102 </div>
103 <div class = "modal-footer">
104 <a href = "<%= request.getContextPath()
  %>/pages/deactivateUser/${user.accID}"><
  button class = "btn btn-primary">Yes</

```

```

105         button></a>
106         <button class = "btn btn-primary" data-
107         dismiss = "modal">No</button>
108     </div>
109 <div class = "modal fade" id = "activ_{$user
110 .accID}">
111 <div class = "modal-body">
112     Are you sure you want to activate this
113     user?
114 </div>
115 <div class = "modal-footer">
116     <a href = "<%= request.getContextPath()
117     %>/pages/activateUser/{$user.accID}">
118     button class = "btn btn-primary">Yes</
119     button></a>
120 <button class = "btn btn-primary" data-
121 dismiss = "modal">No</button>
122 </div>
123 </div>
124 </c:if>
125 </c:forEach>
126 </tbody>
127 </table>
128 </div>
129 </div>
130 </c:forEach>
131 </c:forEach>
132 <c:forEach var = "i" begin = "0" end = "16">
133 <c:forEach var = "provs" items = "{$prov[i]}"
134     varStatus = "stat">
135 <c:forEach var = "user" items = "{$allUsers}"
136     varStatus = "st">
137 <div class = "modal fade" id = "profile3_{$i}_{$
138     stat.index}_{$st.index}">
139 <div class = "modal-body">
140 <c:if test = "{$fn:toLowerCase(user.province)
141     == fn:toLowerCase(provs.province) && user.
142     region == provs.region
143     && (user.userType == 'Provincial Health Worker
144     ' || user.userType == 'Provincial Health
145     Official')}">
146 <div class = "panel panel-info">
147 <div class = "panel-body">
148 <br />
149 <br />
150 <table class = "table table-striped">
151 <tr>
152 <td>First Name:</td>
153 <td>
154     {$user.firstname}
155 </td>
156 </tr>
157 <tr>
158 <td>Middle Name:</td>
159 <td>
160     {$user.middlename}
161 </td>
162 </tr>
163 <tr>
164 <td>Surname:</td>
165 <td>
166     {$user.surname}
167 </td>
168 </tr>
169 <tr>
170 <td>Region:</td>
171 <td>
172     {$user.region}
173 </td>
174 </tr>
175 <tr>
176 <td>Province:</td>
177 <td>
178     {$user.province}
179 </td>
180 </tr>
181 <tr>
182 <td>Institution:</td>
183 <td>
184     {$user.institution}
185 </td>
186 </tr>
187 <tr>
188 <td>Institution's Address:</td>
189 <td>
190     {$user.institutionAddr}
191 </td>
192 </tr>
193 <tr>
194 <td>Email:</td>
195 <td>
196     {$user.email}
197 </td>
198 </tr>
199 <tr>
200 <td>Contact Number:</td>
201 <td>
202     {$user.number}
203 </td>
204 </tr>
205 </table>
206 </div>
207 </c:if>
208 </c:forEach>
209 </c:forEach>
210 </c:forEach>
211 <div class = "modal fade" id = "logs3_{$i}_{$
212     stat.index}_{$st.index}">
213 <div class = "modal-body">
214 <c:if test = "{$fn:toLowerCase(user.province)
215     == fn:toLowerCase(provs.province) && user.
216     region == provs.region
217     && (user.userType == 'Provincial Health Worker
218     ' || user.userType == 'Provincial Health
219     Official')}">
220 <div class = "panel panel-info">
221 <div class = "panel-body">
222 <br />
223 <br />
224 <table class = "table table-striped">
225 <thead>
226 <tr>
227 <th>Date</th>
228 <th>Time</th>
229 <th>Action</th>
230 <th>view Record</th>
231 </tr>
232 </thead>
233 <tbody>
234 <c:forEach var = "logs" items = "{$allLogs
235     }" varStatus = "lgStat">
236 <tr>
237 <c:if test = "{$logs.accountID == user.
238     accID}">
239 <td>{$logs.date}</td>
240 <td>{$logs.time}</td>
241 <td>{$logs.action}</td>
242 <c:if test = "{$logs.recordtable == '
243     muncityrecords' || logs.recordtable == '
244     editmuncityrecords'}">
245 <c:forEach var = "k" begin = "0" end = "$
246     {fn:length(provR) - 1}">
247 <c:forEach var = "provR" items = "{$provR
248     [k]}" varStatus = "prR">
249 <c:if test = "{$logs.recordID == provR.
250     recordID}">
251 <td><button class = "btn btn-primary"
252     href = "#logsRecord3_{$i}_{$st.index}_{$
253     st.index}_{$k}_{$prR.index}" data-toggle =
254     "modal">View</button></td>
255 </c:if>
256 </c:forEach>
257 </c:forEach>
258 </c:if>
259 </tr>
260 </c:forEach>
261 </tbody>
262 </table>
263 </div>
264 </div>
265 </c:if>
266 </c:forEach>
267 </c:forEach>
268 </c:forEach>
269 <c:forEach var = "i" begin = "0" end = "16">
270 <c:forEach var = "provs" items = "{$prov[i]}"
271     varStatus = "stat">
272 <c:forEach var = "user" items = "{$allUsers}"
273     varStatus = "st">
274 <c:if test = "{$fn:toLowerCase(user.province) ==
275     fn:toLowerCase(provs.province) && user.
276     region == provs.region
277     && (user.userType == 'Provincial Health Worker'
278     ' || user.userType == 'Provincial Health
279     Official')}">
280 <c:forEach var = "logs" items = "{$allLogs}"
281     varStatus = "lgStat">
282 <c:if test = "{$logs.recordtable == '
283     muncityrecords'}">
284 <c:forEach var = "k" begin = "0" end = "{$fn:
285     length(provR) - 1}">
286 <c:forEach var = "provR" items = "{$provR[k]}"
287     varStatus = "prR">
288 <div class = "modal fade" id = "logsRecord3_{$i}
289     _{$st.index}_{$st.index}_{$k}_{$prR.index}
290     ">

```



```

259 <div class = "modal-body">
260 <div class = "panel panel-info">
261 <div class = "panel-body">
262 <div class = "panel panel-info">
263 <div class = "panel-heading">
264 Record Information
265 </div>
266 <div class = "panel-body">
267 <div class = "row">
268 <div class = "col-lg-2">
269 <label>Region:</label>
270 </div>
271 <div class = "col-lg-4">
272 <input type = "text" class = "form-
control" readonly = "true" value = "${provR
.region}" />
273 </div>
274 <div class = "col-lg-2">
275 <label>Province:</label>
276 </div>
277 <div class = "col-lg-4">
278 <input type = "text" class = "form-
control" readonly = "true" value = "${provR
.province}" />
279 </div>
280 </div>
281 <br />
282 <div class = "row">
283 <div class = "col-lg-2">
284 <label>Municipality/City:</label>
285 </div>
286 <div class = "col-lg-4">
287 <input type = "text" class = "form-
control" readonly = "true" value = "${provR
.muncity}" />
288 </div>
289 <div class = "col-lg-2">
290 <label>Barangay</label>
291 </div>
292 <div class = "col-lg-4">
293 <input type = "text" class = "form-
control" readonly = "true" value = "${provR
.barangay}" />
294 </div>
295 </div>
296 <br />
297 <div class = "row">
298 <div class = "col-lg-2">
299 <label>Barangay Health Worker:</label>
300 </div>
301 <div class = "col-lg-4">
302 <input type = "text" class = "form-
control" readonly = "true" value = "${provR
.healthworker}" />
303 </div>
304 <div class = "col-lg-2">
305 <label>Date Submitted:</label>
306 </div>
307 <div class = "col-lg-4">
308 <input type = "text" class = "form-
control" readonly = "true" value = "${provR
.dateSubmitted}" />
309 </div>
310 </div>
311 <c:if test = "${provR.status == 1}">
312 <br />
313 <div class = "row">
314 <div class = "col-lg-2">
315 <label>Municipality/City Health Worker
:</label>
316 </div>
317 <div class = "col-lg-4">
318 <input type = "text" class = "form-
control" readonly = "true" value = "${provR
.muncityHW}" />
319 </div>
320 <div class = "col-lg-2">
321 <label>Date Approved:</label>
322 </div>
323 <div class = "col-lg-4">
324 <input type = "text" class = "form-
control" readonly = "true" value = "${provR
.dateApproved}" />
325 </div>
326 </div>
327 </c:if>
328 <c:if test = "${provR.status == 2}">
329 <br />
330 <div class = "row">
331 <div class = "col-lg-2">
332 <label>Municipality/City Health Worker:</
label>
333 </div>
334 <div class = "col-lg-4">
335 <input type = "text" class = "form-
control" readonly = "true" value = "${provR
.muncityHW}" />
336 </div>
337 <div class = "col-lg-2">
338 <label>Date Rejected:</label>
339 </div>
340 <div class = "col-lg-4">
341 <input type = "text" class = "form-
control" readonly = "true" value = "${provR
.dateApproved}" />
342 </div>
343 </div>
344 </c:if>
345 <c:if test = "${provR.editStatus == 1}">
346 <br />
347 <div class = "row">
348 <div class = "col-lg-2">
349 <label>Edited by:</label>
350 </div>
351 <div class = "col-lg-4">
352 <input type = "text" class = "form-control
" readonly = "true" value = "${provR.
editedBy}" />
353 </div>
354 <div class = "col-lg-2">
355 <label>Date Edited:</label>
356 </div>
357 <div class = "col-lg-3">
358 <input type = "text" class = "form-control
" readonly = "true" value = "${provR.
dateEdited}" />
359 </div>
360 </div>
361 </c:if>
362 </div>
363 </div>
364 <div class = "panel panel-info">
365 <div class = "panel-heading">
366 <label>Patient's Information</label>
367 </div>
368 <div class = "panel-body">
369 <c:if test = "${not empty provR.name}">
370 <div class = "row">
371 <div class = "col-lg-1">
372 <label>Name:</label>
373 </div>
374 <div class = "col-lg-5">
375 <input type = "text" class = "form-control"
readonly = "true" value = "${provR.name
}"/>
376 </div>
377 </div>
378 <br />
379 </c:if>
380 <c:if test = "${not empty provR.address}">
381 <div class = "row">
382 <div class = "col-lg-1">
383 <label>Complete Address:</label>
384 </div>
385 <div class = "col-lg-5">
386 <textarea class = "form-control" readonly =
"true" value = "${provR.address}">${provR.
address}</textarea>
387 </div>
388 </div>
389 <br />
390 </c:if>
391 <div class = "row">
392 <div class = "col-lg-1">
393 <label>Age:</label>
394 </div>
395 <div class = "col-lg-2">
396 <input type = "text" class = "form-control"
readonly = "true" value = "${provR.age}"/>
397 </div>
398 <div class = "col-lg-1">
399 <label>Date of Birth:</label>
400 </div>
401 <div class = "col-lg-3">
402 <input class="form-control" type="text"
readonly = "true" value="${provR.
dateofbirth}" />
403 </div>
404 <div class = "col-lg-1">
405 <label>Gender:</label>
406 </div>
407 <div class = "col-lg-3">
408 <input class = "form-control" type = "text"
readonly = "true" value = "${provR.gender
}"/>
409 </div>
410 </div>
411 </div>
412 </div>
413 <div class = "panel panel-info">
414 <div class = "panel-heading">
415 <label>Patient's Clinical Information</label>
416 </div>
417 <div class = "panel-body">
418 <div class = "row">
419 <div class = "col-lg-2">
420 <label>Disease:</label>
421 </div>

```

```

422 <div class = "col-lg-4">
423 <input type = "text" class = "form-control"
    readonly = "true" value = "${provR.disease
    }"/>
424 </div>
425 <div class = "col-lg-2">
426 <label>Date Onset:</label>
427 </div>
428 <div class = "col-lg-4">
429 <input class="form-control" type="text"
    readonly = "true" value="${provR.dateOnset
    }" />
430 </div>
431 </div>
432 <br />
433 <div class = "row">
434 <div class = "col-lg-2">
435 <label>Symptoms:</label>
436 </div>
437 <div class = "col-lg-4">
438 <input type = "text" class = "form-control"
    readonly = "true" value = "${provR.symptoms
    }"/>
439 </div>
440 <div class = "col-lg-2">
441 <label>Case Classification:</label>
442 </div>
443 <div class = "col-lg-4">
444 <input class="form-control" type="text"
    readonly = "true" value="${provR.caseType}"
    />
445 </div>
446 </div>
447 <br />
448 <c:if test = "${provR.admitted == 'No'}">
449 <div class = "row">
450 <div class = "col-lg-2">
451 <label>Patient Admitted?:</label>
452 </div>
453 <div class = "col-lg-4">
454 <input type = "text" class = "form-control"
    readonly = "true" value = "${provR.admitted
    }"/>
455 </div>
456 </div>
457 </c:if>
458 <c:if test = "${provR.admitted == 'Yes'}">
459 <div class = "row">
460 <div class = "col-lg-2">
461 <label>Patient Admitted?:</label>
462 </div>
463 <div class = "col-lg-4">
464 <input type = "text" class = "form-control"
    readonly = "true" value = "${provR.admitted
    }"/>
465 </div>
466 <div class = "col-lg-2">
467 <label>Date Admitted:</label>
468 </div>
469 <div class = "col-lg-4">
470 <input class="form-control" type="text"
    readonly = "true" value="${provR.
    dateAdmitted}" />
471 </div>
472 </div>
473 <br />
474 <div class = "row">
475 <div class = "col-lg-2">
476 <label>Hospital/Clinic:</label>
477 </div>
478 <div class = "col-lg-5">
479 <input type = "text" class = "form-control"
    readonly = "true" value = "${provR.hospital
    }"/>
480 </div>
481 <div class = "col-lg-2">
482 <label>Hospital/Clinic Type:</label>
483 </div>
484 <div class = "col-lg-3">
485 <input class="form-control" type="text"
    readonly = "true" value="${provR.
    hospitalType}" />
486 </div>
487 </div>
488 <br />
489 <div class = "row">
490 <div class = "col-lg-3">
491 <label>Hospital/Clinic Address:</label>
492 </div>
493 <div class = "col-lg-9">
494 <input type = "text" class = "form-control"
    readonly = "true" value = "${provR.
    hospitalAddr}"/>
495 </div>
496 </div>
497 </c:if>
498 <br />
499 <div class = "row">
500 <div class = "col-lg-2">
501 <label>Outcome</label>
502 </div>
503 <div class = "col-lg-4">
504 <input type = "text" class = "form-control"
    readonly = "true" value = "${provR.outcome
    }"/>
505 </div>
506 <c:if test = "${provR.outcome == 'Died'}">
507 <div class = "col-lg-2">
508 <label>Date Died:</label>
509 </div>
510 <div class = "col-lg-4">
511 <input type = "text" class = "form-control"
    readonly = "true" value = "${provR.dateDied
    }"/>
512 </div>
513 </c:if>
514 </div>
515 <br />
516 </div>
517 </div>
518 </div>
519 </div>
520 </div>
521 </div>
522 </c:forEach>
523 </c:forEach>
524 </c:if>
525 </c:forEach>
526 </c:if>
527 </c:forEach>
528 </c:forEach>
529 </c:forEach>
530 <c:forEach var = "i" begin = "0" end = "16">
531 <c:forEach var = "provs" items = "${prov[i]}"
    varStatus = "stat">
532 <c:forEach var = "user" items = "${allUsers}"
    varStatus = "st">
533 <c:if test = "${fn:toLowerCase(user.province) ==
    fn:toLowerCase(provs.province) && user.
    region == provs.region
    && (user.userType == 'Provincial Health Worker'
    || user.userType == 'Provincial Health
    Official')}">
534 <div class = "modal fade" id = "logs3_${i}_${
    stat.index}_${st.index}">
535 <div class = "modal-body">
536 <div class = "panel panel-info">
537 <div class = "panel-body">
538 <br />
539 <br />
540 <table class = "table table-striped">
541 <thead>
542 <tr>
543 <th>Date</th>
544 <th>Time</th>
545 <th>Action</th>
546 <th>view Record</th>
547 </tr>
548 </thead>
549 </tbody>
550 <tbody>
551 <c:forEach var = "logs" items = "${allLogs
    }" varStatus = "lgStat">
552 <c:if test = "${logs.recordtable == '
    muncityrecords'}">
553 <tr>
554 <td>${logs.date}</td>
555 <td>${logs.time}</td>
556 <td>${logs.action}</td>
557 <c:forEach var = "k" begin = "0" end = "${
    fn:length(provR) - 1}">
558 <c:forEach var = "provR" items = "${provR
    [k]}" varStatus = "prR">
559 <c:if test = "${logs.accountID == user.
    accID && logs.recordID == provR.recordID}">
560 <td><button class = "btn btn-primary"
    href = "#logsRecord2_${i}_${stat.index}_${
    st.index}_${k}_${prR.index}" data-toggle =
    "modal">View</button></td>
561 </c:if>
562 </c:forEach>
563 </c:forEach>
564 </tr>
565 </c:if>
566 </c:forEach>
567 </tbody>
568 </table>
569 </div>
570 </div>
571 </div>
572 </div>
573 </c:if>
574 </c:forEach>
575 </c:forEach>
576 </c:forEach>
577 </c:forEach>
578 </div>
579 </div>

```

Listing 62: national.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
  jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
  functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
  springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
  springframework.org/tags/form" %>
5
6 <div class = "container-fluid well-lg" style = "
  background-color: #eee;">
7   <div class = "panel panel-info">
8     <table class="table table-striped" id="
  example1">
9       <thead>
10        <tr>
11          <th>Name</th>
12          <th>User Type</th>
13          <th>Profile</th>
14          <th>User Logs</th>
15          <th>Account Status</th>
16          <th>Activate or Deactivate</th>
17        </tr>
18      </thead>
19      <tbody>
20        <c:forEach var = "userA" items = "${allUsers
  }" varStatus = "st">
21          <c:if test = "${userA.userType == 'National
  Health Official'}">
22            <tr>
23              <td>${userA.firstname} ${userA.middlename}
  ${userA.surname}</td>
24              <td>${userA.userType}</td>
25              <td><button class = "btn btn-primary" href
  = "#profile11_${st.index}" data-toggle = "
  modal">View</button></td>
26              <td><button class = "btn btn-primary" href
  = "#logs11_${st.index}" data-toggle = "
  modal">View</button></td>
27              <td><c:if test = "${userA.status == 1}">
  Active</c:if><c:if test = "${userA.status
  == 0}">Inactive</c:if></td>
28              <td><c:if test = "${userA.status == 0}"><
  button>Activate</button></c:if><c:if test =
  "${userA.status == 1}"><button>Deactivate
  </button></c:if></td>
29            </tr>
30          </c:if>
31        </c:forEach>
32      </tbody>
33    </table>
34  </div>
35  <c:forEach var = "user" items = "${allUsers}"
  varStatus = "st">
36    <c:if test = "${user.userType == 'National
  Health Official'}">
37      <div class = "modal fade" id = "profile11_${st.
  index}" data-width = "700">
38        <div class = "modal-body">
39          <div class = "panel panel-info">
40            <div class = "panel-body">
41              <br />
42              <table class = "table table-striped">
43                <tr>
44                  <td>First Name:</td>
45                  <td>
46                    ${user.firstname}
47                  </td>
48                </tr>
49                <tr>
50                  <td>Middle Name:</td>
51                  <td>
52                    ${user.middlename}
53                  </td>
54                </tr>
55                <tr>
56                  <td>Surname:</td>
57                  <td>
58                    ${user.surname}
59                  </td>
60                </tr>
61                <tr>
62                  <td>Region:</td>
63                  <td>
64                    ${user.region}
65                  </td>
66                </tr>
67                <tr>
68                  <td>Province:</td>
69                  <td>
70                    ${user.province}
71                  </td>
72                </tr>
73                <tr>
74                  <td>Institution:</td>
75                  <td>
76                    ${user.institution}
77                  </td>
78                </tr>
79                <tr>
80                  <td>Institution's Address:</td>
81                  <td>
82                    ${user.institutionAddr}
83                  </td>
84                </tr>
85                <tr>
86                  <td>Email:</td>
87                  <td>
88                    ${user.email}
89                  </td>
90                </tr>
91                <tr>
92                  <td>Contact Number:</td>
93                  <td>
94                    ${user.number}
95                  </td>
96                </tr>
97              </table>
98            </div>
99          </div>
100        </div>
101      </c:if>
102    </c:forEach>
103    <c:forEach var = "user" items = "${allUsers}"
  varStatus = "st">
104      <c:if test = "${user.userType == 'National
  Health Official'}">
105        <div class = "modal fade" id = "logs11_${st.
  index}" data-width = "700">
106          <div class = "modal-body">
107            <div class = "panel panel-info">
108              <div class = "panel-body">
109                <br />
110                <table class = "table table-striped">
111                  <thead>
112                    <tr>
113                      <th>Date</th>
114                      <th>Time</th>
115                      <th>Action</th>
116                      <th>View Record</th>
117                    </tr>
118                  </thead>
119                  <tbody>
120                    <c:forEach var = "logs" items = "${allLogs
  }" varStatus = "lgStat">
121                      <c:if test = "${logs.accountID == user.
  accID}">
122                        <tr>
123                          <td>${logs.date}</td>
124                          <td>${logs.time}</td>
125                          <td>${logs.action}</td>
126                          <td></td>
127                        </tr>
128                      </c:if>
129                    </c:forEach>
130                  </tbody>
131                </table>
132              </div>
133            </div>
134          </div>
135        </div>
136      </c:if>
137    </c:forEach>
138  </div>

```

Listing 63: researcher.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
  jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
  functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
  springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
  springframework.org/tags/form" %>
5
6 <div class = "container-fluid well-lg" style = "
  background-color: #eee;">
7   <div class = "panel panel-info">
8     <c:if test = "${not empty successD}">
9       <div class = "alert alert-success">
10        <strong>${successD}</strong>
11      </div>
12    </c:if>
13    <table class="table table-striped" id="example1
  ">
14      <thead>
15        <tr>
16          <th>Name</th>
17          <th>User Type</th>
18          <th>Profile</th>
19          <th>User Logs</th>
20          <th>Account Status</th>

```

```

21     <th>Activate or Deactivate</th>
22 </tr>
23 </thead>
24 <tbody>
25 <c:forEach var = "userA" items = "${allUsers
26     }" varStatus = "st">
27 <c:if test = "${userA.userType == 'Researcher
28     }'">
29 <tr>
30 <td>${userA.firstname} ${userA.middlename} $
31     {userA.surname}</td>
32 <td>${userA.userType}</td>
33 <td><button class = "btn btn-primary" href =
34     "#profile12_${st.index}" data-toggle = "
35     modal">View</button></td>
36 <td><button class = "btn btn-primary" href =
37     "#logs12_${st.index}" data-toggle = "modal
38     ">View</button></td>
39 <td><c:if test = "${userA.status == 1}">
40     Active</c:if><c:if test = "${userA.status
41     == 0}">Inactive</c:if></td>
42 <td><c:if test = "${userA.status == 0}"><
43     button class = "">Activate</button></c:if><
44     c:if test = "${userA.status == 1}"><button>
45     Deactivate</button></c:if></td>
46 </tr>
47 </c:if>
48 </c:forEach>
49 </tbody>
50 </table>
51 </div>
52 <c:forEach var = "user" items = "${allUsers}"
53     varStatus = "st">
54 <c:if test = "${user.userType == 'Researcher'}">
55 <div class = "modal fade" id = "profile12_${st.
56     index}" data-width = "700">
57 <div class = "modal-body">
58 <div class = "panel panel-info">
59 <div class = "panel-body">
60 <br />
61 <table class = "table table-striped">
62 <thead>
63 <tr>
64 <th>First Name:</th>
65 <td>
66     ${user.firstname}
67 </td>
68 </tr>
69 <tr>
70 <th>Middle Name:</th>
71 <td>
72     ${user.middlename}
73 </td>
74 </tr>
75 <tr>
76 <th>Surname:</th>
77 <td>
78     ${user.surname}
79 </td>
80 </tr>
81 <tr>
82 <th>Region:</th>
83 <td>
84     ${user.region}
85 </td>
86 </tr>
87 <tr>
88 <th>Province:</th>
89 <td>
90     ${user.province}
91 </td>
92 </tr>
93 </thead>
94 <tbody>
95 <c:forEach var = "logs" items = "${allLogs
96     }" varStatus = "lgStat">
97 <c:if test = "${logs.accountID == user.
98     accID}">
99 <tr>
100 <td>${logs.date}</td>
101 <td>${logs.time}</td>
102 <td>${logs.action}</td>
103 </tr>
104 </c:if>
105 </tbody>
106 </table>
107 </div>
108 </div>
109 </div>
110 </div>
111 </div>
112 <div class = "modal-body">
113 <div class = "panel panel-info">
114 <div class = "panel-body">
115 <br />
116 <table class = "table table-striped">
117 <thead>
118 <tr>
119 <th>Date</th>
120 <th>Time</th>
121 <th>Action</th>
122 <th>View Record</th>
123 </tr>
124 </thead>
125 <tbody>
126 <c:forEach var = "logs" items = "${allLogs
127     }" varStatus = "lgStat">
128 <c:if test = "${logs.accountID == user.
129     accID}">
130 <tr>
131 <td>${logs.date}</td>
132 <td>${logs.time}</td>
133 <td>${logs.action}</td>
134 </tr>
135 </c:if>
136 </tbody>
137 </table>
138 </div>
139 </div>
140 </div>
141 </div>
142 </div>
143 </div>

```

Listing 64: insertSuccess.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
2     jsp/jstl/core" %>
3 <%@ taglib uri="http://java.sun.com/jsp/jstl/
4     functions" prefix="fn" %>
5 <%@ taglib prefix="sec" uri="http://www.
6     springframework.org/security/tags" %>
7 <%@ taglib prefix="form" uri="http://www.
8     springframework.org/tags/form" %>
9 <jsp:include page = "../header.jsp" />
10 <div class = "tabbable">
11 <jsp:include page = "../linkings.jsp" />
12 <div class = "tab-content">
13 <sec:authorize access="hasAnyRole('
14     ROLE_BHWorker')">
15 <div class = "panel-heading">Insertion of
16     Records</div>
17 <div class = "panel-body">
18     Insertion successful!
19 <br />
20 <a href = "%= request.getContextPath()
21     %>/pages/system"><button type = "button"
22     class = "btn btn-primary">Add New Records</
23     button></a>
24 </div>
25 </div>
26 </div>
27 </div>
28 </div>
29 <jsp:include page = "../footer.jsp" />

```

Listing 65: changeContactsError.jsp 30

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
  jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
  functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
  springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
  springframework.org/tags/form" %>
5 <jsp:include page = "../header.jsp" />
6
7 <div class = "tabbable">
8 <jsp:include page = "../linkings.jsp" />
9 <div class = "tab-content">
10 <div class = "tab-pane active">
11 <br />
12 <div class = "container-fluid well-lg" style
  = "background-color: #eee;">
13 <div class = "panel panel-info" id = "
  changeUP">
14 <div class = "modal-dialog" style = "
  background-color: #fff;">
15 <div class = "modal-body panel panel-info
  ">
16 <div class = "panel panel-heading">
17 <h4 class = "form-signin-head">Vector
  Borne Diseases Monitoring System</h4>
18 </div>
19 <form:form method="post" class = "form-
  signin" action = "${pageContext.request.
  contextPath}/pages/changeUserContacts"
  modelAttribute = "changeContacts">
20 <table class = "table" >
21 <tr>
22 <td style = "border: none;">
23 <label for="login_password">Email:</
  label>
24 </td>
25 <td style = "border: none;">
26 <form:input path = "email" type="text"
  class="form-control" />
27 <c:set var = "eEmail"><form:errors
  path = "email" /></c:set>
28 <c:if test = "${not empty eEmail}">
29 <div class = "alert alert-danger alert
  -dismissable">

```

```

  <button type="button" class="close"
  data-dismiss="alert" aria-hidden="true">&
  times;</button>
  <strong><form:errors path = "email
  "></form:errors></strong>
  </div>
  </c:if>
  </td>
  </tr>
  <tr>
  <td style = "border: none;">
  <label for="login_password">Contact
  Number:</label>
  </td>
  <td style = "border: none;">
  <form:input path = "number" type="text"
  class="form-control" />
  <c:set var = "eNumber"><form:errors
  path = "number" /></c:set>
  <c:if test = "${not empty eNumber}">
  <div class = "alert alert-danger alert-
  dismissable">
  <button type="button" class="close"
  data-dismiss="alert" aria-hidden="true">&
  times;</button>
  <strong><form:errors path = "number
  "></form:errors></strong>
  </div>
  </c:if>
  </td>
  </tr>
  <tr>
  <td style = "border: none;"><button class
  = "btn btn-primary" type="submit" value =
  "submit">Change</button></td>
  </tr>
  </table>
  </form:form>
  </div>
  </div>
  </div>
  </div>
  <jsp:include page = "../footer.jsp" />

```

Listing 66: changeCredentialsError.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
  jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
  functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
  springframework.org/security/tags" %>
4 <%@ taglib prefix="form" uri="http://www.
  springframework.org/tags/form" %>
5 <jsp:include page = "../header.jsp" />
6
7 <div class = "tabbable">
8 <jsp:include page = "../linkings.jsp" />
9 <div class = "tab-content">
10 <div class = "tab-pane active">
11 <br />
12 <div class = "container-fluid well-lg" style
  = "background-color: #eee;">
13 <div class = "panel panel-info" id = "
  changeUP">
14 <div class = "modal-dialog" style = "
  background-color: #fff;">
15 <div class = "modal-body panel panel-info
  ">
16 <div class = "panel panel-heading">
17 <h4 class = "form-signin-head">Vector
  Borne Diseases Monitoring System</h4>
18 </div>
19 <form:form method="post" class = "form-
  signin" action = "${pageContext.request.
  contextPath}/pages/changeUserDetails"
  modelAttribute = "changeUser">
20 <table class = "table" >
21 <tr>
22 <td style = "border: none;">
23 <label for="login_password">Old
  Password:</label>
24 </td>
25 <td style = "border: none;">
26 <form:input path = "accountPassword"
  type="password" class="form-control" title
  ="Password is required" />
27 <c:set var = "eAccountPassword"><form:
  errors path = "accountPassword" /></c:set>
28 <c:if test = "${not empty
  eAccountPassword}">
29 <div class = "alert alert-danger alert
  -dismissable">
  <button type="button" class="close"
  data-dismiss="alert" aria-hidden="true">&
  times;</button>
  <strong><form:errors path = "
  accountPassword"></form:errors></strong>
  </div>
  </c:if>
  </td>
  </tr>
  <tr>
  <td style = "border: none;">
  <label for="login_password">Re-enter New
  Password:</label>
  </td>
  <td style = "border: none;">
  <form:input path = "accountPassword2"
  type="password" id = "oldPassword" class="
  form-control" title="Password is required"
  />
  <c:set var = "eAccountPassword2"><form:
  errors path = "accountPassword2" /></c:set>
  <c:if test = "${not empty
  eAccountPassword2}">
  <div class = "alert alert-danger alert-
  dismissable">
  <button type="button" class="close"
  data-dismiss="alert" aria-hidden="true">&
  times;</button>
  <strong><form:errors path = "
  accountPassword2"></form:errors></strong>
  </div>
  </c:if>
  </td>
  </tr>
  <tr>
  <td style = "border: none;">
  <label for="login_password">Re-enter New
  Password:</label>
  </td>
  <td style = "border: none;">
  <form:input path = "accountPassword3"
  type="password" id = "newPassword" class="
  form-control" title="Password is required"
  />
  <c:set var = "eAccountPassword3"><form:
  errors path = "accountPassword3" /></c:set>
  <c:if test = "${not empty
  eAccountPassword3}">
  <div class = "alert alert-danger alert-
  dismissable">
  <button type="button" class="close"

```

```

        data-dismiss="alert" aria-hidden="true">&
times;</button>
61 <strong><form:errors path = "
        accountPassword3"></form:errors></strong>
62 </div>
63 </c:if>
64 </td>
65 </tr>
66 <c:if test = "${not empty error || not
        empty error2}">
67 <tr colspan = '6'>
68 <td style = "border: none;" colspan =
        '2'>
69 <div id = "confirmPass" class = "alert
        alert-danger">
70 <strong>${error}</strong>
71 <br />
72 <strong>${error2}</strong>
73 </div>
74 </td>
75 </tr>
76 </c:if>
77 <tr>
78 <td style = "border: none;"><button class =
        "btn btn-primary" type="submit" value = "
        submit">Change</button></td>
79 </tr>
80 </table>
81 </form:form>
82 </div>
83 </div>
84 </div>
85 </div>
86 </div>
87 </div>
88 </div>
89 </div>
90 <jsp:include page = "../..../footer.jsp" />

```

Listing 67: userDeactivated.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
        jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
        functions" prefix="fn" %>
3 <%@ taglib prefix="sec" uri="http://www.
        springframework.org/security/tags" %>
4 <!DOCTYPE html>
5 <html>
6 <head>
7 <title>Vector-Borne Diseases Monitoring System
        </title>
8 <meta name="viewport" content="width=device-
        width, initial-scale=1.0">
9 <jsp:include page="../..../css.jsp"/>
10 <meta http-equiv="refresh" content="5;<%=
        request.getContextPath() %>/
        j_spring_security_logout" />
11 </head>
12 <body style = "background: #f777;">
13 <div class = "container">
14 <div class = "navbar navbar-default" role = "
        navigation">
15 <div class = "container-fluid">
16 <div class = "navbar-header">
17 <button type = "button" class = "navbar-
        toggle" data-toggle = "collapse" data-
        target = ".navbar-collapse">
18 <span class = "sr-only">Toggle Navigation
        </span>
19 <span class = "icon-bar"></span>
20 <span class = "icon-bar"></span>
21 <span class = "icon-bar"></span>
22 </div>
23 <div class = "modal-dialog" style = "
        background-color: #fff;">
24 <div class = "modal-body panel panel-info">
25 <div class = "panel panel-heading">
26 <h4 class = "form-signin-head">Vector Borne
        Diseases Monitoring System</h4>
27 </div>
28 <div class = "alert alert-danger">
29 <center>Sorry, your account was deactivated
        . Please contact vbdsdev@gmail.com for
        information
        on account reactivation. Thank you, good
        day!</center>
30 </div>
31 </div>
32 <div class = "modal-dialog" style = "
        background-color: #fff;">
33 <div class = "modal-body panel panel-info">
34 <div class = "panel panel-heading">
35 <h4 class = "form-signin-head">Vector Borne
        Diseases Monitoring System</h4>
36 </div>
37 <div class = "alert alert-danger">
38 <center>Attention: You will be logged out
        in 5 seconds.</center>
39 </div>
40 </div>
41 </div>
42 </div>
43 </div>
44 <jsp:include page = "../..../footer.jsp" />

```

Listing 68: loginfailed.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
        jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
        functions" prefix="fn" %>
3 <%@ taglib prefix="sec" uri="http://www.
        springframework.org/security/tags" %>
4 <!DOCTYPE html>
5 <html>
6 <head>
7 <title>Vector-Borne Diseases Monitoring System
        </title>
8 <meta name="viewport" content="width=device-
        width, initial-scale=1.0">
9 <jsp:include page="css.jsp"/>
10 </head>
11 <body style = "background: #f777;">
12 <div class = "container">
13 <div class = "navbar navbar-default" role = "
        navigation">
14 <div class = "container-fluid">
15 <div class = "navbar-header">
16 <button type = "button" class = "navbar-
        toggle" data-toggle = "collapse" data-
        target = ".navbar-collapse">
17 <span class = "sr-only">Toggle Navigation
        </span>
18 <span class = "icon-bar"></span>
19 <span class = "icon-bar"></span>
20 <span class = "icon-bar"></span>
21 </button>
22 <a href = "<%= request.getContextPath() %>/
        home" class = "navbar-brand">VBDMS</a>
23 </div>
24 </div>
25 </div>
26 </div>
27 <div class = "modal-dialog" style = "
        background-color: #fff;">
28 <div class = "modal-body panel panel-info">
29 <div class = "panel panel-heading">
30 <h4 class = "form-signin-head">Vector Borne
        Diseases Monitoring System</h4>
31 </div>
32 <div class = "form-signin">
33 <form method="post" class = "form-signin">
34 <div class = "alert alert-danger">
35 <center>Your login credentials didn't
        match! Please try again.</center>
36 </div>
37 <table class = "table" >
38 <tr>
39 <td>
40 <label for="login_username">Username</
        label>
41 <input type="text" name="j_username" id
        ="login_username" class="form-control field
        required" title="Please provide your
        username" />
42 </td>
43 </tr>
44 <tr>
45 <td>
46 <label for="login_password">Password</
        label>
47 <input type="password" name="j_password"
        id="login_password" class="form-control
        field required" title="Password is required
        " />
48 </td>
49 </tr>
50 <tr>
51 <td><button class = "btn btn-primary"
        type="submit" value = "submit">Log in</
        button></td>
52 <td><a href = "<%= request.getContextPath()
        %>/forgotPassword" class = "pull-right">
        Forgot Password?</a></td>
53 </tr>
54 </table>
55 </form>
56 </div>
57 </div>
58 </div>
59 </div>
60 </div>
61 </div>
62 <jsp:include page = "footer.jsp" />

```

Listing 69: generic_error.jsp

```

1 <%@ taglib prefix="c" uri="http://java.sun.com/
  jsp/jstl/core" %>
2 <%@ taglib uri="http://java.sun.com/jsp/jstl/
  functions" prefix="fn"%>
3 <%@ taglib prefix="sec" uri="http://www.
  springframework.org/security/tags" %>
4 <jsp:include page="header.jsp"/>
5 <div class="jumbotron">
6   <c:if test="${not empty exception.errCode}">
7     <h1>${exception.errCode} : System Errors</h1>
8   </c:if>
9
10  <c:if test="${empty exception.errCode}">
11  <div class = "panel panel-info">
12    <div class = "panel-heading"><h4>System Errors
      </h4></div>
13    <div class = "panel-body">
14      <div class = "alert alert-info">
15        <p class = "text-left">
16          <small><em><strong>Error: </strong>${
      exception.errCode}</em></small>
17        </p>
18      </div>
19      <p>
20        Please contact system administrator by
      sending an email to <strong><a href = "
      vbdmsdev@gmail.com">vbdmsdev@gmail.com</a
      ></strong> with the error message. Thank
      you!
21      </p>
22    </div>
23  </div>
24  </c:if>
25
26  <c:if test="${not empty exception.errMsg}">
27    <h4>${exception.errMsg}</h4>
28  </c:if>
29  </div>
30  </div>
31  <jsp:include page="footer.jsp"/>

```

XII. Acknowledgement

Unang-una sa lahat, babatiin ko muna ang sarili ko. :) Yehey! Natapos ko ang SP ko, magtatapos na rin ako. Maraming salamat sa aking sarili dahil hindi ka ganoong tinamad para tapusin ang SP mo. :)

Nagpapasalamat rin ako sa adviser ko, Doc Magboo! Thank you po! Maraming salamat sa medyo medyong pang-aaway niyo sa'kin at pagtitiyaga niyo kahit na minsan paulit-ulit po yung mga mali ko. XD Thank you rin po sa pagtatanggol at pagtulong sa'kin sa defense! Maraming maraming salamat po talaga, the best kayo Doc!:) God bless po sa inyo :)

Syempre, thank you rin sa mga magulang ko :) Thank you Ma, Pa, lalo na sa financial support, na binibigyan niyo po ako kahit hindi ako humihingi. Sorry po kung di ko kayo inuupdate tungkol sa SP ko, ayaw ko lang talaga. Hoho. Thank you po ulit, I love you both. :D Cheesy, di niyo po kasi mababasa 'to. LOL

Tita Mildred, maraming, maraming salamat po sa pagpapaaral sa'kin. :) Kung may inexpect man po kayo sa'kin, sorry po kung di ko nakamit, di ko po talaga kaya. Haha. Ang bait bait niyo po talaga Tita, kaya kayo pinaka-favorite kong Tita eh. :) Para sa inyo po tong paggraduate ko, thank you po talaga, I love you po! :)

Sa mga kaibigan ko ngayong college, Alyra, Ninang Celina, Geraline, at Sarah. Jezra, Liezl, Maris, Nicole, Nay Rheabedette. Pati na rin sa lahat ng Block 12 at ComSci Batch 2010, thank you guys! Mamimiss ko kayo :'(Hahaha. Joke lang. Btw, jokes are half meant right? :)

Thank you rin sa mga ka-chat ko gabi-gabi, Ate Mich, Ate Hainah, Alyra, Zee, Troy. XD

And of course, thank you God! For guiding me since my first day in college until this very day, for all the oppurtunities you laid out to me, for giving me so much blessings. For all the hardships you put me through, for always being there with me even when I doubt you some times, for giving me this wonderful life to live, thank you so much! I wouldn't be here with my strengths alone, and you

helped me become what I am today, thank you. :)