University of the Philippines Manila
College of Arts and Sciences
Department of Physical Sciences and Mathematics

# Schistosomiasis Snail Surveillance and Human Prevalence Geographic Information System

A special problem in partial fulfillment

of the requirements for the degree of

**Bachelor of Science in Computer Science**

Submitted by:

Gerald M. Goco

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 "Schistosomiasis Snail Surveillance and Human Prevalence Geographic Information System" prepared and submitted by Gerald M. Goco in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science has been examined and is recommended for acceptance.

**Geoffrey A. Solano, M.Sc.**
Adviser

**EXAMINERS:**

| | **Approved** | **Disapproved** |
|---|---|---|
| 1. Gregorio B. Baes, Ph.D. (*candidate*) | | |
| 2. Avegail D. Carpio, M.Sc. | | |
| 3. Aldrich Colin K. Co, M.Sc. (*candidate*) | | |
| 4. Ma. Sheila A. Magboo, M.Sc. | | |
| 5. Vincent Peter C. Magboo, M.D., M.Sc. | | |
| 6. Richard Bryann L. Chua, M.Sc. | | |
| 7. Bernie B. Terrado, M.Sc. (*candidate*) | | |

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

Schistosomiasis is a water-borne parasitic disease endemic in tropical areas such as the Philippines. Its transmission is dependent on the presence of intermediate host snails. This study proposes a GIS that visually maps schistosomiasis prevalence in the Philippines by plotting infected snails and human cases by allowing field workers to input snail sample information and by integrating with the DOH schistosomiasis system.

*Keywords*: epidemiology, vector-borne disease, thematic map, schistosomiasis, snail surveillance

# Contents

# List of Figures

# List of Tables

# I. Introduction

## A. Background of the Study

Schistosomiasis is a chronic disease common in tropical areas. It is caused by parasitic flatworms of the genus *Schistosoma* and is endemic in several countries including the Philippines. Out of 17 regions, it is endemic in 12, covering 28 provinces, 190 municipalities and 20 cities in the country. This includes the islands of Mindanao, Samar and Leyte. Despite the progress made in controlling morbidity in these areas, through mass chemotherapy with praziquantel, more places continue to be identified as schistosomiasis endemic areas as indigenous cases and infected *Oncomelania hupensis quadrasi* snail vectors are identified [2, 3, 4].

This species of freshwater snails that act as an intermediate host for the parasite to grow into a cercariae, its infective free-swimming larval stage, is an essential element in the transmission process of schistosomiasis [5]. Prior to the introduction of mass chemotherapy in the Philippines, control strategies were aimed at reducing the number of intermediate host snails by limiting possible exposure to the cercariae by means of manipulating the environment and administering molluscicides. Although these measures have made some significant changes, they were not impressive enough to remarkably reduce the national prevalence of the disease and they were found to be very expensive [6].

Proper temperature and sufficient daylight governs the maturation of the parasite in the snail. After emerging from the snail host, it moves towards the surface, contaminating the water [5]. The parasite may then burrow into a persons skin through contact with the contaminated water [7]. Because the transmission of schistosomiasis is determined by environmental sanitation, access to safe water and the interaction of people during recreational or occupational activities with bodies of water inhabited by host snails, this disease can thus be considered as an environmental disease [3, 8]. Accordingly, having a better understanding of the environmental dynamics that permit schistosomiasis transmission is vital to controlling, and eventually eliminating, morbidity.

With the use of malacological surveillance on snail vectors, geographical distribution and vector density data can be collected to plan and monitor control programmes and make appropriate and timely decisions in interventions [9]. Providing information on which areas have a great concentration of intermediate host snails helps in targeting resources to those areas. Subsequent to the implementation of water and sanitation interventions, a surveillance system can be used to evaluate the efficacy in reducing the disease [10].

Risk mapping is beneficial to achieve this goal since the survival and growth of the parasite within the host and the snails themselves are greatly affected by environmental factors such as vegetation, temperature, land use, and water collections, among others. Schistosomiasis is a disease determined by spatial and temporal factors which can be characterized using satellite imagery and spatial analysis.

Geographic Information Systems (GIS) are systems designed to perform a wide range of data manipulation, including analysis, modelling, storage, and visualization among many other operations, on geospatial information. In 2005, when Google released Google Earth and Google Maps, which included an Application Programming Interface (API) for both, it became possible for anyone to interact with detailed geospatial data and tools and combine it with their own data, even with little computing skills. As it became more popular, it had also expanded the use of GIS in visualizing geographic data [11].

The role of technology in providing data in an easy to analyse form is important in the control and elimination of schistosomiasis. This includes risk mapping using a GIS. With the use of GIS and remote sensing (RS) technology, it is possible to relate the concentration of hosts, along with other risk factors, to the spatial distribution of the cases. Analysis of the relation of prevalence of infection in humans to concentration of vectors and environmental conditions using GIS and RS is a growing research area and is potentially becoming a necessity in schistosomiasis studies.

In this study, GIS technology will be used to identify the geographic location of host snails and its relation to schistosomiasis prevalence in humans. Data on schistosomiasis cases will be shown using a thematic map while vector snails are overlaid using markers.

## B. Statement of the Problem

Visualization is desirable as a supporting material to be used in schistosomiasis studies by the Department of Health (DOH). Currently, snail surveillance is done manually and there is no system that consolidates vector data and schistosomiasis cases in endemic areas from the DOH Schistosomiasis Surveillance Information System in a thematic map. Field workers bring their GPS devices to snail sites and write information on forms. While there is a system that stores and displays data on schistosomiasis cases, it does not have the capability of mapping these data. There is a need for a tool that:

1. Automates data entry of snail sites and samples.

2. Electronically handles the validation of snail sites and samples.

3. Visually correlates snail infection with human cases.

4. Allows users of the current DOH Schistosomiasis System to view visual data on a map.

## C. Objectives of the Study

This study generally aims to develop a GIS that visually maps schistosomiasis prevalence in the Philippines by plotting infected snails and human cases. This system is for the use of field workers, government officials and the surveillance data manager. The system will have the following capabilities:

1. Allows the system administrator to:

   (a) Assign/Change user roles

   (b) Enable/Disable user access to the system

2. Allows data manager to:

   (a) View snail sites and sample information

   (b) Mark snail sites and sample information as “queued” or “final” data

   (c) Import/Export all snail sites and sample information

   (d) View online map with layer of queued or final schistosomiasis cases

   (e) View online map with layer of queued or final snail information

3. Allows field workers to:

   (a) Add/Edit/View snail site information

   (b) Add/Edit/View snail sample information

   (c) View online map with layer of queued or final schistosomiasis cases

   (d) View online map with layer of queued or final snail information

4. Allows government officials (A government official can be from Barangay Health Station (BHS), Rural Health Unit (RHU), Provincial Health Ofce (PHO), Center for Health Development (CHD), Department of Health (DOH), Hospitals (HOS))to:

   (a) View snail site information

   (b) View snail sample information

   (c) View online map with layer of queued or final schistosomiasis cases

   (d) View online map with layer of queued or final snail information

## D. Significance of the Project

Schistosomiasis distribution is influenced by the presence and concentration of intermediate snail hosts and other environmental factors. Planning actions to control and eliminate morbidity relies on understanding these pieces of information. Presenting a visual representation of data using a GIS is essential in data analysis since it gives a clearer view of the relationship between cases and vectors. With the use of GIS, spatial patterns can be examined and can give important information to the underlying disease process. Also, a GIS can help in identifying other likely sites of infection thus preventing an outbreak before its onset. Thus, the implementation of a schistosomiasis snail surveillance and human prevalence GIS as a tool for data entry of snail surveillance as well as for visualizing vectors and cases is beneficial to DOH.

Furthermore, field workers currently use a GPS device for recording the location of vectors in snail surveillance. Having access to the system on their own GPS-enabled mobile device can eliminate the need for a separate GPS device. Ultimately, this project should have an impact that leads to the elimination of schistosomiasis in the Philippines.

## E. Scope and Limitations

1. Only those who have user accounts can access the system.

2. The device running the system must allow GPS access in order to enter colony location automatically. Otherwise, coordinates must be entered manually.

3. Statistics on human cases in the system is limited to the data exported from the schistosomiasis information system, thus, assessing the validity of said data is not included in this system.

4. The fields of the human prevalence data that can be imported into the system must be similar to the format specified in this paper.

5. The system will be accessible online and offline. However, the raster map from Google Maps will only be available online.

6. The system will have an interface to input snail vector data but schistosomiasis cases will be limited only to the imported data from the Schistosomiasis Information System database.

7. The validation and further analysis of data are tasks of the user since the system will only serve as a supporting tool.

## F. Assumptions

1. The Department of Health will provide the shape files to which the data will be overlaid to.

2. The operating system of the device used to setup the offline system is Windows running a XAMPP Server.

# II. Review of Related Literature

## A. GIS in Health Informatics

GIS's have traditionally been used in managing land and natural resources but in recent studies, GIS technologies have served to advance the area of health informatics. Specifically, these tools have been of use in providing geographic data in epidemiological research which requires spatial analysis. It is evidently becoming beneficial in healthcare delivery by providing digital cartography of disease patterns and outbreaks and its relation to geographic differences.

Asif et al. [12] developed web-GIS based decision support system for Public Health Surveillance which was aimed to provide timely access to disease related data and its geographic location. The study included two kinds of data collection systems, one that is desktop based and another that is mobile based. It was able to provide near real time health surveillance through maps and charts.

The use of GIS technology in identifying geographical areas and subpopulations at increased risk for Pneumocystis carinii pneumonia (PCP) was studied by Arno et al. [13] in New York. The data used was hospital discharge records and AIDS surveillance and was retrieved from New York State Department of Health. They were able to show that the PCP index, defined as the number of PCP-related hospitalizations divided by the number of living persons with AIDS, had no statistical correlation with AIDS prevalence rate. This implies that the high PCP index values found and high AIDS prevalence may indicate poor access to health care services. It also suggest that the index provides another level of information other than that reflected by AIDS case density. The mapping of the geographical distribution of PCP index values could make it possible to locate specific risk areas making it valuable for targeting prevention and delivering service in the attempt to reduce morbidity and mortality in infected persons.

Bailey et al. [14] showed how GIS can aid health planners in making informed decisions in interventions to strengthen the emergency referral system of maternal and newborn health. The data set included 797 geo-coded health facilities across two regions in Ethiopia. They have found that around 70% of the population are within a two hour transfer time to a hospital with obstetric surgery which can be increased to 83% if vehicles and communication capabilities are improved. They have observed that GIS mapping of health data is critical to understanding and strengthening the populations access to emergency services.

A study by Abbas et al. [15] investigated the geographic distribution of health care facilities in Kaduna, Nigeria. Global Positioning System (GPS) devices were used to collect the location of health care centres which were then incorporated into a GIS environment. They have found that private centres were concentrated at the north eastern part of the study area. These results were used to propose sites for new health centres, making GIS a valuable tool for site selection of new health centres.

Another study in Nigeria by Kibon and Ahmed [16] examined the relevance of GIS in supporting health planners by mapping the distribution of health care facilities in Kano. Their recommendation is for interdisciplinary scientists to be involved in managing GIS technologies in health.

A study by Kistemann et al. [17] used GIS to study the water-supply structure (WSS) in an area Germany. This study proved to be significant since microbial contamination of drinking water may occur due to bio-films, cracks, animals and sewer leakage in regions with a dense population and highly developed drinking-water supply. As part of an incident plan or outbreak management approach, it is important to prepare GIS facilities. Since water supply infrastructure in densely populated areas is often complicated, it had been helpful to use GIS in calculating distances of features for selected locations. They were able to prove that having the WSS-GIS module is a useful support to decision making in emerging water-borne diseases.

These studies have shown how GIS technologies have been useful in finding patterns in the spatial and temporal aspects of health problems. They have shown that given a reliable data set, a GIS is a valuable decision support in providing healthcare and managing the surveillance of diseases.

## B. GIS in the Study of Tropical Vector-borne Diseases

In the past years, many studies have been made in the advancement of GIS and RS applications in tropical vector-borne disease mapping, including mapping of cases of humans infected with schistosomiasis in various countries. GIS has been used in defining not only the condition but also the location of vector-borne diseases to analyse temporal trends and investigate spatial patterns. These studies also include several reviews that highlight the potential of GIS and RS in spatial analysis targeting the study and control of schistosomiasis.

A GIS-based retrospective study using spatial technology in India by Saxena et al. [18] was initiated using epidemiological data within a three year period for malaria cases. They aimed to identify spatial distribution patterns of two types of malaria, namely Plasodium vivax and Plasodium falciparum. They have found that the first type was randomly distributed while the second was significantly clustered. This was shown by mapping hot spots over the three year period.

To support malaria elimination, Kelly et al. [19] developed a GIS-based spatial decision support system (SDSS) to automatically locate and map the geospatial distribution of malaria cases as a guide in elimination zones. One of the factors that have affected their system is the lack of timely and accurate data. Although they have shown that GIS queries can support the identification and mapping of malaria cases in a detailed level and detect areas which need to be prioritizes, its power is limited by the available data.

Vazquez-Prokopec et al. [20] used GIS in order to quantify the geography of the spread of dengue virus in a 25-week epidemic period. In their visualization, they have shown that 95% of the cases were within a 200 m radius of the cases during the first wave of transmission.

In a related study, Chu et al. [21] proposed integrating GIS into planning insecticide spraying for preventing dengue fever transmission. GIS was used to map the number of cases and the areas where spraying had been done. However, like the preceding study, the factors that were considered in which areas needed insecticide spraying were based on the frequency, duration and intensity of cases in the area and did not include vector surveillance.

On the other hand, Opisa et al. [22] mapped malacological survey sites using a GIS to determine whether active transmission was occurring in settlements with the presence of intermediate host snails. They concluded, with the help of mapping, that transmission patterns are closely related to the concentration and spatial distribution of vector snails.

A study in Brazil by Gomes et al. [23] also used malacological surveys as well as prevalence surveys as data mapped onto satellite images in spatial and temporal analysis of the transmission and environmental factors in schistosomiasis. The satellite images and land use maps helped them quantify the process of occupation of the area and the shifts in the area at risk of schistosomiasis within the given time. They were able to identify the reduction in the areas of permeable ground and vegetation as one of the factors that highlight environmental degradation in the locality and thus favouring the dissemination of schistosomiasis.

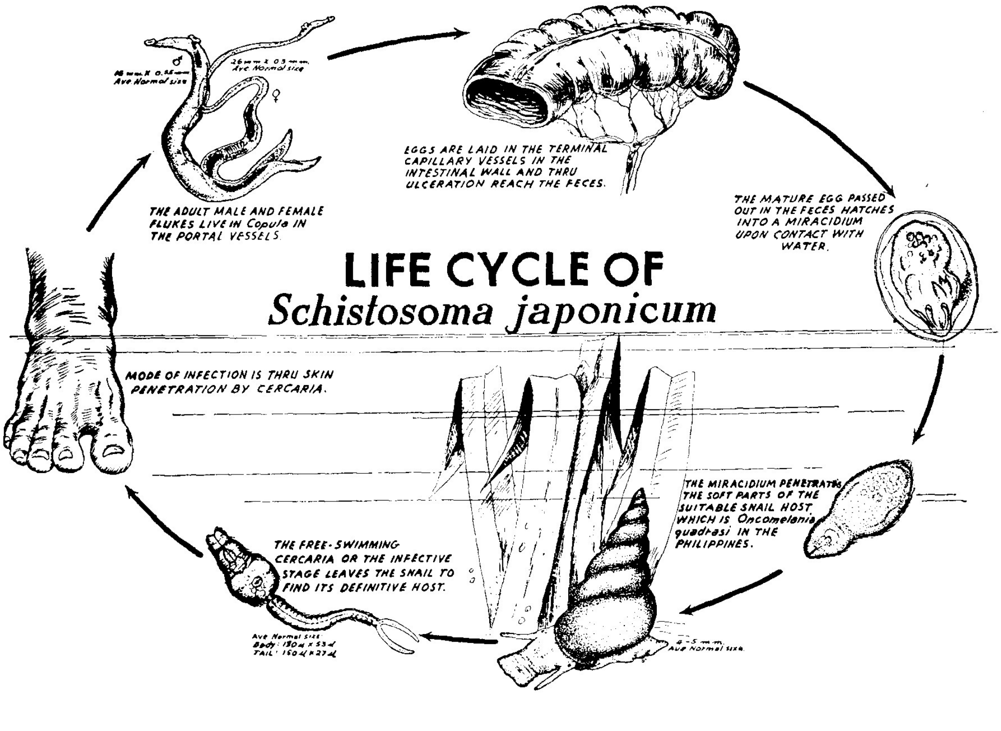
Overall, GIS has been a useful support system for managing vector-borne diseases. Adding the power of GIS in processing data related to vector-borne diseases is valuable in activities that are directed at the control and elimination of such diseases. These studies have established the role of GIS in the control and elimination of tropical vector-borne diseases as it provided readily comprehensible results to aid decision-making.

# III. Theoretical Framework

## A. Schistosomiasis

Symptoms vary in persons infected with schistosomiasis depending on the extent of the infestation. A person with heavy infestation may also experience fever, chills, lymph node enlargement, and liver and spleen enlargement aside from other symptoms such as itching, abdominal pain and diarrhoea, frequent and painful urination and blood in the urine [7].

The disease remains to be a public health burden in endemic provinces as it causes 280,000 deaths per year. As part of their health-related Millennium Development Goals (MDGs), a set of strategic actions to be accomplished by 2015, the Department of Health (DOH) aims to control and eliminate schistosomiasis as a public health problem in all endemic provinces. This goal will be considered to be achieved when the prevalence rate, the proportion of infected population to the total sample population, is reduced to less than one percent for at least five consecutive years [3].



♂
16 mm. X 0.55 mm. Ave. Normal size
26 mm. X 0.3 mm. Ave. Normal size
♀
THE ADULT MALE AND FEMALE FLUKES LIVE IN Copula IN THE PORTAL VESSELS.
EGGS ARE LAID IN THE TERMINAL CAPILLARY VESSELS IN THE INTESTINAL WALL AND THRU ULCERATION REACH THE FECES.
THE MATURE EGG PASSED OUT IN THE FECES HATCHES INTO A MIRACIDIUM UPON CONTACT WITH WATER.
LIFE CYCLE OF
Schistosoma japonicum
MODE OF INFECTION IS THRU SKIN PENETRATION BY CERCARIA.
THE MIRACIDIUM PENETRATES THE SOFT PARTS OF THE SUITABLE SNAIL HOST WHICH IS Oncomelania quadrasi IN THE PHILIPPINES.
THE FREE-SWIMMING CERCARIA OR THE INFECTIVE STAGE LEAVES THE SNAIL TO FIND ITS DEFINITIVE HOST.
Ave Normal size: Body: 130μ x 53μ TAIL: 150μ x 27μ
4-5 mm. Ave Normal size.

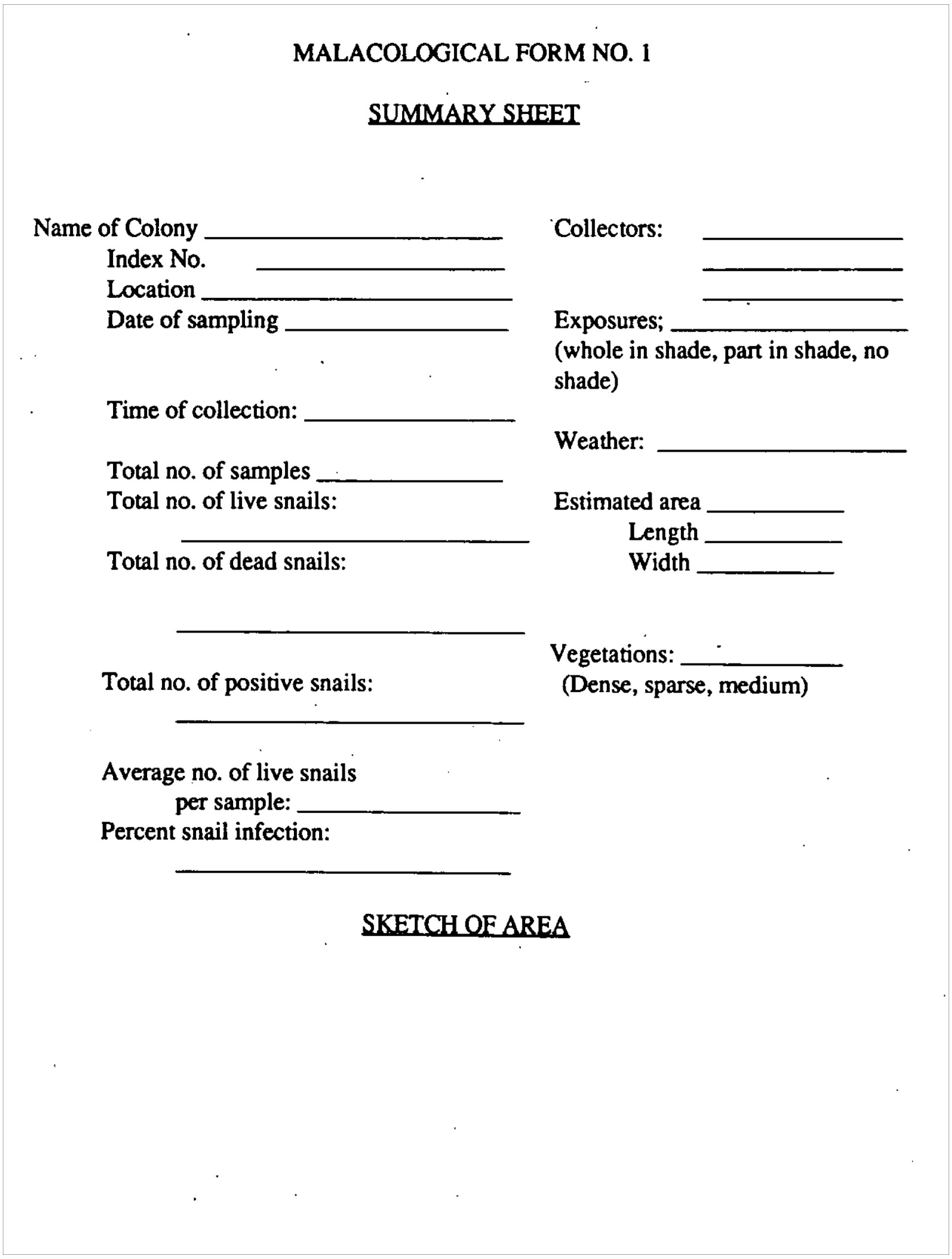
Figure 1: Life Cycle of *Schistosoma Japonicum* [1]

## B. Snail Sampling

Since schistosomiasis is a vector-borne disease transmitted to humans when they become in contact with contaminated water and since stagnating surface waters are the preferred habitat of parasitic worms, anthropogenic activities can heavily impact the transmission of this disease. Tubangui in 1932, as cited by Pesigan et al. [24], first identified *Oncomelania hupensis quadrasi* as the intermediate snail host of *Schistosoma* japonicum in the Philippines after making observations in Palo, Leyte. These vectors are commonly found in wet places such as swamps, rice fields, creeks and borrow-pits.

The presence of snail intermediate hosts is a major factor responsible for the occurrence of the disease [6]. Thus, among other factors, it is important to watch over contaminated waters for infected snails in the control and elimination of schistosomiasis.

Snail surveillance includes the estimated measurement of the area in square meters and the snail density per ring sample. To evaluate the efficacy of snail control efforts, it is necessary to compare population density before and after the control. Snail surveys by sampling are conducted annually on colonies which are subjected to control measures until snails are eradicated. Sampling aims to determine the characteristic of the snail population without the need to examine each individual snail. There are several ways to collect snail samples such as random collection using metal rings or tubes, or using snail attractants. Among these methods, tube sampling is considered to be most effective. This method is accomplished by pushing a brass tube, 13.5 cm in diameter, into the mud and collecting the snails from the mud sample. The data gathered, recorded in a form shown in Figure 2, includes the location, date and time of collection, total number of samples taken, total number of infected snails, and average number of live and dead snails collected [1].



MALACOLOGICAL FORM NO. 1

SUMMARY SHEET

Name of Colony ____________________
Index No. ________________
Location ____________________
Date of sampling ______________

Time of collection: ______________

Total no. of samples ___________
Total no. of live snails:
____________________
Total no. of dead snails:

____________________

Total no. of positive snails:
____________________

Average no. of live snails
per sample: ______________
Percent snail infection:
____________________

Collectors: ____________
____________
____________

Exposures; ______________
(whole in shade, part in shade, no shade)

Weather: ______________

Estimated area _________
Length _________
Width _________

Vegetations: _________
(Dense, sparse, medium)

SKETCH OF AREA

Figure 2: Malacological Form

## C. Geographic Information Systems

A GIS is an operational system that allows resource managers to store, retrieve, and edit geospatial data [11]. Geographic data can either be stored in raster or vector format. In a raster system, areas are divided into grids with a value for each cell in the grid. Each cell is coded according to the factor of interest. On the other hand, a vector system is based on the actual shapes and boundaries of regions. It is used when physical information is of importance [25].

In this study, both raster and vector representations will be used to visualize the spatial distribution of snail intermediate hosts and schistosomiasis cases. Since data is presented using maps, GIS allows quick and easy identification of patterns and relationships given a complex mass of information.

## D. Google Maps API

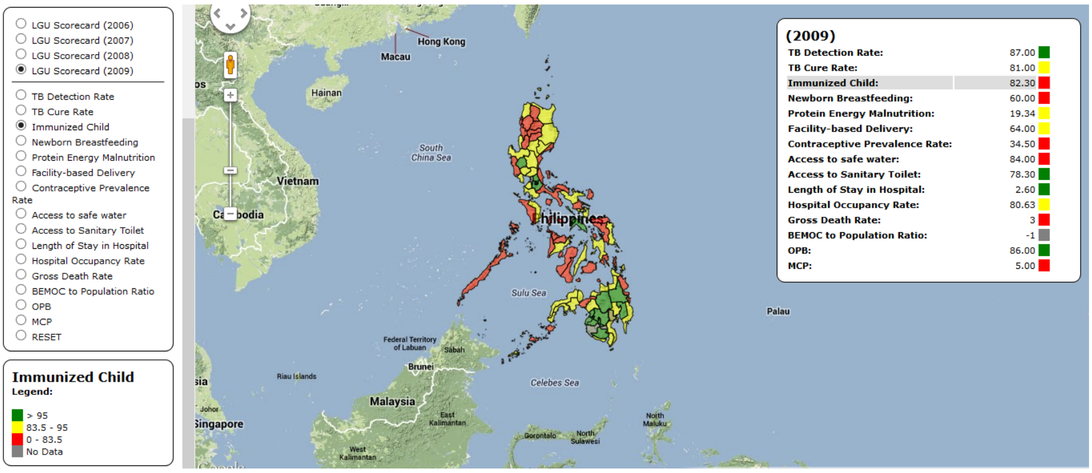
Google Maps is a leading web-based mapping tool owned by Google Incorporated. They claim to display the best imagery available. The satellite imagery they provide, however, are not updated in real time but are updated on a regular basis and are no older than three years [26].

The Google Maps application programming interface (API) was launched in 2005 to allow developers to combine their own data with Google Maps. Overall, it is the most popular API on the Internet since 39 percent of all mashups use Google Maps API [27].

Mashups are web pages or applications that are based on combined data or functionalities from different sources. Geospatial data visualization tools are important in many of the popular mashups and Google Maps API provides a way for programmers to use the power of Google Maps into their own application and integrate their own data efficiently [28].

## E. Thematic Maps

Thematic mapping, a type of heat mapping, is a visualization method in GIS for geographically clustering data. Concentrations are grouped into polygons of which values may be represented using labels, width of line segments, or colors [29]. For instance, an overview of concentration levels of schistosomiasis cases and hosts can be shown by displaying different intensities of color corresponding to the concentration while each polygon would represent barangays. Varying color intensities aid in quickly identifying risk areas given the large dataset. The following figure, obtained from DOH [30], is an example of a thematic map showing the number of immunized children in the Philippines per province. It is a mashup of Google Maps' raster map and a vector map of data on immunized children.



○ LGU Scorecard (2006)
○ LGU Scorecard (2007)
○ LGU Scorecard (2008)
● LGU Scorecard (2009)

○ TB Detection Rate
○ TB Cure Rate
● Immunized Child
○ Newborn Breastfeeding
○ Protein Energy Malnutrition
○ Facility-based Delivery
○ Contraceptive Prevalence Rate
○ Access to safe water
○ Access to Sanitary Toilet
○ Length of Stay in Hospital
○ Hospital Occupancy Rate
○ Gross Death Rate
○ BEMOC to Population Ratio
○ OPB
○ MCP
○ RESET

**Immunized Child**
**Legend:**

🟩 > 95
🟨 83.5 - 95
🟥 0 - 83.5
⬛ No Data

**(2009)**

| | | |
|---|---|---|
| **TB Detection Rate:** | 87.00 | 🟩 |
| **TB Cure Rate:** | 81.00 | 🟨 |
| **Immunized Child:** | 82.30 | 🟥 |
| **Newborn Breastfeeding:** | 60.00 | 🟥 |
| **Protein Energy Malnutrition:** | 19.34 | 🟨 |
| **Facility-based Delivery:** | 64.00 | 🟨 |
| **Contraceptive Prevalence Rate:** | 34.50 | 🟥 |
| **Access to safe water:** | 84.00 | 🟥 |
| **Access to Sanitary Toilet:** | 78.30 | 🟩 |
| **Length of Stay in Hospital:** | 2.60 | 🟩 |
| **Hospital Occupancy Rate:** | 80.63 | 🟨 |
| **Gross Death Rate:** | 3 | 🟥 |
| **BEMOC to Population Ratio:** | -1 | ⬛ |
| **OPB:** | 86.00 | 🟩 |
| **MCP:** | 5.00 | 🟥 |

Figure 3: Sample Thematic Map

## F. Model-View-Controller

The Model-View-Controller (MVC) is an object-oriented framework designed to provide the user an abstracted access to large and complex data sets. This abstraction is implemented by dividing the software into three parts: domain simulation, visual presentation and input interpretation. These three interconnected roles are respectively called model, view and controller.

The model and view works together through a subscribe/notify protocol while the controller serves as an intermediary between the two. For every change of state in a model, it notifies a corresponding view that depends on it to reflect the changes made. This protocol between the two makes it possible to reuse a model in multiple views without the need to rewrite the model for each [31].

This framework aids in avoiding the tendency of lumping interface designs together. Decoupling this triad with the use of MVC, makes the code more maintainable and flexible.

## G. Relational Database Management System

A database is a set of records with component fields which was generally presented in the early days of computing as a flat file, a single file divided into data blocks that consisted of records and fields within records. Due to the inadequacy of the flat file model for complex applications, the relational model was created. As opposed to the flat file model that views data as files containing records, the relational model views data as a set of tables containing columns as fields and rows as individual entities.

Database Management Systems (DBMS) serve the purpose of adding, updating, retrieving, manipulating and reporting information from the given database [32]. Correspondingly, a Relational Database Management System (RDBMS) is a type of DBMS that manages relational databases.

In RDBMS, the central component is the table which is referred to as a relation. The rows in a relation are called tuples while the columns are called attributes. The RDBMS is a software that handles a programmers requests to manage relational databases [33].

## H. Hypertext Preprocessor

Hypertext Preprocessor (PHP) is a popular server-side scripting language primarily for web development. It was created in 1994 by Rasmus Lerdorf with the intention of replacing a set of Perl scripts in managing his personal home page; thus, the acronym PHP.

PHP has a combination of sophisticated features making it powerful and easy to use. First, its code is preprocessed on the server-side and rendered as HTML making it secure and fast. Second, although HTML is static, PHP provides a work around by giving support to object-oriented concepts. This includes numerous libraries open-source objects and functions that allow PHP to access database servers, handle popular web formats and other common internet tasks. Lastly, due to its popularity, technical support is widely available. A wide range of PHP resources are available to programmers [34].

## I. MySQL

Virtually every RDBMS designed for multiple users uses Structured Query Language (SQL) as the programming language in manipulating relational databases. SQL is a standard computer language designed for accessing, querying and manipulating data contained in each relations. It gives programmers a powerful set of query tools, allowing them to retrieve, delete, update or add to specific information [35].

MySQL is an open source software, available through the GNU General Public License. The software consists of a server that is responsible for managing the database system. All interactions with the database are done by sending SQL commands to the MySQL server using PHP. It is an ideal tool in web development due to its speed and small size [36].

## J. CodeIgniter

CodeIgniter is a PHP-based open source application development framework written by Rick Ellis for rapid development of dynamic websites. It reduces the amount of coding time by removing the need for repetitive code and providing standard functionalities beforehand. Furthermore, it provides support for MySQL and is loosely based on the MVC pattern [37] thus cutting down even more the amount of code written by a programmer.

# IV. Design and Implementation

## A. Use Cases

Figure 4 shows the top level use case diagram of the system. It has four actors: government officials, health workers, data managers, and system administrators.

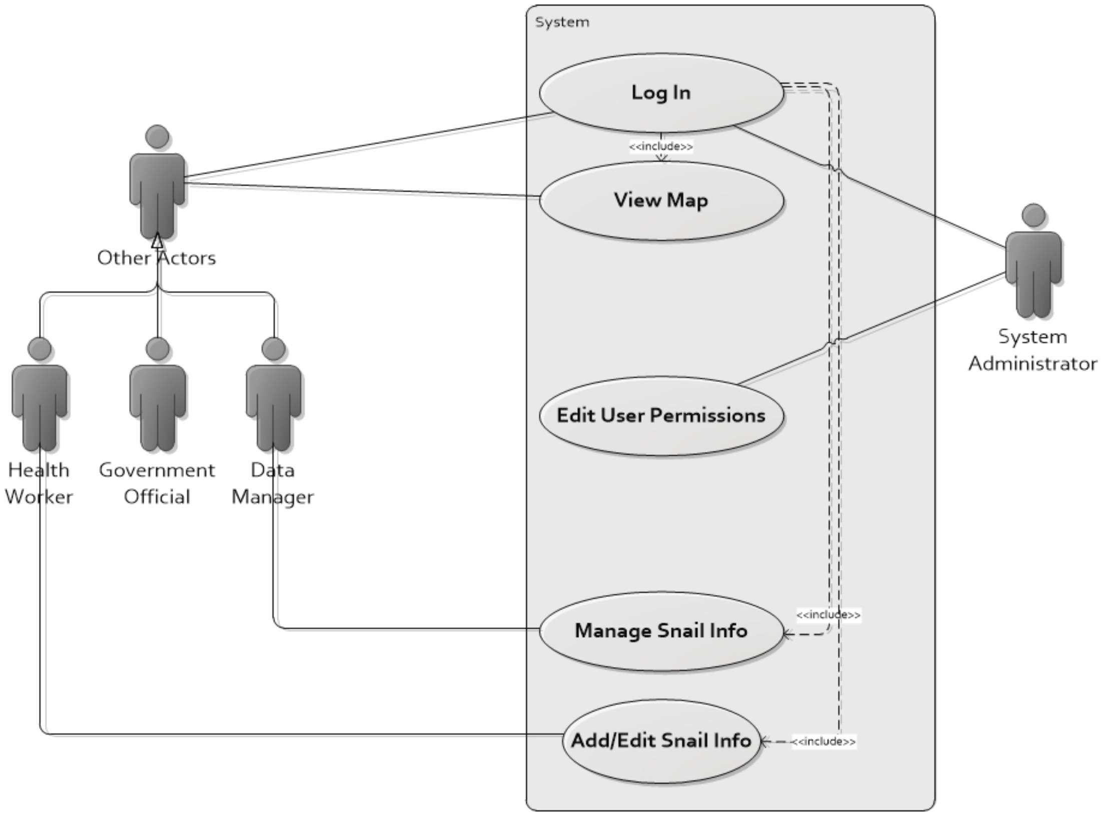


Figure 4: Top Level Use Case Diagram

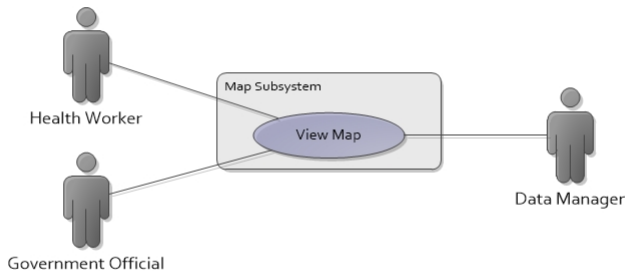


Figure 5: Map Subsystem Use Case Diagram

Upon logging in, Government officials, health workers, and data managers are allowed to view data on the map.

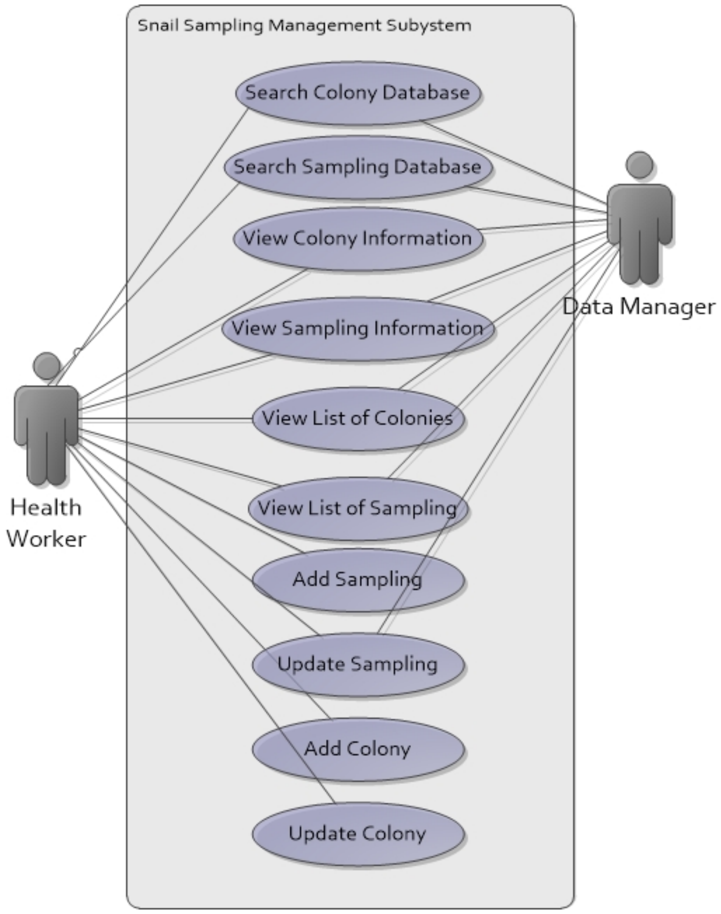


Figure 6: Snail Sampling Subsystem Use Case Diagram

Health workers and data managers can view snail information in detail. Additionally, Health workers can enter information regarding the colony site and sampling per colony while data managers can mark the data as either queued or final.

## B. Database Design

### Entity Relationship Diagram

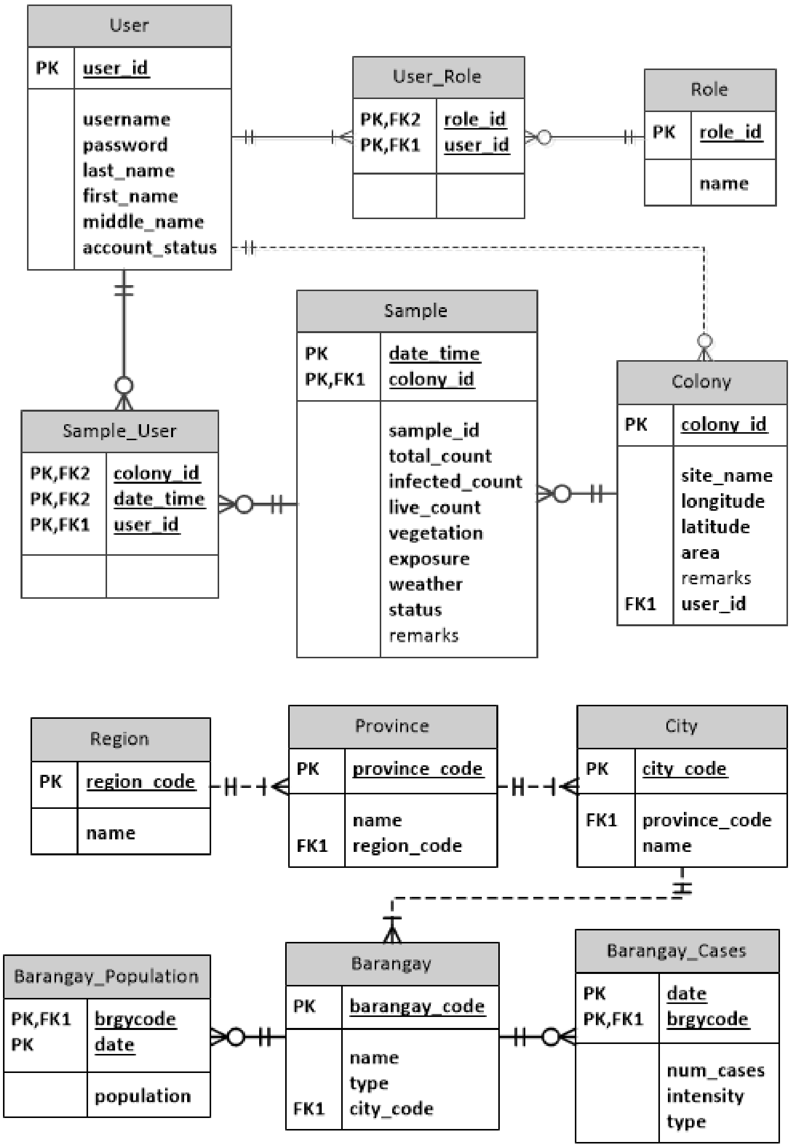


Figure 7: Schistosomiasis GIS Entity Relationship Diagram

The preceding figure shows an entity relationship diagram of the whole Schistosomiasis GIS. Each user of the system is assigned with one or more roles. The Colony table contains the basic information about each snail site. A snail site can have none to multiple samples and for each sample, there is at least one health worker who collected the sample.

**Data Dictionary**

| **Field** | **Data Type** | **Description** |
|---|---|---|
| user_id | int(5); Primary Key | auto-incrementing user identification number |
| username | varchar(20) | username used in logging in |
| password | varchar(50) | password used in logging in |
| last_name | varchar(20) | last name of the user |
| first_name | varchar(20) | first name of the user |
| middle_name | varchar(20) | middle name of the user |
| email | varchar(50) | email address of the user |
| account_status | enum('enabled', 'disabled') | account status |

Table 1: User Table

The User Table contains information about the users of the system.

| **Field** | **Data Type** | **Description** |
|---|---|---|
| role_id | int(3); Primary Key | auto-incrementing role identification number |
| name | varchar(20) | readable name that identifies the role |

Table 2: Role Table

The Role Table contains information about the roles of users of the system.

| **Field** | **Data Type** | **Description** |
|---|---|---|
| user_id | int(5); Primary Key; Foreign Key | user identification number |
| role_id | int(3); Primary Key; Foreign Key | role identification number |

Table 3: User_Role Table

The User_Role Table contains the assigned roles to each user of the system.

| **Field** | **Data Type** | **Description** |
|---|---|---|
| colony_id | int(5); Primary Key | auto-incrementing colony identification number |
| site_name | varchar(20) | name of the site of the snail colony |
| longitude | double | longitude location of the colony |
| latitude | double | latitude location of the colony |
| area | int(5) | estimated area in square meters |
| status | enum('queued', 'final') | whether the data has been verified by the data manager |
| remarks | text | comments that the data manager makes for each snail site |
| user_id | int(5) | identification number of the last field worker to make changes to a specific snail site |

Table 4: Colony Table

The Colony Table contains details about the snail colonies.

| Field | Data Type | Description |
|---|---|---|
| colony_id | int(5); Primary Key; Foreign Key | identifier for the snail colony associated with the collection data |
| date_time | datetime(); Primary Key; Foreign Key | date and time of collection |
| total_count | int(5) | total number of snails collected in the area |
| infected_count | int(5) | number of infected snails in the collection |
| live_count | int(5) | number of live snails in the collection |
| vegetation | enum('dense', 'sparse', 'medium') | vegetation status of the colony as of collection time |
| exposure | enum('whole in shade', 'part in shade', 'no shade') | exposure of the colony as of collection time |
| weather | text | description of the weather as of collection time |
| status | enum('queued', 'final') | whether the data has been verified by the data manager |
| remarks | text | comments that the data manager makes for each sample information |

Table 5: Sample Table

The Sample Table contains details about the snail sampling per colony.

| Field | Data Type | Description |
|---|---|---|
| colony_id | int(5); Primary Key; Foreign Key | identifies the snail colony associated with data |
| date_time | datetime(); Primary Key; Foreign Key | date and time of collection |
| user_id | int(5); Primary Key; Foreign Key | identifies the collector associated with the data |

Table 6: Sample_User Table

The Sample_User Table contains details about the collectors per snail sample.

| Field | Data Type | Description |
|---|---|---|
| region_code | int(5); Primary Key | region identification number |
| name | varchar(10) | name of the region |

Table 7: Region Table

The Region Table identifies a region by its name and code.

| Field | Data Type | Description |
|---|---|---|
| province_code | int(5); Primary Key | province identification number |
| region_code | int(5); Foreign Key | region identification number |
| name | varchar(20) | name of province |

Table 8: Province Table

The Province Table identifies a province to a region by its name and code.

| Field | Data Type | Description |
|---|---|---|
| city_code | int(7); Primary Key | city identification number |
| province_code | int(5); Foreign Key | province identification number |
| name | varchar(20) | name of city |

Table 9: City Table

The City Table identifies a city to a province by its name and code.

| Field | Data Type | Description |
|---|---|---|
| barangay_code | int(11); Primary Key | barangay identification number |
| city_code | int(7); Foreign Key | city identification number |
| name | varchar(20) | name of barangay |
| type | enum('urban', 'rural') | type of barangay; urban or rural |

Table 10: Barangay Table

The Barangay Table identifies a barangay to a city by its name and code.

| Field | Data Type | Description |
|---|---|---|
| barangay_code | int(11); Primary Key; Foreign Key | barangay identification number |
| date | date; Primary Key | date of survey |
| num_cases | int(5) | number of patients |
| intensity | int(5) | average number of parasite eggs per patient |
| type | enum('queued', 'final') | type of data from the source |

Table 11: Barangay_Cases Table

The Barangay_Cases Table contains the number of schistosomiasis cases per barangay on a specified date.

| Field | Data Type | Description |
|---|---|---|
| barangay_code | int(11); Primary Key; Foreign Key | barangay identification number |
| date | date(7); Primary Key | date of census |
| population | int(5) | number of population |

Table 12: Barangay_Population Table

The Barangay_Population Table contains details about a barangays population on a specified date.

# V. Technical Architecture

The server machine needs the following applications:

1. PHP Server (Apache 2.4.7)

2. MySQL (5.5 and up)

The users of the system should have the following specifications:

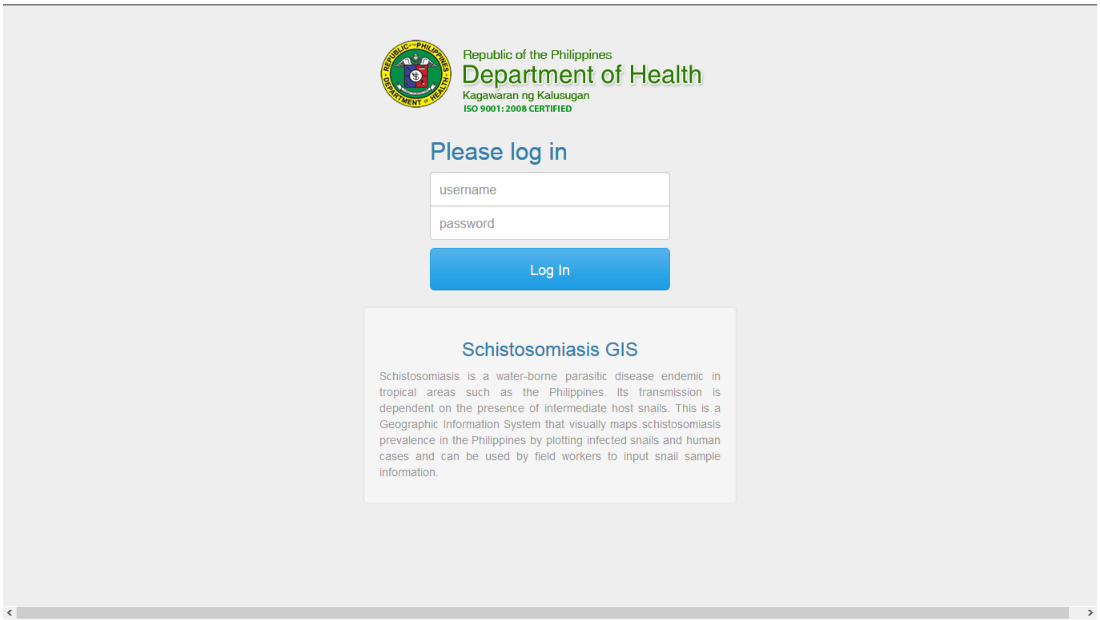
1. Web Browser (Firefox 25/Google Chrome 31/Internet Explorer 8)

2. Internet connection (to access the map)

# VI. Results

## A. General View

### Log In Page



Republic of the Philippines
Department of Health
Kagawaran ng Kalusugan
ISO 9001:2008 CERTIFIED

Please log in

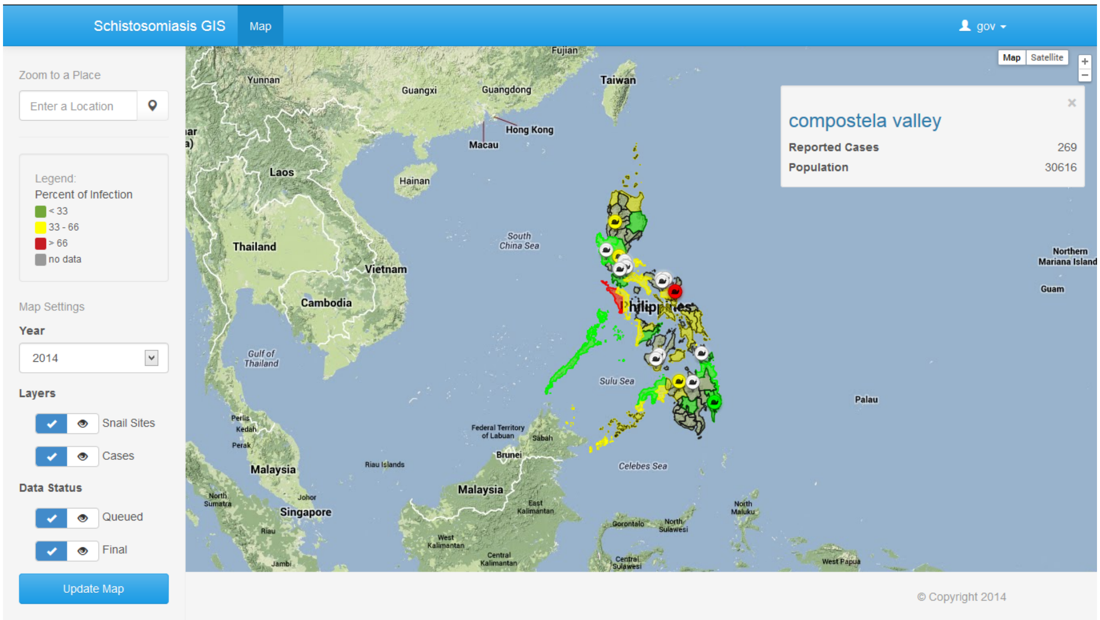
Schistosomiasis GIS

Schistosomiasis is a water-borne parasitic disease endemic in tropical areas such as the Philippines. Its transmission is dependent on the presence of intermediate host snails. This is a Geographic Information System that visually maps schistosomiasis prevalence in the Philippines by plotting infected snails and human cases and can be used by field workers to input snail sample information.

Figure 8: Log In Page

Figure 8 shows the first page that shows up when the system is accessed. This is the home page which contains basic information about the system and a log-in form. Unless a user has an account in the schistosomiasis system and is enabled for the this system, they will not be able to use any of the system's functionalities.

## Map



Zoom to a Place

Enter a Location

Legend:
Percent of Infection
🟩 < 33
🟨 33 - 66
🟥 > 66
⬛ no data

Map Settings

Year

2014

Layers

☑ Snail Sites

☑ Cases

Data Status

☑ Queued

☑ Final

Update Map

compostela valley

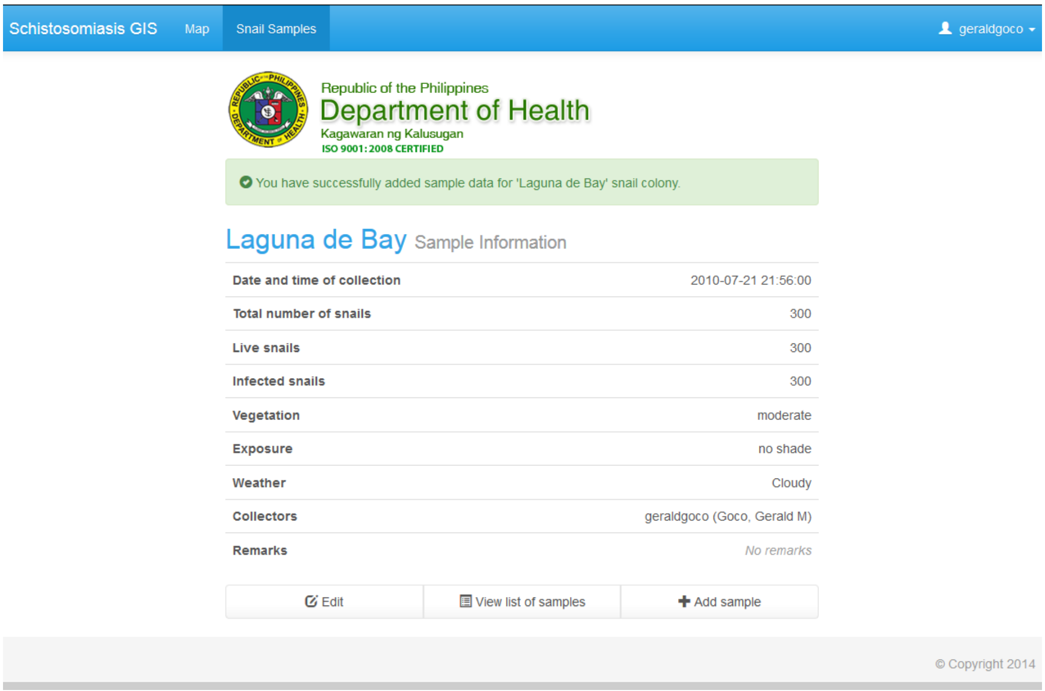
| | |
|---|---|
| Reported Cases | 269 |
| Population | 30616 |

© Copyright 2014

Figure 9: Map

Every user except for the administrator has access to the map. In this page, the user may select which data to show or hide on the map and may view specific snail sites and case summaries in detail.

### Snail Sample Information



You have successfully added sample data for 'Laguna de Bay' snail colony.

Laguna de Bay Sample Information

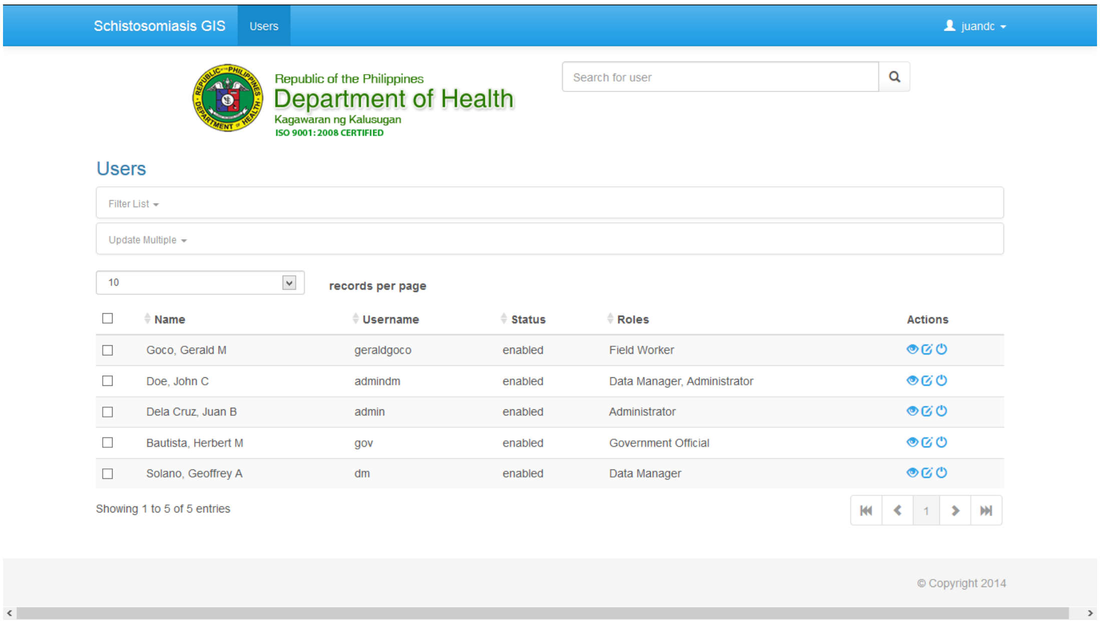
| | |
|---|---|
| Date and time of collection | 2010-07-21 21:56:00 |
| Total number of snails | 300 |
| Live snails | 300 |
| Infected snails | 300 |
| Vegetation | moderate |
| Exposure | no shade |
| Weather | Cloudy |
| Collectors | geraldgoco (Goco, Gerald M) |
| Remarks | *No remarks* |

Figure 10: Snail Sample Page

Each sample on the map, when clicked, leads to a page that displays the details for that specific item. From this view, a field worker can edit or add a sample while a data manager can add/edit remarks.

# B. Administrator View

## Users List



Users

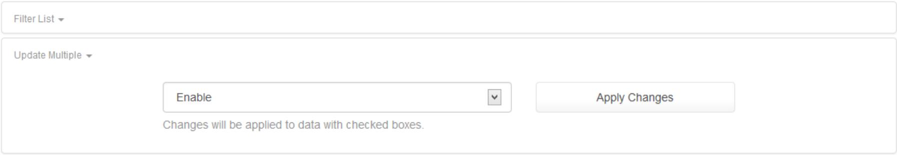
10 records per page

| □ | Name | Username | Status | Roles | Actions |
|---|---|---|---|---|---|
| □ | Goco, Gerald M | geraldgoco | enabled | Field Worker | 👁 ✎ ⏻ |
| □ | Doe, John C | admindm | enabled | Data Manager, Administrator | 👁 ✎ ⏻ |
| □ | Dela Cruz, Juan B | admin | enabled | Administrator | 👁 ✎ ⏻ |
| □ | Bautista, Herbert M | gov | enabled | Government Official | 👁 ✎ ⏻ |
| □ | Solano, Geoffrey A | dm | enabled | Data Manager | 👁 ✎ ⏻ |

Showing 1 to 5 of 5 entries

© Copyright 2014

Figure 11: User Management Page

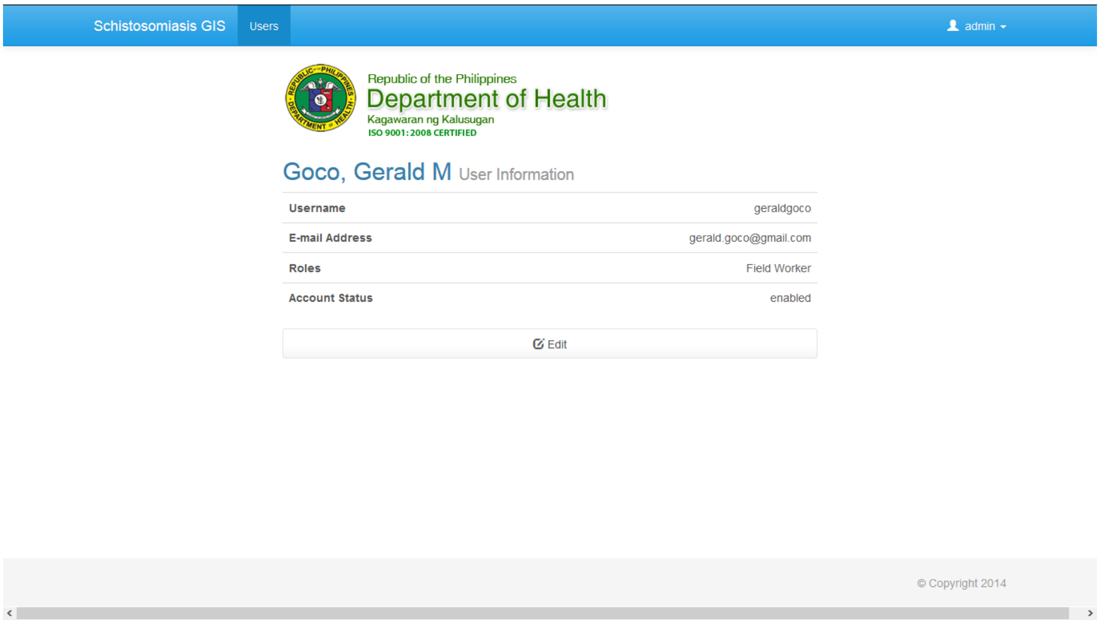


Changes will be applied to data with checked boxes.

Figure 12: User Management Update Options

Figures 11 and 12 show the screen for the administrator upon log in. Administrators can manage user roles and enable or disable other user accounts except their own.

## View Account



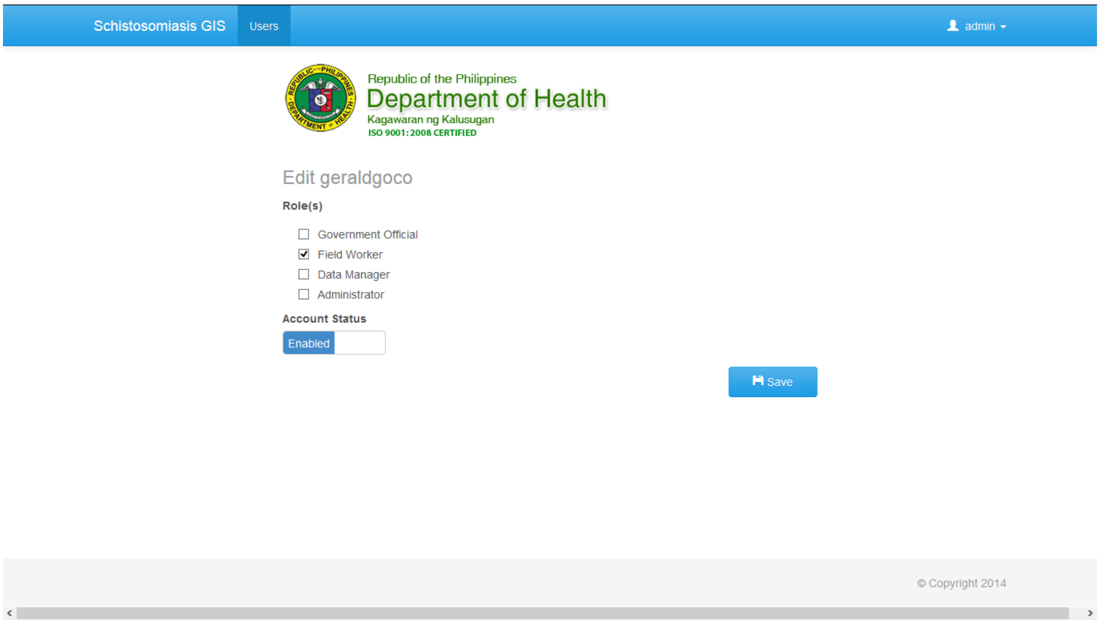
Goco, Gerald M User Information

|  |  |
|---|---|
| Username | geraldgoco |
| E-mail Address | gerald.goco@gmail.com |
| Roles | Field Worker |
| Account Status | enabled |

Figure 13: View Account

An administrator can also view each existing account individually along with their basic information.

## Update Account



Edit geraldgoco

Role(s)

- ☐ Government Official
- ☑ Field Worker
- ☐ Data Manager
- ☐ Administrator

Account Status

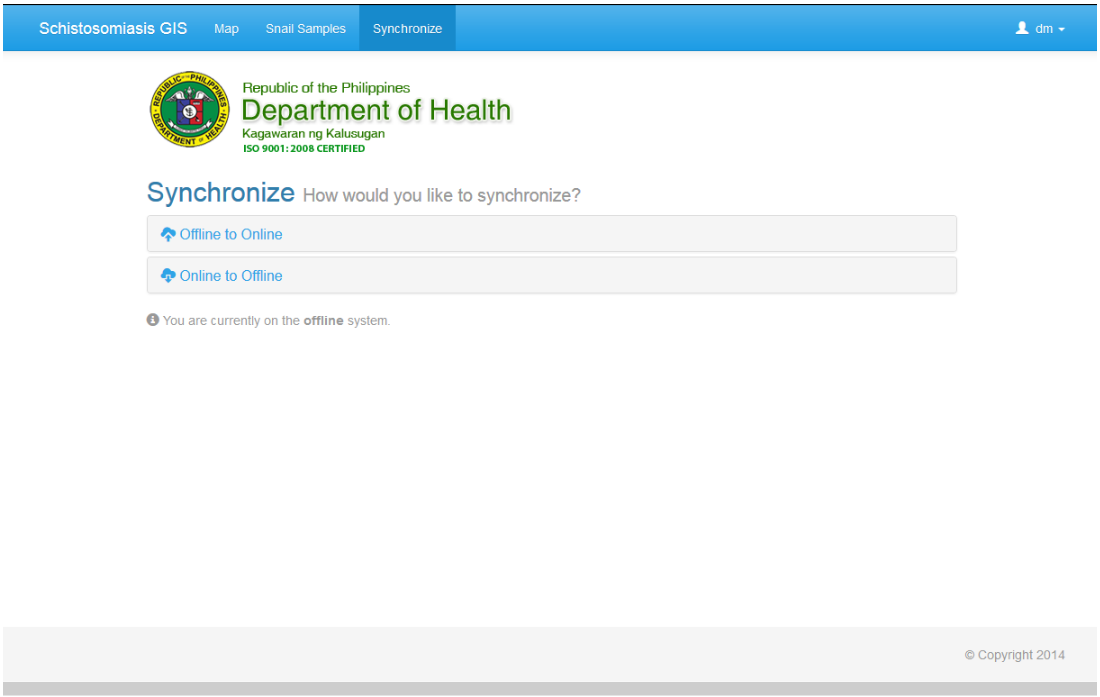
Enabled

Figure 14: Update Account

The administrator can only edit the user roles and account status in this system.

# C. Data Manager View

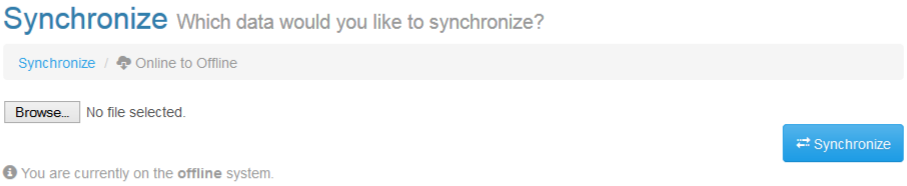
## Synchronization



Synchronize How would you like to synchronize?

You are currently on the **offline** system.

Figure 15: Synchronization Page



Synchronize Which data would you like to synchronize?

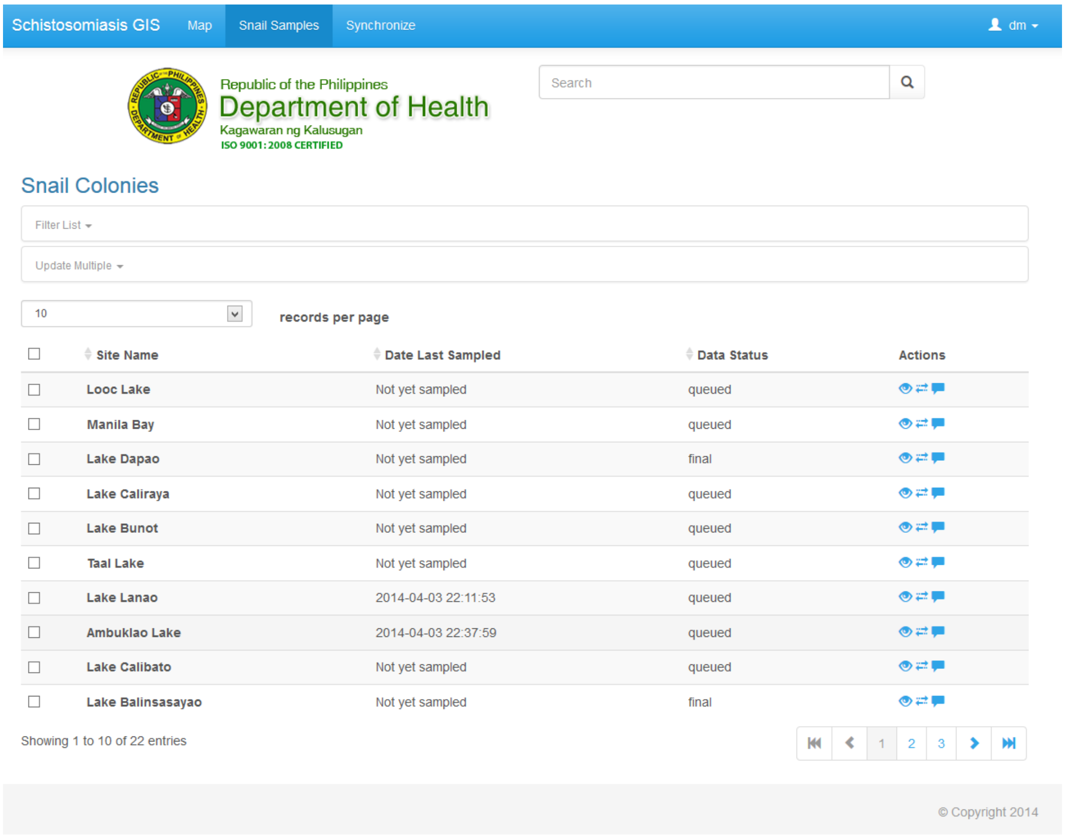
No file selected.

You are currently on the **offline** system.

Figure 16: Synchronization: File Selection Page

Data administrators can import or export snail iformation from the system using this page.

### List of Snail Colonies



Snail Colonies

| □ | Site Name | Date Last Sampled | Data Status | Actions |
|---|---|---|---|---|
| □ | Looc Lake | Not yet sampled | queued | |
| □ | Manila Bay | Not yet sampled | queued | |
| □ | Lake Dapao | Not yet sampled | final | |
| □ | Lake Caliraya | Not yet sampled | queued | |
| □ | Lake Bunot | Not yet sampled | queued | |
| □ | Taal Lake | Not yet sampled | queued | |
| □ | Lake Lanao | 2014-04-03 22:11:53 | queued | |
| □ | Ambuklao Lake | 2014-04-03 22:37:59 | queued | |
| □ | Lake Calibato | Not yet sampled | queued | |
| □ | Lake Balinsasayao | Not yet sampled | final | |

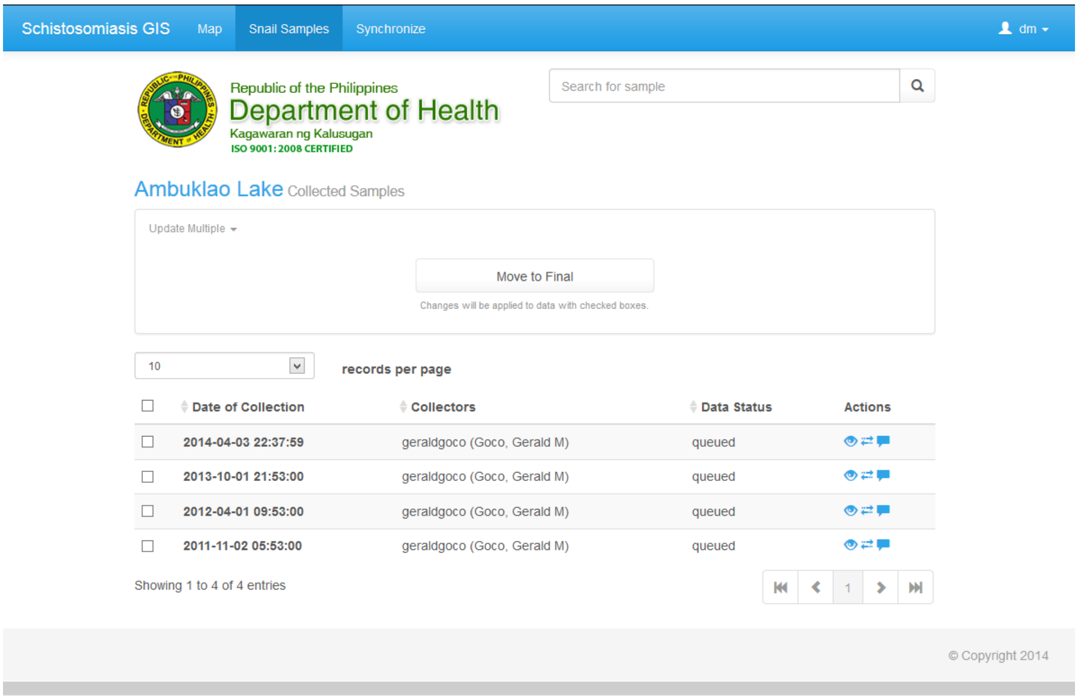
Showing 1 to 10 of 22 entries

© Copyright 2014

Figure 17: Snail Colony List for Data Managers

Data administrators can view all snail colonies that have been added to the system. They may also mark these data as either "queued" or "final".

### List of Snail Samples



Ambuklao Lake Collected Samples

Changes will be applied to data with checked boxes.

records per page

| □ | Date of Collection | Collectors | Data Status | Actions |
|---|---|---|---|---|
| □ | 2014-04-03 22:37:59 | geraldgoco (Goco, Gerald M) | queued | 👁 ⇄ 💬 |
| □ | 2013-10-01 21:53:00 | geraldgoco (Goco, Gerald M) | queued | 👁 ⇄ 💬 |
| □ | 2012-04-01 09:53:00 | geraldgoco (Goco, Gerald M) | queued | 👁 ⇄ 💬 |
| □ | 2011-11-02 05:53:00 | geraldgoco (Goco, Gerald M) | queued | 👁 ⇄ 💬 |

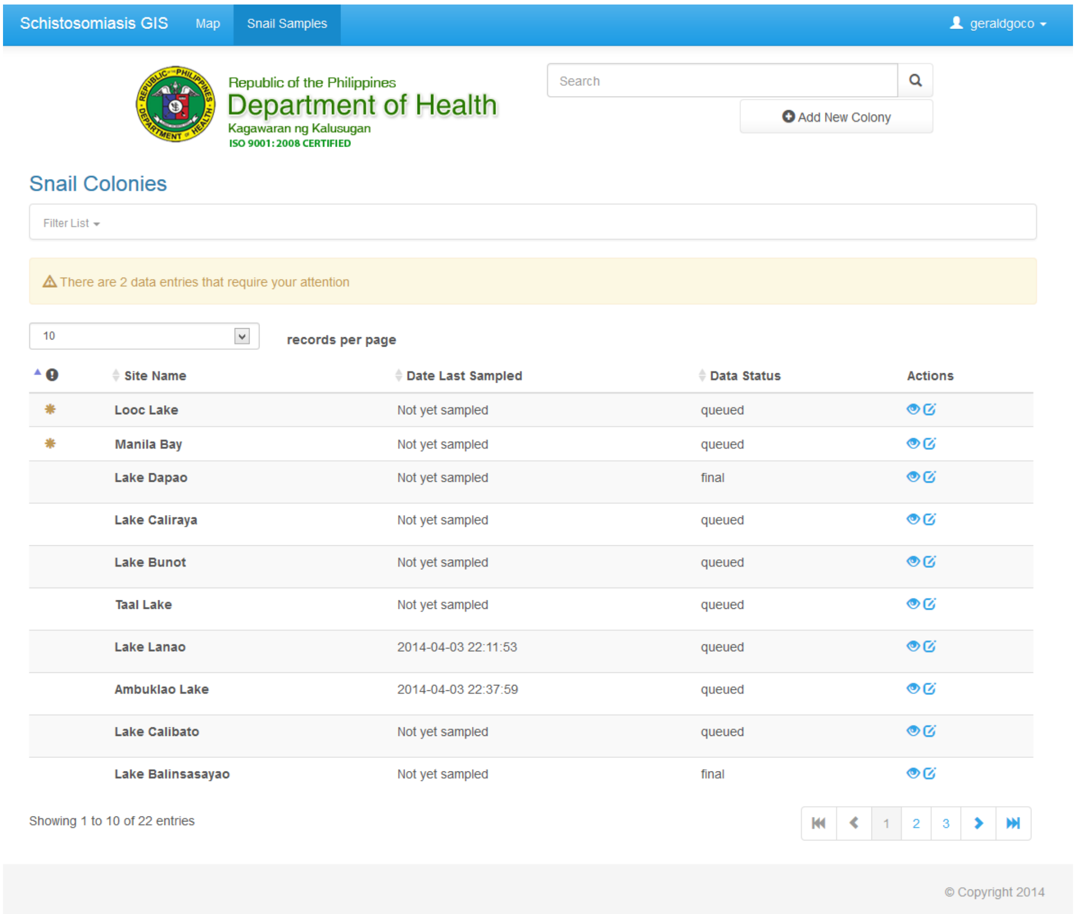
Showing 1 to 4 of 4 entries

Figure 18: Snail Samples List for Data Managers

Similar to the snail colonies page, data administrators can also view all snail sample information for each colony. They may also mark each sample data as either “queued” or “final”.

# D. Field Worker View

## List of Snail Colonies



Snail Colonies

⚠ There are 2 data entries that require your attention

10 records per page

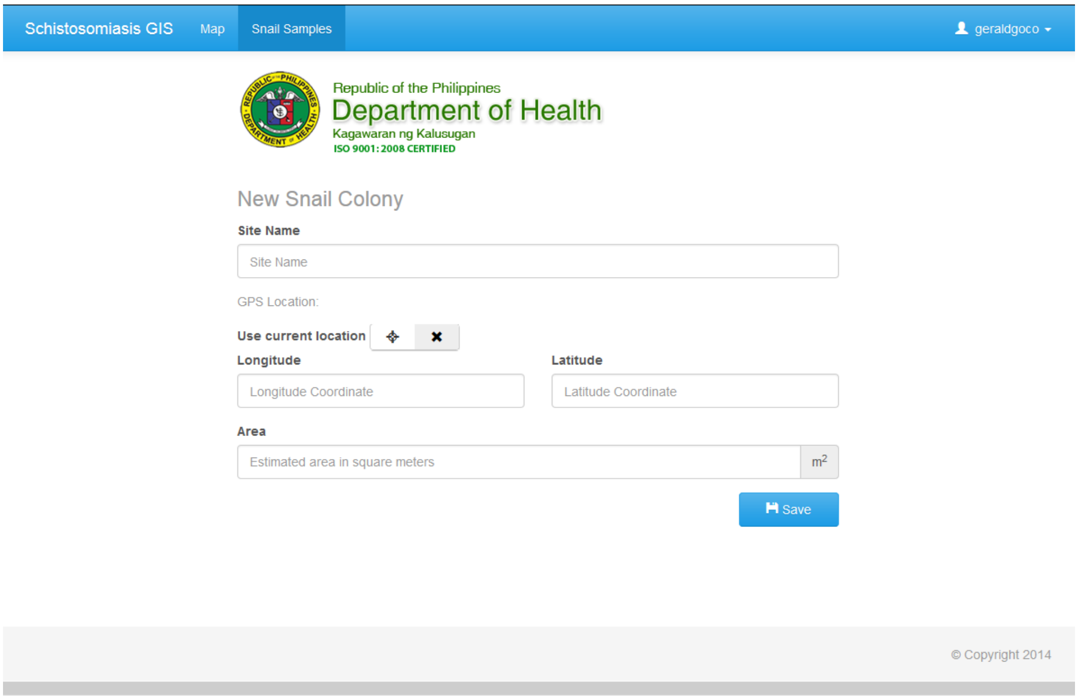
| ! | Site Name | Date Last Sampled | Data Status | Actions |
|---|---|---|---|---|
| ✱ | Looc Lake | Not yet sampled | queued | 👁 ✎ |
| ✱ | Manila Bay | Not yet sampled | queued | 👁 ✎ |
| | Lake Dapao | Not yet sampled | final | 👁 ✎ |
| | Lake Caliraya | Not yet sampled | queued | 👁 ✎ |
| | Lake Bunot | Not yet sampled | queued | 👁 ✎ |
| | Taal Lake | Not yet sampled | queued | 👁 ✎ |
| | Lake Lanao | 2014-04-03 22:11:53 | queued | 👁 ✎ |
| | Ambuklao Lake | 2014-04-03 22:37:59 | queued | 👁 ✎ |
| | Lake Calibato | Not yet sampled | queued | 👁 ✎ |
| | Lake Balinsasayao | Not yet sampled | final | 👁 ✎ |

Showing 1 to 10 of 22 entries

Figure 19: Snail Colony List for Field Workers

Field workers can also view all snail colonies that have been added to the system. However, they are allowed to edit other information except whether it is marked as “queued” or “final”.

## Add Snail Colony



New Snail Colony

Site Name: Site Name

GPS Location:

Use current location ✖

Longitude: Longitude Coordinate

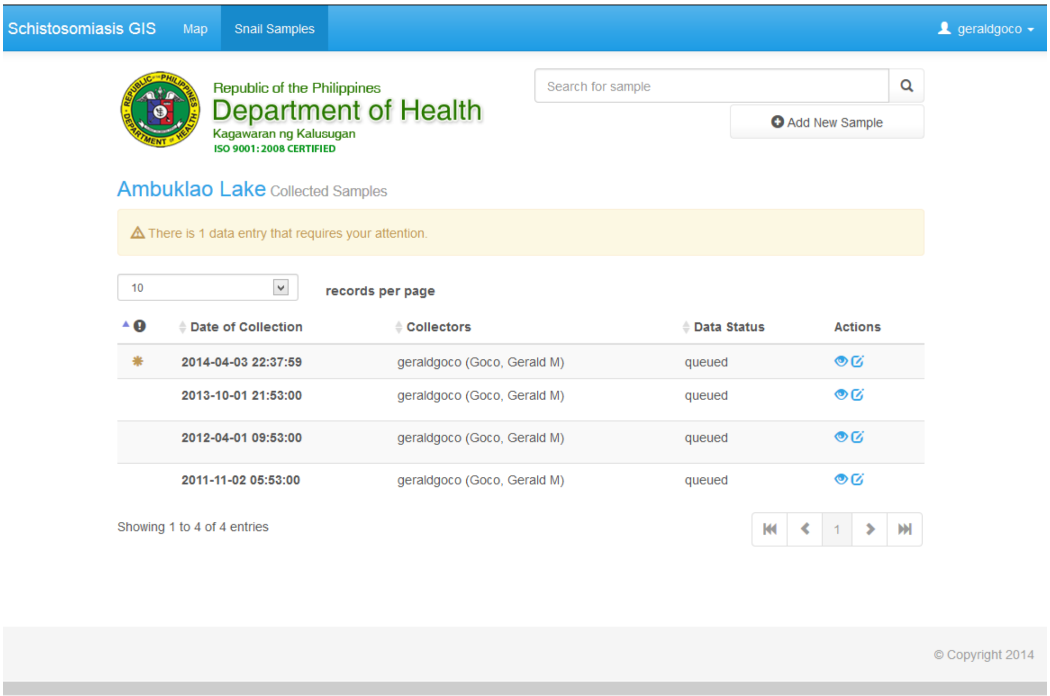
Latitude: Latitude Coordinate

Area: Estimated area in square meters m²

Figure 20: Add/Edit Snail Colony

Field workers can add snail colonies and can use their current location as input. This form is also used for the Edit Colony function.

## List of Snail Samples



Ambuklao Lake Collected Samples

⚠ There is 1 data entry that requires your attention.

| ! | Date of Collection | Collectors | Data Status | Actions |
|---|---|---|---|---|
| ✱ | 2014-04-03 22:37:59 | geraldgoco (Goco, Gerald M) | queued | 👁 ✎ |
|  | 2013-10-01 21:53:00 | geraldgoco (Goco, Gerald M) | queued | 👁 ✎ |
|  | 2012-04-01 09:53:00 | geraldgoco (Goco, Gerald M) | queued | 👁 ✎ |
|  | 2011-11-02 05:53:00 | geraldgoco (Goco, Gerald M) | queued | 👁 ✎ |

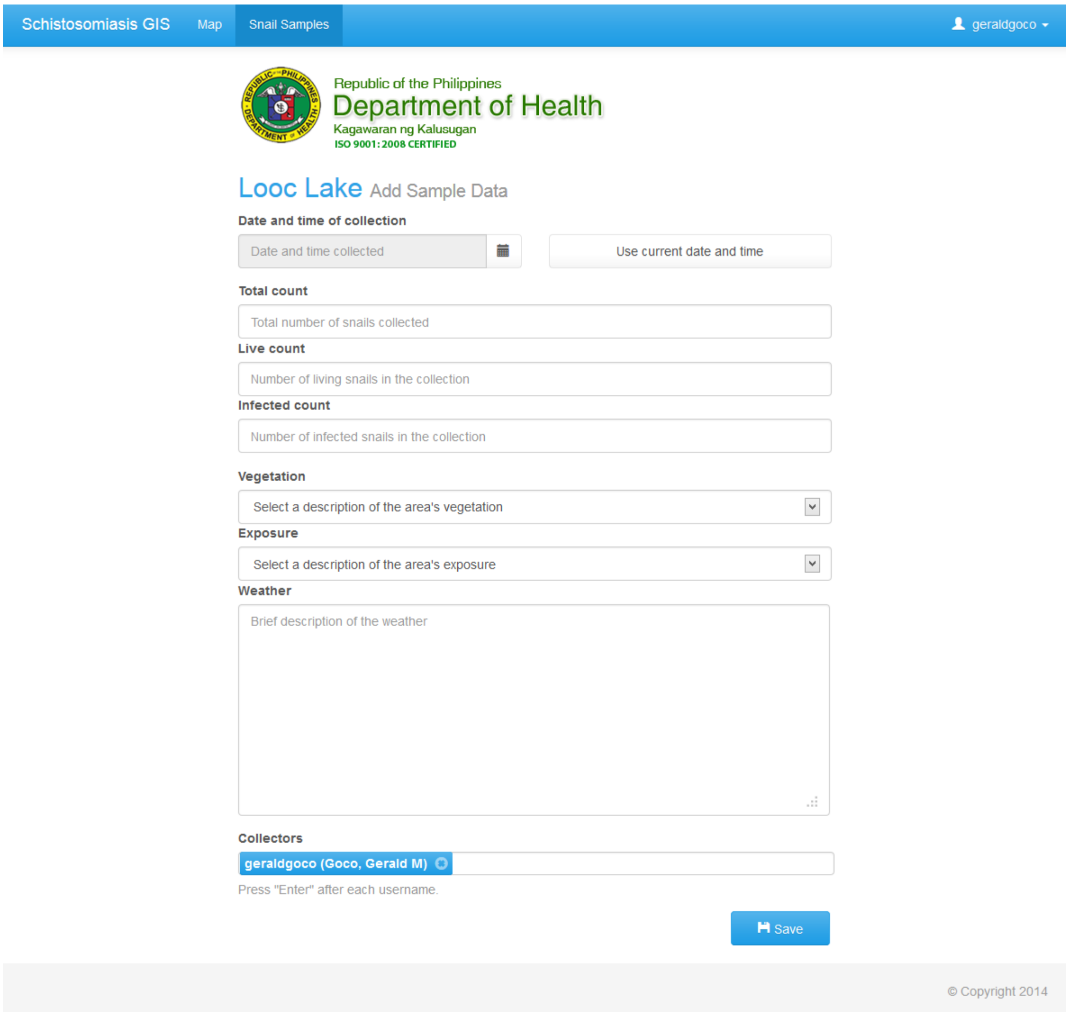
Showing 1 to 4 of 4 entries

© Copyright 2014

Figure 21: Snail Samples List for Field Workers

Field workers can also view all snail sample information for each colony. They may add/edit details about the samples except change whether it is “queued” or “final”.

### Add Snail Sample


Schistosomiasis GIS   Map   Snail Samples   geraldgoco

Republic of the Philippines
Department of Health
Kagawaran ng Kalusugan
ISO 9001:2008 CERTIFIED

Looc Lake Add Sample Data

Date and time of collection

Date and time collected   Use current date and time

Total count

Total number of snails collected

Live count

Number of living snails in the collection

Infected count

Number of infected snails in the collection

Vegetation

Select a description of the area's vegetation

Exposure

Select a description of the area's exposure

Weather

Brief description of the weather

Collectors

geraldgoco (Goco, Gerald M)

Press "Enter" after each username.

Save

© Copyright 2014

Figure 22: Add/Edit Snail Sample

Field workers can add snail sample to any snail colony and have their account automatically added as a collector. This form is also used for the Edit Sample function.

# VII. Discussions

The Schistosomiasis Snail Surveillance and Human Prevalence Geographic Information System is a tool for visualizing schistosomiasis in the Philippines by mapping data collected through surveillance. In the generated map, two datasets are considered: data on snail samples on human cases. This special problem is an automation of snail sampling records that are being collected by health workers. Through this system, field workers are able to collect information on snail sites without having to manually fill out forms since adding and editing snail information are done electronically. Moreover, through Internet connection they can use their smart-phones as input devices for location instead of recording from a separate GPS device. Unlike the current manual data storage, data are more updated and securely stored through the online database. This system also uses a database for schistosomiasis cases that conforms with the current DOH system for schistosomiasis. It is integrated with other surveillance systems under DOH.

The map that is produced by this system makes it easier for DOH to act on the endemic disease since it graphically shows them where the cases are and where the intermediate hosts might cause a spread. Through this system, regular monitoring of the spread of schistosomiasis for each province and for each municipality in the Philippines is possible.

With the use of CodeIgniter's MVC pattern, it wasn't very difficult to create this system. Plug-ins were readily available and were easily integrated. The GoogleMaps API for CodeIgniter made it easy to initialize the map and setup its layers.

# VIII. Conclusions

The Schistosomiasis Snail Surveillance and Human Prevalence Geographic Information System is a web application that is able to store and retrieve snail site and snail sample information. It also allows its users to view schistosomiasis data in a visual manner. This was achieved by showing a map implemented through GoogleMaps API that displayed a layer of markers that indicates the snail sites and a layer of polygons that indicates the prevalence of the disease in humans. Users can access the system if they already have an account with the DOH system and if they are allowed by the administrator.

Administrators create accounts for users through the DOH system and assigns them roles that dictate their permissions for this system. Data managers indicate which data are verified and not. They are also responsible for synchronizing offline setups with the online system. Field workers use the system for data entry on snail sites and snail samples. They can also use the system to see on a map the data that they have gathered. Finally, government officials use the system to understand possible risks in their areas so that they can be a part in the planning of control and elimination of schistosomiasis.

# IX. Recommendations

One of the most useful features of this system is that it uses the device's current location as an input when entering snail site information. Hence, when it is accessed via mobile phone, it can be used by field workers as an alternative to having a separate GPS device. However, the availability of this system for data entry in the field is limited to the capacity of the field worker's mobile device used to access the system. Without Internet connection via 3G or other means, the system cannot be used. Furthermore, if the web browser used to access the system is not capable of getting the field worker's location, the location grabbing feature of this system is also useless. To reduce the limitations of this system, a mobile app that can be available offline and can temporarily store data on the mobile device until the user is online, at which point the data will be automatically uploaded to the system, can be developed.

Also, it uses GoogleMaps in order to display raster and vector maps. However, when GoogleMaps service is unavailable, neither the raster nor the vector map can be viewed. Hence, providing a map that is as reliable as GoogleMaps and yet is also available offline will be an improvement to the system.

# X. Bibliography

[1] B. L. Blas, *Handbook for the Control of Schistosomiasis Japonica in the Philippines.* Manila: Department of Health, 1988-89.

[2] J. Utzinger, E. K. Ngoran, C. R. Caffrey, and J. Keiser, "From innovation to application: Social-ecological context, diagnostics, drugs and integrated control of schistosomiasis.," *Acta Trop*, vol. 120 Suppl 1, pp. S121–37, 2011.

[3] Department of Health, "Schistosomiasis," in *2011-2016 National Objectives for Health, Health Sector Reform Agenda Monographs*, pp. 73–75, Manila: Health Policy Development and Planning Bureau (HPDPB) Department of Health, July 2012.

[4] R. M. Olveda, B. L. Daniel, B. D. Ramirez, G. D. Aligui, L. P. Acosta, P. Fevidal, E. Tiu, F. de Veyra, P. A. Peters, R. Romulo, E. Domingo, P. M. Wiest, and G. R. Olds, "Schistosomiasis japonica in the philippines: the long-term impact of population-based chemotherapy on infection, transmission, and morbidity.," *J Infect Dis*, vol. 174, no. 1, pp. 163–72, 1996.

[5] L.-P. Sun, Y.-S. Liang, H.-H. Wu, Z.-X. Tian, J.-R. Dai, K. Yang, Q.-B. Hong, X.-N. Zhou, and G.-J. Yang, "A google earth-based surveillance system for schistosomiasis japonica implemented in the lower reaches of the yangtze river, china.," *Parasit Vectors*, vol. 4, p. 223, 2011.

[6] D. U. Olveda, Y. Li, R. M. Olveda, A. K. Lam, D. P. McManus, T. N. P. Chau, D. A. Harn, G. M. Williams, D. J. Gray, and A. G. P. Ross, "Bilharzia in the philippines: past, present, and future.," *Int J Infect Dis*, vol. 18, pp. 52–56, 2013.

[7] ADAM Encyclopedia, "Schistosomiasis." `http://www.ncbi.nlm.nih.gov/pubmedhealth/PMH0002298/`, October 2012. Accessed: 10 January 2014.

[8] A. N. Akullian, “Spatial patterns of schistosomiasis: A watershed approach to measuring s. japonicum environmental risk and human and animal disease outcomes,” B.Sc Thesis, Center for Environmental Studies, Brown University, 2007.

[9] World Health Organization, “Vector surveillance and control,” in *Dengue Haemorrhagic Fever: Diagnosis, Treatment, Prevention and Control - 2nd edition*, pp. 48–59, Geneva: World Health Organization Office of Publications, July 1997.

[10] E. Funari, T. Kistemann, S. Herbst, and A. Rechenburg, *Technical Guidance on Water-related Disease Surveillance.* Copenhagen, Denmark: World Health Organization Regional Office for Europe, 2011.

[11] “Information Technology as Megaengineering: The Impact of GIS,” in *Engineering Earth: The Impacts of Megaengineering Project* (S. D. Brunn, ed.), pp. 37–47, Dordrecht: Springer, 2011.

[12] M. Asif, N. K. Tripathi, and S. Ahmed, “Towards near real time public health surveillance (a decision support system for public health surveillance),” *International Journal of Computer Applications*, vol. 61, pp. 45–50, January 2013. Published by Foundation of Computer Science, New York, USA.

[13] P. S. Arno, M. N. Gourevitch, E. Drucker, J. Fang, M. Goldberg, M. Memmot, K. Bonuck, N. Deb, and E. Schoenbaum, “Analysis of a population-based pneumocystis carinii pneumonia index as an outcome measure of access and quality of care for the treatment of HIV disease,” *American Journal of Public Health*, vol. 92, no. 3, pp. 395–398, 2002.

[14] P. E. Bailey, E. B. Keyes, C. Parker, M. Abdullah, H. Kebede, and L. Freedman, “Using a gis to model interventions to strengthen the emergency referral system for maternal and newborn health in ethiopia.,” *Int J Gynaecol Obstet*, 2011.

[15] I. I. Abbas, S. Z. Auta, and R. M. Naiya, “Health care facilities mapping and database creation using gis in chikun local government, kaduna state, nigeria,” *Global Journal of Human Social Science Geography and Environmental GeoSciences*, vol. 12, no. 10, pp. 11–18, 2012.

[16] U. A. Kibon and M. Ahmed, “Distribution of primary health care facilities in kano metropolis using gis (geographic information system),” *Research Journal of Environmental and Earth Sciences*, vol. 5, no. 4, pp. 167–176, 2013.

[17] T. Kistemann, S. Herbst, F. Dangendorf, and M. Exner, “GIS-based analysis of drinking-water supply structures: a module for microbial risk assessment.,” *Int J Hyg Environ Health*, vol. 203, no. 4, pp. 301–10, 2001.

[18] R. Saxena, B. N. Nagpal, M. K. Das, A. Srivastava, S. K. Gupta, A. Kumar, A. T. Jeyaseelan, and V. K. Baraik, “A spatial statistical approach to analyze malaria situation at micro level for priority control in ranchi district, jharkhand.,” *Indian J Med Res*, vol. 136, no. 5, pp. 776–82, 2012.

[19] G. C. Kelly, E. Hale, W. Donald, W. Batarii, H. Bugoro, J. Nausien, J. Smale, K. Palmer, A. Bobogare, G. Taleo, A. Vallely, M. Tanner, L. S. Vestergaard, and A. C. Clements, “A high-resolution geospatial surveillance-response system for malaria elimination in solomon islands and vanuatu.,” *Malar J*, vol. 12, no. 1, p. 108, 2013.

[20] G. M. Vazquez-Prokopec, U. Kitron, B. Montgomery, P. Horne, and S. A. Ritchie, “Quantifying the spatial dimension of dengue virus epidemic spread within a tropical urban environment.,” *PLoS Negl Trop Dis*, vol. 4, no. 12, p. e920, 2010.

[21] H.-J. Chu, T.-C. Chan, and F.-J. Jao, “Gis-aided planning of insecticide spraying to control dengue transmission.,” *Int J Health Geogr*, vol. 12, no. 1, p. 42, 2013.

[22] S. Opisa, M. R. Odiere, W. G. Z. O. Jura, D. M. S. Karanja, and P. N. M. Mwinzi, “Malacological survey and geographical distribution of vector snails for schistosomiasis within informal settlements of kisumu city, western kenya.,” *Parasit Vectors*, vol. 4, p. 226, 2011.

[23] E. C. de Souza Gomes, O. B. Leal-Neto, J. Albuquerque, H. P. da Silva, and C. S. Barbosa, “Schistosomiasis transmission and environmental change: a spatio-temporal analysis in porto de galinhas, pernambuco - brazil,” *International Journal of Health Geographics*, vol. 11, pp. 1–11, November 2012.

[24] T. P. Pesigan, N. G. Hairston, J. J. Jauregui, E. G. Garcia, A. T. Santos, B. C. Santos, and A. A. Besa, “Studies on schistosoma japonicum infection in the philippines. 2. the molluscan host.,” *Bull World Health Organ*, vol. 18, no. 4, pp. 481–578, 1958.

[25] “Geographical Information Systems (GIS),” in *Encyclopedia of Computer Science and Technology, Revised Edition* (H. Henderson, ed.), pp. 208–209, New York: Facts on File, Inc., 2009.

[26] Google, Inc., “Google Earth: Frequently Asked Questions.” `https://support.google.com/earth/answer/187961`, January 2014. Accessed: 30 January 2014.

[27] ProgrammableWeb, “API dashboard.” `http://www.programmableweb.com/apis`, January 2014. Accessed: 30 January 2014.

[28] G. Svenneberg, *Beginning Google Maps API 3*. New York: Apress Media LLC, 2010.

[29] Y.-H. Wu and M.-C. Hung, “Non-connective linear cartograms for mapping traffic conditions,” *Cartographic Perspectives*, vol. 0, no. 65, 2012.

[30] Department of Health, Republic of the Philippines, “LGU Scoreboard (Provinces).” `http://www.maps.doh.gov.ph/scorecard.html`, 2009. Accessed: 30 January 2014.

[31] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, "Design patterns in smalltalk mvc," in *Design Patterns: Elements of Reusable Object-oriented Software*, pp. 14–16, Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1995.

[32] "Database Management System," in *Encyclopedia of Computer Science and Technology, Revised Edition* (H. Henderson, ed.), pp. 131–133, New York: Facts on File, Inc., 2009.

[33] K. Kemp, "Database Management System (DBMS)," in *Encyclopedia of Geographic Information Science*, A Sage Reference Publication, pp. 69–75, SAGE Publications, 2008.

[34] "PHP," in *Encyclopedia of Computer Science and Technology, Revised Edition* (H. Henderson, ed.), pp. 372–373, New York: Facts on File, Inc., 2009.

[35] M. N. DeMers, "Searching with SQL in any GIS," in *GIS for Dummies*, pp. 103–104, New York: Wiley Publishing, Inc., 2009.

[36] J. Valade, "Chapter 1: Introducion to PHP and MySQL," in *PHP & MySQL for Dummies, 3rd Edition*, pp. 9–20, Wiley Publishing Inc., 2007.

[37] EllisLab, Inc., "Codeigniter user guide version 2.1.4." `http://ellislab.com/codeigniter/user-guide/`, 2008. Accessed: 30 January 2014.

# XI. Appendix

## A. Source Code

### Cascading Style Sheets

Listing 1: switch.css

```
/* ========================================================================
 * bootstrap-switch - v3.0.0
 * http://www.bootstrap-switch.org
 * ========================================================================
 * Copyright 2012-2013 Mattia Larentis
 *
 * ========================================================================
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *     http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 * ========================================================================
 */

.bootstrap-switch {
  display: inline-block;
  cursor: pointer;
  border-radius: 4px;
  border: 1px solid;
  border-color: #cccccc;
  position: relative;
  text-align: left;
  overflow: hidden;
  line-height: 8px;
  -webkit-user-select: none;
  -moz-user-select: none;
  -ms-user-select: none;
  -o-user-select: none;
  user-select: none;
  vertical-align: middle;
  min-width: 100px;
  -webkit-transition: border-color ease-in-out .15s, box-shadow ease-in-out .15s;
  transition: border-color ease-in-out .15s, box-shadow ease-in-out .15s;
}
.bootstrap-switch.bootstrap-switch-mini {
  min-width: 72px;
}
.bootstrap-switch.bootstrap-switch-mini > div > span,
.bootstrap-switch.bootstrap-switch-mini > div > label {
  padding-bottom: 4px;
  padding-top: 4px;
  font-size: 10px;
  line-height: 9px;
}
.bootstrap-switch.bootstrap-switch-mini .bootstrap-switch-mini-icons {
  height: 1.20em;
  line-height: 9px;
  vertical-align: text-top;
  text-align: center;
  transform: scale(0.6);
  margin-top: -1px;
  margin-bottom: -1px;
}
.bootstrap-switch.bootstrap-switch-small {
  min-width: 80px;
}
.bootstrap-switch.bootstrap-switch-small > div > span,
.bootstrap-switch.bootstrap-switch-small > div > label {
  padding-bottom: 3px;
  padding-top: 3px;
  font-size: 12px;
  line-height: 18px;
}
.bootstrap-switch.bootstrap-switch-large {
  min-width: 120px;
}
.bootstrap-switch.bootstrap-switch-large > div > span,
.bootstrap-switch.bootstrap-switch-large > div > label {
  padding-bottom: 9px;
  padding-top: 9px;
  font-size: 16px;
  line-height: normal;
}
.bootstrap-switch.bootstrap-switch-animate > div {
  -webkit-transition: margin-left 0.5s;
  transition: margin-left 0.5s;
}
.bootstrap-switch.bootstrap-switch-on > div {
  margin-left: 0%;
}
.bootstrap-switch.bootstrap-switch-on > div > label {
  border-bottom-right-radius: 3px;
  border-top-right-radius: 3px;
}
.bootstrap-switch.bootstrap-switch-off > div {
  margin-left: -50%;
}
.bootstrap-switch.bootstrap-switch-off > div > label {
  border-bottom-left-radius: 3px;
  border-top-left-radius: 3px;
}
.bootstrap-switch.bootstrap-switch-disabled,
.bootstrap-switch.bootstrap-switch-readonly {
  opacity: 0.5;
  filter: alpha(opacity=50);
  cursor: default !important;
}
.bootstrap-switch.bootstrap-switch-disabled > div > span,
.bootstrap-switch.bootstrap-switch-readonly > div > span,
.bootstrap-switch.bootstrap-switch-disabled > div > label,
.bootstrap-switch.bootstrap-switch-readonly > div > label {
  cursor: default !important;
}
.bootstrap-switch.bootstrap-switch-focused {
  border-color: #66afe9;
  outline: 0;
  -webkit-box-shadow: inset 0 1px 1px rgba(0,0,0,.075), 0 0 8px rgba(102, 175, 233, 0.6);
  box-shadow: inset 0 1px 1px rgba(0,0,0,.075), 0 0 8px rgba(102, 175, 233, 0.6);
}
.bootstrap-switch > div {
  display: inline-block;
  width: 150%;
  top: 0;
  border-radius: 4px;
  -webkit-transform: translate3d(0, 0, 0);
  transform: translate3d(0, 0, 0);
}
.bootstrap-switch > div > span,
.bootstrap-switch > div > label {
  -webkit-box-sizing: border-box;
  -moz-box-sizing: border-box;
  box-sizing: border-box;
  cursor: pointer;
  display: inline-block !important;
  height: 100%;
  padding-bottom: 4px;
  padding-top: 4px;
  font-size: 14px;
  line-height: 20px;
}
.bootstrap-switch > div > span {
  text-align: center;
  z-index: 1;
  width: 33.333333333%;
}
.bootstrap-switch > div > span.bootstrap-switch-handle-on {
  color: #f00;
  border-bottom-left-radius: 3px;
  border-top-left-radius: 3px;
}
.bootstrap-switch > div > span.bootstrap-switch-handle-off {
  color: #000;
  background: #eeeeee;
  border-bottom-right-radius: 3px;
  border-top-right-radius: 3px;
}
.bootstrap-switch > div > span.bootstrap-switch-primary {
  color: #fff;
  background: #428bca;
}
.bootstrap-switch > div > span.bootstrap-switch-info {
  color: #fff;
  background: #5bc0de;
}
.bootstrap-switch > div > span.bootstrap-switch-success {
  color: #fff;
  background: #5cb85c;
}
.bootstrap-switch > div > span.bootstrap-switch-warning {
  background: #f0ad4e;
  color: #fff;
}
.bootstrap-switch > div > span.bootstrap-switch-danger {
  color: #fff;
  background: #d9534f;
}
.bootstrap-switch > div > span.bootstrap-switch-default {
  color: #000;
  background: #eeeeee;
}
.bootstrap-switch > div > label {
  text-align: center;
  margin-top: -1px;
  margin-bottom: -1px;
  z-index: 100;
  width: 33.333333333%;
  color: #333333;
  background: #ffffff;
}
.bootstrap-switch input[type='radio'],
.bootstrap-switch input[type='checkbox'] {
  position: absolute !important;
  top: 0;
  left: 0;
  z-index: -1;
}
```

login.css

## Listing 2: login.css

```
body{
    padding-top: 40px;
    padding-bottom: 40px;
    background-color: #eee;
}

form{
    max-width: 330px;
    padding: 15px;
    margin: 0 auto;
}
h2, .checkbox{
    margin-bottom: 10px;
}

.checkbox{
    font-weight: normal;
}

.form-control{
    position: relative;
    height: auto;
    -webkit-box-sizing: border-box;
    -moz-box-sizing: border-box;
    box-sizing: border-box;
    padding: 10px;
    font-size: 16px;
}

.form-control:focus {
    z-index: 2;
}

input[type="username"] {
    margin-bottom: -1px;
    border-bottom-right-radius: 0;
    border-bottom-left-radius: 0;
}
input[type="password"] {
    margin-bottom: 10px;
    border-top-left-radius: 0;
    border-top-right-radius: 0;
}
```

map.css

## Listing 3: map.css

```
html, body {height:100%;}
body{
    margin: 0;
    padding-top:50px;
}

/*
 * Sidebar
 */

@media (min-width: 768px) {
    .sidebar {
        position: fixed;
        top: 51px;
        bottom: 0;
        left: 0;
        z-index: 1000;
        display: block;
        padding: 20px;
        overflow-x: hidden;
        overflow-y: auto; /* Scrollable contents if viewport is shorter than content. */
        background-color: #f5f5f5;
        border-right: 1px solid #eee;
    }
}

/* Sidebar navigation */
.nav-sidebar {
    margin-right: -21px; /* 20px padding + 1px border */
    margin-bottom: 20px;
    margin-left: -20px;
}
```

```
.nav-sidebar > li > a {
    padding-right: 20px;
    padding-left: 20px;
}
.nav-sidebar > .active > a {
    color: #fff;
    background-color: #428bca;
}

.bubble
{
    position: relative;
    width: 400px;
    height: 150px;
    padding: 10px;
    background: #eeeeee;
    -webkit-border-radius: 10px;
    -moz-border-radius: 10px;
    border-radius: 10px;
}

.bubble:after
{
    content: '';
    position: absolute;
    border-style: solid;
    border-width: 15px 20px 0;
    border-color: #eeeeee transparent;
    display: block;
    width: 0;
    z-index: 1;
    bottom: -15px;
    left: 172px;
}

#map_canvas{
    margin-top: -20px;
}
```

style.css

Listing 4: style.css

```
/* Sticky footer styles
-------------------------------------------------- */
html {
    position: relative;
    min-height: 100%;
}
body {
    /* Margin bottom by footer height */
    margin-bottom: 60px;
}
#footer {
    position: absolute;
    bottom: 0;
    width: 100%;
    /* Set the fixed height of the footer here */
    height: 60px;
    background-color: #f5f5f5;
}
```

## Javascript

Listing 5: colony_add.js

```
var success = function(pos){
    var lat = pos.coords.latitude,
        long= pos.coords.longitude;
    $("#longitude").attr("readonly", "readonly");
    $("#latitude").attr("readonly", "readonly");
    $("#longitude").val(long);
    $("#latitude").val(lat);
    $("#loading-icon").addClass("hidden");        //hide loading icon
}
function error(){
    alert("could not find your location");
}

function getLocation(){
    if(this.checked){
        $("#loading-icon").removeClass("hidden");    //show loading icon
        navigator.geolocation.getCurrentPosition(success,error);
    }else{
        $("#loading-icon").addClass("hidden");
        $("#longitude").removeAttr("readonly");
        $("#latitude").removeAttr("readonly");
    }

    event.preventDefault();
}
$("#gps_checkbox").bootstrapSwitch();
$("#gps_checkbox").on("switchChange",getLocation);
```

Listing 6: map.js

```
var data_status = $("input[name='data_status']").val().split(",");
console.log(data_status);
function zoom_to_place(){
    var address = $("#zoom").val();
    var geocoder = new google.maps.Geocoder();
    $("body").css("cursor", "wait");    // show loading cursor
    // Get LatLng information by name
    geocoder.geocode({
        "address": address
    }, function(results, status){
        map.setCenter(results[0].geometry.location);
        map.setZoom(8);
        $("body").css("cursor", "default");
    });
}

var mouseX;
var mouseY;
var show_bubble = false;
$(document).mousemove( function(e) {
    mouseX = e.pageX;
    mouseY = e.pageY;
    if(show_bubble) $('#info-box').css({'top':mouseY-175,'left':mouseX-$(".bubble").width()
        /2});
});
function show_details(title,area,infection){
    $("#info-box #title").html(title);
    $("#info-box #area").html(area);
    if(infection!=null){
        var infection_span = "<span id='infection-span'></span>"
        $("#info-box #infection-form").removeClass("hidden");
        $("#info-box #infection-container").html(infection_span);
        if(infection>66){
            $("#info-box #infection-span").addClass("text-danger");
        }else if(infection>33){
            $("#info-box #infection-span").addClass("text-warning");
        }else{
            $("#info-box #infection-span").addClass("text-success");
        }
        $("#info-box #infection-span").html(infection + "%");
        $("#info-box #default").addClass("hidden");
    }else{
        $("#info-box #infection-form").addClass("hidden");
        $("#info-box #default").removeClass("hidden");
    }

    $("#info-box").removeClass("hidden");
    $(".bubble:after").css({"left":$(".bubble").width()/2});
    show_bubble = true;
}
function hide_details(){
    $('#info-box').addClass("hidden");
    show_bubble = false;
}

$(".switch").bootstrapSwitch();
$("#map-params :input").change(function() {
    $("#notif").removeClass("hidden");
});
$(".switch").on("switchChange",function() {
    if(this.checked){
        data_status.push(this.id);
    }else if(data_status.indexOf(this.id)!=-1){
        data_status.splice(data_status.indexOf(this.id),1);
    }

    $("input[name='data_status']").val(data_status.toString())
    $("#notif").removeClass("hidden");
});
```

## Views

Listing 7: footer.php

```
<br/>
<div id="footer">
    <div class="container">
        <p class="text-muted pull-right" style="padding-top:20px;">&copy; Copyright 2014</p>
    </div>
</div>
</body>
</html>
```

Listing 8: header.php

```
<!DOCTYPE html>
<html lang="en">
<head>
    <!-- Meta, title, CSS, favicons, etc. -->
    <meta charset="utf-8">
    <link rel="icon" href="<?php echo base_url("/favicon.ico")?>">
    <title><?php if(isset($title)) echo $title . " | "; ?>Schistosomiasis GIS</title>
    <?php echo link_tag(’assets/css/bootstrap.min.css’); ?>
    <?php echo link_tag(’assets/css/style.css’); ?>
    <!-- HTML5 shim and Respond.js IE8 support of HTML5 elements and media queries -->
    <!--[if lt IE 9]>
    <script src=”https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js”></script>
    <script src=”https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js”></script>
     <![endif]-->
    <script src=”<?php echo base_url(); ?>assets/js/jquery-2.0.3.min.js”></script>
    <script src=”<?php echo base_url(); ?>assets/js/bootstrap.js”></script>
</head>
<body style=”padding-top:70px;”>
<noscript><h2 class=”text-center jumbotron”><span class=”glyphicon glyphicon-warning-sign text
    -warning”></span> This website needs javascript to work properly. Please enable javascript
    .</h2><style>div{display:none;}</style></noscript>
    <!-- navbar -->
    <div class=”navbar navbar-default navbar-fixed-top” role=”navigation”>
        <div class=”container”>
            <div class=”navbar-header”>
                <button type=”button” class=”navbar-toggle” data-toggle=”collapse” data-target
                    =”.navbar-collapse”>
                    <span class=”sr-only”>Toggle Navigation</span>
                    <span class=”icon-bar”></span>
                    <span class=”icon-bar”></span>
                    <span class=”icon-bar”></span>
                </button>
                <a class=”navbar-brand” href=”#”>Schistosomiasis GIS</a>
            </div>
            <div class=”navbar-collapse collapse”>
                <ul class=”nav navbar-nav”>
                    <?php if(    in_array(1,$this->session->userdata(”user_roles”))||
                                 in_array(2,$this->session->userdata(”user_roles”))||
                                 in_array(3,$this->session->userdata(”user_roles”))){  //if
                                     user is a data manager, gov official, health worker ?>
                    <li<?php if(strtolower($this->uri->segment(1))==”map”) echo ’ class=”
                         active”’ ?>><a href=”<?php echo base_url(’map’); ?>”>Map</a></li>
                    <?php
                    }
                    ?>
                    <?php if(    in_array(2,$this->session->userdata(”user_roles”))||
                                 in_array(3,$this->session->userdata(”user_roles”))){  //if
                                     user is a data manager or health worker ?>
                    <li class=”dropdown <?php if(strtolower($this->uri->segment(1))==”colony
                        ”||strtolower($this->uri->segment(1))==”sample”) echo ’active’ ?>”><a
                        class=”dropdown-toggle” data-toggle=”dropdown” href=”#”>Snail Sampling
                        </a>
                        <ul class=”dropdown-menu” role=”menu”>
                            <li><a href=”<?php echo base_url(’colony’); ?>”><span class=”
                                 glyphicon glyphicon-list-alt”></span> View Colony List</a></li
                                 >
                            <li><a href=”<?php echo base_url(’colony/add’); ?>”><span class=”
                                 glyphicon glyphicon-plus”></span> Add New Colony</a></li>
                        </ul>
                    </li>
                        <?php if(in_array(3,$this->session->userdata(”user_roles”))){ ?>
                    <li<?php if(strtolower($this->uri->segment(1))==”cases”) echo ’ class=”
                         active”’ ?>><a href=”<?php echo base_url(’cases’); ?>”>Cases</a></li>
                            <?php } ?>
                    <?php
                    }
                    ?>
                    <?php if(in_array(3,$this->session->userdata(”user_roles”))){  //if user
                         is a data manager ?>
                    <li<?php if(strtolower($this->uri->segment(1))==”synchronize”) echo ’
                         class=”active”’ ?>><a href=”<?php echo base_url(’synchronize’); ?>”>
                         Synchronize</a></li>
                    <?php
                    }
                    ?>
                    <?php if(in_array(4,$this->session->userdata(”user_roles”))){  //if user
                         is admin ?>
                    <li class=”dropdown <?php echo (strtolower($this->uri->segment(1))==”user
                        ”)? ”active” : ”” ?>”>
                        <a href=”#” class=”dropdown-toggle” data-toggle=”dropdown”>Users
                        </a>
                        <ul class=”dropdown-menu” role=”menu”>
                            <li><a href=”<?php echo base_url(’user’); ?>”><span class=”
                                 glyphicon glyphicon-list-alt”></span> View User List</a></li>
                            <li><a href=”<?php echo base_url(’user/add’); ?>”><span class=”
                                 glyphicon glyphicon-plus”></span> Add New User</a></li>
                        </ul>
                    <?php
                    }
                    ?>
                </ul>
                <ul class=”nav navbar-nav navbar-right”>
                    <li class=”dropdown”>
                        <a href=”#” class=”dropdown-toggle” data-toggle=”dropdown”>
                            <span class=”glyphicon glyphicon-user”></span>&nbsp;&nbsp;<?php
                                 echo  $this->session->userdata(’username’); ?> <b class=”caret
                                 ”></b>
                        </a>
                        <ul class=”dropdown-menu dropdown-menu-right” role=”menu”>
                            <li><a href=”<?php echo base_url(). ’user/view/’ . $this->session
                                 ->userdata(’user_id’)?>”><span class=”glyphicon glyphicon-cog
                                 text-info”></span> My Account</a></li>
                            <li><a href=”<?php echo base_url(’user/logout’); ?>”><span class=”
                                 glyphicon glyphicon-log-out help-block”></span> Log Out</a></
                                 li>
                        </ul>
                </li>
            </ul>

        </div>
    </div>
</div>
```

## Listing 9: sample_add.php

```
<?php echo link_tag('assets/css/bootstrap-datetimepicker.min.css'); ?>
<?php echo link_tag('assets/css/bootstrap-tags.css'); ?>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<div class="row">
    <div class="col-md-3"></div>
    <div class="col-md-6">
        <?php
        echo form_open(base_url($action), array("role"=>"form"));
        echo form_hidden("colony_id",$colony_id);
        echo form_hidden("site_name",$site_name);
        if(isset($validation_errors)){
            ?>
            <div class="alert alert-danger text-center">
                <?php echo $validation_errors; ?>
            </div>
        <?php
        }
        echo '<h2><a href="'.base_url().'colony/view/'.$colony_id.'" data-toggle="tooltip"
             data-placement="top" title="View colony information">'.rawurldecode($site_name)
             .'</a> <small> '. $task . ' Sampling Data</small></h2>';
        echo form_hidden("colony_id",$colony_id);

        ?>
        <div class="form-group">
            <label for="date_time">Date and time of collection</label>
            <div class="row">
                <div class="col-md-6">
                    <div class="input-group" id="datepicker">
                        <?php echo form_input(array("id" => "date_time", "name" => "date_time
                             ","required" => "","class"=>"form-control","placeholder"=>"Date
                             and time collected","data-format"=>"YYYY-MM-DD HH:mm:ss","readonly
                             "=>"true"),$sampledata["date_time"])?>
                        <span class="input-group-btn">
                            <button class="btn btn-default" type="button"><span class="
                                 glyphicon glyphicon-calendar"></span></button>
                        </span>
                    </div><!-- /input-group -->
                </div><!-- /.col-lg-6 -->
                <div class="col-md-6">
                    <a href="#" class="btn btn-default col-sm-12" id="use_current">Use current
                         date and time</a>
                </div><!-- /.col-lg-6 -->
            </div>
        </div>
        <div class="form-group">
            <label for="total_count">Total count</label>
            <?php echo form_input(array("id" => "total_count", "name" => "total_count","
                 required" => "","class"=>"form-control","placeholder"=>"Total number of snails
                  collected"),$sampledata["total_count"])?>
            <label for="live_count">Live count</label>
            <?php echo form_input(array("id" => "live_count", "name" => "live_count","required
                 " => "","class"=>"form-control","placeholder"=>"Number of living snails in the
                  collection"),$sampledata["live_count"])?>
            <label for="infected_count">Infected count</label>
            <?php echo form_input(array("id" => "infected_count", "name" => "infected_count","
                 required" => "","class"=>"form-control","placeholder"=>"Number of infected
                 snails in the collection"),$sampledata["infected_count"])?>
        </div>
        <div class="form-group">
            <label for="vegetation">Vegetation</label>
            <?php echo form_dropdown("vegetation", array(""=>"Select a description of the area
                 's vegetation","sparse"=>"Sparse","moderate"=>"Moderate","dense"=>"Dense"),
                 $sampledata["vegetation"],'class="form-control"');?>
            <label for="exposure">Exposure</label>
            <?php echo form_dropdown("exposure", array(""=>"Select a description of the area's
                  exposure","no shade"=>"No Shade","part in shade"=>"Part in Shade","whole in
                 shade"=>"Whole in Shade"),$sampledata["exposure"],'class="form-control"');?>
            <label for="weather">Weather</label>
            <?php echo form_textarea(array("id" => "weather", "name" => "weather","class"=>"
                 form-control","placeholder"=>"Brief description of the weather"),$sampledata["
                 weather"])?>
        </div>

        <div class="form-group">
            <label for="collectors">Collectors</label><br/>
            <div id="users-list" class="tag-list"><div class="tags"></div></div>
            <p class="help-block">Press "Enter" after each username.</p>
        </div>
        <?php echo form_input(array("id" => "collectors_names", "name" => "collectors_names","
             class"=>"hidden"),$sampledata["collectors_names"])?>

        <?php
        echo form_button(array("id"=>"submit","name"=>"submit","type"=>"submit","class"=>"btn
             btn-primary col-lg-2 col-lg-offset-10","content"=>"<span class='glyphicon
             glyphicon-floppy-disk'></span> Save"));
        echo form_close();
        ?>
    </div>
```

```
</div>
<script src=”<?php echo base_url()?>assets/js/moment-2.4.0.js”></script>
<script src=”<?php echo base_url()?>assets/js/bootstrap-datetimepicker.js”></script>

<script src=”<?php echo base_url()?>assets/js/bootstrap-tags.min.js”></script>
<script>
    $(function () {
        $(’#users-list’).tags({
            tagData:           [<?php echo $sampledata[”collectors_names”];?>],
            suggestions:    <?php echo json_encode($users_list); ?>,
            restrictTo:     <?php echo json_encode($users_list); ?>,
            promptText:     ”Enter usernames of collectors”,
            tagClass:       ”btn-primary”,
            afterAddingTag: function(tag){
                $(”#collectors_names”).val(function(i,val) {
                    return val + (val ? ’, ’ : ’’ ) + ’\”’ + tag +’\”’;
                });
            },
            afterDeletingTag: function(tag){
                $(”#collectors_names”).val(function(i,val) {
                    return val.replace(’\”, \”’,’\”\”’);
                });
                $(”#collectors_names”).val(function(i,val) {
                    return val.replace(’\”’+tag+’\”’,’’);
                });

                $(”#collectors_names”).val(function(i,val) {
                    return val.replace(’\”\”’,’\”, \”’);
                });
            }
        });
        $(’#users-list’).keypress(function(event) {if(event.keyCode==13) event.preventDefault
            (); });

        $(’#datepicker’).datetimepicker();
    });
    $(”#use_current”).click(
        function(){
            change_date();
            event.preventDefault();
        }
    );
    function change_date(){
        var currentdate = new Date();
        var currDay = padZero(currentdate.getDate());
        var currMonth = padZero(currentdate.getMonth()+1);
        var currHour = padZero(currentdate.getHours());
        var currMin = padZero(currentdate.getMinutes());
        var currSec = padZero(currentdate.getSeconds());

        $(”#date_time”).val(currentdate.getFullYear() + ”-”
            + currMonth + ”-”
            + currDay   + ” ”
            + currHour  + ”:”
            + currMin   + ”:”
            + currSec
        );
    }
    function padZero(num){
        return num>9? num : ”0”+ num.toString();
    }
</script>
```

## Listing 10: sample_all.php

```
<?php echo link_tag(’assets/css/DT_bootstrap.css’); ?>

<?php
echo form_open(””, array(”role”=>”form”));
?>
<div class=”row”>
    <div class=”col-lg-4 col-lg-offset-3”>
        <div class=”input-group”>
            <input type=”text” class=”form-control” placeholder=”Search for sampling data” id
                =”sample-search” name=”site_name”>
            <span class=”input-group-btn”>
                <button class=”btn btn-default” type=”submit” name=”search” value=”search”><
                    span class=”glyphicon glyphicon-search”></span></button>
            </span>
        </div>
    </div>
    <div class=”col-lg-2”>
        <button class=”btn btn-default col-md-12” type=”submit” name=”add” value=”add”><span
            class=”glyphicon glyphicon-plus-sign”></span> Add New Sampling Data</button>
    </div>
    <div class=”col-lg-8 col-lg-offset-2”>
    <h3><a href=”<?php echo base_url(’colony/view/’.$this->uri->segment(3));?>” data-toggle=”
        tooltip” data-placement=”top” title=”View colony information”><?php echo $sitename
        ?></a><small> Collected Samples</small></h3>
        <?php
        if(in_array(3,$this->session->userdata(”user_roles”))){
        ?>
        <div class=”panel-group” id=”accordion”>
            <div class=”panel panel-default”>
                <a data-toggle=”collapse” data-parent=”#accordion” href=”#collapseTwo”>
                    <div class=”panel-heading”>
```

```
                        <h4 class="panel-title"><small>Update Multiple <span class="caret"></
                             span></small></h4>
                    </div>
                </a>
                <div id="collapseTwo" class="panel-collapse collapse<?php echo strlen($this->
                     input->post("filter"))? " in": ""?>">
                    <div class="panel-body">
                        <div class="row">
                            <div class="col-lg-8 col-lg-offset-2">
                                <div class="alert alert-warning alert-dismissable hidden" id="
                                     update-error">
                                    <button type="button" class="close" onclick="$('#update-
                                         error').addClass('hidden');" aria-hidden="true">&times
                                         ;</button>
                                    You haven't selected any data yet.
                                </div>
                            </div>
                            <div class="col-lg-5 col-lg-offset-2">
                                <?php echo form_dropdown("update_action",array("Move to Queue
                                     ","Move to Final"),"Move to Queue","class='form-control'
                                     id='update_action'"); ?>
                                <span class="help-block">Changes will be applied to data with
                                     checked boxes.</span>
                            </div>
                            <div class="col-lg-3">
                                <?php
                                echo form_button(array("content"=>"Apply Changes","class"=>'
                                     btn btn-default col-lg-12',"onclick"=>"update_multiple()
                                     ;"));
                                ?>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
        <?php } ?>
    </div>
    <div class="col-lg-8 col-lg-offset-2">
        <div class="table-responsive">
            <table class="table table-striped table-hover">
                <thead>
                    <tr>
                        <?php if(in_array(3,$this->session->userdata("user_roles"))&&count(
                             $samples)){ ?>
                            <th class="sorting_disabled"><?php echo form_checkbox('checkall',
                                 'checkall', FALSE, 'id="checkall"')?></th>
                        <?php } ?>
                        <th>Date of Collection</th>
                        <th>Collectors</th>
                        <th>Data Status</th>
                        <th class="sorting_disabled">Actions</th>
                    </tr>
                </thead>
                <tbody id="sample-list">
                <?php
                for($i=0;$i<count($samples);$i++){
                    $num = $i+1;
                    $c_id = $samples[$i]->colony_id.'/'.rawurlencode($samples[$i]->date_time);
                    echo    "<tr id='".$c_id."'>";
                    if(in_array(3,$this->session->userdata("user_roles")))
                        echo        "<td>" .form_checkbox('selected_samples[]',$samples[$i]->
                             date_time,false,'id ="'.$i.'_checkbox"')."</td>";
                    echo        '<td><b>'.$samples[$i]->date_time.'</b></td>';
                    echo    "<td>". ((strlen($samples[$i]->collectors_names))?$samples[$i]->
                         collectors_names : "No collectors provided") ."</td>";
                    echo    "<td>".$samples[$i]->status."</td>";
                    echo    "<td>";
                    echo            "<a href='". base_url() ."sample/view/".$c_id."'><span
                         class='glyphicon glyphicon-eye-open' title='View'></span></a>&nbsp;<a
                         href='". base_url() ."sample/edit/".$samples[$i]->colony_id.'/'.
                         rawurlencode($sitename).'/'.rawurlencode($samples[$i]->date_time)."'><
                         span class='glyphicon glyphicon-edit' title='Edit'></span></a>";

                    if(in_array(3,$this->session->userdata("user_roles"))){
                        echo    "&nbsp;<a href='#' data-toggle='modal' data-target='#move-data
                             '";
                        echo    "   onclick = 'update_single(".$samples[$i]->colony_id.",\"".
                             $i;
                        echo    "\",\"".$samples[$i]->date_time."\",\"".$samples[$i]->status
                             ."\",\"$sitename\")' title='Change status'><span class='glyphicon
                             glyphicon-transfer'></span></a>";
                    echo    "</td>";
                    }
                    echo    '</tr>';
                }
                if(count($samples)==0) echo "<tr><td colspan='3' class='text-center text-muted
                     '><em>No samples found for '$sitename'</em></td></tr>";
                ?>
                </tbody>
            </table>

        </div>
    </div>
</div>
<div class="modal fade" id="move-data" tabindex="-1" role="dialog" aria-labelledby="Confirm
     action" aria-hidden="true">
    <div class="modal-dialog">
        <div class="modal-content">
```

```
            <div class="modal-header">
                <button type="button" class="close" data-dismiss="modal" aria-hidden="true">
                     </button>
                <h4 class="modal-title">Confirm action</h4>
            </div>
            <div class="modal-body">
                <p id="message"></p>
            </div>
            <div class="modal-footer">
                <?php
                echo form_button(array("data-dismiss"=>"modal","content"=>"Cancel","class"=>'
                     btn btn-default'));
                echo form_button(array("name"=>"update_form","id"=>"update_form","content"=>"
                     Continue","value"=>"update","class"=>'btn btn-primary',"type"=>"submit"));
                ?>
            </div>
        </div>
    </div>
</div>

<?php
echo form_close();
?>
<?php if(count($samples)!=0){ ?>
    <script src="<?php echo base_url(); ?>assets/js/jquery.dataTables.min.js"></script>
    <script src="<?php echo base_url(); ?>assets/js/my_dataTables.js"></script>
    <script src="<?php echo base_url()?>assets/js/jquery.fastLiveFilter.js"></script>
    <script>
        $(function() { $('#sample-search').fastLiveFilter('#sample-list'); });
    </script>
    <?php
        if(in_array(3,$this->session->userdata("user_roles"))){
    ?>
        <script>
            $('#checkall').on('click', function () {
                $('tr input:checkbox').prop('checked', this.checked);
            });
            function update_multiple(){
                var inputs = $("td input:checked");
                if(inputs.length==0){
                    $("#update-error").removeClass("hidden");
                }else{
                    $("#update-error").addClass("hidden");
                    $(".modal").modal();
                    var mark = $("#update_action").val()==0? "queued" : "final";
                    var msg =   "The following data will be marked as <span class='text-info
                         '>" + mark + "</span>:"
                        +   "<div class='well'><ul>";
                    for(var i=0;i<inputs.length;i++){
                        msg += "<li>" + inputs[i].value + "</li>";
                    }
                    msg +=   "</ul></div>";
                    $("#message").html(msg);
                }
            }
            function update_single(id,iter,date,status,name){
                var mark;
                if(status=="queued"){
                    mark = "final";
                    $("#update_action").val(1);
                }else{
                    mark = "queued";
                    $("#update_action").val(0);
                }
                $('tr input:checkbox').prop('checked', false);  // uncheck others
                $("#"+iter+"_checkbox").prop("checked",true)
                $("#message").html("You are about to mark data collected on <strong>" + date +
                      "</strong> for '" + name + "' as <span class='text-info'>" + mark + "</
                     span>.")
            }
        </script>
    <?php } ?>
<?php } ?>
```

## Listing 11: sample_id.php

```
<div class="row">
    <div class="col-md-3"></div>
    <div class="col-md-6">
        <?php
        if(isset($success_msg)&&$success_msg!=-1){
            echo '<div class="alert alert-success">';
            echo "<span class='glyphicon glyphicon-ok-sign'></span>";
            echo '<span id="success-msg"> ';
            switch($success_msg){
                case 1:
                    echo "You have successfully added sample data for '$sitename' snail colony
                         .";
                    break;
                case 2:
                    echo "The changes you have made to sample data for '$sitename' snail
                         colony have been saved.";
                    break;
            }
            echo '</span>';
            echo "</div>";
        }
        ?>
        <h2><a href=”<?php echo base_url(’colony/view/’.$sampledata->colony_id)?>”><?php echo
             $sitename ?></a> <small>Sample Information</small></h2>
        <table class=”table”>
            <tr>
                <td><b>Date and time of collection</b></td>
                <td class=”text-right”><?php  echo $sampledata->date_time?></td>
            </tr>
            <tr>
                <td><b>Total number of snails</b></td>
                <td class=”text-right”><?php echo $sampledata->total_count?></td>
            </tr>
            <tr>
                <td><b>Live snails</b></td>
                <td class=”text-right”><?php echo $sampledata->live_count ?></td>
            </tr>
            <tr>
                <td><b>Infected snails</b></td>
                <td class=”text-right”><?php echo $sampledata->infected_count ?></td>
            </tr>
            <tr>
                <td><b>Vegetation</b></td>
                <td class=”text-right”><?php echo $sampledata->vegetation ?></td>
            </tr>
            <tr>
                <td><b>Exposure</b></td>
                <td class=”text-right”><?php echo $sampledata->exposure ?></td>
            </tr>
            <tr>
                <td><b>Weather</b></td>
                <td class=”text-right”><?php echo $sampledata->weather ?></td>
            </tr>
            <tr>
                <td><b>Collectors</b></td>
                <td class=”text-right”><?php echo $collectors_names ?></td>
            </tr>
        </table>
        <?php
        echo form_open(”edit_view_list_add”);
        echo form_hidden(”colony_id”,$sampledata->colony_id);
        echo form_hidden(”site_name”,$sitename);
        echo form_hidden(”date_time”,$sampledata->date_time);
        ?>
        <div class=”btn-group btn-group-justified”>
            <a href=”<?php echo base_url(); ?>sample/edit/<?php echo $sampledata->colony_id
                 .”/”. rawurlencode($sitename) .”/”.rawurlencode($sampledata->date_time); ?>”
                 class=”btn btn-default”><span class=”glyphicon glyphicon-edit”></span> Edit</a
                 >
            <!--<a href=”<?php echo base_url(); ?>map/view/<?php echo $sampledata->colony_id
                 .”/”.rawurlencode($sampledata->date_time) ?>” class=”btn btn-default”><span
                 class=”glyphicon glyphicon-map-marker”></span> View on map</a>-->
            <a href=”<?php echo base_url(); ?>sample/view_all/<?php echo $sampledata->
                 colony_id?>” class=”btn btn-default”><span class=”glyphicon glyphicon-list-alt
                 ”></span> View list of samples</a>
            <a href=”<?php echo base_url(); ?>sample/add/<?php echo $sampledata->colony_id
                 .”/”. rawurlencode($sitename); ?>” class=”btn btn-default”><span class=”
                 glyphicon glyphicon-plus”></span> Add sampling data</a>
        </div>
        <?php echo form_close(); ?>
    </div>
</div>
```

## Listing 12: colony_add.php

```
<?php echo link_tag(’assets/css/bootstrap-switch.css’); ?>
<div class=”row”>
    <div class=”col-md-3”></div>
    <div class=”col-md-6”>
        <h1><small><?php echo $title?></small></h1>
    <?php
    echo form_open(base_url($action), array(”role”=>”form”));
    echo form_hidden(”colony_id”,$colony[”colony_id”]);
    if(isset($validation_errors)){
        ?>
        <div class=”alert alert-danger text-center”>
            <?php echo $validation_errors;           ?>
        </div>
    <?php
    }
    ?>
    <div class=”form-group”>
        <label for=”site_name”>Site Name</label>
        <?php echo form_input(array(”id” => ”site_name”, ”name” => ”site_name”, ”required” =>
             ”true”,”class”=>”form-control”,”placeholder”=>”Site Name”),$colony[”site_name”])?>
    </div>
    <div class=”form-group”>
        <div class=”form-horizontal”>
            <span class=”help-block”>GPS Location:</span>
            <label class=”control-label” for=”gps_checkbox”>Use current location</label>
            <input type=”checkbox” id=”gps_checkbox” data-on-text=”<span class=’glyphicon
                 glyphicon-ok’></span>” data-off-text=”<span class=’glyphicon glyphicon-remove
                 ’></span>” data-label-text=”<span class=’glyphicon glyphicon-screenshot’></
                 span>”>&nbsp;<img src=”<?php echo base_url();?>assets/images/loading.gif”
                 class=”hidden” id=”loading-icon”/>
        </div>
        <div class=”form-group row”>
            <div class=”col-lg-6”>
```

```
                <label for="longitude">Longitude</label>
                <?php echo form_input(array("id" => "longitude", "name" => "longitude", "
                     required" => "true","class"=>"form-control","placeholder"=>"Longitude
                     Coordinate"),$colony["longitude"])?>
            </div>
            <div class="col-lg-6">
                <label for="latitude">Latitude</label>
                <?php echo form_input(array("id" => "latitude", "name" => "latitude", "
                     required" => "true","class"=>"form-control","placeholder"=>"Latitude
                     Coordinate"),$colony["latitude"])?>
            </div>
        </div>
    </div>
    <div class="form-group">
        <label for="area">Area</label>
        <div class="input-group">
            <?php echo form_input(array("id" => "area", "name" => "area", "required" => "true
                 ","class"=>"form-control","placeholder"=>"Estimated area in square meters"),
                 $colony["area"])?>
            <span class="input-group-addon">m<sup>2</sup></span>
        </div>
    </div>
    <?php
        echo form_button(array("id"=>"submit","name"=>"submit","type"=>"submit","class"=>"btn
             btn-primary col-lg-2 col-lg-offset-10","content"=>"<span class='glyphicon
             glyphicon-floppy-disk'></span> Save"));
        echo form_close();
    ?>
    </div>
</div>
<script src="<?php echo base_url("assets/js/bootstrap-switch.js"); ?>"></script>
<script src="<?php echo base_url("assets/js/colony_add.js"); ?>"></script>
```

## Listing 13: colony_all.php

```
<?php echo link_tag('assets/css/DT_bootstrap.css'); ?>
<?php
echo form_open("", array("role"=>"form"));
?>
<div class="row">
    <div class="col-lg-4 col-lg-offset-3">
        <div class="input-group">
            <input type="text" class="form-control" placeholder="Search" id="colony-search"
                 name="site_name">
            <input type="hidden" id="search-term" name="search-term" value="<?php echo $term;
                 ?>">
            <span class="input-group-btn">
                <button class="btn btn-default" type="submit" name="search" value="search"><
                     span class="glyphicon glyphicon-search"></span></button>
            </span>
        </div>
    </div>
    <div class="col-lg-2"><button class="btn btn-default col-md-12" type="submit" name="add"
         value="add"><span class="glyphicon glyphicon-plus-sign"></span> Add New Colony</button
         ></div>
    <div class="col-lg-10 col-lg-offset-1">
        <h3>Snail Colonies</h3>
        <div class="panel-group" id="accordion">
            <div class="panel panel-default">
                <a data-toggle="collapse" data-parent="#accordion" href="#collapseOne">
                    <div class="panel-heading">
                        <h6 class="panel-title"><small>Filter List <span class="caret"></span
                             ></small></h6>
                    </div>
                </a>
                <div id="collapseOne" class="panel-collapse collapse<?php echo strlen($this->
                     input->post("filter"))||count($colonies)==0? " in": ""?>">
                    <div class="panel-body">
                        <div class="row">
                            <div class="col-lg-6 form-horizontal">
                                <label>Site Name</label>
                                <ul class="nav nav-pills">
                                    <?php
                                    foreach(range('A','Z') as $i){
                                        if($starting==$i){
                                            echo '<li class="active">';
                                        }else{
                                            echo '<li>';
                                        }
                                        echo '<a href="'.base_url().'colony/view_all/'.$i
                                             .'">'.$i.'</a>';
                                        echo '</li>';
                                    }
                                    if($starting=="ALL"){
                                        echo '<li class="active">';
                                    }else{
                                        echo '<li>';
                                    }
                                    echo '<a href="'.base_url().'colony/view_all/all">ALL</a
                                         >';
                                    echo '</li>';
                                    ?>
                                </ul>
                                <!--
                                <div class="form-group">
                                    <label class="col-sm-2 control-label">Region</label>
                                    <div class="col-sm-10">
                                    <?php
                                    $regions = array("any"=>"Any Region") + $regions;
                                    echo form_dropdown("region",$regions,"any","class='form-
                                         control'");
                                    ?>
                                    </div>
                                </div>
                                <div class="form-group">
                                    <label class="col-sm-2 control-label">Province</label>
                                    <div class="col-sm-10">
                                    <?php
                                //print_r($provinces);
                                $regions = array("Any Province");
                                echo form_dropdown("region",$regions,"Any Region","class='form
                                    -control'");
                                    ?>
                                    </div>
                                </div>
                                <div class="form-group">
                                    <label class="col-sm-2 control-label">Barangay</label>
                                    <div class="col-sm-10">
                                    <?php
                                    $regions = array("Any Barangay");
                                    echo form_dropdown("region",$regions,"Any Region","class='
                                         form-control'");
                                    ?>
                                    </div>
                                </div>
                                -->
                            </div>
                            <div class="col-lg-6">
                                <label class="col-sm-offset-2">Data Status</label>
                                <div class="form-group">
                                    <div class="col-sm-offset-2 col-sm-10">
                                        <div class="checkbox">
                                            <label>
                                                <?php
                                                echo form_checkbox('status[]', 'queued', (
                                                     $this->input->post("status"))? in_array("
                                                     queued",$this->input->post("status")):TRUE
                                                     ); ?> Queued
                                            </label>
                                        </div>
                                    </div>
                                </div>
                                <div class="form-group">
                                    <div class="col-sm-offset-2 col-sm-10">
                                        <div class="checkbox">
                                            <label>
                                                <?php
                                                echo form_checkbox('status[]', 'final', ($this
                                                     ->input->post("status"))? in_array("final
                                                     ",$this->input->post("status")):TRUE); ?>
                                                     Final
                                            </label>
                                        </div>
                                    </div>
                                </div>
                            </div>
                            <button class="btn btn-default col-lg-2 col-lg-offset-5" type="
                                 submit" name="filter" value="filter"><span class="glyphicon
                                 glyphicon-filter"></span> Filter</button>
                        </div>
                    </div>
                </div>
            </div>
            <?php
                if(in_array(3,$this->session->userdata("user_roles"))){
            ?>
            <div class="panel panel-default">
                <a data-toggle="collapse" data-parent="#accordion" href="#collapseTwo">
                    <div class="panel-heading">
                        <h4 class="panel-title"><small>Update Multiple <span class="caret"></
                             span></small></h4>
                    </div>
                </a>
                <div id="collapseTwo" class="panel-collapse collapse<?php echo strlen($this->
                     input->post("filter"))? " in": ""?>">
                    <div class="panel-body">
                        <div class="row">
                            <div class="col-lg-8 col-lg-offset-2">
                                <div class="alert alert-warning alert-dismissable hidden" id="
                                    update-error">
                                    <button type="button" class="close" onclick="$('#update-
                                         error').addClass('hidden');" aria-hidden="true">&times
                                         ;</button>
                                    You haven't selected any data yet.
                                </div>
                            </div>
                            <div class="col-lg-5 col-lg-offset-2">
                                <?php echo form_dropdown("update_action",array("Move to Queue
                                    ","Move to Final"),"Move to Queue","class='form-control'
                                    id='update_action'"); ?>
                                <span class="help-block">Changes will be applied to data with
                                    checked boxes.</span>
                            </div>
                            <div class="col-lg-3">
                                <?php
```

```
                                echo form_button(array("content"=>"Apply Changes","class"=>'
                                     btn btn-default col-lg-12',"onclick"=>"update_multiple()
                                     ;"));
                                ?>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
            <?php
                }
            ?>
        </div>
    </div>
    <div class="col-lg-10  col-lg-offset-1 table-responsive">
        <table class="table table-striped table-hover">
            <thead>
            <tr>
                <?php if(in_array(3,$this->session->userdata("user_roles"))){ ?>
                <th class="sorting_disabled"><?php echo form_checkbox('checkall', 'checkall',
                     FALSE, 'id="checkall"')?></th>
                <?php } ?>
                <th>Site Name</th>
                <th>Date Last Sampled</th>
                <th>Data Status</th>
                <th class="sorting_disabled"><strong>Actions</strong></th>
            </tr>
            </thead>
            <tbody id="colony-list">
            <?php
                foreach($colonies as $colony){
                    $c_id = $colony->colony_id;
                    echo    "<tr>";
                    if(in_array(3,$this->session->userdata("user_roles")))
                        echo        "<td>" .form_checkbox('selected_colonies[]',$colony->
                             site_name,false,'id ="'.$c_id.'_checkbox"')."</td>";
                    echo        "<td><strong>$colony->site_name</strong></td>";
                    echo        "<td>$colony->last_sample</td>";
                    echo        "<td>$colony->status</td>";
                    echo        "<td>";
                    echo            "<a href='". base_url() ."colony/view/".$c_id."'><span
                         class='glyphicon glyphicon-eye-open' title='View'></span></a>&nbsp;<a
                         href='". base_url() ."colony/edit/".$c_id."'><span class='glyphicon
                         glyphicon-edit' title='Edit'></span></a>";
                    if(in_array(3,$this->session->userdata("user_roles")))
                        echo    "&nbsp;<a href='#' data-toggle='modal' data-target='#move-data
                             ' onclick='update_single($c_id,\"$colony->status\",\"$colony->
                             site_name\")' title='Change status'><span class='glyphicon
                             glyphicon-transfer'></span></a>";
                    echo    "</td>";
                    echo "</tr>";
                }
                if(count($colonies)==0) echo "<tr><td colspan='5' class='text-center text-
                     muted'><em>No colonies found under the given filters</em></td></tr>";
            ?>
            </tbody>
        </table>
    </div>
</div>
<div class="modal fade" id="move-data" tabindex="-1" role="dialog" aria-labelledby="Confirm
     action" aria-hidden="true">
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-dismiss="modal" aria-hidden="true">
                     </button>
                <h4 class="modal-title">Confirm action</h4>
            </div>
            <div class="modal-body">
                <p id="message"></p>
            </div>
            <div class="modal-footer">
                <?php
                echo form_button(array("data-dismiss"=>"modal","content"=>"Cancel","class"=>'
                     btn btn-default'));
                echo form_button(array("name"=>"update_form","id"=>"update_form","content"=>"
                     Continue","value"=>"update","class"=>'btn btn-primary',"type"=>"submit"));
                ?>
            </div>
        </div>
    </div>
</div>
<?php
echo form_close();
?>
<?php if(count($colonies)!=0){ ?>
    <script src="<?php echo base_url(); ?>assets/js/jquery.dataTables.min.js"></script>
    <script src="<?php echo base_url(); ?>assets/js/my_dataTables.js"></script>
    <script src="<?php echo base_url()?>assets/js/jquery.fastLiveFilter.js"></script>
    <script>
        $(function() {
            $('#colony-search').fastLiveFilter('#colony-list',{
                callback: function(total){
                    $("#shown-results").html(total);
                }
            });
            $('#checkall').on('click', function () {
                $('tr input:checkbox').prop('checked', this.checked);
            });
        });
        $("th").not(".sorting_disabled").css("cursor","pointer");
        function update_multiple(){
            var inputs = $("td input:checked");
            if(inputs.length==0){
                $("#update-error").removeClass("hidden") ;
            }else{
                $("#update-error").addClass("hidden") ;
                $(".modal").modal();
                var mark = $("#update_action").val()==0? "queued" : "final";
                var msg =   "The following data will be marked as <span class='text-info'>" +
                     mark + "</span>:"
                            +   "<div class='well'><ul>";
                for(var i=0;i<inputs.length;i++){
                    msg += "<li>" + inputs[i].value + "</li>";
                }
                msg +=   "</ul></div>";
                $("#message").html(msg);
            }
        }
        function update_single(id,status,name){
            var mark;
            if(status=="queued"){
                mark = "final";
                $("#update_action").val(1);
            }else{
                mark = "queued";
                $("#update_action").val(0);
            }
            $('tr input:checkbox').prop('checked', false);
            $("#"+id+"_checkbox").prop("checked",true)
            $("#message").html("You are about to mark '" + name + "' as <span class='text-info
                '>" + mark + "</span>.");
        }
    </script>
<?php } ?>
<script>
    $("#search-keyword").click(
        function(){
            $("#user-search").val("");
            $("#search-term").val("");
            $("form").submit();
        }
    )
</script>
```

## Listing 14: colony_id.php

```
<div class="row">
    <div class="col-md-3"></div>
    <div class="col-md-6">
        <?php
        if(isset($success_msg)){
            echo '<div class="alert alert-success">';
            echo "<span class='glyphicon glyphicon-ok-sign'></span>";
            echo '<span id="success-msg"> ';
            switch($success_msg){
                case 1:
                    echo "You have successfully added '".$colony->site_name."' snail colony.";
                    break;
                case 2:
                    echo "The changes you have made to '".$colony->site_name."' snail colony
                        have been saved.";
                    break;
            }
            echo '</span>';
            echo "</div>";
        }
        ?>
        <h2><?php echo $colony->site_name ?> <small>Colony Information</small></h2>
        <table class="table">
            <tr>
                <td><b>Longitude</b></td>
                <td class="text-right"><?php echo $colony->longitude; ?></td>
            </tr>
            <tr>
                <td><b>Latitude</b></td>
                <td class="text-right"><?php echo $colony->latitude; ?></td>
            </tr>
            <tr>
                <td><b>Area</b></td>
                <td class="text-right"><?php echo $colony->area; ?> m<sup>2</sup></td>
            </tr>
            <tr>
                <td><b>Number of times sampled</b></td>
                <td class="text-right"><?php echo $colony->sample_times; ?></td>
            </tr>
        </table>
        <div class="btn-group btn-group-justified">
            <a href="<?php echo base_url(); ?>colony/edit/<?php echo $colony->colony_id?>"
                 class="btn btn-default"><span class="glyphicon glyphicon-edit"></span> Edit</a
                 >
            <a href="<?php echo base_url(); ?>map/view/<?php echo $colony->latitude . "/" .
                 $colony->longitude?>" class="btn btn-default"><span class="glyphicon glyphicon
                 -map-marker"></span> View on map</a>
            <a href="<?php echo base_url(); ?>sample/view_all/<?php echo $colony->colony_id?>"
                  class="btn btn-default"><span class="glyphicon glyphicon-list-alt"></span>
                 View sampling data</a>
            <a href=”<?php echo base_url(); ?>sample/add/<?php echo $colony->colony_id . ”/”.
                 rawurlencode($colony->site_name) ?>” class=”btn btn-default”><span class=”
                 glyphicon glyphicon-plus”></span> Add sampling data</a>
        </div>
    </div>
</div>
```

## Listing 15: sync_file.php

```
<div class=”row”>
    <?php
        echo form_open(””,array(”class”=>”form-horizontal”,”role”=>”form”));
    ?>
    <div class=”col-md-offset-2 col-md-8 alert alert-info”>
        <strong><?php echo $colony[”total”]; ?></strong> entries were found under the <strong>
             colony</strong> database.<br/>
        <?php echo $colony[”total”] - count($colony[”changed”]) - $colony[”success”]; ?>
             entries were unchanged.<br/>
        <span class=”text-success”><strong><?php echo $colony[”success”]; ?></strong> entries
             were successfully added to this database.<br/></span>
        <?php
            if(count($colony[”changed”])){
                echo form_hidden(”colonies”,json_encode($colony[”changed”]));
        ?>
        <span class=”text-warning”><strong><?php echo count($colony[”changed”]); ?></strong>
             discrepancies were found.</span>
            <div class=”form-group”>
                <label for=”colony_action” class=”col-sm-3 control-label”>Action</label>
                <div class=”col-sm-4”>
                    <?php echo form_input(”colony_action”,””, ’id = ”colony_action” readonly
                         class=”form-control” placeholder=”No action selected.”’) ?>
                </div>
                <a href=”#colony-actions” id=”showColony” class=”col-sm-3 btn btn-default”>
                     View data and select action.</a>
            </div>
        <?php } ?>
    </div>
    <div class=”col-md-offset-2 col-md-8 alert alert-info”>
        <strong><?php echo $sample[”total”]; ?></strong> entries were found under the <strong>
             sample</strong> database.<br/>
        <?php echo $sample[”total”] - count($sample[”changed”]) - $sample[”success”] ?>
             entries were unchanged.<br/>
        <span class=”text-success”><strong><?php echo $sample[”success”]; ?></strong> entries
             were successfully added to this database.<br/></span>
        <!--
        <?php if(count($sample[”changed”])){ ?>
        <span class=”text-warning”><strong><?php echo count($sample[”changed”]); ?></strong>
             discrepancies were found. <a href=”#” onclick=”showSample();”>View and select an
             action.</a><br/></span>
        <?php
        }
        ?>
        -->
    </div>
    <div class=”col-md-offset-1 col-md-10 table-responsive” id=”colony-table”>
        <table class=”table table-striped table-hover”>
            <thead>
                <tr>
                    <th>Source</th>
                    <th>Data ID</th>
                    <th>Site Name</th>
                    <th>Longitude</th>
                    <th>Latitude</th>
                    <th>Area</th>
                    <th>Data Status</th>
                </tr>
            </thead>
            <tbody>
            <?php
                foreach($colony[”changed”] as $c){
                    echo ”<tr>”;
                    echo ”<td>Online<br/>Offline</td>”;
                    echo ”<td>” . $c[”online”][’colony_id’] .”<br/>”. $c[”offline”][’
                         colony_id’]. ”</td>”;
                    echo ”<td>” . $c[”online”][’site_name’] .”<br/>”. $c[”offline”][’
                         site_name’]. ”</td>”;
                    echo ”<td>” . $c[”online”][’longitude’] .”<br/>”. $c[”offline”][’
                         longitude’]. ”</td>”;
                    echo ”<td>” . $c[”online”][’latitude’] .”<br/>”. $c[”offline”][’latitude
                         ’]. ”</td>”;
                    echo ”<td>” . $c[”online”][’area’] .”<br/>”. $c[”offline”][’area’]. ”</td
                         >”;
                    echo ”<td>” . $c[”online”][’status’] .”<br/>”. $c[”offline”][’status’].
                         ”</td>”;
                    echo ”</tr>”;
                }
            ?>
            </tbody>
        </table>
        <div class=”pull-right” id=”colony-actions”>
            <a href=”#” class=”btn btn-default” onclick=”$(’#colony_action’).val(’Discard
                 Changes’);hideTables();”>Discard Changes</a>
            <a href=”#” class=”btn btn-default” onclick=”$(’#colony_action’).val(’Apply
                 Changes’);hideTables();”>Apply Changes</a>
        </div>
    </div>
    <!--
    <div class=”col-md-offset-1 col-md-10 table-responsive” id=”sample-table”>
        <table class=”table table-striped table-hover”>
            <thead>
            <tr>
                <th>Source</th>
                <th>Data ID</th>
                <th>Site Name</th>
                <th>Total Snails</th>
                <th>Live Snails</th>
                <th>Infected Snails</th>
                <th>Vegetation</th>
                <th>Exposure</th>
                <th>Weather</th>
                <th>Collectors</th>
            </tr>
            </thead>
            <tbody>
            </tbody>
        </table>
        <div class=”pull-right”>
            <button class=”btn btn-default”>Discard Changes</button>
            <button class=”btn btn-default”>Apply Changes</button>
        </div>
    </div>
    -->
    <div class=”col-md-8 col-md-offset-2”>
        <br/>
    <?php
        echo form_button(array(”content”=>”Continue”,”type”=>”submit”,”class”=>”btn btn-
            primary pull-right”,”name”=>”continue”,”value”=>”continue”));
    ?>
    </div>
    <?php
        echo form_close();
    ?>
</div>

<script>
    $(”th”).addClass(”sorting_disabled”);
    $(function() {
        $(”#colony-table”).hide();
        $(”#sample-table”).hide();
    });

    $(”#showColony”).click(function(){
        $(”#colony-table”).show(”fast”);
        $(”#sample-table”).hide();
    });
    function showSample(){
        $(”#sample-table”).show(”fast”);
        $(”#colony-table”).hide();
    }
    function hideTables(){
        $(”#colony-table”).hide();
        $(”#sample-table”).hide();
    }
</script>
<script src=”<?php echo base_url(); ?>assets/js/jquery.dataTables.min.js”></script>
<script src=”<?php echo base_url(); ?>assets/js/my_dataTables.js”></script>
```

## Listing 16: sync_options.php

```
<?php
    $to_file = ”<p>Select this option if you wish to extract data from this system onto an
        excel file and take it to the $ext system.</p>”;
    $from_file = ”<p>Select this option if you have data in an excel file from the $ext system
        that you wish to add to this system.</p>”;
?>
<div class=”row”>
    <?php
        if(isset($msg)){
    ?>
            <div class=”alert alert-success”><span class=”glyphicon glyphicon-ok-sign”></span>
                <?php echo $msg?></div>
    <?php
        }
    ?>
    <div class=”col-md-8 col-md-offset-2”>
        <h2>Synchronize <small id=”sub”>How would you like to synchronize?</small></h2>
    </div>
    <div class=”col-md-8 col-md-offset-2” id=”step1”>
        <div class=”panel-group” id=”accordion”>
            <div class=”panel panel-default”>
                <div class=”panel-heading”>
                    <a data-toggle=”collapse” data-parent=”#accordion” href=”#collapseOne”
                        title=”Click to view details”>
                        <h4 class=”panel-title”>
                            <span class=”glyphicon glyphicon-cloud-upload”></span>&nbsp;
                                Offline to Online
                        </h4>
                    </a>
                </div>

                <div id=”collapseOne” class=”panel-collapse collapse”>
                    <div class=”panel-body”>
                        <?php echo $int==”offline”? $to_file : $from_file; ?>
                        <button type="button" class="btn btn-primary pull-right" onclick="
                             hide_show('#step2-<?php echo $int=="offline"? "tofile" : "fromfile
                             "; ?>');"> Continue</button>
                    </div>
                </div>
            </div>
            <div class="panel panel-default">
                <div class="panel-heading">
                    <a data-toggle="collapse" data-parent="#accordion" href="#collapseTwo"
                        title="Click to view details">
                        <h4 class="panel-title"><span class="glyphicon glyphicon-cloud-
                             download"></span>&nbsp;Online to Offline
                        </h4>
                    </a>
                </div>
                <div id="collapseTwo" class="panel-collapse collapse">
                    <div class="panel-body">
                        <?php echo $int=="offline"? $from_file : $to_file; ?>
                        <button type="button" class="btn btn-primary pull-right" onclick="
                             hide_show('#step2-<?php echo $int=="offline"? "fromfile" : "tofile
                             "; ?>');"> Continue</button>
                    </div>
                </div>
            </div>

        </div>

    </div>
    <div class="col-md-8 col-md-offset-2" id="step2-tofile">
        <ol class="breadcrumb">
            <li><a href="#" onclick="hide_show('#step1')">Synchronize</a></li>
            <li class="active"><?php
                echo $int=="offline"? '<span class="glyphicon glyphicon-cloud-upload"></span>
                     Offline to Online':'<span class="glyphicon glyphicon-cloud-download"></
                     span> Online to Offline';
            ?></li>
        </ol>
        <?php
            echo form_open("synchronize/to_file");
        ?>
        <div class="form-group">
            <div class="checkbox">
                <label>
                    <?php
                    echo form_checkbox("data[]","colony", true); ?> Colony Information<!-- <
                        span class="glyphicon glyphicon-info-sign text-info"></span>--><br/>
                </label>
            </div>
        </div>
        <div class="form-group">
            <div class="checkbox">
                <label>
                    <?php
                    echo form_checkbox("data[]","sample", true); ?> Sampling Information<!-- <
                        span class="glyphicon glyphicon-info-sign text-info"></span>--><br/>
                </label>
            </div>
        </div>
        <!--
        <div class="form-group">
            <div class="checkbox">
                <label>
                    <?php
                    echo form_checkbox("data[]","case", true); ?> Cases <span class="glyphicon
                          glyphicon-info-sign text-info"></span><br/>
                </label>
            </div>
        </div>
        -->
        <?php
            echo form_button(array("content"=>"<span class='glyphicon glyphicon-transfer'></
                 span> Synchronize","class"=>"btn btn-primary pull-right","type"=>"submit"));
            echo form_close();
        ?>
    </div>
    <div class="col-md-8 col-md-offset-2" id="step2-fromfile">
        <ol class="breadcrumb">
            <li><a href="#" onclick="hide_show('#step1')">Synchronize</a></li>
            <li class="active"><?php
                echo $ext=="offline"? '<span class="glyphicon glyphicon-cloud-upload"></span>
                     Offline to Online':'<span class="glyphicon glyphicon-cloud-download"></
                     span> Online to Offline';
                ?></li>
        </ol>
        <?php
            if(isset($err)){
                echo "<div class='alert alert-danger'>$err</div>";
        ?>
            <script>hide_show('#step2-fromfile'); </script>
        <?php
            }
            echo form_open_multipart("synchronize/from_file");
            echo form_upload("userfile");
            echo form_button(array("content"=>"<span class='glyphicon glyphicon-transfer'></
                 span> Synchronize","class"=>"btn btn-primary pull-right","type"=>"submit"));
            echo form_close();
        ?>
    </div>
    <div class="col-md-8 col-md-offset-2" id="info">
```

```
        <span class="text-muted"><span class="glyphicon glyphicon-info-sign"></span> You are
             currently on the <b><?php echo $int; ?></b> system.</span>
    </div>
</div>
<script>
    var curr_window = "#step1";
    $("#step2-tofile").hide();
    $("#step2-fromfile").hide();
    function hide_show(s){
        $('#sub').html(s=="#step1"? "How would you like to synchronize?" : 'Which data would
             you like to synchronize?');
        $(curr_window).hide("fast")
        $(s).show("slow");
        curr_window = s;
    }
</script>
```

### Listing 17: user_add.php

```
<?php echo link_tag('assets/css/bootstrap-switch.css'); ?>
<div class="row">
    <div class="col-md-3"></div>
    <div class="col-md-6">
        <h1><small><?php echo $title;?></small></h1>
        <?php
            if($action=="edit"){
                echo form_open(base_url("user/edit_attempt"),array("onsubmit"=>"set_as();"));
                echo form_hidden("user_id",$new_user["user_id"]);
            }else{
                echo form_open(base_url("user/add_attempt"),array("onsubmit"=>"return
                     password_match;"));
            }
            if(isset($validation_errors)){
                ?>
                <div class="alert alert-danger text-center">
                    <?php echo $validation_errors; ?>
                </div>
<?php
    }
?>
        <div class="form-group">
            <label for="last_name">Last Name</label>
            <?php echo form_input(array("id" => "last_name", "name" => "last_name", "required"
                 => "true","class"=>"form-control","placeholder"=>"Last Name"),$new_user["
                last_name"])?>
            <label for="first_name">First Name</label>
            <?php echo form_input(array("id" => "first_name", "name" => "first_name", "
                required" => "true","class"=>"form-control","placeholder"=>"First Name"),
                $new_user["first_name"])?>
            <label for="middle_name">Middle Name</label>
            <?php echo form_input(array("id" => "middle_name", "name" => "middle_name", "
                required" => "true","class"=>"form-control","placeholder"=>"Middle Name"),
                $new_user["middle_name"])?>
        </div>
        <div class="form-group">
            <label for="email">Email</label>
            <?php echo form_input(array("type"=>"email","id" => "email", "name" => "email", "
                required" => "true","class"=>"form-control","placeholder"=>"E-mail Address"),
                $new_user["email"])?>
        </div>
        <label for="role">Role(s)</label><br/>
        <div class="checkbox">
            <?php
            foreach($roles as $role){ ?>
                <div class="checkbox"><label><?php echo form_checkbox('role[]', $role->role_id
                     ,($curr_roles)? in_array($role->role_id,$curr_roles):false) . "&nbsp;" .
                     $role->name ; ?></label></div>
            <?php } ?>
        </div>
        <div class="form-group">
            <label for="username">Username</label>
            <?php echo form_input(array("id" => "username", "name" => "username", "required"
                => "true","class"=>"form-control","placeholder"=>"Username"),$new_user["
                username"])?>
        </div>

        <div class="form-group row">
            <?php
                if($action=="add"){
            ?>
            <div id="divCheckPasswordMatch"></div>
            <?php
                }
            ?>
            <label for="password"<?php if($action=="edit"){ echo ' class="col-md-3"'; }?>>
                Password</label>
            <?php
                if($action=="edit"){
            ?>
                    <a href='#' data-toggle='modal' data-target='#modal-changepass' class="col
                        -md-9">Change Password</a>
            <?php
            }else{
                echo form_password(array("id" => "password", "name" => "password", "required"
                    => "true","class"=>"form-control","placeholder"=>"Password"))?>
            <label for="username">Re-type Password</label>
            <?php echo form_password(array(”id” => ”retype_password”, ”name” => ”
                 retype_password”, ”required” => ”true”,”class”=>”form−control”,”placeholder
                 ”=>”Verify Your Password”))?>

        <?php
        }
        ?>
        </div>
        <?php
        if($action==”edit”){
        ?>
        <div class=”form−group row”>
            <label for=”account_status” class=”col−md−3”>Account Status</label><div class=”col
                 −md−9”><?php echo form_checkbox(”as_checkbox”,”active”,$new_user[”
                 account_status”]==”enabled”,’id=”as_checkbox” data−on−text=”&nbsp;Enabled&nbsp
                 ;” data−off−text=”&nbsp;Disabled&nbsp;”’)?></div>
        </div>
        <?php
            echo form_input(”account_status”,$new_user[”account_status”],”id=’account_status’
                 class=’hidden’”);
        }
        ?>
        <?php
        echo form_button(array(”id”=>”submit”,”name”=>”submit”,”type”=>”submit”,”class”=>”btn
             btn−primary col−lg−2 col−lg−offset −10”,”content”=>”<span class=’glyphicon
             glyphicon−floppy−disk’></span> Save”));
        echo form_close();
        ?>

    </div>
</div>
<?php  if($action==”edit”){ ?>
<div class=”modal fade” id=”modal−changepass” tabindex=”−1” role=”dialog” aria−labelledby=”
     Change password” aria−hidden=”true”>
    <div class=”modal−dialog”>
        <?php
        echo form_open(”user/change_password”, array(”class”=>”form−horizontal”,”onsubmit”=>”
             return password_match;”,”id”=>”change_pass”));
        echo form_hidden(”user_id”,$new_user[”user_id”]);
        ?>
        <div class=”modal−content”>
            <div class=”modal−header”>
                <button type=”button” class=”close” data−dismiss=”modal” aria−hidden=”true”>
                     </button>
                <h4 class=”modal−title”>Change Password</h4>
            </div>
            <div class=”modal−body”>
                <div class=”form−group”>
                    <label for=”old” class=”col−sm−4 control−label”>Current Password</label>
                    <div class=”col−sm−7”>
                        <input type=”password” class=”form−control” id=”old” placeholder=”Old
                             Password” required=”” name=”old”>
                    </div>
                </div>
                <div class=”col−sm−7 col−sm−offset −4” id=”divCheckPasswordMatch”></div>
                <div class=”form−group”>
                    <label for=”password” class=”col−sm−4 control−label”>New Password</label>
                    <div class=”col−sm−7”>
                        <input type=”password” class=”form−control” id=”password” placeholder
                             =”New Password” required=”” name=”password”>
                    </div>
                </div>
                <div class=”form−group”>
                    <label for=”retype_password” class=”col−sm−4 control−label”>Verify New
                         Password</label>
                    <div class=”col−sm−7”>
                        <input type=”password” class=”form−control” id=”retype_password”
                             placeholder=”New Password” required=”” name=”retype_password”>
                    </div>
                </div>
            </div>
            <div class=”modal−footer”>
                <?php
                echo form_button(array(”data−dismiss”=>”modal”,”content”=>”Cancel”,”class”=>’
                     btn btn−default’));
                echo form_button(array(”name”=>”change_password”,”id”=>”change_password”,”
                     content”=>”Save Changes”,”value”=>”change_password”,”class”=>’btn btn−
                     primary’,”type”=>”submit”));
                ?>
            </div>
        </div>
        <?php echo form_close(); ?>
    </div>
</div>
<?php  }  ?>

<script src=”<?php echo base_url(”assets/js/bootstrap−switch.js”); ?>”></script>
<script>
    var password_match = false;
    $(”#as_checkbox”).bootstrapSwitch();
    function set_as(){
        var status =”enabled”;
        if(!$(’#as_checkbox’).bootstrapSwitch(’state’)){
            status =”disabled”;
        }

        $(”#account_status”).val(status);

    }
    $(document).ready(function () {
        $("#retype_password").keyup(checkPasswordMatch);
    });
    function checkPasswordMatch() {
        var password = $("#password").val();
        var confirmPassword = $("#retype_password").val();

        if (password != confirmPassword){
            $("#divCheckPasswordMatch").html("<div class='alert alert-danger text-center'><
                span class='glyphicon glyphicon-remove-circle'></span>" +
                " Passwords do not match." +
                "</div>");
            password_match = false;
            $("#retype_password").addClass("text-warning");
        }else{
            $("#divCheckPasswordMatch").html("<div class='alert alert-success text-center'><
                span class='glyphicon glyphicon-ok-circle'></span>" +
                " Passwords match." +
                "</div>");
            password_match = true;
        }
    }
</script>
```

## Listing 18: user_all.php

```
<?php
echo link_tag('assets/css/DT_bootstrap.css');
echo form_open("", array("role"=>"form"));
?>
<div class="row">
    <div class="col-lg-4 col-lg-offset-3">
        <div class="input-group">
            <input type="text" class="form-control" placeholder="Search for user" id="user-
                search" name="username">
            <input type="hidden" id="search-term" name="search-term" value="<?php echo $term;
                ?>">
            <span class="input-group-btn">
                <button class="btn btn-default" type="submit" name="search" value="search"
                    onclick='$("#search-term").val("");'><span class="glyphicon glyphicon-
                    search"></span></button>
            </span>
        </div>
    </div>
    <div class="col-lg-2">
        <button class="btn btn-default col-md-12" type="submit" name="add" value="add"><span
            class="glyphicon glyphicon-plus-sign"></span> Add New User</button>
    </div>
    </div>
</div>
<div id="search-help" class="row hidden">
    <div class="col-lg-4 col-lg-offset-3 help-block text-center"><em>Hit the search button to
        get more results.</em></div>
</div>

<?php
    //echo json_encode($users);
?>
<div class="row">
    <div class="col-lg-1"></div>
    <div class="col-lg-10">
        <h3>Users
            <?php
            echo strlen($term)? "<small style='cursor: pointer' id='search-keyword'><div class
                ='label label-default'>Search results for '$term'&nbsp;&nbsp;<span id='close'
                title='dismiss'>&times;</span></div></small>" : "";
            ?>
        </h3>
        <div class="panel-group" id="accordion">
            <div class="panel panel-default">
                <a data-toggle="collapse" data-parent="#accordion" href="#collapseOne">
                <div class="panel-heading">
                    <h4 class="panel-title"><small>Filter List <span class="caret"></span></
                        small></h4>
                </div>
                </a>
                <div id="collapseOne" class="panel-collapse collapse">
                    <div class="panel-body">
                        <p class="help-block">Show only users where</p>
                        <div class="row">
                            <div class="col-lg-6">
                                <label for="role">Role:</label><br/>
                                <div class="checkbox">
                                    <?php foreach($roles as $role){ ?>
                                        <div class="checkbox"><label><?php echo form_checkbox
                                            ('role[]', $role->role_id,($this->input->post("
                                            role"))? in_array($role->role_id,$this->input->
                                            post("role")):TRUE) . "&nbsp;" . $role->name ;
                                            ?></label></div>
                                    <?php } ?>
                                </div>
                            </div>
                            <div class="col-lg-6">
                                <label for="role">Status:</label><br/>
                                <div class="form-group">
                                    <div class="col-sm-10">
                                        <div class="checkbox">
```

```
                                            <label>
                                        <?php
                                        echo form_checkbox('status[]', 'enabled', ($this->
                                             input->post("status"))? in_array("enabled",$this->
                                             input->post("status")):TRUE); ?> Enabled
                                            </label>
                                        </div>
                                    </div>
                                </div>
                                <div class="form-group">
                                    <div class="col-sm-10">
                                        <div class="checkbox">
                                            <label>
                                                <?php echo form_checkbox('status[]', 'disabled
                                                     ', ($this->input->post("status"))?
                                                     in_array("disabled",$this->input->post("
                                                     status")):TRUE); ?>
                                                Disabled
                                            </label>
                                        </div>
                                    </div>
                                </div>
                            </div>
                            <button class="btn btn-default col-lg-2 col-lg-offset-5" type="
                                 submit" name="filter" value="filter"><span class="glyphicon
                                 glyphicon-filter"></span> Filter</button>
                        </div>
                    </div>
                </div>
            </div>
            <div class="panel panel-default">
                <a data-toggle="collapse" data-parent="#accordion" href="#collapseTwo">
                    <div class="panel-heading">
                        <h4 class="panel-title"><small>Update Multiple <span class="caret"></
                             span></small></h4>
                    </div>
                </a>
                <div id="collapseTwo" class="panel-collapse collapse<?php echo strlen($this->
                     input->post("filter"))? " in": ""?>">
                    <div class="panel-body">
                        <div class="row">
                            <div class="col-lg-8 col-lg-offset-2">
                                <div class="alert alert-warning alert-dismissable hidden" id="
                                     update-error">
                                    <button type="button" class="close" onclick="$('#update-
                                         error').addClass('hidden');" aria-hidden="true">&times
                                         ;</button>
                                    You haven't selected any data yet.
                                </div>
                            </div>
                            <div class="col-lg-5 col-lg-offset-2">
                                <?php echo form_dropdown("update_action",array("enabled"=>"
                                     Enable","disabled"=>"Disable"),"enabled","class='form-
                                     control' id='update_action'"); ?>
                                <span class="help-block">Changes will be applied to data with
                                     checked boxes.</span>
                            </div>
                            <div class="col-lg-3">
                                <?php
                                echo form_button(array("content"=>"Apply Changes","class"=>'
                                     btn btn-default col-lg-12',"onclick"=>"update_multiple()
                                     ;"));
                                ?>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>
<div class="row">
    <div class="col-lg-1"></div>
    <div class="col-lg-10 table-responsive">
        <table class="table table-striped table-hover">
            <thead>
            <tr>
                <th class="sorting_disabled"><?php echo form_checkbox('checkall', 'checkall',
                     FALSE, 'id="checkall"')?></th>
                <th>Name</th>
                <th>Username</th>
                <th>Status</th>
                <th>Roles</th>
                <th class="sorting_disabled"><strong>Actions</strong></th>
            </tr>
            </thead>
            <tbody id="user-list">

    <?php
        foreach($users as $user){
            $user_id = $user->user_id;
            echo "<tr>";
            echo    "<td>".form_checkbox('selected_users[]',$user->username,false,'id="'.
                 $user_id.'_checkbox"')."</td>";
            echo    "<td>$user->last_name, $user->first_name $user->middle_name</td>";
            echo    "<td>$user->username</td>";
            echo    "<td>$user->account_status</td>";
            echo    "<td>$user->roles</td>";
            echo    "<td>";
            echo        ”<a href=’”. base_url() .”user/view/”.$user_id.”’><span class=’
                 glyphicon glyphicon-eye-open’ title=’View’></span></a>&nbsp;<a href=’”.
                 base_url() .”user/edit/”.$user_id.”’><span class=’glyphicon glyphicon-edit’
                 title=’Edit’></span></a>&nbsp;”;
            echo        ”<a href=’#’ data-toggle=’modal’ data-target=’#modal-update’ onclick=’
                 update_single($user_id,\”$user->account_status\”, \”$user->username\”)’ title
                 =’Change status’><span class=’glyphicon glyphicon-off’></span> </a>”;
            echo    ”</td>”;
            echo ”</tr>”;
        }
        if(count($users)==0) echo ”<tr><td colspan=’6’ class=’text-center text-muted’><em>No
             user accounts found under the given filters.</em></td></tr>”;
    ?>
            </tbody>
        </table>
    </div>
</div>
<div class=”modal fade” id=”modal-update” tabindex=”-1” role=”dialog” aria-labelledby=”Confirm
      action” aria-hidden=”true”>
    <div class=”modal-dialog”>
        <div class=”modal-content”>
            <div class=”modal-header”>
                <button type=”button” class=”close” data-dismiss=”modal” aria-hidden=”true”>
                     </button>
                <h4 class=”modal-title”>Confirm action</h4>
            </div>
            <div class=”modal-body”>
                <p id=”message”>The following users will be <span class=”text-info” id=”modal-
                     action”></span>.</p>
                <div class=”well well-sm”><ul id=”modal-users”></ul></div>
            </div>
            <div class=”modal-footer”>
                <?php
                echo form_button(array(”data-dismiss”=>”modal”,”content”=>”Cancel”,”class”=>’
                     btn btn-default’));
                echo form_button(array(”name”=>”update_form”,”id”=>”update_form”,”content”=>”
                     Continue”,”value”=>”update”,”class”=>’btn btn-primary’,”type”=>”submit”));
                ?>
            </div>
        </div>
    </div>
</div>
<?php
echo form_close();
if(count($users)!=0){
?>
    <script src=”<?php echo base_url(”assets/js/jquery.dataTables.min.js”); ?>”></script>
    <script src=”<?php echo base_url(”assets/js/my_dataTables.js”); ?>”></script>
    <script src=”<?php echo base_url(”assets/js/jquery.fastLiveFilter.js”); ?>”></script>
    <script>
    $(function() {
        $(’#user-search’).fastLiveFilter(’#user-list’,{
            callback: function(){
                ($(’#user-search’).val().length==0)? $(”#search-help”).addClass(”hidden”) : $
                     (”#search-help”).removeClass(”hidden”);}
        });
    });
    $(”th”).not(”.sorting_disabled”).css(”cursor”,”pointer”);
    $(function () {
        $(’#checkall’).on(’click’, function () {
            $(’tr input:checkbox’).prop(’checked’, this.checked);
        });
    });
    function update_single(id,status,name){
        var mark = (status==”enabled”)? ”disabled” : ”enabled”;
        $(”#update_action”).val(mark);
        $(’tr input:checkbox’).prop(’checked’, false);      // uncheck other boxes
        $(”#”+id+”_checkbox”).prop(”checked”,true);

        $(”#modal-action”).html($(”#update_action”).val());
        $(”#modal-users”).html(”<li>”+name+”</li>”);
    }
    function update_multiple(){
        var inputs = $(”td input:checked”);
        if(inputs.length==0){
            $(”#update-error”).removeClass(”hidden”) ;
        }else{
            var users = ””;
            for(var i=0;i<inputs.length;i++){
                users += ”<li>” + inputs[i].value + ”</li>”;
            }
            $(”#modal-action”).html($(”#update_action”).val());
            $(”#modal-users”).html(users);
            $(”.modal”).modal();
        }
    }
    </script>
<?php
}
?>
<script>
    $(”#search-keyword”).click(
        function(){
            $(”#user-search”).val(””);
            $(”#search-term”).val(””);
            $(”form”).submit();
        }
    )
</script>
```

## Listing 19: user_id.php

```
<div class="row">
    <div class="col-md-3"></div>
    <div class="col-md-6">
        <?php
        if(isset($success_msg)){
            echo '<div class="alert alert-success text-center">';
            echo "<span class='glyphicon glyphicon-ok-sign'></span>";
            echo '<span id="success-msg"> ';
            switch($success_msg){
                case 1:
                    echo "You have successfully created an account for '".$user->username
                        ."'.";
                    break;
                case 2:
                    echo "The changes you have made to '".$user->username."' have been saved
                        .";
                    break;
                case 3:
                    echo "Password has been changed for '".$user->username."'.";
                    break;
            }
            echo '</span>';
            echo "</div>";
        }
        if(isset($fail_msg)){
            echo '<div class="alert alert-danger text-center">';
            echo "<span class='glyphicon glyphicon-remove-circle'></span>";
            echo '<span id="fail-msg"> '.$fail_msg.'</span>';
            echo "</div>";
        }
        ?>
        <h2><?php echo $user->last_name .", ". $user->first_name . " ". $user->middle_name?> <
            small>User Information</small></h2>
        <table class="table">
            <tr>
                <td><b>Username</b></td>
                <td class="text-right"><?php echo $user->username; ?></td>
            </tr>
            <tr>
                <td><b>E-mail Address</b></td>
                <td class="text-right"><?php echo $user->email; ?></td>
            </tr>
            <tr>
                <td><b>Roles</b></td>
                <td class="text-right">
                    <?php foreach($user->roles as $role){
                        echo $role . "<br/>";
                    } ?>
                </td>
            </tr>
            <tr>
                <td><b>Account Status</b></td>
                <td class="text-right"><?php echo $user->account_status; ?></td>
            </tr>
        </table>
        <div class="btn-group btn-group-justified">
            <a href="<?php echo base_url(); ?>user/edit/<?php echo $user->user_id?>" class="
                btn btn-default"><span class="glyphicon glyphicon-edit"></span> Edit</a>
            <?php if(array_key_exists(2,$user->roles)){ //if user is a health worker?>
            <!--<a href="<?php echo base_url(); ?>sample/search/<?php echo $user->user_id?>/
                all" class="btn btn-default"><span class="glyphicon glyphicon-list-alt"></span
                > View samples collected by <?php echo $user->username; ?></a>-->
            <?php
            }
            ?>
        </div>
    </div>
</div>
```

## Listing 20: user_login.php

```
<!DOCTYPE HTML>
<html>
<head>
    <title>Schistosomiasis GIS</title>
    <link rel="icon" href="<?php  echo base_url("/favicon.ico")?>">
    <?php
    echo link_tag('assets/css/bootstrap.min.css');
    echo link_tag('assets/css/login.css');
    ?>
</head>
<body>
<div class="text-center col-lg-12"><img src="<?php echo base_url()?>/assets/images/doh_logo.
    png"/></div>
<div class="container" id="login-box">
<?php
echo form_open(base_url("user/login_attempt"), array("class"=>"form-login"));
?>
    <h2 style="padding-top:90px;">Please log in</h2>
<?php
if(isset($errors)){
    ?>
    <div class="alert alert-danger text-center">
        <?php
        echo $errors;
        ?>
    </div>
<?php
}

    echo form_input(array(”id” => ”username”, ”name” => ”username”, ”required” => ””,”class
        ”=>”form-control”,”type”=>”username”,”autofocus”=>””,”placeholder”=>”username”));
    echo form_password(array(”id” => ”password”, ”name” => ”password”, ”required” => ””,”class
        ”=>”form-control”,”type”=>”password”,”autofocus”=>””,”placeholder”=>”password”));
?>
<?php
    echo form_submit(array(”class”=>”btn btn-lg btn-primary col-sm-12”,”type”=>”submit”,”value
        ”=>”Log In”));
    echo form_close();
?>
    <label class=”col-sm-12 text-center forgot-pass”><a href=”#”>Forgot password?</a></label>
</div>

<div class=”container hidden” id=”forgot-box”>
<?php
echo form_open(base_url(”user/reset_password”), array(”class”=>”form-login”));
?>
    <h2 style=”padding-top:90px;”>Reset Password</h2>
    <p class=”help-block ”>Enter the email address associated with your account and we’ll
         email you a reset link.</p>
<?php
    echo form_input(array(”id” => ”email”, ”name” => ”email”, ”type”=>”email”,”required” =>
        ””,”class”=>”form-control”,”placeholder”=>”Enter email”));
    echo form_submit(array(”class”=>”btn btn-lg btn-default col-sm-12”,”type”=>”submit”,”value
        ”=>”Send me a link”));
    echo form_close();
?>
</div>
</body>
<script src=”<?php echo base_url(); ?>assets/js/jquery-2.0.3.min.js”></script>
<script src=”<?php echo base_url(); ?>assets/js/bootstrap.min.js”></script>
<script>
    $(”.forgot-pass”).click(toggle_forgot);
    function toggle_forgot(){
        $(”#login-box”).toggleClass(”hidden”);
        $(”#forgot-box”).toggleClass(”hidden”);
        event.preventDefault();
    }
</script>
</html>
```

## Controllers

Listing 21: cases.php

```
<?php
/**
 * Created by PhpStorm.
 * User: Gerald
 * Date: 3/7/14
 * Time: 9:15 PM
 */

class Cases extends CI_Controller{
    var $data;
    function index(){
        $this->load->library(”permission”);
        //$this->permission->restrict_to($this->session->userdata(”user_roles”),0);
        $this->show_import();
    }
    function show_import($data=array()){
        $data[”view”] = ”view_import”;
        $this->load->view(”view_template”,$data);
    }
    function load_file(){
        $this->load->model(”model_import”);

        if($this->input->post(’upload’)){
            $this->model_import->do_upload();  //upload the file
            $upload_data     = $this->upload->data();
            $file_name       = $upload_data[’file_name’];
            $sheetData       = $this->model_import->do_read($file_name);

            //get indices
            foreach($sheetData[1] as $key => $value){
                $indices[$value] = $key;
            }
            $barangay_data = array();   // array containing the data per barangay
            for($row=2;$row<=count($sheetData);$row++){
                $curr_row = $sheetData[$row];
                $curr_bgy = $curr_row[$indices[”bgycode”]];
                $curr_date = $curr_row[$indices[”cfdate”]];
                $curr_scepg =  $curr_row[$indices[”scepg”]];
                if(array_key_exists($curr_bgy,$barangay_data)&&array_key_exists($curr_date,
                     $barangay_data[$curr_bgy])){
                    $barangay_data[$curr_bgy][$curr_date][”total_scepg”] += $curr_scepg;
                    $barangay_data[$curr_bgy][$curr_date][”total_cases”]++;
                }else{
                    //init array
                    $barangay_data[$curr_bgy] = array($curr_date=> array());
                    $barangay_data[$curr_bgy][$curr_date] = array(
                        ”total_scepg”   => $curr_scepg,
                        ”total_cases”   => 1,
                        ”average_scepg” => 0
                    );
                }
            }
            $this->data = array(
                ”barangay_data” => $barangay_data
            );
        }
        $this->show_import($this->data);
    }
    function confirm(){
        $this->load->model(”model_import”);
        $arr = $this->input->post(”data”);
        for($i=0;$i<count($arr);$i++){
            $data = array(
                ”barangay_code” => $arr[$i][0],
                ”date” => $arr[$i][1],
                ”num_cases” => $arr[$i][2],
                ”intensity” => $arr[$i][3],
            );
            $query = $this->db->insert(”barangay_cases”, $data);
        }
    }
}
```

Listing 22: colony.php

```
<?php
/**
 * Created by PhpStorm.
 * User: Gerald
 * Date: 2/20/14
 * Time: 8:54 PM
 */

class Colony extends CI_Controller{
    public function index(){
        redirect(”colony/view_all/all”);
    }
    public function add($data=array()){
        $colony = array(
            ”colony_id”   => $this->input->post(”colony_id”),
            ”site_name”   => $this->input->post(”site_name”),
            ”longitude”     => $this->input->post(”longitude”),
            ”latitude”      => $this->input->post(”latitude”),
            ”area”          => $this->input->post(”area”),

        );
        $_data = array(
            ”action”    => ”colony/add_attempt”,
            ”colony”    => $colony,
            ”title”     => ”New Snail Colony”,
            ”view”      => ”colony/view_colony_add”
        );
        $this->load->view(”view_template”,array_merge($data,$_data));
    }
    public function add_attempt(){
        $this->load->model(”model_colonies”);

        $this->load->library(”form_validation”);
        $this->form_validation->set_rules(”site_name”, ”Site Name”, ”required|is_unique[colony
            .site_name]”);
        $this->form_validation->set_rules(”longitude”, ”Longitude”, ”required|trim”);
        $this->form_validation->set_rules(”latitude”, ”Latitude”, ”required|trim”);
        $this->form_validation->set_rules(”area”, ”Area”, ”required|trim”);
        $this->form_validation->set_message(”is_unique”,”A colony with that name already
             exists.”);

        $colony = array(
            'site_name' => $this->input->post('site_name'),
            'longitude' => $this->input->post('longitude'),
            'latitude' => $this->input->post('latitude'),
            'area' => $this->input->post('area')
        );

        if($this->form_validation->run()){
            $query =$this->model_colonies->insert($colony);
            if($query!=-1){
                redirect(”colony/view/$query/1”);
            }else{
                echo ”db error”;
            }

        }else{
            $this->add(array(”validation_errors” => validation_errors()));
        }
    }
    public function edit($id=null){
        if(isset($id)){
            $this->load->model(”model_colonies”);
            $colony = (array)$this->model_colonies->get($id);
            if(count($colony)){
                $data = array(
                    ”colony”    => $colony,
                    ”title”      => ”Edit Snail Colony”,
                    ”action”    => ”colony/edit_attempt”,
                    ”view”       => ”colony/view_colony_add”
                );
                $this->load->view(”view_template”,$data);
            }else{
                echo ”hi”;
                echo ”404: Colony not found”;
            }
        }else{
            redirect(”colony/add”);
        }
    }
    public function name_check($name, $id){
        $this->load->model(”model_colonies”);
        if(!$this->model_colonies->unique_edit($id, $name)){
            $this->form_validation->set_message(’name_check’, ’A colony with the name ”’.
                 $name . ’” already exists.’);
            return false;
        }
        return true;
    }
    public function edit_attempt(){
        $this->load->model(”model_colonies”);
        $this->load->library(”form_validation”);
        $colony_id = $this->input->post(’colony_id’);
        $this->form_validation->set_rules(”site_name”, ”Site Name”, ”required|
             callback_name_check[$colony_id]”);     //add custom callback for uniqueness
        $this->form_validation->set_rules(”longitude”, ”Longitude”, ”required|trim”);
        $this->form_validation->set_rules(”latitude”, ”Latitude”, ”required|trim”);
        $this->form_validation->set_rules(”area”, ”Area”, ”required|trim”);

        $colony = array(
            ’colony_id’ => $colony_id,
            ’site_name’ => $this->input->post(’site_name’),
            ’longitude’ => $this->input->post(’longitude’),
            ’latitude’ => $this->input->post(’latitude’),
            ’area’ => $this->input->post(’area’),
        );
        if($this->form_validation->run()){
            $this->model_colonies->update($colony);
            redirect(”colony/view/”.$this->input->post(’colony_id’).”/2”);
        }else{
            $data = array(
                ”colony”             => $colony,
                ”action”             => ”colony/edit_attempt”,
                ”validation_errors” => validation_errors(),
                ”title”              => ”Edit Snail Colony”,
                ”view”               => ”colony/view_colony_add”
            );
            $this->load->view(”view_template”,$data);
        }
    }
    public function view($id,$success_msg=null){
        if(!isset($id)){
            redirect(”colony/view_all/all”);
        }else{
            $this->load->model(”model_colonies”);
            $this->load->model(”model_samples”);
            $colony = $this->model_colonies->get($id);
            $colony->sample_times = count($this->model_samples->get_by_id($colony->colony_id))
                ;
            $data = array(
                ”colony”        => $colony,
                ”success_msg”   => $success_msg,
                ”title”         => $colony->site_name . ” Snail Colony”,
                ”view”          => ”colony/view_colony_id”
            );
            $this->load->view(”view_template”,$data);
        }
    }
    public function view_all($starting=”all”){
        $term = strlen($this->input->post(”site_name”))? $this->input->post(”site_name”) :
             $this->input->post(”search-term”);
        if(strlen($this->input->post(’add’))){
            strlen($term)? $this->add() : redirect(”colony/add”);
        }else{
            $this->load->model(”model_colonies”);
            if(strlen($this->input->post(’update_form’))){
                foreach($this->input->post(’selected_colonies’) as $site_name){
                    $this->model_colonies->move($site_name,$this->input->post(’update_action’)
                        );
                }
            }
            if(!isset($starting)||preg_match(’/^([a-z]|[A-Z]|all)$/’, $starting) === 0) {   //
                 if invalid starting i.e. not single alphabet or ”all”
                $starting = ”A”;
            }
            $this->load->model(”model_samples”);
            $this->load->model(”model_cases”);
            $status = $this->input->post(”status”);
            $status = count($status)==1?$status[0] : null;     //check if status should be
                 filtered
            $colonies = $this->model_colonies->get_all($starting,’colony_id, site_name, status
                 ’, $status);
            foreach($colonies as $colony){
                $colony->last_sample = $this->model_samples->get_date_by_id($colony->colony_id
                    );
            }
            $data = array(
                ”starting”  => strtoupper($starting),
                ”colonies”  => $colonies,
                ”title”     => ”List of Snail Colonies”,
                ”view”      => ”colony/view_colony_all”,
                ”term”      => $term,
                ”regions”   => json_decode(file_get_contents(base_url(”assets/js/regions.json
                    ”)),true), //$this->model_cases->get_all_regions()
                ”provinces” => json_decode(file_get_contents(base_url(”assets/js/provinces.
                    json”)),true)
            );
            $this->load->view(”view_template”,$data);
        }
    }
    public function search($term){

    }
}
```

Listing 23: main.php

```
<?php if ( ! defined(’BASEPATH’)) exit(’No direct script access allowed’);

class Main extends CI_Controller {

        /**
         * Index Page for this controller.
         *
         * Maps to the following URL
         *              http://example.com/index.php/welcome
         *      - or -
         *              http://example.com/index.php/welcome/index
         *      - or -
         * Since this controller is set as the default controller in
         * config/routes.php, it’s displayed at http://example.com/
         *
         * So any other public methods not prefixed with an underscore will
         * map to /index.php/welcome/<method_name>
         * @see http://codeigniter.com/user_guide/general/urls.html
         */
        public function index(){
        if($this->session->userdata(”is_logged_in”)){
            redirect(”map”);
        }else{
            redirect(”user/login”);
        }
        }
    public function test(){
        $search     = ”Malabon”;
        $line_number = false;
        $file = file(base_url(”assets/js/municipality_shapes.json”));
        foreach($file as $line){
            echo $line;
        }

        /*
        if ($handle = fopen(base_url(”assets/js/municipality_shapes.json”), ”r”)) {
            echo count($handle);

            $count = 0;
            $line = fgets($handle);
            echo $line;
            echo ”<br/>”;
            while ($line !== FALSE and !$line_number) {
                $count++;
                $line_number = strpos($line, $search);
                $prev_line = $line;
                $line = fgets($handle, 4096);
            }
            echo $count;
            echo ”<br/>”;
            echo $line;

            //fseek($handle, $line_number);


            fclose($handle);
        }                       */
    }
    public function restricted(){
        $this->load->view(”restricted”);
        $this->load->view(”user/view_user_login”);
    }
}

/* End of file main.php */
/* Location: ./application/controllers/main.php */
```

Listing 24: map.php

```
<?php if ( ! defined(’BASEPATH’)) exit(’No direct script access allowed’);

class Map extends CI_Controller {
    var $year;
    /**
     * Index Page for this controller.
     *
```

```
 * Maps to the following URL
 *         http://example.com/index.php/welcome
 *  - or -
 *         http://example.com/index.php/welcome/index
 *  - or -
 * Since this controller is set as the default controller in
 * config/routes.php, it's displayed at http://example.com/
 *
 * So any other public methods not prefixed with an underscore will
 * map to /index.php/welcome/<method_name>
 * @see http://codeigniter.com/user_guide/general/urls.html
 */
public function index()
{
    $this->loadMap();
}
public function view($lat,$long){
    $this->loadMap($lat,$long);
}
public function getColonies($stat){
    $this->load->model("model_colonies");
    $colonies = (count($stat)==1)? $this->model_colonies->get_all("all",null,$stat[0]) :
         $this->model_colonies->get_all("all");

    $this->load->model("model_samples");
    $samples = $this->model_samples->get_by_date($this->year);
    //create a marker for each colony
    $colony_markers = array();
    foreach($colonies as $colony){
        $colony_markers[$colony->colony_id] = array(
            "position"       => $colony->latitude . "," . $colony->longitude,
            "icon"           => base_url("assets/images/snail_plain_icon.png"),
            "title"          => "Click to view details for " . $colony->site_name,
            "onmouseover"    => "show_details('$colony->site_name','".$colony->area."')",
            "onmouseout"     => "hide_details()",
            "onclick"        => "window.location.href = '". base_url(). 'colony/view/'.
                 $colony->colony_id ."';",
            "site_name"    => $colony->site_name,
            "colony_area"    => $colony->area
            //"info_window_content" => "HELLO WORLD"
        );

    }
    foreach($samples as $sample){
        if(isset($colony_markers[$sample->colony_id])){

            $c_id = $sample->colony_id;
            $average = ($sample->total_count >0)? $sample->live_count/$sample->total_count
                 : 0;  // avoid division by 0
            if($average >0.66){
                $icon = base_url("assets/images/snail_red_icon.png");
            }elseif($average >0.33){
                $icon = base_url("assets/images/snail_yellow_icon.png");
            }elseif($average >0){
                $icon = base_url("assets/images/snail_green_icon.png");
            }
            $colony_markers[$c_id]["onclick"] = "window.location.href = '". base_url("
                 sample/view/".$c_id) . "/". rawurlencode($sample->date_time) . "';";
            $colony_markers[$c_id]["icon"] = $icon;
            $colony_markers[$sample->colony_id]["onmouseover"] = "show_details(".
                "'".$colony_markers[$sample->colony_id]["site_name"]."',".
                $colony_markers[$sample->colony_id]["colony_area"] . "," .
                $average*100 .
            ")";

        }
    }
    return $colony_markers;
}
function array_depth(array $array) {
    $max_depth = 1;

    foreach ($array as $value) {
        if (is_array($value)) {
            $depth = $this->array_depth($value) + 1;

            if ($depth > $max_depth) {
                $max_depth = $depth;
            }
        }
    }

    return $max_depth;
}
public function createPoly($points){
    $poly = array(
        "points"         => $points,
        "fillColor"      => "#333333",
        "strokeColor"    => "#333333",
        "strokeWeight"   => 1,
        "num_cases"      => 0,
        "population"     => 0,
        "intensity"      => 0
    );
    $this->googlemaps->add_polygon($poly);
    return $poly;
}
public function getShapesPerDivision($division,$status){
    /*
    $file = file(base_url("assets/js/municipality_shapes.json"));
    $count = 0;
    foreach($file as $line){
        $shape = null;
        $shape = json_decode($line);
        $count++;
        if($shape){
            if($this->array_depth($shape->points)==3){
                $s = array();
                foreach($shape->points[0] as $points){
                    $s[] = "\"".implode($points,",")."\"";
                }
                $polys[strtolower($shape->name)] = $this->createPoly($s);
            }else{
                $c = 1;
                foreach($shape->points as $points){
                    $s = array();
                    foreach($points[0] as $p){
                        $s[] = "\"".implode($p,",")."\"";
                    }
                    $polys[strtolower($shape->name)."_" .$c] = $this->createPoly($s);
                }
            }
        }
        if($count>2){
            break;
        }
    }*/
    $division = (strlen($division))? $division : "province";
    $divisions = $division . "_shapes";
    $shapes = json_decode(file_get_contents(base_url("assets/js/$divisions.json")),true);
    $polys = array();
    foreach($shapes as $shape){
        $polys[strtolower($shape["name"])] = $this->createPoly($shape["points"]);
    }
    $this->load->model("model_cases");
    $cases = $this->model_cases->get_all_cases($status);
    switch($division){
        case "barangay":
            return $this->getCasesPerBarangay($cases,$polys);
            break;
        case "region":
            return $this->getCasesPerRegion($cases,$polys);
            break;
        default:

            return $this->getCasesPerProvince($cases,$polys);
            break;
    }
}
public function getCasesPerRegion($cases,$reg_polys){
    $reg_cases = array();
    $this->load->model("model_cases");
    foreach($cases as $case){
        $reg_name = strtolower($this->model_cases->regname_from_bgycode($case->
             barangay_code));
        $reg_cases[$reg_name] = $reg_polys[$reg_name];
        $reg_cases[$reg_name]["num_cases"]+=$case->num_cases;
        if($reg_cases[$reg_name]["num_cases"]>300){
            $reg_cases[$reg_name]["fillColor"]="#FF0000";
            $reg_cases[$reg_name]["strokeColor"]="#FF0000";
        }elseif($reg_cases[$reg_name]["num_cases"]>200){
            $reg_cases[$reg_name]["fillColor"]="#FFFF00";
            $reg_cases[$reg_name]["strokeColor"]="#FFFF00";
        }else{
            $reg_cases[$reg_name]["fillColor"]="#00FF00";
            $reg_cases[$reg_name]["strokeColor"]="#00FF00";
        }
        $reg_cases[$reg_name]["onmouseover"]="alert(".$reg_cases[$reg_name]["num_cases
             "].")";
        //$reg_polys[$reg_name]["intensity"]+=$case->intensity;
    }
    return $reg_cases;
}
public function getCasesPerProvince($cases,$prov_polys){
    $prov_cases = $prov_polys;
    $this->load->model("model_cases");

    foreach($cases as $case){
        $bgycode = $case->barangay_code;
        $prov_name = strtolower($this->model_cases->provname_from_bgycode($bgycode));
        $prov_polys[$prov_name]["num_cases"] += $case->num_cases;
        $prov_polys[$prov_name]["population"] += $this->model_cases->get_population(
             $bgycode,$this->year);
        $average = $prov_polys[$prov_name]["num_cases"]/$prov_polys[$prov_name]["
             population"];
        $prov_polys[$prov_name]["average"] = $average;
        //$prov_polys[$prov_name]["onmouseover"]  = "show_prov_details('$prov_name','".
             $prov_polys[$prov_name]["num_cases"]."')";
        //$prov_polys[$prov_name]["onmouseout"]   = "hide_details()";
        if($average>0.66){
            $prov_polys[$prov_name]["fillColor"]="#FF0000";
            $prov_polys[$prov_name]["strokeColor"]="#FF0000";
        }elseif($average>0.33){
            $prov_polys[$prov_name]["fillColor"]="#FFFF00";
            $prov_polys[$prov_name]["strokeColor"]="#FFFF00";
        }else{
            $prov_polys[$prov_name]["fillColor"]="#00FF00";
            $prov_polys[$prov_name]["strokeColor"]="#00FF00";
            }

            //$province_polys[$prov_name][”intensity”]+=$case->intensity;
        }
        return $prov_polys;
    }
    public function getCasesPerBarangay($cases,$bgy_polys){
        $bgy_cases = array();
        $this->load->model(”model_cases”);
        foreach($cases as $case){
            $bgy_name = strtolower($this->model_cases->bgyname_from_bgycode($case->
                 barangay_code));
            $bgy_cases[$bgy_name] = $bgy_polys[$bgy_name];
            $bgy_cases[$bgy_name][”num_cases”]+=$case->num_cases;
            $bgy_cases[$bgy_name][”fillColor”]=”#FF0000”;
            $bgy_cases[$bgy_name][”strokeColor”]=”#FF0000”;
            $bgy_cases[$bgy_name][”onmouseover”]=”alert(”.$bgy_cases[$bgy_name][”num_cases
                 ”].”)”;
            //$bgy_polys[$bgy_name][”intensity”]+=$case->intensity;
        }
        return $bgy_cases;
    }
    public function loadMap($lat=”Manila”,$long=”Philippines”){
        $status = strlen($this->input->post(”data_status”))? explode(”,”,$this->input->post(”
             data_status”)) : array(”queued”,”final”);

        $this->year = strlen($this->input->post(”year”)==0)? date(”Y”) : $this->input->post(”
             year”);
        $this->load->library(’googlemaps’);
        $this->load->library(’excel’);

        $objReader = PHPExcel_IOFactory::createReader(’Excel5’);
        $config = array(
            ”center” => $lat. ”, ” . $long,
            ”zoom” => $lat==”Manila”? 5: 8,
            ”places” => "TRUE",
            ’placesAutocompleteInputID’ => ’zoom’,
            ’placesAutocompleteBoundsMap’ => "TRUE",
            ’placesAutocompleteOnChange’ => ’zoom_to_place();’,
            ”map_type”  => "TERRAIN",
            ”map_height” =>”100%”
            //”minifyJS”  => "TRUE"
            //,”zoom” => ”auto”
        );

        $this->googlemaps->initialize($config);

        $polys = $this->getShapesPerDivision(strtolower($this->input->post(”division”)),
             $status);

        if(isset($polys)){
            foreach($polys as $poly){
                $this->googlemaps->add_polygon($poly);
            }
        }

        $colony_markers = $this->getColonies($status);
        foreach($colony_markers as $marker){
            $this->googlemaps->add_marker($marker);
        }

        $data = array(
            ”map” => $this->googlemaps->create_map(),
            ”data_status”   => $status
        );
        $this->show_map($data);
    }
    public function show_map($data=array()){
        $data[”title”] = ”Map”;
        $data[”view”] = ”view_map”;
        $this->load->view(”view_template”,$data);
    }

}
```

Listing 25: sample.php

```
<?php
/**
 * Created by PhpStorm.
 * User: Gerald
 * Date: 2/20/14
 * Time: 9:20 PM
 */

class sample extends CI_Controller{
    public function index(){
        redirect(”colony/view_all”);
    }
    public function get_userslist(){
        $this->load->model(”model_users”);
        $users = $this->model_users->get_all();
        foreach($users as $user){
            $users_list[] = $user->username . ” (” . $user->last_name . ”, ” . $user->
                 first_name . ” ” . $user->middle_name . ”)”;
        }
        return $users_list;
```

```
}
public function add($id,$name){
    $data = array(
        "colony_id"      => $id,
        "site_name"      => $name,
        "action"         => "sample/add_attempt",
        "task"           => "Add",
        "title"          => "New Sample Data",
        "users_list"     => $this->get_userslist(),
        "view"           => "sample/view_sample_add",
        "sampledata"     => array(
            "date_time"          => $this->input->post("date_time"),
            "total_count"        => $this->input->post("total_count"),
            "live_count"         => $this->input->post("live_count"),
            "infected_count"     => $this->input->post("infected_count"),
            "vegetation"         => $this->input->post("vegetation"),
            "exposure"           => $this->input->post("exposure"),
            "weather"            => $this->input->post("weather"),
            "collectors_names"   => $this->input->post("collectors_names")
        )
    );
    $this->load->view("view_template",$data);
}
public function add_attempt(){
    $this->load->library("form_validation");
    $this->load->model("model_samples");
    $this->form_validation->set_rules("date_time", "Date and Time","required");
    $this->form_validation->set_rules("total_count", "Total Count","required|numeric|trim
        ");
    $this->form_validation->set_rules("live_count", "Live Count","required|numeric|trim");
    $this->form_validation->set_rules("infected_count", "Infected Count","required|numeric
        |trim");
    $this->form_validation->set_rules("collectors_names", "Collectors","required");
    $this->form_validation->set_rules("vegetation", "Vegetation","required");
    $this->form_validation->set_rules("exposure", "Exposure","required");

    $sample = array(
        "colony_id" => $this->input->post("colony_id"),
        "date_time" => $this->input->post("date_time"),
        "total_count" => $this->input->post("total_count"),
        "live_count" => $this->input->post("live_count"),
        "infected_count" => $this->input->post("infected_count"),
        "vegetation" => $this->input->post("vegetation"),
        "exposure" => $this->input->post("exposure"),
        "weather" => $this->input->post("weather"),
    );
    $collectors = $this->input->post("collectors_names");
    if($this->form_validation->run()){
        $query =$this->model_samples->add($sample,$collectors);
        if(!$query){
            redirect("sample/view/".$query['colony_id']."/".rawurlencode($query['date_time
                ']) ."/1");
        }else{
            redirect("sample/view/".$query['colony_id']."/".rawurlencode($query['date_time
                ']) ."/1");
        }
    }else{
        $data = array(
            "colony_id"      => $sample["colony_id"],
            "site_name"    => $this->input->post("site_name"),
            "action"         => "sample/add_attempt",
            "task"           => "Add",
            "title"          => "New Sample Data",
            "validation_errors" =>validation_errors(),
            "users_list"     => $this->get_userslist(),
            "sampledata"     => array_merge($sample,array("collectors_names"=>$collectors))
                ,
            "view"           => "sample/view_sample_add"
        );

        $this->load->view("view_template",$data);
    }
}
public function edit($id,$name,$date_time){
    $this->load->model("model_samples");
    $date_time = rawurldecode($date_time);
    $sampledata = $this->model_samples->get($id,$date_time);
    $sampledata = (array)$sampledata[0];
    $collectors = $this->model_samples->get_collectors($id,$date_time);
    $sampledata["collectors_names"] = (count($collectors))?'"'.implode($collectors,'", "')
        .'"' : "";
    $data = array(
        "colony_id"      => $id,
        "site_name"    => $name,
        "sampledata"     => $sampledata,
        "action"         => "sample/edit_attempt",
        "task"           => "Edit",
        "title"          => "Edit Sample Data",
        "users_list"     => $this->get_userslist(),
        "view"           => "sample/view_sample_add"
    );
    $this->load->view("view_template",$data);
}
public function edit_attempt(){
    $this->load->model("model_samples");

    $this->load->library("form_validation");
    $this->form_validation->set_rules("date_time", "Date and Time","required");
```

```
        $this->form_validation->set_rules(”total_count”, ”Total Count”,”required|numeric|trim
            ”);
        $this->form_validation->set_rules(”live_count”, ”Live Count”,”required|numeric|trim”);
        $this->form_validation->set_rules(”infected_count”, ”Infected Count”,”required|numeric
            |trim”);
        $this->form_validation->set_rules(”collectors_names”, ”Collectors”,”required”);
        $this->form_validation->set_rules(”vegetation”, ”Vegetation”,”required”);
        $this->form_validation->set_rules(”exposure”, ”Exposure”,”required”);

        $sampledata    = array(
            ”colony_id”          => $this->input->post(”colony_id”),
            ”date_time”          => $this->input->post(”date_time”),
            ”total_count”        => $this->input->post(”total_count”),
            ”live_count”         => $this->input->post(”live_count”),
            ”infected_count”     => $this->input->post(”infected_count”),
            ”vegetation”         => $this->input->post(”vegetation”),
            ”exposure”           => $this->input->post(”exposure”),
            ”weather”            => $this->input->post(”weather”)
        );
        $collectors = html_entity_decode($this->input->post(”collectors_names”));
        if($this->form_validation->run()){
            $query =$this->model_samples->update($sampledata,$collectors);
            if($query!=false){
                redirect(”sample/view/”.$sampledata[’colony_id’].”/”.rawurlencode($sampledata
                    [’date_time’]).”/1”);
            }else{
                echo ”db error”;
            }
        }else{
            $data = array(
                ”colony_id”      => $sampledata[”colony_id”],
                ”site_name”   => $this->input->post(”site_name”),
                ”action”         => ”sample/edit_attempt”,
                ”task”           => ”Add”,
                ”title”          => ”New Sample Data”,
                ”validation_errors” =>validation_errors(),
                ”users_list”     => $this->get_userslist(),
                ”sampledata”     => array_merge($sampledata,array(”collectors_names”=>
                    $collectors)),
                ”view”           => ”sample/view_sample_add”
            );
            $this->load->view(”view_template”,$data);
        }
    }
    public function view($id,$date_time,$msg=-1){
        $this->load->model(”model_samples”);
        $this->load->model(”model_colonies”);
        $date_time = rawurldecode($date_time);
        $sitename = $this->model_colonies->get($id)->site_name;
        $sampledata = $this->model_samples->get($id,$date_time);
        $data = array(
            ”success_msg”   => $msg,
            ”sampledata”    => $sampledata[0],
            ”sitename”    => $sitename,
            ”title”          => ”Sample Data for ” . $sitename,
            ”collectors_names” => implode($this->model_samples->get_collectors($id,$date_time)
                ,”<br/>”),
            ”view”               => ”sample/view_sample_id”
        );
        $this->load->view(”view_template”,$data);
    }
    public function view_all($colony_id){
        $this->load->model(”model_colonies”);
        $sitename = $this->model_colonies->get($colony_id)->site_name;
        if(strlen($this->input->post(’add’))){
            redirect(”sample/add/$colony_id/”. rawurlencode($sitename));
        }else{
            $this->load->model(”model_samples”);
            if(strlen($this->input->post(’update_form’))){
                //print_r($this->input->post(’selected_samples’));
                foreach($this->input->post(’selected_samples’) as $date_time){
                    $this->model_samples->move($colony_id,$date_time,$this->input->post(’
                        update_action’));
                }
            }

            $samples = $this->model_samples->get_all($colony_id);
            foreach($samples as $sample){
                $sample->collectors_names = implode($this->model_samples->get_collectors(
                    $sample->colony_id,$sample->date_time),”, ”);
            }

            $data = array(
                ”samples”   => $samples,
                ”sitename”  => $sitename,
                ”title”     => ”List of samples for $sitename”,
                ”view”      => ”sample/view_sample_all”
            );
            $this->load->view(”view_template”,$data);
        }
    }
    public function search($term){

    }
}
```

Listing 26: stat.php

```
<?php
/**
 * Created by PhpStorm.
 * User: Gerald
 * Date: 2/21/14
 * Time: 8:38 AM
 */

class stat extends CI_Controller{
    var $data;
    function index(){
        $this->import();
    }
    function import(){
        $this->load->view("view_stat");
    }
    function load_file(){
        $this->load->model("model_import");

        if($this->input->post('upload')){
            $this->model_import->do_upload();  //upload the file
            $upload_data     = $this->upload->data();
            $file_name       = $upload_data['file_name'];
            $sheetData       = $this->model_import->import_barangay($file_name);
            /*
            //get indices
            foreach($sheetData[1] as $key => $value){
                $indices[$value] = $key;
            }
            $prev_provcode = "x";
            $provinces = array();
            for($row=2;$row<=count($sheetData);$row++){
                $curr_row = $sheetData[$row];
                $curr_provcode = $curr_row[$indices["provcode"]];
                if($curr_provcode!=$prev_provcode){
                    $provinces["$curr_provcode"] =$curr_row[$indices["province"]];
                }
            }

            print_r($provinces);
            */
        }

        $this->import($this->data);
    }
    function confirm(){
        $this->load->model("model_import");
        $arr = $this->input->post("data");

    }

    public function show($maincontent,$data=array()){
        $data["userdata"] = $this->session->userdata;
        //$this->restrict();
        $this->load->view("includes/header",$data);
        if(is_string($maincontent)){
            $this->load->view($maincontent,$data);
        }else{
            foreach($maincontent as $page){
                $this->load->view($page,$data);
            }
        }
        $this->load->view("includes/footer");
    }
}
```

Listing 27: sync.php

```
<?php
/**
 * Created by PhpStorm.
 * User: Gerald
 * Date: 2/19/14
 * Time: 7:10 PM
 */

class Synchronize extends CI_Controller{
    var $int = "offline";
    var $ext = "online";

    public function index(){
        $this->options();
    }
    public function options($data = array()){
        $this->load->view("view_template",array_merge($data,array(
            "title" => "Synchronize",
            "view"  => "synchronize/view_sync_options",
            "int"   => $this->int,
            "ext"   => $this->ext
        )));
    }
    public function to_file(){
        $this->load->library('excel');
        $includes = $this->input->post("data");

        $sheet = new PHPExcel();                // Load Excel library and create an instance
             of a PHPExcel object.
        $sheet->getProperties()->setTitle('Colonies')->setDescription('Colonies');
        $sheet->createSheet();
        $sheet->createSheet();

        if(in_array("colony",$includes)){
            $this->load->model("model_colonies");
            $sheet->setActiveSheetIndex(0)->setTitle("Colonies");          // Set Title of the
                 Excel file and the sheet to write to.
            $sheet = $this->write_file($sheet,$this->model_colonies->get_all());
        }
        if(in_array("sample",$includes)){
            $this->load->model("model_samples");
            $sheet->setActiveSheetIndex(1)->setTitle("Samples");          // Set Title of the
                 Excel file and the sheet to write to.
            $sheet = $this->write_file($sheet,$this->model_samples->get_all());

            $sheet->createSheet();                  // create a sheet for user_sample
            $sheet->createSheet();                  // create a sheet for selected users
            $sheet->setActiveSheetIndex(3)->setTitle("Samples_Users");          // Set Title of
                  the Excel file and the sheet to write to.
            $user_sample = $this->model_samples->get_user_sample();

            $sheet = $this->get_titles($sheet,$user_sample);

            $row = 2;
            $user_ids = array();
            foreach ($user_sample as $table_row) {   // Populate values on rows and columns.
                $col = 0;
                $user_ids[] = $table_row->user_id;
                foreach ($table_row as $field_val) {
                    $sheet->getActiveSheet()->setCellValueByColumnAndRow($col, $row,
                         $field_val);
                    $col++;
                }
                $row++;
            }
            $user_ids = array_unique($user_ids);

            $users = array();

            $this->load->model("model_users");
            foreach($user_ids as $user_id){
                $u = 0;
                $users[] = $this->model_users->get($user_id)[0];
            }
            $sheet->setActiveSheetIndex(4)->setTitle("Users");          // Set Title of the
                 Excel file and the sheet to write to.
            $sheet = $this->write_file($sheet,$users);
        }
        if(in_array("case",$includes)){
            $this->load->model("model_cases");
            $sheet->setActiveSheetIndex(2)->setTitle("Cases");          // Set Title of the
                 Excel file and the sheet to write to.
            $sheet = $this->write_file($sheet,$this->model_cases->get_all_cases());

        }

        $sheet_writer = PHPExcel_IOFactory::createWriter($sheet, 'Excel5'); // Create a writer
             to write values to.

        // Setup headers to download the excel file.
        header('Content-Type: application/vnd.ms-excel');
        header('Content-Disposition: attachment;filename="schisto-GIS-data-'.date('dMy').'.xls
            "');
        header('Cache-Control: max-age=0');

        $sheet_writer->save('php://output');
    }
    public function write_file($sheet,$table){
        $sheet = $this->get_titles($sheet,$table);
        $sheet = $this->get_rows($sheet,$table);
        return $sheet;
    }
    public function get_titles($sheet,$table){
        $col = 0;                               // Populate column titles on first row.
        foreach ($table[0] as $field=>$value) {
            $sheet->getActiveSheet()->setCellValueByColumnAndRow($col, 1, $field);
            $col++;
        }
        return $sheet;
    }
    public function get_rows($sheet,$table){
        $row = 2;
        foreach ($table as $table_row) {   // Populate values on rows and columns.
            $col = 0;
            foreach ($table_row as $field_val) {
                $sheet->getActiveSheet()->setCellValueByColumnAndRow($col, $row, $field_val);
                $col++;
            }
            $row++;
        }
        return $sheet;
    }
    public function compare_data($a,$b){
        $has_changed = false;
        foreach($a as $key=>$val){
            $has_changed = $val!=$b[$key];
            if($has_changed) break;
        }
        return $has_changed;
    }
    public function read_file(){

        $this->load->model("model_import");
        $this->model_import->do_upload();  //upload the file
        $upload_data    = $this->upload->data();
        if($upload_data){
            $file_name  = $upload_data['file_name'];
            $objPE      = $this->model_import->get_objPHPExcel($file_name);

            $objPE->setActiveSheetIndex(2);
            $cases      = $objPE->getActiveSheet()->toArray(null,true,true,true);

            $objPE->setActiveSheetIndex(0);
            $colonies   = $objPE->getActiveSheet()->toArray(null,true,true,true);
            $colony_conflict = array();
            $colony_success = 0;
            if(count($colonies)){
                $this->load->model("model_colonies");
                for($row=2;$row<=count($colonies);$row++){
                    $curr_row = array_combine($colonies[1], array_values($colonies[$row]));
                    // check if ID is taken
                    $fromdb = $this->model_colonies->get($curr_row["colony_id"]);

                    // if ID is taken
                    if(isset($fromdb)){
                        // check for changes
                        // if changed, add to discrepancy list, else, do nothing
                        if($this->compare_data($fromdb,$curr_row)){
                            $curr_row["conflict_type"] = "same_id";
                            $colony_conflict[] = array($this->int=>(array)$fromdb,$this->ext=>
                                $curr_row);
                        }
                    }else{
                        // check if name is taken
                        $fromdb = $this->model_colonies->getby_name($curr_row["site_name"]);
                        if(isset($fromdb)){
                            // there is a discrepancy in colony_id
                            $curr_row["conflict_type"] = "same_name";
                            $curr_row["old_id"] = $fromdb->colony_id;
                            $colony_conflict[] = array($this->int=>(array)$fromdb,$this->ext=>
                                $curr_row);
                        }elseif($this->model_colonies->insert($curr_row)!=-1){
                            $colony_success++;
                        }
                    }
                }
            }

            $objPE->setActiveSheetIndex(1);
            $sample_conflict = array();
            $samples    =   $objPE->getActiveSheet()->toArray(null,true,true,true);
            $sample_success = 0;
            if(count($samples)){
                //re_organize sample_user
                $objPE->setActiveSheetIndex(3);
                $temp = $objPE->getActiveSheet()->toArray(null,true,true,true);
                $sample_user = array();
                foreach($temp as $pair){
                    if(!isset($sample_user[$pair["A"]])){
                        $sample_user[$pair["A"]] = array();
                    }
                    $sample_user[$pair["A"]][$pair["B"]][] = $pair["C"];
                }
                $this->load->model("model_samples");

                for($row=2;$row<=count($samples);$row++){
                    $curr_row = array_combine($samples[1], array_values($samples[$row]));
                    // check if data for that date and id exists
                    $c_id   = $curr_row["colony_id"];
                    $dt     = $curr_row["date_time"];
                    $fromdb = $this->model_samples->get($c_id,$dt);
                    if(count($fromdb)){
                        // check if conflict
                        if($this->compare_data($fromdb[0],$curr_row)){
                            $sample_conflict[] = array($this->int=>(array)$fromdb[0],$this->
                                ext=>$curr_row);
                        }elseif(isset($sample_user[$c_id][$dt])){
                            // sample data already exists

                            // if sample_collector from db and from xls contain data
                            $su_fromdb = $this->model_samples->get_userid($c_id,$dt);
                            $su_count = count($su_fromdb);
                            if($su_count){
                                $su_curr = $sample_user[$c_id][$dt];
                                if(count($su_curr)==$su_count){
                                    for($srow=0;$srow<$su_count;$srow++){
                                        if($su_curr[$srow]!=$su_fromdb[$srow]->user_id){
                                            $sample_conflict[] = array($this->int=>(array)
                                                $fromdb[0],$this->ext=>$curr_row);
                                            break;
                                        }
                                    }
                                }else{
                                    // collectors for this sample were changed
                                    $sample_conflict[] = array($this->int=>(array)$fromdb[0],
                                        $this->ext=>$curr_row);
                                }
                            }else{
                                // simply add sample_user
                                foreach($sample_user[$c_id][$dt] as $user){
                                    $this->model_samples->insert_user(array(
                                        "colony_id" => $c_id,
                                        "date_time" => $dt,
                                        "user_id"    => $user
                                    ));
                                }
                            }
                        }
                    }else{

                        if($this->model_samples->insert_sample($curr_row)) $sample_success++;
                        if(isset($sample_user[$c_id][$dt])){
                            foreach($sample_user[$c_id][$dt] as $user){
                                $this->model_samples->insert_user(array(
                                    "colony_id" => $c_id,
                                    "date_time" => $dt,
                                    "user_id"    => $user
                                ));
                            }
                        }
                    }
                }
            }

            $data = array(
                "title" => "Synchronize",
                "view"  => "synchronize/view_sync_from_file",
                "int"   => $this->int,
                "ext"   => $this->ext,
                "colony"=> array(
                    "total"      => count($colonies)-1,
                    "success"    => $colony_success,
                    "changed"  => $colony_conflict
                ),
                "sample"=> array(
                    "total"      => count($samples)-1,
                    "success"    => $sample_success,
                    "changed"  => $sample_conflict
                )
            );
            $this->load->view("view_template",$data);
        }else{
            $this->options(array("err"=>"Error in uploading file."));
        }

    }
    public function from_file(){
        if(strlen($this->input->post("continue"))){
            if($this->input->post("colony_action")=="Apply Changes"){
                $colonies = json_decode($this->input->post("colonies"));
                $this->load->model("model_colonies");
                $this->load->model("model_samples");
                foreach($colonies as $colony){
                    $arr = (array)$colony;
                    if($arr[$this->ext]->conflict_type=="same_id"){
                        //replace colony with given id
                        unset($arr[$this->ext]->conflict_type);
                        $this->model_colonies->update((array)$arr[$this->ext]);
                    }else{
                        $id = $arr[$this->ext]->old_id;
                        unset($arr[$this->ext]->old_id);
                        unset($arr[$this->ext]->conflict_type);

                        //add new colony to database
                        $this->model_colonies->insert($arr[$this->ext]);

                        //change id of samples for old colony
                        $this->model_samples->update_sample_id($id,$arr[$this->ext]->colony_id
                            );

                        //remove old colony with same name using its id
                        $this->model_colonies->delete(array("colony_id"=>$id));
                    }
                }
                $this->options(array("msg"=>" Synchronization complete."));
            }
            if($this->input->post("sample_action")=="Apply Changes"){

            }
            redirect("colony");
        }else{
            $this->read_file();
        }
    }
}
```

Listing 28: user.php

```
<?php
class User extends CI_Controller{
    public function index(){
        redirect("user/view_all");
    }
```

```
public function login($data=array()){
    if($this->session->userdata("is_logged_in")){
        redirect("map");
    }else{
        $this->load->view("user/view_user_login",$data);
    }
}

public function login_attempt(){
    $this->load->library("form_validation");
    $this->form_validation->set_rules("username","Username","required|alpha_numeric|trim")
        ;   //trim - remove white spaces
    $this->form_validation->set_rules("password","Password","required|md5");

    if($this->form_validation->run()){  // if all the rules have been verified
        $this->load->model("model_users");
        $this->load->model("model_roles");
        $username = $this->input->post("username");
        if($this->model_users->can_log_in()){
            $id = $this->model_users->get_id($username);
            if($this->model_users->is_enabled($id)){
                $data = array(
                    "user_id"       => $id,
                    "username"      => $username,
                    "is_logged_in"  => true,
                    "user_roles"    => $this->model_roles->get_user_roles($id)
                );
                $this->session->set_userdata($data);
                redirect($this->session->userdata["redirect"]);
            }else{
                $this->login(array("errors"=>"Sorry, this account has been disabled.
                     Please contact your system administrator for more information."));
            }
        }else{
            $this->login(array("errors"=>"Sorry, unrecognized username or password."));
        }
    }else{
        $this->login(array("errors"=>validation_errors()));
    }
}
public function add($data=array()){
    $this->load->model("model_roles");
    $_data = array(
        "title"         => "New User",
        "roles"         => $this->model_roles->get_roles(),
        "view"          => "user/view_user_add",
        "action"        => "add",
        "curr_roles"    => array(),
        "new_user"      => array(
            "last_name"     => "",
            "first_name"    => "",
            "middle_name"   => "",
            "email"         => "",
            "username"      => $this->input->post("username")
        )
    );
    $this->load->view("view_template",array_merge($data,$_data));
}
public function add_attempt(){
    $this->load->model("model_users");
    $this->load->model("model_roles");

    $this->load->library("form_validation");
    $this->form_validation->set_rules("email","Email","trim|valid_email");
    $this->form_validation->set_rules("role","Role(s)","required");
    $this->form_validation->set_rules("username","Username","required|alpha_numeric|trim|
         is_unique[user.username]");   //trim - remove white spaces
    $this->form_validation->set_rules("password","Password","required|md5");
    $this->form_validation->set_rules("retype_password","Retype Password","required|md5");

    $this->form_validation->set_message("is_unique","That username is already taken.");

    $new_user = array(
        "last_name" => $this->input->post("last_name"),
        "first_name" => $this->input->post("first_name"),
        "middle_name" => $this->input->post("middle_name"),
        "username" => $this->input->post("username"),
        "email" => $this->input->post("email"),
        "password" => md5($this->input->post("password")),
        "account_status" => "enabled"
    );
    $roles = $this->input->post("role");
    if($this->form_validation->run()){  // if all the rules have been verified
        $key = md5(uniqid());

        // send an email
        if(strlen($this->input->post("email"))){
            $this->load->library("email");
            $this->email->from("admin@doh.gov.ph","Schistosomiasis GIS");
            $this->email->to($this->input->post("email"));
            $this->email->subject("New Account at doh.gov.ph");

            $message =  "<p>Thank you for signing up!</p>".
                "<p><a href='".base_url('user/confirm/' . $key)."'> Click here</a> to
                     confirm your account</p>";
            $this->email->message($message);
            $this->email->send();
        }
        //add to database
```

```
            $query = $this->model_users->add_user($new_user,$roles);
            if(isset($query["id"])){
                redirect("user/view/". $query["id"]);
            }
        }else{
            $this->load->view("view_template", array(
                "new_user"=>$new_user,
                "validation_errors" => validation_errors(),
                "title" => "New User",
                "view"  => "user/view_user_add",
                "roles" => $this->model_roles->get_roles(),
                "action"    => "add",
                "curr_roles"=> $this->input->post("role")
            ));
        }
    }
    public function logout(){
        $this->session->sess_destroy();
        $this->login();
    }

    public function view_all(){
        $term = strlen($this->input->post("username"))? $this->input->post("username") : $this
            ->input->post("search-term");
        if(strlen($this->input->post('add'))){
            strlen($term)? $this->add() : redirect("user/add");
        }else{
            $this->load->model("model_users");
            $this->load->model("model_roles");
            if(strlen($this->input->post("update_form"))){
                foreach($this->input->post("selected_users") as $username){
                    $this->model_users->change_status($username,$this->input->post("
                        update_action"));
                }
            }
            $status = $this->input->post("status");
            $status = count($status)==1? $status[0] : null;    //check if status should be
                filtered
            $role_input = $this->input->post("role");
            $roles = is_array($role_input)&&count($role_input)<4&&count($role_input)>0?
                $role_input : null;

            $users = $this->model_users->get_all($term, $status,$roles);

            foreach($users as $user){
                $user->roles = implode($this->model_roles->get_user_role_names($user->user_id)
                    ,", ");
            }
            $data = array(
                "roles" => $this->model_roles->get_roles(),
                "view"  => "user/view_user_all",
                "users" => $users,
                "term"  => $term
            );
            $this->load->view("view_template",$data);

        }
    }
    public function view($id){
        if(isset($id)){
            $this->load->model("model_users");
            $this->load->model("model_roles");
            $curr_user = $this->model_users->get($id);
            if(count($curr_user)){
                $curr_user = $curr_user[0];
                $curr_user->roles = $this->model_roles->get_user_role_names($id);
                $data = array(
                    "user"      =>  $curr_user,
                    "title"     =>  $curr_user->username,
                    "view"      =>  "user/view_user_id"
                );
                $this->load->view("view_template",$data);
            }else{
                redirect("user/view_all");
            }
        }else{
            redirect("user/view_all");
        }
    }
    public function edit($id=null){
        if(isset($id)){
            $this->load->model("model_users");
            $this->load->model("model_roles");
            $curr_user = $this->model_users->get($id);
            $curr_user = (array)$curr_user[0];
            $data = array(
                "new_user"      =>  $curr_user,
                "title"     =>  "Edit " . $curr_user["username"],
                "view"      =>  "user/view_user_add",
                "action"    => "edit",
                "curr_roles"=>$this->model_roles->get_user_roles($id),
                "roles"     => $this->model_roles->get_roles()
            );
            $this->load->view("view_template",$data);
        }else{
            redirect("user/add");
        }
    }
    public function change_password(){
        $this->load->library("form_validation");
        $this->form_validation->set_rules("old","Old Password","required|md5");
        $this->form_validation->set_rules("password","New Password","required|md5");
        $this->form_validation->set_rules("retype_password","Retyped New Password","required|
            md5");

        $this->load->model("model_users");
        $this->load->model("model_roles");
        $id = $this->input->post("user_id");
        $curr_user = $this->model_users->get($id);
        $curr_user = $curr_user[0];
        $curr_user->roles = $this->model_roles->get_user_role_names($id);

        if($this->form_validation->run()){
            $old_pass = $curr_user->password;
            if($old_pass==$this->input->post("old")){
                $this->model_users->update_user(array("user_id"=>$id,"password"=>$this->input
                    ->post("password")));
                $this->load->view("view_template",array(
                    "user"      =>  $curr_user,
                    "title"     =>  "Password Changed",
                    "view"      =>  "user/view_user_id",
                    "success_msg"   => 3
                ));
            }else{
                $this->load->view("view_template",array(
                    "user"      =>  $curr_user,
                    "title"     =>  "Password Change Failed",
                    "view"      =>  "user/view_user_id",
                    "fail_msg"   => "Incorrect password."
                ));
            }
        }else{
            //unhandled error
            $this->load->view("view_template",array(
                "user"      =>  $curr_user,
                "title"     =>  "Password Change Failed",
                "view"      =>  validation_errors(),
                "fail_msg"   => "Incorrect password."
            ));
        }
    }

    public function disable($id){
        $this->load->model("model_users");
        $this->model_users->disable($id);
    }
    public function enable($id){
        $this->load->model("model_users");
        $this->model_users->enable($id);
    }
    public function username_check($username,$id){
        $this->load->model("model_users");
        if(!$this->model_users->unique_edit($id, $username)){
            $this->form_validation->set_message('username_check', 'The username "'. $username
                . '" is already being used by another account.');
            return false;
        }
        return true;
    }
    public function edit_attempt(){
        echo validation_errors();
        $this->load->model("model_users");

        $this->load->library("form_validation");
        $this->form_validation->set_rules("email","Email","trim|valid_email");
        $this->form_validation->set_rules("role","Role(s)","required");

        $user_id = $this->input->post("user_id");
        $this->form_validation->set_rules("username","Username","required|trim|alpha_numeric|
            callback_username_check[$user_id]");    //trim - remove white spaces

        $this->load->model("model_users");
        $this->load->model("model_roles");
        $new_user = array(
            "user_id" => $user_id,
            "last_name" => $this->input->post("last_name"),
            "first_name" => $this->input->post("first_name"),
            "middle_name" => $this->input->post("middle_name"),
            "username" => $this->input->post("username"),
            "email" => $this->input->post("email"),
            "account_status" => $this->input->post("account_status")
        );
        $roles = $this->input->post("role");
        if($this->form_validation->run()){  // if all the rules have been verified
            if($this->model_users->update($new_user,$roles)){
                redirect("user/view/$user_id");
            }else{
                echo "db error";
            }
        }else{
            $this->load->view("view_template", array(
                "new_user"=>$new_user,
                "validation_errors" => validation_errors(),
                "title"         => "Edit User",
                "view"          => "user/view_user_add",
                "roles"         => $this->model_roles->get_roles(),
                "action"        => "edit",
                "curr_roles"    => $this->input->post("role")
            ));
        }
    }
    public function search($arg1='',$arg2='',$offset=0){
        $term = $this->input->post(”username”);

        $this->load->library(”pagination”);
        $config = array(
            ”page_base” => base_url(”user/search”)
        );
    }
    public function reset_password(){
    }
}
```

## Models

Listing 29: cases.php

```
<?php
/**
 * Created by PhpStorm.
 * User: Gerald
 * Date: 3/1/14
 * Time: 12:07 PM
 */

class Model_cases extends CI_Model{
    public function get_all_cases($status){
        $year = $this->input->post(”year”);
        $year = (strlen($year)==0)? 2014 : $year;
        $this->db->like(”date”,$year,”after”);
        if(count($status)==1) $this->db->where(”status”,$status[0]);
        $query = $this->db->get(”barangay_cases”);
        return $query->result();
    }
    public function provname_from_bgycode($bgycode){
        $city_code = $this->db->get_where(”barangay”,array(”barangay_code”=>$bgycode))->result
            ();
        $city_code = $city_code[0]->city_code;
        $prov_code = $this->db->get_where(”city”,array(”city_code”=>$city_code))->result();
        $prov_code = $prov_code[0]->province_code;
        $query = $this->db->get_where(”province”,array(”province_code”=>$prov_code))->result()
            ;
        return $query[0]->name;
    }
    public function bgyname_from_bgycode($bgycode){
        $query = $this->db->get_where(”barangay”,array(”barangay_code”=>$bgycode))->result();
        return $query[0]->city_code;
    }
    public function get_all_provinces(){
        return $this->db->get(”province”)->result();
    }
    public function get_population($bgycode, $year){
        $this->db->like(”date”,$year,”after”);
        $query = $this->db->get_where(”barangay_population”,array(”barangay_code”=>$bgycode))
            ->result();
        return $query[0]->population;
    }

}
```

Listing 30: colonies.php

```
<?php
/**
 * Created by PhpStorm.
 * User: Gerald
 * Date: 2/15/14
 * Time: 8:36 PM
 */

class Model_colonies extends CI_Model{
    public function insert($data){
        $query = $this->db->insert(”colony”,$data);
        if($query){
            return $this->db->insert_id();
        }
        return -1;
    }
    public function delete($data){
        $this->db->delete(”colony”,$data);
    }
    public function get($id){
        $q = $this->db->get_where(”colony”,array(”colony_id”=>$id))->result();
        return (count($q))? $q[0] : null;
    }
    public function getby_name($name){
        $q = $this->db->get_where(”colony”,array(”site_name”=>$name))->result();
        return (count($q))? $q[0] : null;
    }
    public function get_all($starting=”all”,$select=null,$status=null){
        if(isset($select))      $this->db->select($select);
        if($starting!=”all”)    $this->db->like(”site_name”,$starting,”after”);
        if(isset($status))      $this->db->where("status",$status);

        $this->db->order_by("site_name","asc");
        return $this->db->get("colony")->result();
    }
    public function move($site_name, $status){
        $this->db->update("colony", array("status"=>$status? "final":"queued"), "site_name = '
            $site_name'");
    }
    public function search($term){
        $this->$db->like("site_name",$term);
        $this->$db->or_like("longitude",$term);
        $this->$db->or_like("latitude",$term);
        $this->$db->or_like("area",$term);
        return $this->db->get("colony")->result();
    }
    public function unique_edit($id,$name){
        $colony = $this->db->get_where("colony",array("site_name"=>$name))->result();
        if(count($colony)==0||(count($colony)&&$id==$colony[0]->colony_id)){      // not yet
             used or name didn't change and belongs to the original colony
            return true;
        }
        return false;
    }
    public function update($data){
        if(isset($data)){
            $this->db->update("colony", $data, "colony_id = " . $data["colony_id"]);
            return true;
        }
        return false;
    }

}
```

## Listing 31: roles.php

```
<?php
class model_roles extends CI_Model{

    public function get_user_roles($user_id){
        $user_roles = $this->db->get_where("user_role",array("user_id"=>$user_id))->result();
        $role_ids = array();
        foreach($user_roles as $user_role){
            $role_ids[] = $user_role->role_id;
        }
        return $role_ids;
    }
    public function get_user_role_names($user_id){
        $user_roles =$this->db->get_where("user_role",array("user_id"=>$user_id))->result();
        $role_names = array();
        foreach($user_roles as $user_role){
            $query = $this->db->get_where("role",array("role_id"=>$user_role->role_id))->
                result();
            $role_names[$user_role->role_id] = $query[0]->name;
        }
        return $role_names;
    }

    public function get_roles(){
        return $this->db->get("role")->result();
    }

}
```

## Listing 32: imports.php

```
<?php
/**
 * Created by PhpStorm.
 * User: Gerald
 * Date: 2/11/14
 * Time: 11:26 AM
 */

//namespace application\models;


class Model_import extends CI_Model{
    var $file_path;
    var $sheetData;

    function Model_import(){
        $this->file_path = realpath(APPPATH . "../excel_files");
    }
    function do_upload(){
        delete_files($this->file_path); //delete any previously uploaded file
        $config = array(
            "allowed_types" => "xls",
            "upload_path" => $this->file_path
        );
        $this->load->library("upload",$config);

        return $this->upload->do_upload();
    }
```

```
function do_read($filename){
    return $this->get_objPHPExcel($filename)->getActiveSheet()->toArray(null,true,true,
        true);
}
function get_objPHPExcel($filename){
    $this->load->library(”excel”);
    set_time_limit(0);
    ini_set('memory_limit', '-1');
    $cacheMethod = PHPExcel_CachedObjectStorageFactory::cache_to_phpTemp;
    $cacheSettings =    array( ' memoryCacheSize ' =>'8MB');
    //create a phpexcel reader
    $objReader = PHPExcel_IOFactory::createReader('Excel5');
    //load the file form file_path
    return $objReader->load($this->file_path. ”\\” . $filename);
}

function import_provinces($filename){
    $this->load->library(”excel”);
    set_time_limit(0);
    ini_set('memory_limit', '-1');
    $cacheMethod = PHPExcel_CachedObjectStorageFactory::cache_to_phpTemp;
    $cacheSettings =    array( ' memoryCacheSize ' =>'8MB');
    //create a phpexcel reader
    $objReader = PHPExcel_IOFactory::createReader('Excel5');
    //load the file form file_path
    $objPHPExcel = $objReader->load($this->file_path. ”\\” . $filename);
    $objWorksheet = $objPHPExcel->setActiveSheetIndex(”0”);
    $highestRow = $objWorksheet->getHighestRow();
    $provcode =”128”;
    for ($row = 2; $row <= $highestRow; ++$row) {
        // Fetch the data of the columns you need
        if($provcode!=$objWorksheet->getCellByColumnAndRow(3, $row)->getValue()){
            $provcode=$objWorksheet->getCellByColumnAndRow(3, $row)->getValue();
            $data = array(
                ”province_code” => $objWorksheet->getCellByColumnAndRow(3, $row)->getValue
                    (),
                ”name” => $objWorksheet->getCellByColumnAndRow(2, $row)->getValue(),
                ”region_code” => $objWorksheet->getCellByColumnAndRow(1, $row)->getValue()
            );
            $this->db->insert(”province”,$data);
        }

    }
    echo $row;
    return $objPHPExcel->getActiveSheet()->toArray(null,true,true,true);

}
function import_cities($filename){
    $this->load->library(”excel”);
    set_time_limit(0);
    ini_set('memory_limit', '-1');
    $cacheMethod = PHPExcel_CachedObjectStorageFactory::cache_to_phpTemp;
    $cacheSettings =    array( ' memoryCacheSize ' =>'8MB');
    //create a phpexcel reader
    $objReader = PHPExcel_IOFactory::createReader('Excel5');
    //load the file form file_path
    $objPHPExcel = $objReader->load($this->file_path. ”\\” . $filename);
    $objWorksheet = $objPHPExcel->setActiveSheetIndex(”0”);
    $highestRow = $objWorksheet->getHighestRow();
    $citycode =”0”;
    for ($row = 2; $row <= $highestRow; ++$row) {
        // Fetch the data of the columns you need
        if($citycode!=$objWorksheet->getCellByColumnAndRow(5, $row)->getValue()){
            $citycode=$objWorksheet->getCellByColumnAndRow(5, $row)->getValue();
            $data = array(
                ”province_code” => $objWorksheet->getCellByColumnAndRow(3, $row)->getValue
                    (),
                ”name” => $objWorksheet->getCellByColumnAndRow(4, $row)->getValue(),
                ”city_code” => $objWorksheet->getCellByColumnAndRow(5, $row)->getValue()
            );
            $this->db->insert(”city”,$data);
        }

    }
    echo $row;
    return $objPHPExcel->getActiveSheet()->toArray(null,true,true,true);

}
function import_barangay($filename){
    $this->load->library(”excel”);
    set_time_limit(0);
    ini_set('memory_limit', '-1');
    $cacheMethod = PHPExcel_CachedObjectStorageFactory::cache_to_phpTemp;
    $cacheSettings =    array( ' memoryCacheSize ' =>'8MB');
    //create a phpexcel reader
    $objReader = PHPExcel_IOFactory::createReader('Excel5');
    //load the file form file_path
    $objPHPExcel = $objReader->load($this->file_path. ”\\” . $filename);
    $objWorksheet = $objPHPExcel->setActiveSheetIndex(”0”);
    $highestRow = $objWorksheet->getHighestRow();
    $citycode =”0”;
    for ($row = 36520; $row <= $highestRow; ++$row) {
        // Fetch the data of the columns you need
            $data = array(
                ”type” => $objWorksheet->getCellByColumnAndRow(8, $row)->getValue(),
                ”barangay_code” => $objWorksheet->getCellByColumnAndRow(7, $row)->getValue
                    (),
                ”name” => $objWorksheet->getCellByColumnAndRow(6, $row)->getValue(),
                ”city_code” => $objWorksheet->getCellByColumnAndRow(5, $row)->getValue()
                );
            $data2 = array(
                ”barangay_code” => $objWorksheet->getCellByColumnAndRow(7, $row)->getValue(),
                ”population” => $objWorksheet->getCellByColumnAndRow(9, $row)->getValue(),
                ”date” => date(”Y-m-d H:i:s”)
            );
                $this->db->insert(”barangay”,$data);
                $this->db->insert(”barangay_population”,$data2);


        }
        echo $row;
        return $objPHPExcel->getActiveSheet()->toArray(null,true,true,true);

    }
}
```

## Listing 33: samples.php

```
<?php
/**
 * Created by PhpStorm.
 * User: Gerald
 * Date: 2/15/14
 * Time: 9:08 PM
 */

class Model_samples extends CI_Model{
    public function get($colony_id,$date_time){
        return $this->db->get_where(”sample”,array(”colony_id”=>$colony_id,”date_time”=>
             rawurldecode($date_time)))->result();
    }
    public function get_all($colony_id=null){
        $this->db->order_by(”date_time”,”desc”);
        if(isset($colony_id)){
            $this->db->where(”colony_id”,$colony_id);
        }
        $query = $this->db->get(”sample”);
        return $query->result();
    }
    public function get_by_date($date){
        $this->db->like(”date_time”,”$date”,”after”);
        return $this->db->get(”sample”)->result();
    }
    public function get_by_id($id){
        $this->db->order_by(”date_time”,”desc”);
        return $this->db->get_where(”sample”,array(”colony_id”=>$id))->result();
    }
    public function get_date_by_id($id){
        $query = $this->get_by_id($id);
        if(count($query)){
            return $query[0]->date_time;
        }
        return ”Not yet sampled”;
    }
    public function get_userid($colony_id,$date_time){
        if($colony_id)  $this->db->where(”colony_id”,$colony_id);
        if($date_time)  $this->db->where(”date_time”,$date_time);
        return $this->get_user_sample();
    }
    public function get_collectors($id=null,$date_time=null){
        $users = $this->get_userid($id,$date_time);
        $usernames = array();
        foreach($users as $user){
            $userdata = $this->db->get_where(”user”,array(”user_id”=>$user->user_id))->result
                ();
            $userdata = $userdata[0];
            $usernames[] = $userdata->username . ” (” . $userdata->last_name . ”, ” .$userdata
                ->first_name . ” ” . $userdata->middle_name . ”)”;
        }
        return $usernames;
    }
    public function get_user_sample(){
        return $this->db->get(”sample_collector”)->result();
    }
    public function search($term){

    }
    public function insert_sample($data){

        return $this->db->insert(”sample”,$data);
    }
    public function add($sample,$collectors){
        if(!$this->insert_sample($sample)){
            return false;
        }

        $users = array_unique(explode('”, ”',$collectors));
        foreach($users as $user){
            $username = str_replace('”','',strstr($user,” (”,true));
            $user_id = $this->db->get_where(”user”,array(”username”=>$username))->result();
            $user_id = $user_id[0]->user_id;
            $data = array(
                ”colony_id” => $sample[”colony_id”],
                ”date_time” => $sample[”date_time”],
                ”user_id” => $user_id,
            );
            if(!$this->insert_user($data)){
                return false;
            }
        }
        return $data;
    }
    public function insert_user($data){
        return $this->db->insert(”sample_collector”,$data);
    }
    public function move($id,$date_time,$status){
        $this->db->where(”colony_id”,$id);
        $this->db->where(”date_time”,$date_time);
        $this->db->update(”sample”, array(”status”=>$status? ”final”:”queued”));
    }
    public function update_samples($sample){
        $this->db->where(”colony_id”,$sample[”colony_id”]);
        $this->db->update(”sample”, $sample);
    }
    public function update_sample_id($from,$to){
        $samples = $this->db->get_where(”sample”,array(”colony_id”=>$from))->result();

        foreach($samples as $sample){
            $sample = (array)$sample;
            $sample[”colony_id”] = $to;
            $this->db->insert(”sample”,$sample);
        }
        $this->db->flush_cache();
        $collectors = $this->db->get_where(”sample_collector”,array(”colony_id”=>$from))->
             result();
        foreach($collectors as $user){
            $user = (array)$user;
            $user[”colony_id”] = $to;
            $this->db->insert(”sample_collector”,$user);
        }
        $this->db->query(’SET FOREIGN_KEY_CHECKS=0’);
        $this->db->delete(”sample_collector”,array(”colony_id”=>$from));
        $this->db->delete(”sample”,array(”colony_id”=>$from));
        $this->db->query(’SET FOREIGN_KEY_CHECKS=1’);
    }
    public function update($sample=null,$collectors=null){
        if(isset($sample)){
            $this->db->where(”date_time”,$sample[”date_time”]);
            $this->update_samples($sample);
        }
        if(isset($collectors)){
            $this->db->where(array(”colony_id”=>$sample[”colony_id”],”date_time”=>$sample[”
                 date_time”]));
            $this->db->delete(”sample_collector”);

            $users = array_unique(explode(’”, ”’,$collectors));
            foreach($users as $user){
                $username = str_replace(’”’,’’,strstr($user,” (”,true));
                $user_id = $this->db->get_where(”user”,array(”username”=>$username))->result()
                     ;
                $user_id = $user_id[0]->user_id;
                $data = array(
                    ”colony_id” => $sample[”colony_id”],
                    ”date_time” => $sample[”date_time”],
                    ”user_id” => $user_id,
                );
                $query = $this->db->insert(”sample_collector”,$data);
                if(!$query){
                    return false;
                }
            }
        }
        return true;
    }
}
```

## Listing 34: users.php

```
<?php
class User extends CI_Controller{
    public function index(){
        redirect(”user/view_all”);
    }
    public function login($data=array()){
        if($this->session->userdata(”is_logged_in”)){
            redirect(”map”);
        }else{
            $this->load->view(”user/view_user_login”,$data);
        }
    }

    public function login_attempt(){
        $this->load->library(”form_validation”);
        $this->form_validation->set_rules(”username”,”Username”,”required|alpha_numeric|trim”)
             ;   //trim - remove white spaces
        $this->form_validation->set_rules(”password”,”Password”,”required|md5”);

        if($this->form_validation->run()){  // if all the rules have been verified
            $this->load->model(”model_users”);
            $this->load->model(”model_roles”);
            $username = $this->input->post(”username”);
            if($this->model_users->can_log_in()){
```

```
            $id = $this->model_users->get_id($username);
            if($this->model_users->is_enabled($id)){
                $data = array(
                    ”user_id”       => $id,
                    ”username”      => $username,
                    ”is_logged_in”  => true,
                    ”user_roles”    => $this->model_roles->get_user_roles($id)
                );
                $this->session->set_userdata($data);
                redirect($this->session->userdata[”redirect”]);
            }else{
                $this->login(array(”errors”=>”Sorry, this account has been disabled.
                     Please contact your system administrator for more information.”));
            }
        }else{
            $this->login(array(”errors”=>”Sorry, unrecognized username or password.”));
        }
    }else{
        $this->login(array(”errors”=>validation_errors()));
    }
}
public function add($data=array()){
    $this->load->model(”model_roles”);
    $_data = array(
        ”title”         => ”New User”,
        ”roles”         => $this->model_roles->get_roles(),
        ”view”          => ”user/view_user_add”,
        ”action”        => ”add”,
        ”curr_roles”    => array(),
        ”new_user”      => array(
            ”last_name”     => ””,
            ”first_name”    => ””,
            ”middle_name”   => ””,
            ”email”         => ””,
            ”username”      => $this->input->post(”username”)
        )
    );
    $this->load->view(”view_template”,array_merge($data,$_data));
}
public function add_attempt(){
    $this->load->model(”model_users”);
    $this->load->model(”model_roles”);

    $this->load->library(”form_validation”);
    $this->form_validation->set_rules(”email”,”Email”,”trim|valid_email”);
    $this->form_validation->set_rules(”role”,”Role(s)”,”required”);
    $this->form_validation->set_rules(”username”,”Username”,”required|alpha_numeric|trim|
         is_unique[user.username]”);   //trim - remove white spaces
    $this->form_validation->set_rules(”password”,”Password”,”required|md5”);
    $this->form_validation->set_rules(”retype_password”,”Retype Password”,”required|md5”);

    $this->form_validation->set_message(”is_unique”,”That username is already taken.”);

    $new_user = array(
        ”last_name” => $this->input->post(”last_name”),
        ”first_name” => $this->input->post(”first_name”),
        ”middle_name” => $this->input->post(”middle_name”),
        ”username” => $this->input->post(”username”),
        ”email” => $this->input->post(”email”),
        ”password” => md5($this->input->post(”password”)),
        ”account_status” => ”enabled”
    );
    $roles = $this->input->post(”role”);
    if($this->form_validation->run()){  // if all the rules have been verified
        $key = md5(uniqid());

        // send an email
        if(strlen($this->input->post(”email”))){
            $this->load->library(”email”);
            $this->email->from(”admin@doh.gov.ph”,”Schistosomiasis GIS”);
            $this->email->to($this->input->post(”email”));
            $this->email->subject(”New Account at doh.gov.ph”);

            $message =  ”<p>Thank you for signing up!</p>”.
                ”<p><a href=’”.base_url(’user/confirm/’ . $key).”’> Click here</a> to
                     confirm your account</p>”;
            $this->email->message($message);
            $this->email->send();
        }
        //add to database
        $query = $this->model_users->add_user($new_user,$roles);
        if(isset($query[”id”])){
            redirect(”user/view/”. $query[”id”]);
        }
    }else{
        $this->load->view(”view_template”, array(
            ”new_user”=>$new_user,
            ”validation_errors” => validation_errors(),
            ”title” => ”New User”,
            ”view”  => ”user/view_user_add”,
            ”roles” => $this->model_roles->get_roles(),
            ”action”    => ”add”,
            ”curr_roles”=> $this->input->post(”role”)
        ));
    }
}
public function logout(){
    $this->session->sess_destroy();
    $this->login();
```

```
}

public function view_all(){
    $term = strlen($this->input->post("username"))? $this->input->post("username") : $this
        ->input->post("search-term");
    if(strlen($this->input->post('add'))){
        strlen($term)? $this->add() : redirect("user/add");
    }else{
        $this->load->model("model_users");
        $this->load->model("model_roles");
        if(strlen($this->input->post("update_form"))){
            foreach($this->input->post("selected_users") as $username){
                $this->model_users->change_status($username,$this->input->post("
                    update_action"));
            }
        }
        $status = $this->input->post("status");
        $status = count($status)==1? $status[0] : null;    //check if status should be
            filtered
        $role_input = $this->input->post("role");
        $roles = is_array($role_input)&&count($role_input)<4&&count($role_input)>0?
            $role_input : null;

        $users = $this->model_users->get_all($term, $status,$roles);

        foreach($users as $user){
            $user->roles = implode($this->model_roles->get_user_role_names($user->user_id)
                ,", ");
        }
        $data = array(
            "roles" => $this->model_roles->get_roles(),
            "view"  => "user/view_user_all",
            "users" => $users,
            "term"  => $term
        );
        $this->load->view("view_template",$data);

    }
}
public function view($id){
    if(isset($id)){
        $this->load->model("model_users");
        $this->load->model("model_roles");
        $curr_user = $this->model_users->get($id);
        if(count($curr_user)){
            $curr_user = $curr_user[0];
            $curr_user->roles = $this->model_roles->get_user_role_names($id);
            $data = array(
                "user"      =>  $curr_user,
                "title"     =>  $curr_user->username,
                "view"      =>  "user/view_user_id"
            );
            $this->load->view("view_template",$data);
        }else{
            redirect("user/view_all");
        }
    }else{
        redirect("user/view_all");
    }
}
public function edit($id=null){
    if(isset($id)){
        $this->load->model("model_users");
        $this->load->model("model_roles");
        $curr_user = $this->model_users->get($id);
        $curr_user = (array)$curr_user[0];
        $data = array(
            "new_user"      =>  $curr_user,
            "title"     =>  "Edit " . $curr_user["username"],
            "view"      =>  "user/view_user_add",
            "action"    => "edit",
            "curr_roles"=>$this->model_roles->get_user_roles($id),
            "roles"     => $this->model_roles->get_roles()
        );
        $this->load->view("view_template",$data);
    }else{
        redirect("user/add");
    }
}
public function change_password(){
    $this->load->library("form_validation");
    $this->form_validation->set_rules("old","Old Password","required|md5");
    $this->form_validation->set_rules("password","New Password","required|md5");
    $this->form_validation->set_rules("retype_password","Retyped New Password","required|
        md5");

    $this->load->model("model_users");
    $this->load->model("model_roles");
    $id = $this->input->post("user_id");
    $curr_user = $this->model_users->get($id);
    $curr_user = $curr_user[0];
    $curr_user->roles = $this->model_roles->get_user_role_names($id);

    if($this->form_validation->run()){
        $old_pass = $curr_user->password;
        if($old_pass==$this->input->post("old")){
            $this->model_users->update_user(array("user_id"=>$id,"password"=>$this->input
                ->post("password")));
            $this->load->view("view_template",array(
                    "user"      =>  $curr_user,
                    "title"     =>  "Password Changed",
                    "view"      =>  "user/view_user_id",
                    "success_msg"   => 3
                ));
            }else{
                $this->load->view("view_template",array(
                    "user"      =>  $curr_user,
                    "title"     =>  "Password Change Failed",
                    "view"      =>  "user/view_user_id",
                    "fail_msg"   => "Incorrect password."
                ));
            }
        }else{
            //unhandled error
            $this->load->view("view_template",array(
                "user"      =>  $curr_user,
                "title"     =>  "Password Change Failed",
                "view"      =>  validation_errors(),
                "fail_msg"   => "Incorrect password."
            ));
        }
    }

    public function disable($id){
        $this->load->model("model_users");
        $this->model_users->disable($id);
    }
    public function enable($id){
        $this->load->model("model_users");
        $this->model_users->enable($id);
    }
    public function username_check($username,$id){
        $this->load->model("model_users");
        if(!$this->model_users->unique_edit($id, $username)){
            $this->form_validation->set_message('username_check', 'The username "'. $username
                . '" is already being used by another account.');
            return false;
        }
        return true;
    }
    public function edit_attempt(){
        echo validation_errors();
        $this->load->model("model_users");

        $this->load->library("form_validation");
        $this->form_validation->set_rules("email","Email","trim|valid_email");
        $this->form_validation->set_rules("role","Role(s)","required");

        $user_id = $this->input->post("user_id");
        $this->form_validation->set_rules("username","Username","required|trim|alpha_numeric|
             callback_username_check[$user_id]");   //trim - remove white spaces

        $this->load->model("model_users");
        $this->load->model("model_roles");
        $new_user = array(
            "user_id" => $user_id,
            "last_name" => $this->input->post("last_name"),
            "first_name" => $this->input->post("first_name"),
            "middle_name" => $this->input->post("middle_name"),
            "username" => $this->input->post("username"),
            "email" => $this->input->post("email"),
            "account_status" => $this->input->post("account_status")
        );
        $roles = $this->input->post("role");
        if($this->form_validation->run()){  // if all the rules have been verified
            if($this->model_users->update($new_user,$roles)){
                redirect("user/view/$user_id");
            }else{
                echo "db error";
            }
        }else{
            $this->load->view("view_template", array(
                "new_user"=>$new_user,
                "validation_errors" => validation_errors(),
                "title"         => "Edit User",
                "view"          => "user/view_user_add",
                "roles"         => $this->model_roles->get_roles(),
                "action"        => "edit",
                "curr_roles"    => $this->input->post("role")
            ));
        }
    }
    public function search($arg1='',$arg2='',$offset=0){
        $term = $this->input->post("username");

        $this->load->library("pagination");
        $config = array(
            "page_base" => base_url("user/search")
        );
    }
    public function reset_password(){
    }
}
```

# XII. Acknowledgement

I would like to thank the Lord God Almighty for giving me the strength and the wisdom to finish this special problem.

I would like to thank my family for being supportive; especially my mom, for always praying for me and trying to help even though she never understood what I was doing, and my sister, for being my shock absorber when I'm really stressed and moody.

I would like to thank all of my friends, especially Abie, Jason, Robby, Hainah, Zach, Loann, and Jennifer, for being there when I needed help or when I needed a distraction.

I would like to thank the people at the Department of Clinical Epidemiology for coming up with this project and for helping me along the way.

I would like to thank the panelists for being thorough and for making me smarter.

I would like to thank Sir Solano for taking me as his advisee. Thank you, sir, for being so patient and supportive, and for guiding me all along.

I thank you, all, for your support and for making this SP possible. May God bless you more so that much more may be blessed through you.

Tapos na!