

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

Disease Outbreak Detection using Time Series Analysis

A Special Problem in partial fulfillment  
of the requirements for the degree of  
Bachelor of Science in Computer Science

Submitted by:

Richard John M. Buendia  
April 2010

ACCEPTANCE SHEET

The Special Problem entitled “Disease Outbreak Detection using Time Series Analysis” is prepared and submitted by Richard John M. Buendia 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.S.**  
Adviser

EXAMINERS	APPROVED	DISAPPROVED
1. Gregorio B. Baes, Ph.D (Candidate)	_____	_____
2. Avegail D. Carpio, MS	_____	_____
3. Richard Bryann L.Chua, MS	_____	_____
4. Aldrich Colin K. Co, MS (Candidate)	_____	_____
5. Ma. Sheila A. Magboo, MS	_____	_____
6. Vincent Peter C. Magboo, MD, MS	_____	_____
7. Bernie B. Terrado , MS (Candidate)	_____	_____

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

\_\_\_\_\_  
**Geoffrey A. Solano, M.S.**  
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

\_\_\_\_\_  
**Reynaldo H. Imperial, Ph.D.**  
Dean

**ABSTRACT**

Disease Outbreak Detection System is an online system that helps epidemiologist analyze the behavior of a certain disease outbreak by providing them prediction values for a specific time interval. The system is able to perform such feature with the aid of R software which performs computations of Time Series analysis using Autoregressive Moving Averages (ARMA) Model to generate values based on the present condition of the outbreak. These generated values will serve as basis to know how the outbreak will turn out thus giving the epidemiologist sufficient time to respond to major public health threats and formulate preventive measures to control and solve the outbreak.

The system has four main users. First is the National Epidemiological Center (NEC) medical officer. He can perform several computations such as ARIMA forecast and Partial AutoCorrelation Function in order to produce accurate prediction values. He can also assess the present condition of a certain outbreak by using several status indicators such as case mapping and graphs that would help identify the behaviour of the outbreak in a faster and easier way. Second is the Local Field Health officer. He is the one responsible in managing the case records stored in the system's database which in turn will be the basis in the assessment that will be conducted by the NEC medical officer. System Administrator is the next user. His duty is to manage the user accounts of each registered users and also, he is responsible in managing the contents of the site. Lastly, the guest or the anonymous user. He is able to view the published health news about disease outbreaks.

**KEYWORDS:** Disease Outbreak Detection, Time Series Analysis, ARMA, NEC

## TABLE OF CONTENTS

<b>I. INTRODUCTION.....</b>	<b>5</b>
A. BACKGROUND.....	5
B. STATEMENT OF THE PROBLEM.....	6
C. OBJECTIVE.....	7
D. SIGNIFICANCE OF THE STUDY.....	8
E. SCOPE AND LIMITATIONS.....	9
F. ASSUMPTIONS.....	9
<b>II. REVIEW OF RELATED LITERATURE.....</b>	<b>10</b>
<b>III. THEORETICAL FRAMEWORK.....</b>	<b>14</b>
DISEASE OUTBREAK DETECTION SYSTEM.....	14
AUTOREGRESSIVE MOVING AVERAGES (ARMA).....	14
TIME SERIES ANALYSIS.....	15
R SOFTWARE.....	15
CASE FATALITY RATE (CFR).....	15
AKAIKE'S INFORMATION CRITERION (AIC).....	15
.....	16
COMMUNICABLE/INFECTIOUS DISEASE.....	16
TUBERCULOSIS.....	16
MALARIA.....	16
SCHISTOSOMIASIS.....	16
LEPROSY.....	17
NATIONAL EPIDEMIC SENTINEL SURVEILLANCE SYSTEM (NESSS).....	17
NATIONAL EPIDEMIOLOGICAL CENTER (NEC).....	17
FIELD HEALTH SERVICE INFORMATION SYSTEM (FHSIS).....	18
FPDF.....	18
PRADO.....	18
DATABASE MANAGEMENT SYSTEM (DBMS).....	18
INFORMATION SYSTEM (IS).....	19
<b>IV. DESIGN AND IMPLEMENTATIONS.....</b>	<b>20</b>
CONTEXT DIAGRAM.....	21
TOP-LEVEL DATA FLOW DIAGRAM.....	23
ENTITY- RELATIONSHIP DIAGRAM.....	31
SYSTEM ENTITIES.....	32
DATA DICTIONARY.....	33
TECHNICAL ARCHITECTURE.....	34
<b>V. RESULTS.....</b>	<b>36</b>
<b>VI. DISCUSSION.....</b>	<b>51</b>
<b>VII. CONCLUSION.....</b>	<b>53</b>
<b>VIII. RECOMMENDATION.....</b>	<b>54</b>
<b>IX. BIBLIOGRAPHY.....</b>	<b>55</b>
<b>XI. ACKNOWLEDGEMENT.....</b>	<b>103</b>

## **I. INTRODUCTION**

### **A. Background**

The Philippines obtains its health data from two major sources: surveys and censuses and from administrative records of health and health related agencies.

Health Surveys and Census are limited to the following:

The National Demographic and Health Survey (NDHS) is undertaken by the National Statistics Office (NSO) once every 5 years to provide national and regional estimates of levels and trends of fertility as well as examines the differentials and determinants of fertility. It also yields information on family planning, childhood and adult mortality, maternal and child health, and knowledge and attitudes related to HIV/AIDS and other sexually transmitted infections.

In addition to the surveys and censuses, the Department of Health (DOH) and other government agencies separately collect and maintain data that can be used for monitoring and assessing the health status of the Philippines. The administrative data serves as supplementary data to augment the extensive data required for planning, policy formulation and program interventions on health. With the large contribution of the private sector, monitoring and evaluation of their services are best obtained through administrative data.

The DOH continues to provide / generate the bulk of data for the health sector which are obtained mostly from administrative reports regularly furnished by public hospitals, rural health units (RHUs) and other local government health units.

A major source of data for the DOH is the Field Health Service Information System (FHSIS). It provides information on the different public health programs such as: maternal and child health; nutrition; family planning; Expanded Program on Immunization; Mental health; Communicable Disease Prevention and Control (Tuberculosis, Malaria, Schistosomiasis, Leprosy); Environmental Health; Vital Statistics

(natality, mortality, population); and notifiable disease reporting system. Data comes from local field health personnel through the regional and provincial health offices and consolidated at the central office. These are presented by province, city and region in the Annual FHSIS Report. [1]

A team of trained surveillance personnel conducts daily hospital rounds and weekly analysis of data. Everyday, they will proceed to the patient information desk where the 24 hours logbook is located. Names of patients admitted within the last 24 hours and whose diagnosis is included among the diseases being monitored are copied. Patients are physically examined to see if they fit the case definition. Analysis of surveillance data and writing of morbidity reports are done weekly.

Surveillance data from each Regional Epidemiology and Surveillance Unit are sent to NEC by mail (diskettes) or e-mail (for regions with access to Internet) monthly for collation and merging. NEC collates all surveillance data from the regions and produces an annual report. [2]

Disease outbreak detecting system is an online system that aims to generate reports on the status of a disease outbreak based on the reported cases from hospitals and other health units. It also provides additional tools that can help different health sectors in formulating actions to prevent and control disease outbreaks.

## **B. Statement of the Problem**

Disease outbreak detection is a very delicate procedure to be considered and it needs to be accurate and precise because the output of such method will be a great help in preventing further problem. Each scenario can produce different outcome. Many factors are to be considered to acquire the correct information. Several algorithms emerge claiming to be the best in supplying solutions to a number of situations. This leads to the problem of choosing the appropriate technique in computing the rate of disease outbreak. If all techniques will be applied in the actual environment, it will be costly and expected to consume a lot of valuable time putting public health at risk.

Though several systems have been developed having the same purpose, the system has not been able to detect disease outbreak locally considering local conditions. And if there's any system that's already been used, it offers limited features in evaluating disease outbreaks.

Also, according to an epidemiologist, there are still deviations between what is indicated in FHSIS annual reports and what is actually happening at the health centers with respect to data collection and recording system. [3] The deviation may be explained by some missing or delayed records while consolidating all of them in the NEC or errors due to duplication of records.

### **C. Objective**

To create a disease outbreak detection system that:

- 1) allow the National Epidemiological Center medical officer to:
  - a) approximate the number of infected cases for a given time interval by Time Series Analysis using the Autoregressive Moving Averages Model.
  - b) view outbreak status through annual reports and different generated graphs
  - c) view case-mapping tool with visual representation of risk levels.
  - d) manage disease records allowing him/her to:
    - i. Add new disease record
    - ii. Delete an existing disease record
    - iii. Update an existing disease record
  - e) publish health news
  
- 2) allow the local field health personnel to:
  - a) manage reported cases allowing him/her to :
    - i. Add a new reported case
    - ii. Delete an existing case
    - iii. Update an existing case
  
  - b) perform a database search on the reported cases

- 3) allow the system administrator to:
  - a) create and modify the accounts of registered users
  - b) manage the contents of the site specifically health news
  
- 4) allow guests to :
  - a) view posted health news

#### **D. Significance of the Study**

The disease outbreak detection system enables the user to generate an approximation on how many will be infected in a given time utilizing the set of data gathered from different time intervals. These will be beneficial to the health sector for the reason that they will have an idea on what will be the status of the outbreak for a given time interval therefore providing them ideas on what degree the outbreak will turn out. As a result, it will provide our health sectors enough time to plan an action on preventing or controlling the situation.

For disease outbreak detection, the public health community has historically relied on the watchful eyes of doctors, who have reported individual cases or clusters of cases of particular diseases to the authorities. But these days, the availability of electronic health-care data should facilitate more automated and earlier outbreak detection and intervention. Besides diagnoses of known diseases, other indicators--such as primary complaints of patients coming to the emergency room or calling a nurse hotline--are being collected in electronic formats and could be analyzed if suitable methods existed [4]. These solve one of the causes of inconsistency among data gathered. It minimizes the load on reporting cases therefore it will be easy to submit data which result to a lesser deviation.

Accurate and timely detection of infectious disease outbreaks provides valuable information which can enable public health officials to respond to major public health threats in a timely fashion. However, disease outbreaks are often not directly observable. For surveillance systems used to detect outbreaks, noises caused by routine behavioural patterns and special events, can further complicate the detection task [5]. For this system, only the set



of data gathered in a given time interval are analyzed, excluding all other factors contributing to disease outbreak detection for the purpose of avoiding this situation.

### **E. Scope and Limitations**

1. Only the set of data reflecting the number of infected case in a specific time interval will be used in the computation.
2. The system is limited and applied only to cities within the National Capital Region.
3. Also, the graphs that will be generated and the data that is to be used for the computation will be limited to the data recorded in the database.
4. The system will focus only on communicable diseases specifically Tuberculosis, Leprosy, Malaria and Schistosomiasis.
5. User accounts are created by signing up the registration form and with the approval of the system administrator.
6. The format of each output will depend on the functions and will follow the standards of the reports from FHSIS.

### **F. Assumptions**

1. It is assumed that all the inputs provided by the users are all valid and error-free.
2. The users are expected to have knowledge of how the said techniques in generating the approximated number of infected cases.
3. The users have an internet connection and other required system components installed.

## II. REVIEW OF RELATED LITERATURE

The Disease Outbreak Detection System uses the Autoregressive Moving Averages model to generate prediction values utilizing past records as basis for the computation. This system is just one of the many applications where ARMA can be used. The following are examples of systems and studies implementing the model.

SPSS13.0 software was used to construct the ARIMA model based on the monthly malaria incidence of Huaiyuan and Tongbai countries in Huaihe River Valley, from Jan. 1998 to Dec. 2005. The constructed model was then applied to predict the monthly malaria incidence in 2006 and the incidence from ARIMA model was compared with the actual incidence, so as to evaluate the model's validity. Malaria incidence of 2007 was predicted by ARIMA model based on malaria incidence from 1998 to 2006. As a result, ARIMA (1,0,0) (0,1,1) model exactly fitted the incidence of the previous monthly incidence from Jan. 1998 to Dec. 2005, and the predicted monthly incidence in 2006 by the model was consistent with the actual incidence. The model of ARIMA seems to be an appropriate model to fit exactly the changes of malaria incidence and to predict the future incidence trend, with a high prediction precision of short term time series. [6]

A univariate time-series analysis method has been used to model and forecast the monthly number of dengue haemorrhagic fever (DHF) cases in southern Thailand. They developed autoregressive integrated moving average (ARIMA) models on the data collected between 1994–2005 and then validated the models using the data collected between January–August 2006. The results showed that the regressive forecast curves were consistent with the pattern of actual values. The ARIMA (1,0,1) model fitting was adequate for the data with the Q-statistic ( $Q=4.446$ ). [7]

To identify the stochastic autoregressive integrated moving average (ARIMA) model for short term forecasting of hepatitis C virus (HCV) seropositivity among volunteer blood donors in Karachi, Pakistan, ninety-six months (1998-2005) data on HCV seropositive cases among male volunteer blood donors tested at four major blood banks in Karachi, Pakistan

were subjected to ARIMA modelling. Subsequently, a fitted ARIMA model was used to forecast HCV seropositive donors for 91-96 mos to contrast with observed series of the same months. To assess the forecast accuracy, the mean absolute error rate (%) between the observed and redicted HCV seroprevalence was calculated. Finally, a fitted ARIMA model was used for short-term forecasts beyond the observed series. This short-term forecast beyond the observed series adequately captured the pattern in the data and showed increasing tendency of HCV seropositivity over the forecast interval. [8]

In this study, a mathematically causal time domain technique, autoregressive moving average (ARMA) analysis, was used to parameterize the relations of respiration and arterial blood pressure to heart rate in eight humans before and during total cardiac autonomic blockade. Impulse-response curves thus generated showed the relation of respiration to heart rate to be characterized by an immediate increase in heart rate of  $9.1 \pm 1.8$  beats.min<sup>-1</sup>.l<sup>-1</sup>, followed by a transient mild decrease in heart rate to  $-1.2 \pm 0.5$  beats.min<sup>-1</sup>.l<sup>-1</sup> below baseline. ARMA analysis may be useful as a time domain representation of autonomic heart rate control for cardiovascular modeling. [9]

The main objective of the study is to apply autoregressive integrated moving average (ARIMA) models to make real-time predictions on the number of beds occupied in Tan Tock Seng Hospital, during the recent SARS outbreak.

This is a retrospective study design. Hospital admission and occupancy data for isolation beds was collected from Tan Tock Seng hospital for the period 14th March 2003 to 31st May 2003. The main outcome measure was daily number of isolation beds occupied by SARS patients. Among the covariates considered were daily number of people screened, daily number of people admitted (including observation, suspect and probable cases) and days from the most recent significant event discovery.

They found that the ARIMA (1,0,3) model was able to describe and predict the number of beds occupied during the SARS outbreak well Furthermore, the model also provided three-day forecasts of the number of beds required. Total number of admissions and probable cases admitted on the previous day were also found to be independent prognostic factors of bed occupancy.

ARIMA models provide useful tools for administrators and clinicians in planning for real-time bed capacity during an outbreak of an infectious disease such as SARS. The model could well be used in planning for bed-capacity during outbreaks of other infectious diseases as well. [10]

The oil industry has used decline curve analysis with limited success in estimating crude oil reserves and in predicting future behavior of oil and gas wells. The paper explores the possibility of using the Autoregressive Integrated Moving Average (ARIMA) technique in forecasting and estimating crude oil reserves. The authors compare this approach with the traditional decline method using real oil production data from twelve (12) oil wells in South Louisiana. The Box and Jenkins methodology is used to develop forecast functions for the twelve wells under study. These forecast functions are used to predict future oil productions. The forecast values generated are then used to determine the remaining crude oil reserves for each well. The accuracy of the forecasts relative to the actual values for both ARIMA and decline curve methods is determined by various statistical error analyses. The conditions, under which one method gives better results than the other, are fully investigated. In almost all the cases considered, the ARIMA method is found to perform better than the decline curve method. [11]

The application of seasonal time series models to the single-interval traffic flow forecasting problem for urban freeways is addressed. Seasonal time series approaches have not been used in previous forecasting research. However, time series of traffic flow data are characterized by definite periodic cycles. Seasonal autoregressive integrated moving average (ARIMA) and Winters exponential smoothing models were developed and tested on data sets belonging to two sites: Telegraph Road and the Woodrow Wilson Bridge on the inner and outer loops of the Capital Beltway in northern Virginia. Data were 15-min flow rates and were the same as used in prior forecasting research by B. Smith. Best-fit Winters exponential smoothing models were also developed for each site. The single-step forecasting results indicate that seasonal ARIMA models outperform the nearest-neighbour, neural network, and historical average models as reported by Smith. [12]

Mathematical modelling of medical resource requirements during military operations requires analyzing the underlying relationships between Disease and Non-Battle Injury (DNBI) rates and Wounded-In-Action (WIA) rates. DNBI and WIA data were extracted from Marine Corps unit diaries for a 150-day period of the Korean War and from a 90 day period of the Okinawa operation during World War 2 and a 123 day period of the war in Vietnam. The time series data were set up in a bivariate autoregressive integrated moving average analysis. All the univariate models are best represented with a moving average term. This study has shown that WIA rates can be a useful predictor of DNBI rates when using a bivariate ARIMA model. High levels of WIA incidence will affect DNBI rates the immediate day and the following day. These results were consistent throughout the three military conflicts examined and should prove indicative for future military campaigns. [13]

Monthly test-day milk yields of 1200 dairy Sarda ewes were analyzed by time-series methods. Autocorrelation functions were calculated for lactations within parity classes and altitude of location of flocks. Spectral analysis of the successions of data was developed by Fourier transformation, and different Box-Jenkins autoregressive integrated moving average models were fitted. The forecasting power of autoregressive integrated moving average models was tested by predicting total milk production for a standardized lactation length of 225 days from only a few test day records. Results indicated a greater forecasting capacity in comparison with standard methods and suggested further development of time-series analysis for studying lactation curves with more sophisticated methods, such as wavelet decomposition and neural network models. [14]

In the study, time series model of ARIMA is used to make short-term forecasting of property crime for one city of China. With the given data of property crime for 50 weeks, an ARIMA model is determined and the crime amount of 1 week ahead is predicted. The model's fitting and forecasting results are compared with exponential smoothing. It is shown that the ARIMA model has higher fitting and forecasting accuracy than exponential smoothing. This work will be helpful for the local police stations and municipal governments in decision making and crime suppression. [15]

### **III. THEORETICAL FRAMEWORK**

#### **DISEASE OUTBREAK DETECTION SYSTEM**

An online system that enables user to generate an approximated number of infected cases provided that it was given a set of data from reported and existing cases. The output will give the public health workers an idea on how the outbreak will be in the given amount of time. As a result, the public health workers will have a chance to integrate procedures in controlling and preventing the outbreak.. Results can be easily interpreted because of the graphs that will be generated using the information the users have inputted.

The system also records and stores information regarding reported cases indicating their personal information, location and condition. This will bring several benefits to the health workers because of faster retrieval of data that will be needed, more organize and clean reports because of easier updating and a faster response to the situation.

#### **AUTOREGRESSIVE MOVING AVERAGES (ARMA)**

ARMA models are mathematical models of the persistence, or autocorrelation, in a time series. ARMA models are widely used in hydrology, dendrochronology, and many other fields. There are several possible reasons for fitting ARMA models to data. Modeling can contribute to understanding the physical system by revealing something about the physical process that builds persistence into the series. ARMA models can also be used to *predict* behavior of a time series from past values alone. Such a prediction can be used as a baseline to evaluate possible importance of other variables to the system. ARMA models are widely used for prediction of economic and industrial time series. [16]

## **TIME SERIES ANALYSIS**

In statistics, signal processing, and many other fields, a time series is a sequence of data points, measured typically at successive times, spaced at (often uniform) time intervals. Time series analysis comprises methods that attempt to understand such time series, often either to understand the underlying context of the data points (Where did they come from? What generated them?), or to make forecasts (predictions). Time series forecasting is the use of a model to forecast future events based on known past events: to forecast future data points before they are measured. A standard example in econometrics is the opening price of a share of stock based on its past performance. [17]

## **R SOFTWARE**

R is a freely available language and environment for statistical computing and graphics providing a wide variety of statistical and graphical techniques. It is very similar to a commercial statistics package called S-Plus, which is widely used. [18]

## **CASE FATALITY RATE (CFR)**

epidemiological term for death rate of a disease.[19]

Formula:

$$CFR = ( \text{Deaths} / \text{Case Count} ) * 100$$

## **AKAIKE'S INFORMATION CRITERION (AIC)**

is a measure of the goodness of fit of an estimated statistical model. Given a data set, several competing models may be ranked according to their AIC, with the one having the lowest AIC being the best.

In the general case, the AIC is

$$AIC = 2k - 2\ln(L)$$

where k is the number of parameters in the statistical model, and L is the maximized value of the likelihood function for the estimated model. [20]

## **COMMUNICABLE/INFECTIOUS DISEASE**

An **infectious disease** is a clinically evident disease resulting from the presence of pathogenic microbial agents, including pathogenic viruses, pathogenic bacteria, fungi, protozoa, multicellular parasites and aberrant proteins known as prions. These pathogens are able to cause disease in animals and/or plants. Infectious pathologies are usually qualified as **contagious diseases** (also called communicable diseases) due to their potential of transmission from one person or species to another. Transmission of an infectious disease may occur through one or more of diverse pathways including physical contact with infected individuals. These infecting agents may also be transmitted through liquids, food, body fluids, contaminated objects, airborne inhalation, or through vector-borne spread. [21]

## **TUBERCULOSIS**

is a common and often deadly infectious disease caused by mycobacteria. In humans, *Mycobacterium tuberculosis* is the primary causative. Tuberculosis usually attacks the lungs (as pulmonary TB) but can also affect the central nervous system, the lymphatic system, the circulatory system, the genitourinary system, the gastrointestinal system, bones, joints, and even the skin. [22]

## **MALARIA**

Malaria is a vector-borne infectious disease caused by a eukaryotic protist of the genus *Plasmodium*. It is widespread in tropical and subtropical regions, including parts of the Americas, Asia, and Africa. Each year, there are approximately 350–500 million cases of malaria, killing between one and three million people, the majority of whom are young children in Sub-Saharan Africa. Malaria is commonly associated with poverty, but is also a cause of poverty and a major hindrance to economic development. . [23]

## **SCHISTOSOMIASIS**

Schistosomiasis is a parasitic disease caused by several species of fluke of the genus *Schistosoma*. Although it has a low mortality rate, schistosomiasis often is a chronic illness that can damage internal organs and, in children, impair growth and cognitive



development. The urinary form of schistosomiasis is associated with increased risks for bladder cancer in adults. Schistosomiasis is the second most socioeconomically devastating parasitic disease after malaria.

This disease is most commonly found in Asia, Africa, and South America, especially in areas where the water contains numerous freshwater snails, which may carry the parasite.

The disease affects many people in developing countries, particularly children who may acquire the disease by swimming or playing in infected water. [24]

## **LEPROSY**

Leprosy is a chronic disease caused by the bacteria *Mycobacterium leprae* and *Mycobacterium lepromatosis*. Leprosy is primarily a granulomatous disease of the peripheral nerves and mucosa of the upper respiratory tract; skin lesions are the primary external symptom. Left untreated, leprosy can be progressive, causing permanent damage to the skin, nerves, limbs and eyes. Leprosy does not directly cause body parts to fall off on their own accord; instead they become disfigured or auto amputated as a result of disease symptoms. [25]

## **NATIONAL EPIDEMIC SENTINEL SURVEILLANCE SYSTEM (NESSS)**

is a network of hospital sentinel sites nationwide. These sentinel sites are linked to the Regional Epidemiology and Surveillance Units (RESUs). The sentinel sites are hospitals in the regions and are selected according to the following criteria:

1. The hospital should have at least 250 bed capacity (a hospital with lesser bed capacity can be a sentinel site if it is the only hospital in the province)
2. The hospital should have a laboratory capability to do malarial smears; blood and stool cultures; hepatitis serology
3. The hospital authorities should be willing to participate
4. Available personnel willing and interested to do surveillance; and
5. Adequate means of communication between sentinel site and RESU should exist. [2]

## **NATIONAL EPIDEMIOLOGICAL CENTER (NEC)**

NEC tasks is to develop and evaluate surveillance systems and other health information systems: collect, analyze and disseminate reliable and timely information on the health status; investigate disease outbreak and other threats to public health, network of public health laboratories in support of epidemiological and surveillance activities.[26]

### **FIELD HEALTH SERVICE INFORMATION SYSTEM (FHSIS)**

FHSIS serves as a major source of data for all its different public health programs to monitor activities in each of these programs on a routine basis (weekly, monthly, quarterly, or annually).[27]

### **FPDF**

FPDF is a PHP class which allows generating PDF files with pure PHP that is to say without using the PDFlib library. F from FPDF stands for Free: you may use it for any kind of usage and modify it to suit your needs. [28]

### **PRADO**

PRADO is a component-based and event-driven programming framework for developing Web applications in PHP 5. PRADO stands for **PHP Rapid Application Development Object-oriented**. [29]

### **DATABASE MANAGEMENT SYSTEM (DBMS)**

A set of computer programs that controls the creation, maintenance, and the use of the database of an organization and its end users. It allows organizations to place control of organization wide database development in the hands of Database Administrators (DBAs) and other specialist. DBMSes may use any of a variety of database models, such as the network model or relational model. In large systems, a DBMS allows users and other software to store and retrieve data in a structured way. It helps you specify the logical organization for a database and access and use the information within a database. It provides facilities for controlling data access, enforcing data integrity, managing concurrency controlled restoring database [30].

## **INFORMATION SYSTEM (IS)**

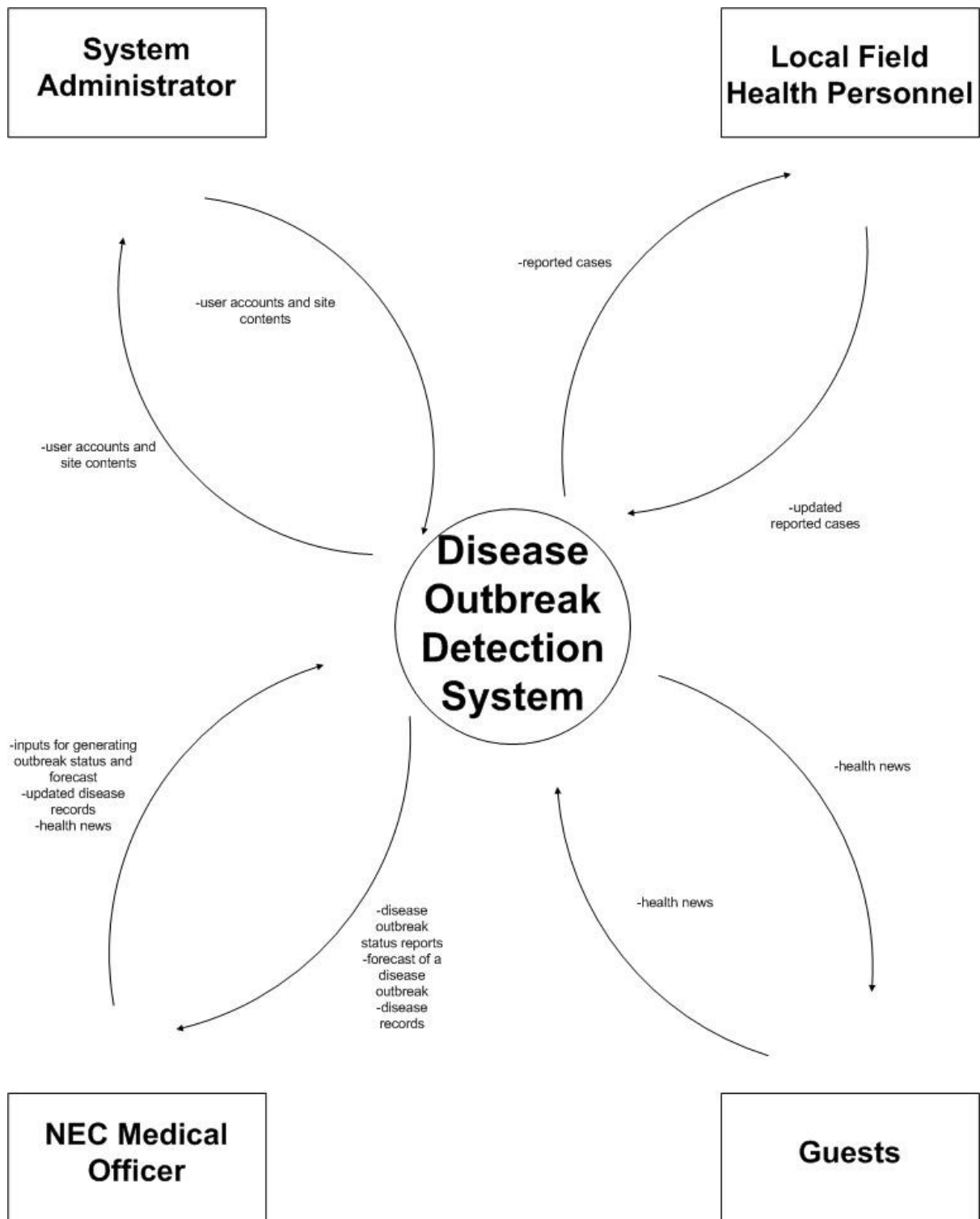
It refers to a system of people, data records and activities that process the data and information in an organization, and it includes the organization's manual and automated processes. In a narrow sense, the term *information system* (or **computer-based information system**) refers to the specific application software that is used to store data records in a computer system and automates some of the information-processing activities of the organization. Computer-based information systems are in the field of information technology. The discipline of business process modeling describes the business processes supported by information systems [31].

#### **IV. DESIGN AND IMPLEMENTATIONS**

The context diagram of the system is shown in Figure 1. Here, the overview of the interactions of all the users with the system is shown. The role of the NEC medical officer is to perform the computations to generate an approximation and provide status reports that reflect the condition of the outbreak given particular inputs and maintain the disease records. While the local field health personnel only have the ability to manage the patient's record that is stored in the system. Only they have the access to these records. Only the NEC medical officer has the capability to post health news and updates regarding disease outbreak. The role of the System Administrator is to manage the accounts of the users and the contents of the site. Viewing the summary reports about the status of the outbreak whether it is for a particular city or reports in the form of graphs is accessible only to the NEC medical officer.

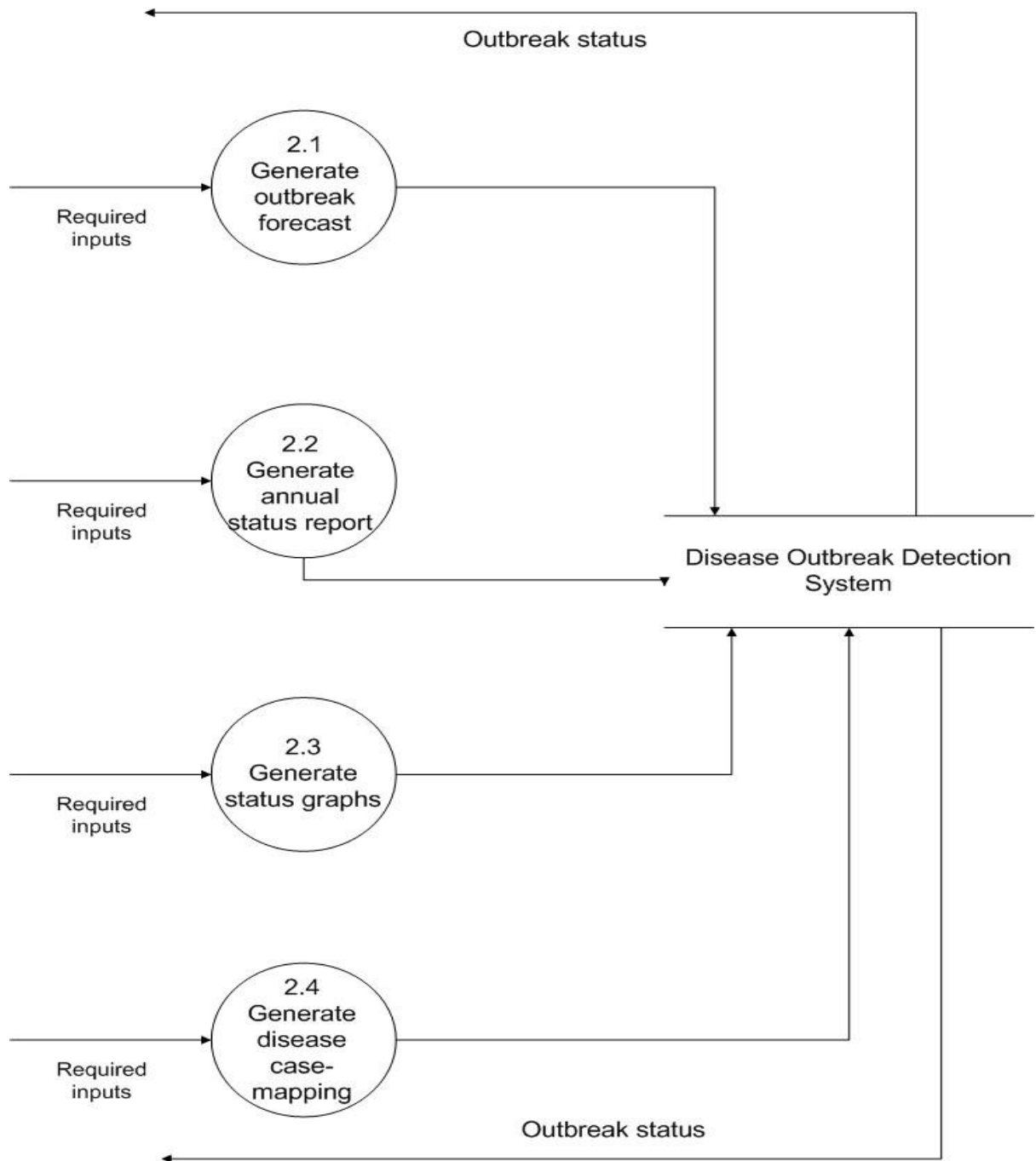
The Top-Level Data Flow Diagram is shown in Figure 2. Here, several processes are shown for a particular user. Process 1: Login is for all the users except the guests. This is done to access the functions that are restricted to be use only by registered users. Processes such as the Update and Search Patients are only for the local field health personnel. These process aims to manage the records that are stored in the system. Viewing summary reports in the form of different graphs that reflects the status of an outbreak is for the NEC medical officer only. The process of Managing User Accounts and Site Contents are restricted only for the use of the System Administrator.

## **CONTEXT DIAGRAM**



**FIGURE 1: CONTEXT DIAGRAM OF DISEASE OUTBREAK DETECTION SYSTEM**

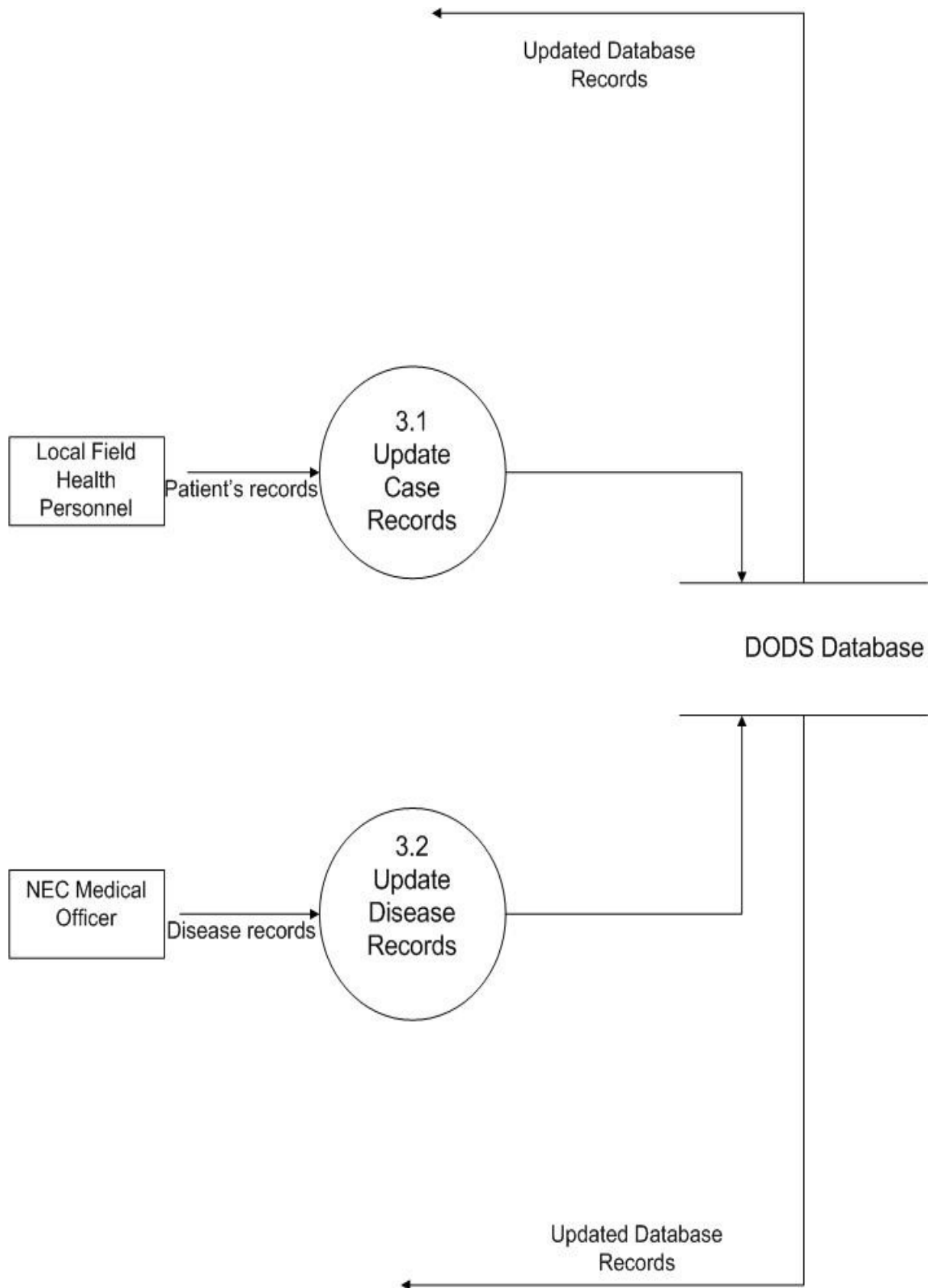




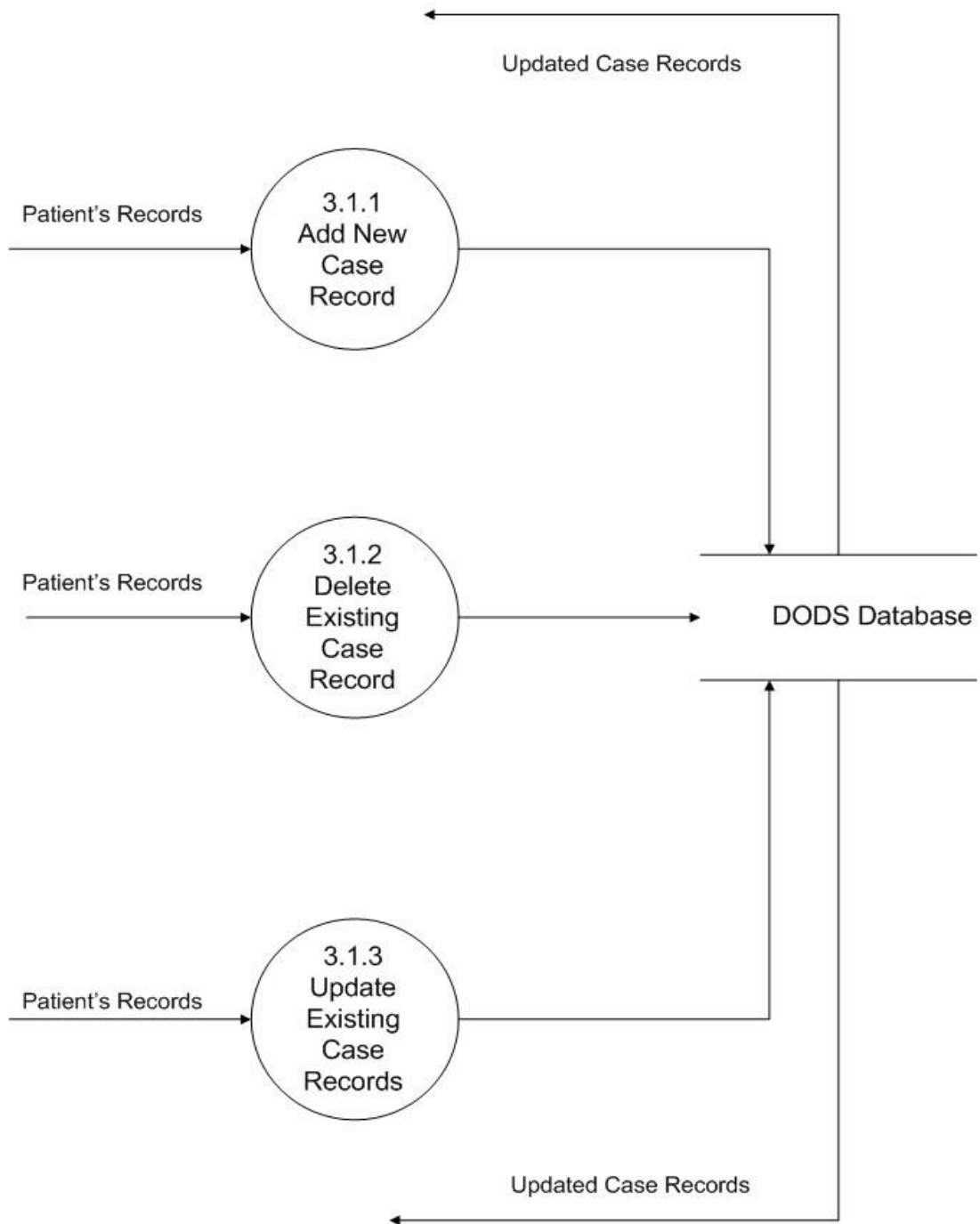
**FIGURE 3. SUBEXPLOSION OF PROCESS 2: GENERATE OUTBREAK STATUS**

Figure 3 shows the sub explosion of Process 2 Generate Outbreak Status. Here, the functionalities that would help epidemiologist in assessing the status of an outbreak can be done. Note that only the NEC medical officer can access these functionalities.

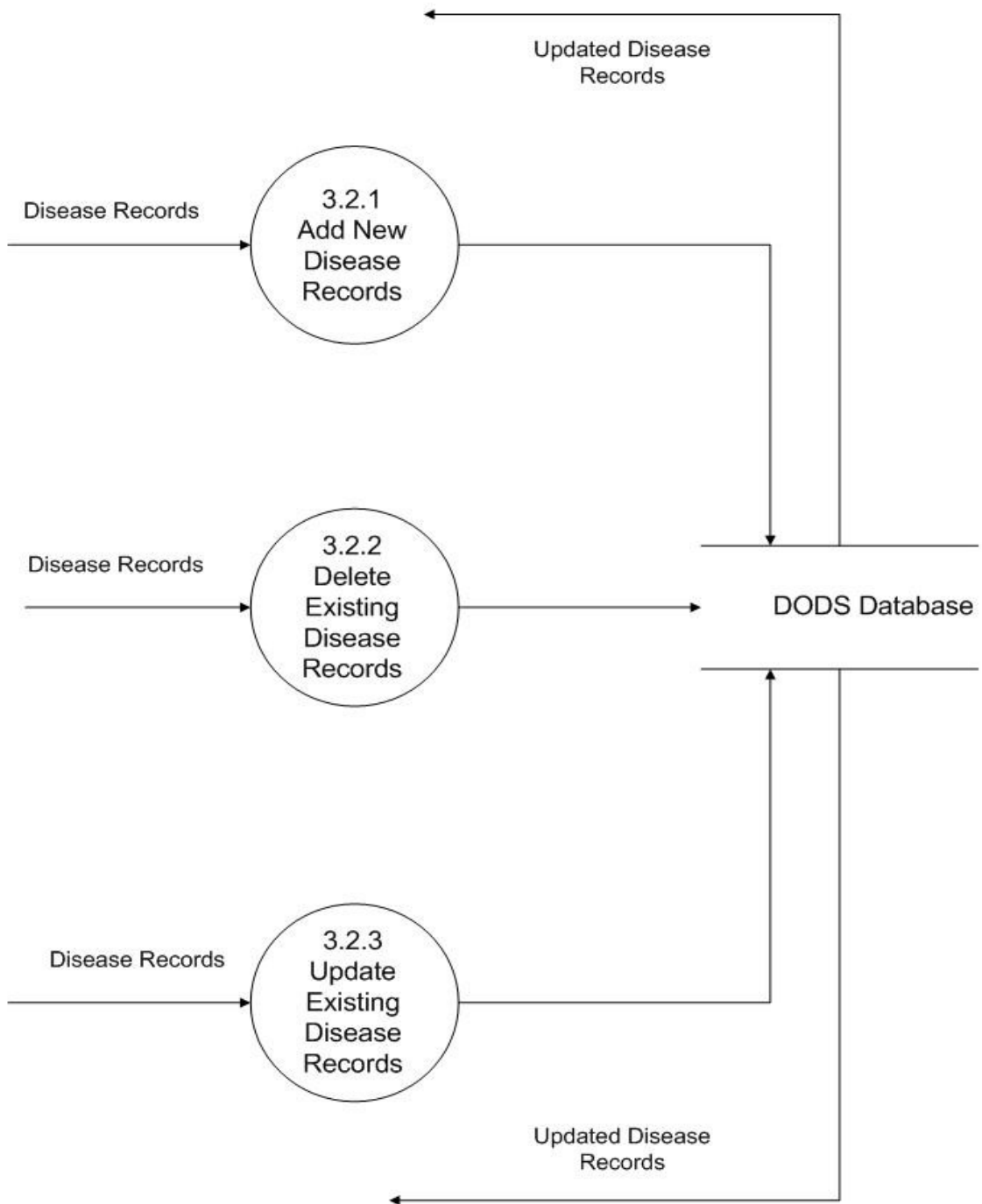




**FIGURE 4. SUBEXPLOSION OF PROCESS 3: MANAGE SYSTEM RECORDS**



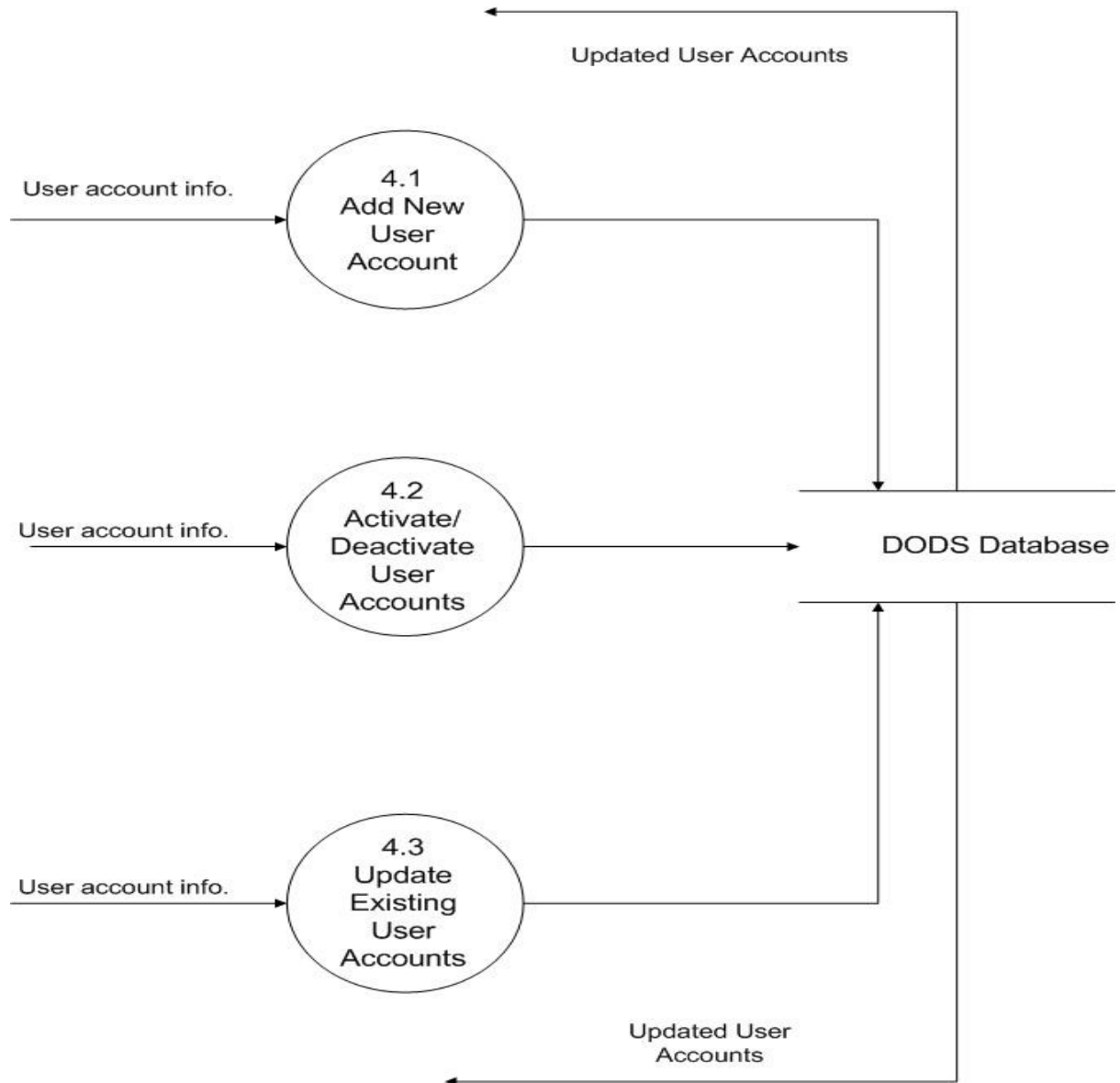
**FIGURE 5. SUBEXPLOSION OF PROCESS 3.1 : UPDATE CASE RECORDS**



**FIGURE 6. SUBEXPLOSION OF PROCESS 3.2 : UPDATE DISEASE RECORDS**

Figure 4 shows the sub explosion of Process 3 Manage System Records. These two main sub processes namely Update Case Records and Update Disease records where their sub processes are shown in Figure 5 and 6 respectively. The sub processes of Update case Records are namely Add New Record, Delete Existing Case Records and Update Existing Case Records. And for Update Disease Records, its sub processes are namely Add New Disease Record, Delete Existing Disease Records and Update Existing Disease Records. These processes give the users the ability to manage the patient's records as well as the disease records.

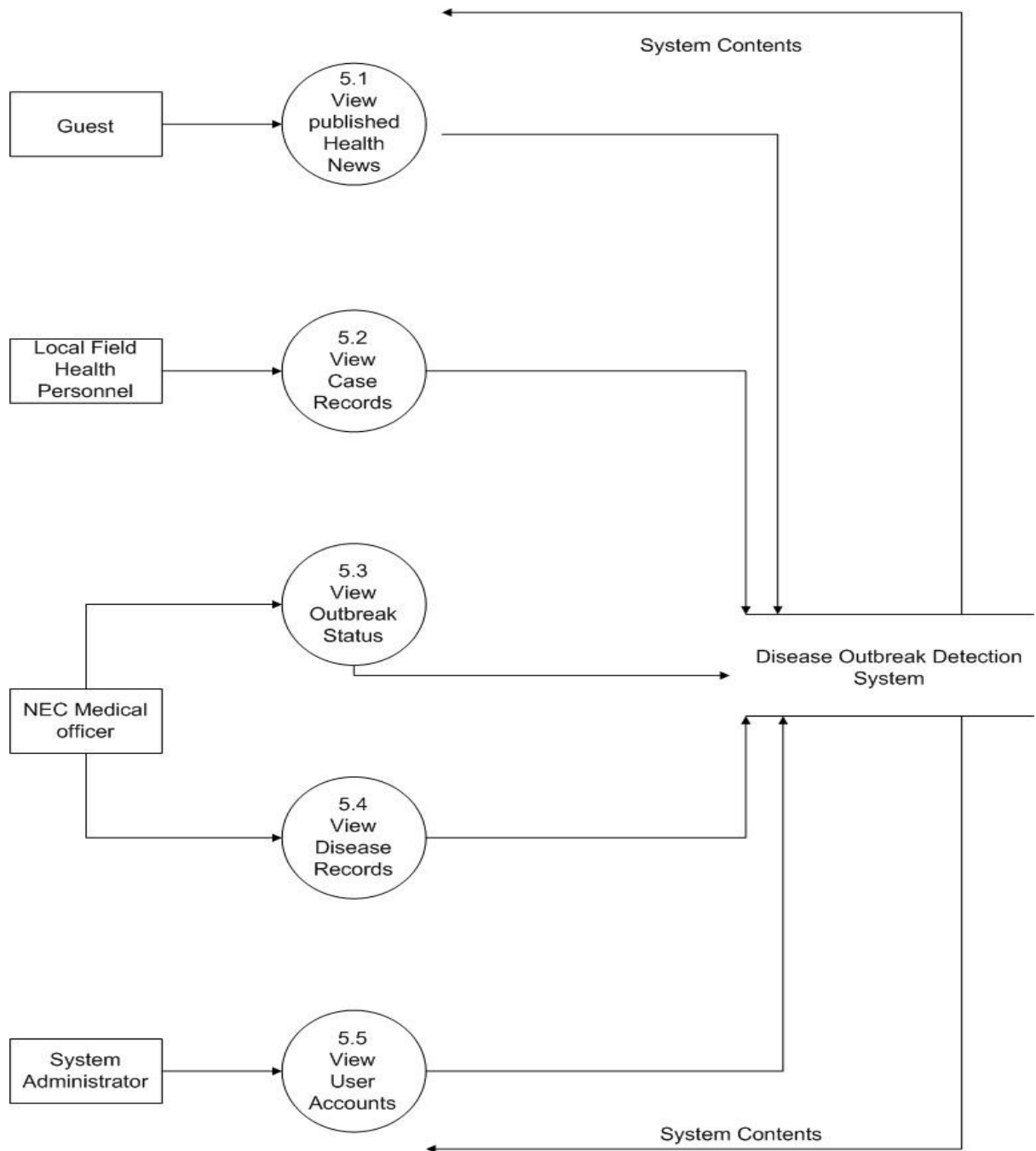
Take note that Process 3.1 is accessible only to the local field health personnel while Process 3.2 is accessible only to the NEC medical officer. They are allowed to view the details of each record to assess it and to perform necessary operations.



**FIGURE 7. SUBEXPLOSION OF PROCESS 4: MANAGE USER ACCOUNTS**

Figure 7 shows the sub explosion of Process 4 Manage User Accounts. These three main sub processes namely Add New User Account, Activate/Deactivate Existing User Account and Update Existing User Account.

Note that only the System Administrator is responsible in managing all the user accounts within the system.

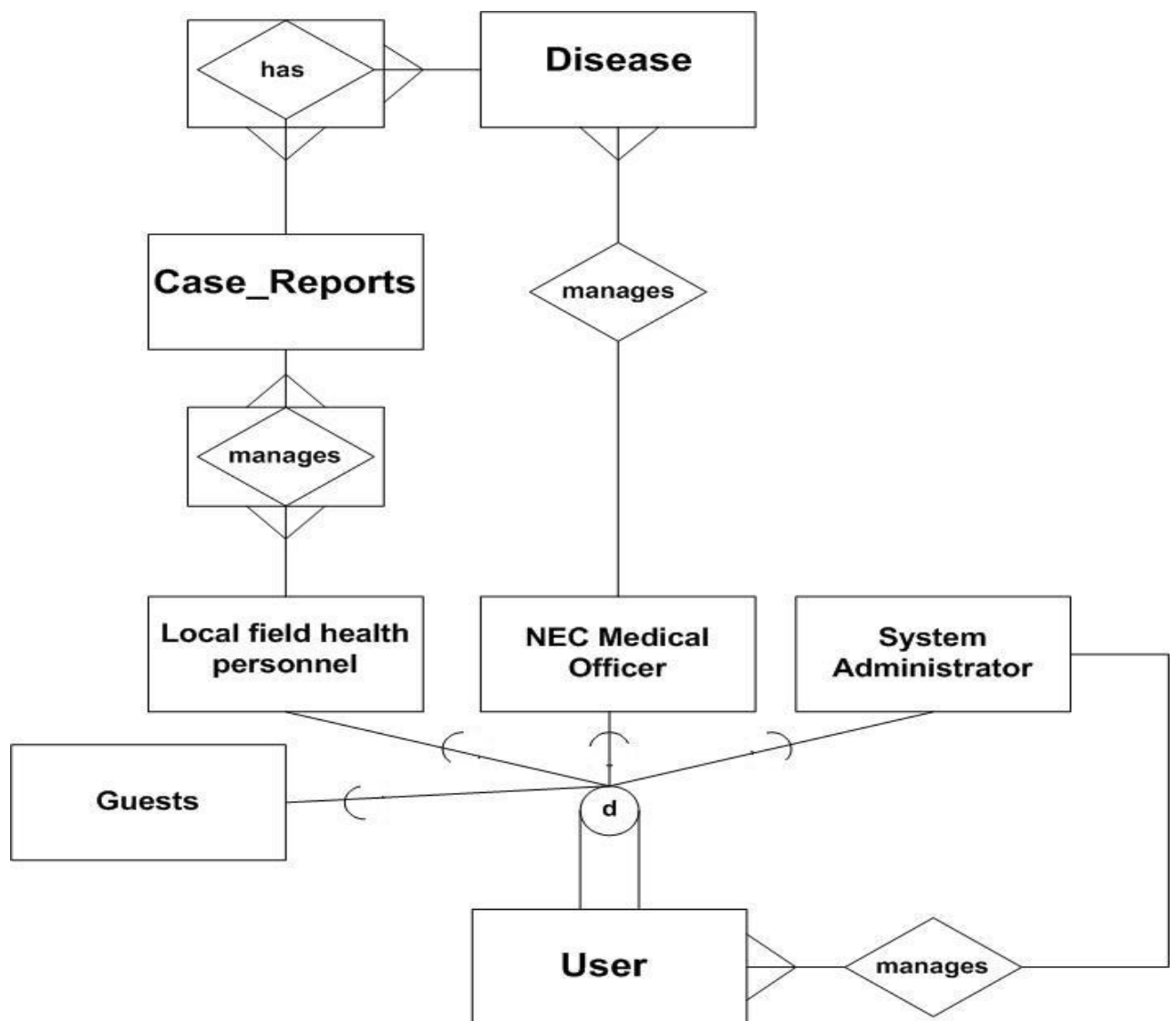


**FIGURE 8. SUBEXPLOSION OF PROCESS 5: VIEW SYSTEM CONTENTS**

Figure 8 shows the different contents that a particular user can view. Here, it shows that guests can only view the published health news while the local field health personnel can view case records. On the other hand, the NEC medical officer is the only one that can view the outbreak status and also the disease records stored in the system. While the System Administrator can view the accounts of the users registered in the system.

## ENTITY- RELATIONSHIP DIAGRAM

The Entity-Relationship Diagram of the system is shown in Figure 8. Here, the local field health personnel are responsible to manage the patient's record while the NEC medical officer is responsible for maintaining the disease records. The system administrator is responsible to manage several user accounts and to update the contents of the site.



**FIGURE 9: ENTITY-RELATIONSHIP DIAGRAM OF DISEASE OUTBREAK  
DETECTION SYSTEM**

## SYSTEM ENTITIES

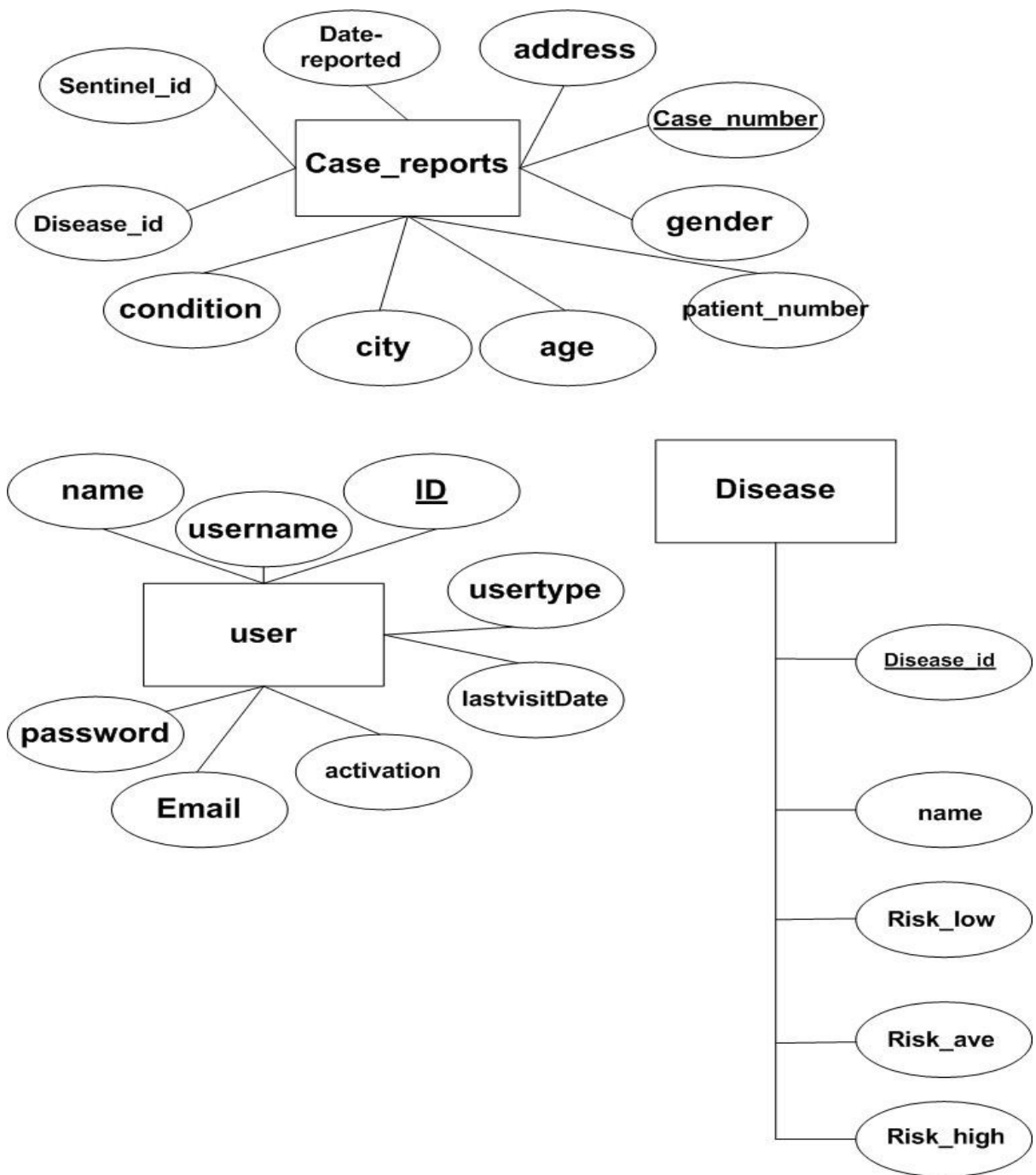


FIGURE 10: DISEASE OUTBREAK DETECTION SYSTEM ENTITIES



## DATA DICTIONARY

**TABLE 1: CASE\_REPORTS**-contains the patient's records

Field	Type	Key Index	Definition
case_number	Integer(11)	Primary key	unique id that refers to a specific case. Auto_increment
Patient_number	VarChar(50)		Patient's number
Age	Integer(5)		Age of the patient's
City	ENUM(LP,PQUE,MNLA,PAT,TAG ,MAR,MAL,QUE,PAS,SAN,CAL, MUN,VAL,NAV)		The city where the patient's resides
Gender	ENUM(M,F)		patient's gender
Disease_id	VarChar(5)	Foreign Key	Patient's disease
condition	ENUM(INFECTED,CURED,DECEASED)		Patient's condition
Sentinel_id	VarChar (20)		Id of hospital
Date_reported	Date		Date when the case was reported

**TABLE 2: USERS**-contains the records of the users of the system

Field	Type	Key Index	Definition
ID	Integer(11)	Primary Key	Unique id. Auto_increment
Name	Varchar(50)		User's name
username	Varchar(25)		User's username
Email	Varchar(100)		User's email
password	Varchar(100)		User's password
usertype	Varchar(25)		User's position
activation	ENUM(Enable,Disable )		User's registration status
lastvisitDate	Datetime		date and time the last visit of the user

**TABLE 3: DISEASE-** contains the details of diseases recorded

Field	Type	Key Index	Definition
Disease_id	VarChar(20)	Primary Key	ICD10 code of the disease
Name	VarChar(50)		Name of the disease
Risk_high	Integer(5)		High risk threshold
Risk_ave	Integer(5)		Average risk threshold
Risk_low	Integer(5)		Low risk threshold

## TECHNICAL ARCHITECTURE

The following tools are used to create and run the system:

- FPDF 1.6
- Joomla 1.5
- PRADO 3.1.3
- R-2.10.0.0
- Windows 98/XP/Vista
- XAMPP 1.6.4
- Any internet browser but preferably Mozilla Firefox that support Javascript

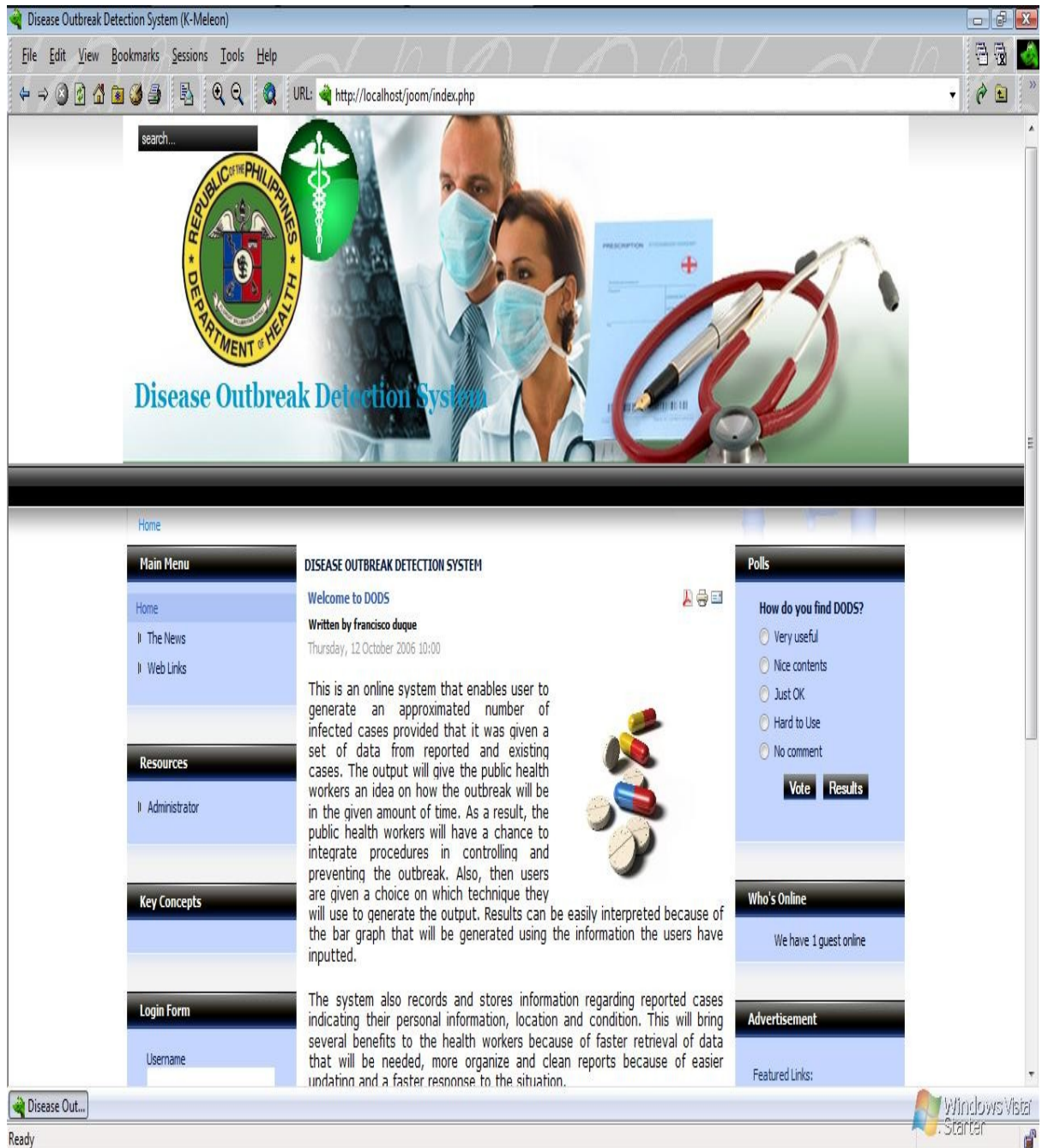
The following are the server requirements for the system:

- Pentium processor of speed at least 833 MHz or higher
- 5GB HDD or more
- 16MB video card or higher
- 128 MB RAM

The following are the clients requirement for the system:

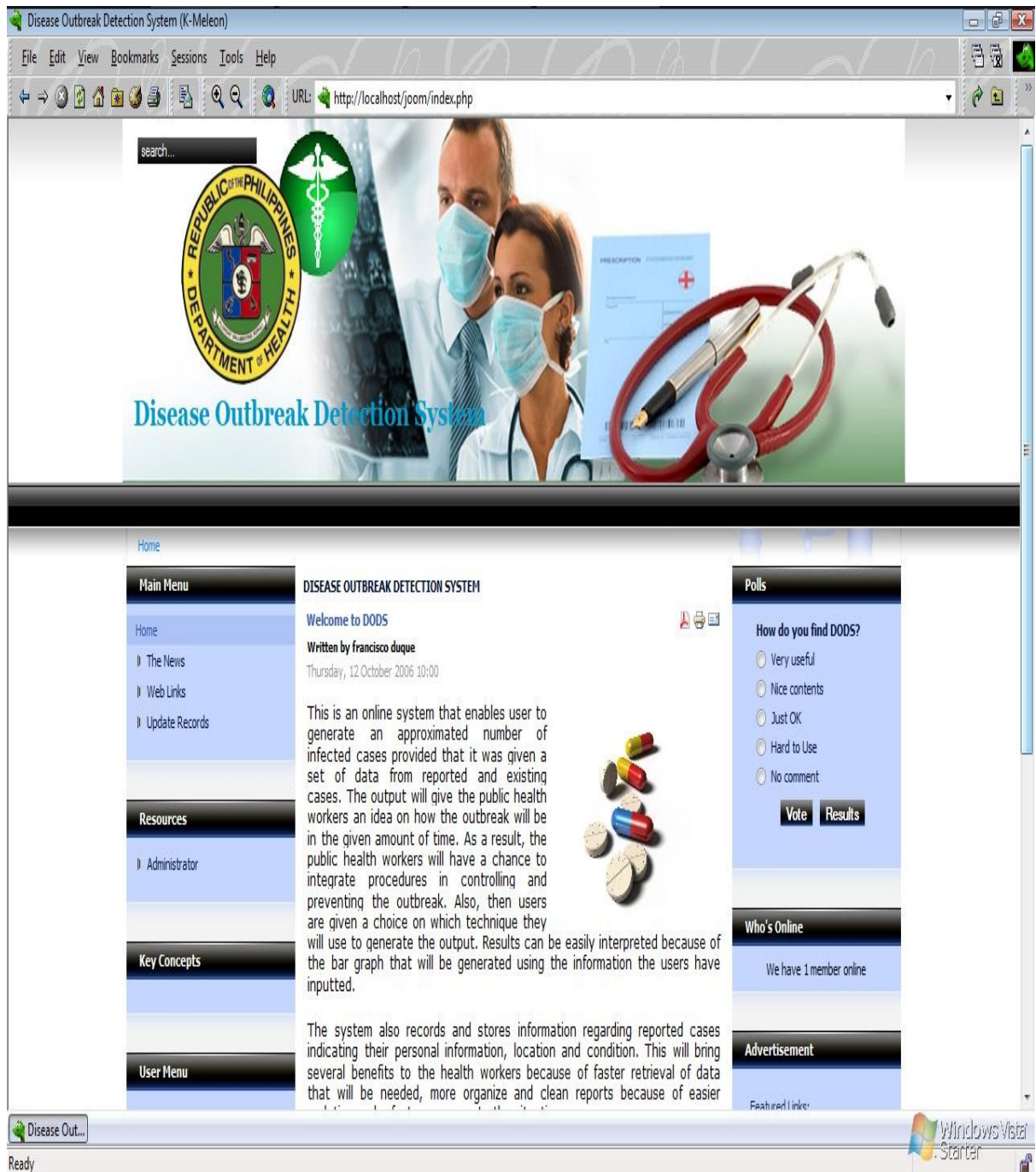
- Pentium processor of speed at least 833 MHz or higher
- 16MB video card or higher
- 128 MB RAM
- Internet connection

## V. RESULTS



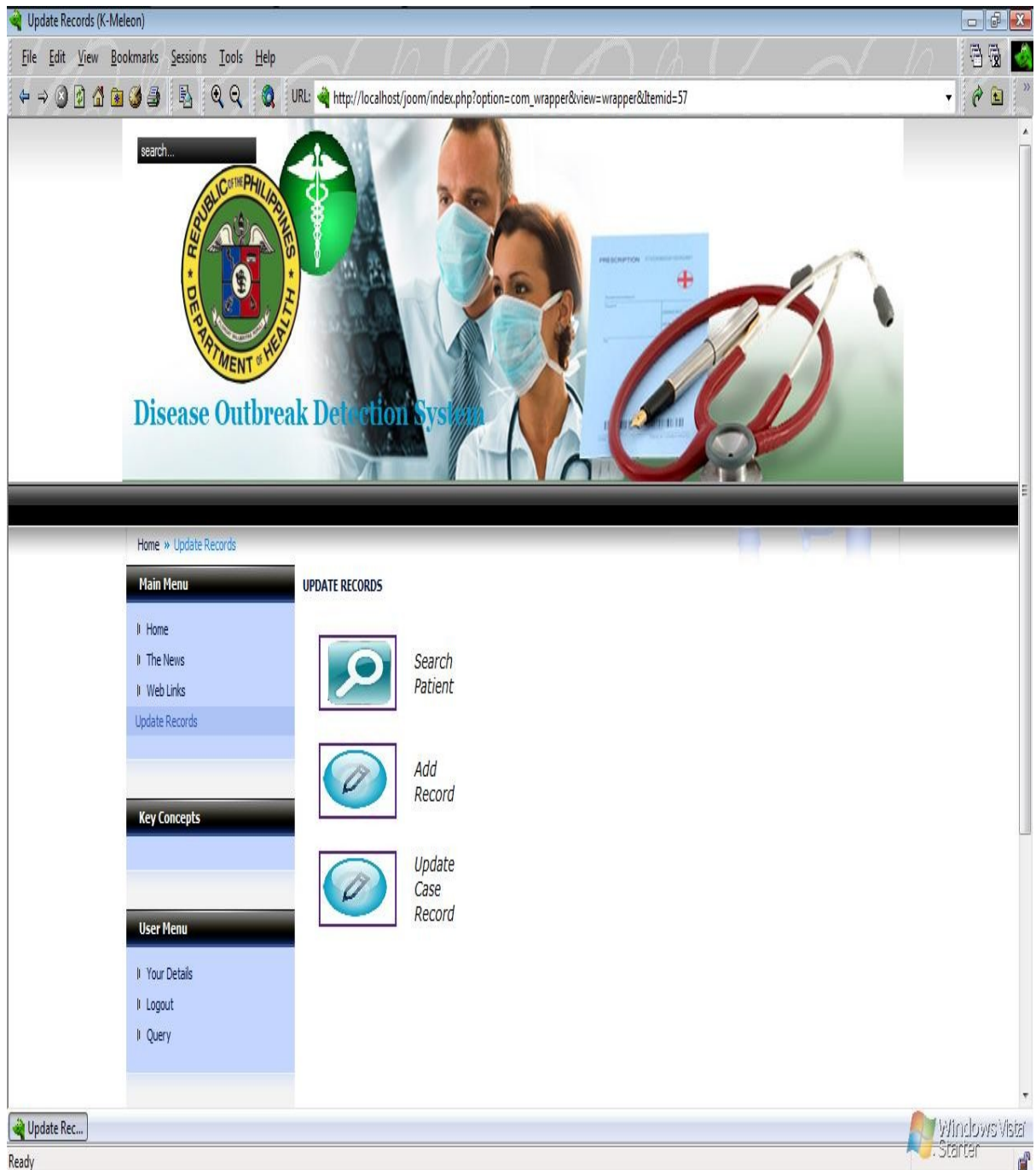
**FIGURE 11: DISEASE OUTBREAK DETECTION SYSTEM HOMEPAGE**

The Disease Outbreak Detection System homepage is shown in Figure 11. Here, health news is displayed. This is the only page online guests can access.



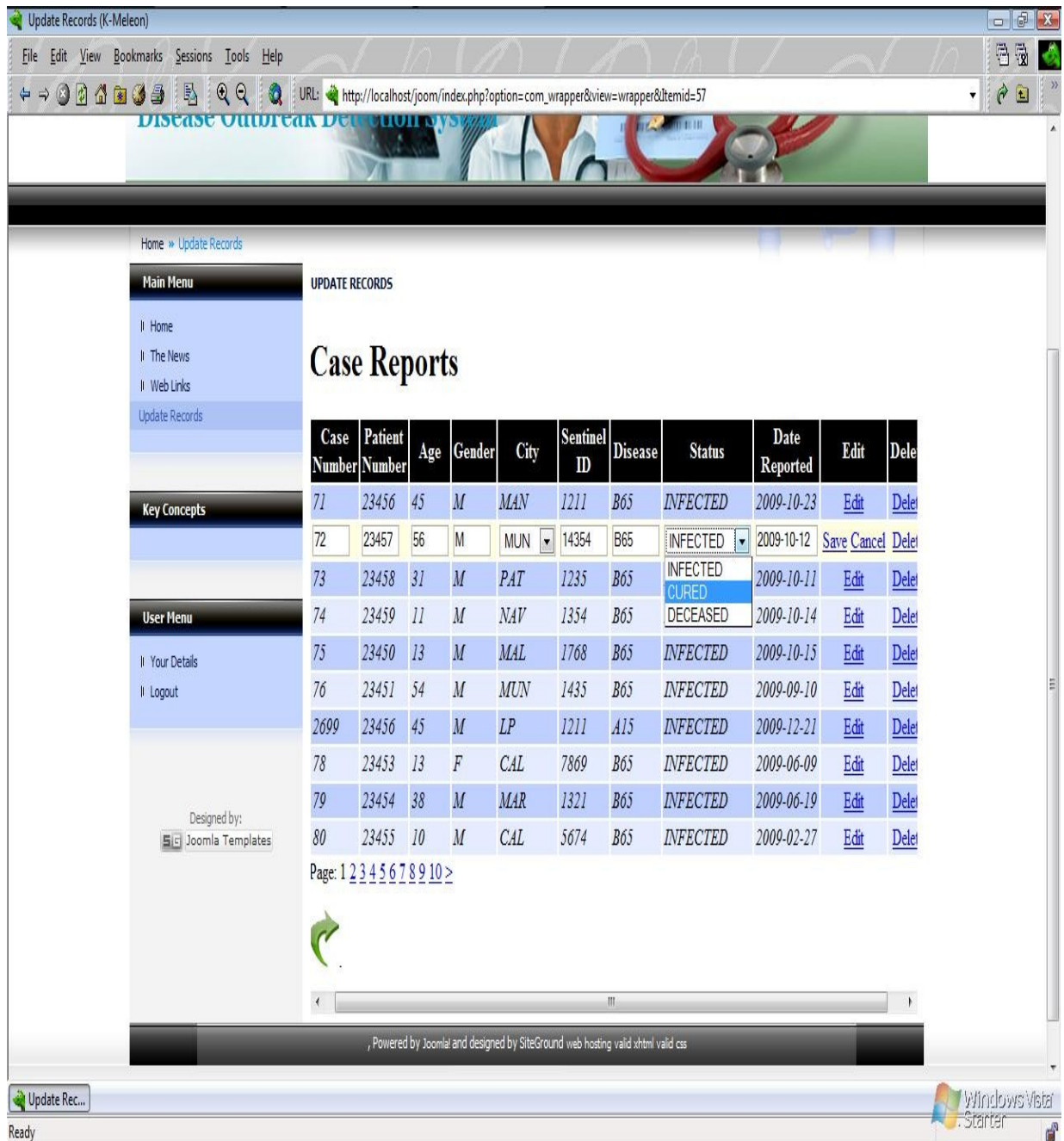
**FIGURE12: LOCAL FIELD HEALTH PERSONNEL PAGE ,DODS**

The page which the local field health personnel can view is shown in Figure 12. Here, only one additional feature is added and that is the Update Records. This function gives the user the ability to manage reported Case Records.



**FIGURE 13: UPDATE RECORD PAGE FOR LOCAL FIELD HEALTH PERSONNEL, DODS**

Following the previous page, the Update Record Page shown in Figure 13 offers several functionalities to the user. These functionalities are: Search Patient, Add Record and Update Case Record.



**FIGURE 14: UPDATE CASE RECORD PAGE, DODS**

The Update Case Record is shown in Figure 14. This page is also accessible only for registered users particularly the local field health personnel. This page allows the users to easily manage all the records that the system stores. This page gives the capability of deleting or editing a particular patient record. Also, this page displays some of the details of the patient for easy referencing.

Update Records (K-Meleon)

File Edit View Bookmarks Sessions Tools Help

URL: http://localhost/joom/index.php?option=com\_wrapper&view=wrapper&Itemid=57

## Disease Outbreak Detection System

Home » Update Records

UPDATE RECORDS

### Case Reports

Case Number	Patient Number	Age	Gender	City	Sentinel ID	Disease	Status	Date Reported
71	23456	45	M	MAN	1211	B65	INFECTED	2009-10-23
72	23457	56	M	MUN	14354	B65	INFECTED	2009-10-12
73	23458	31	M	PAT	1235	B65	INFECTED	2009-10-11
74	23459	11	M	NAV	1354	B65	INFECTED	2009-10-14
75	23450	13	M	MAL	1768	B65	INFECTED	2009-10-15
76	23451	54	M	MUN	1435	B65	INFECTED	2009-09-10
2699	23456	45	M	LP	1211	A15	INFECTED	2009-12-21
78	23453	13	F	CAL	7869	B65	INFECTED	2009-06-09
79	23454	38	M	MAR	1321	B65	INFECTED	2009-06-19
80	23455	10	M	CAL	5674	B65	INFECTED	2009-02-27

Page: 1 2 3 4 5 6 7 8 9 10 >

Designed by: Joomla Templates

Powered by Joomla! and designed by SiteGround web hosting valid xhtml valid css

**FIGURE 15: SEARCH PATIENT RESULT PAGE, DODS**

The Search patient result page is shown in Figure 15. Here, a few details of a case are displayed. The page displays at the maximum of ten patient's record per page in case of multiple results.





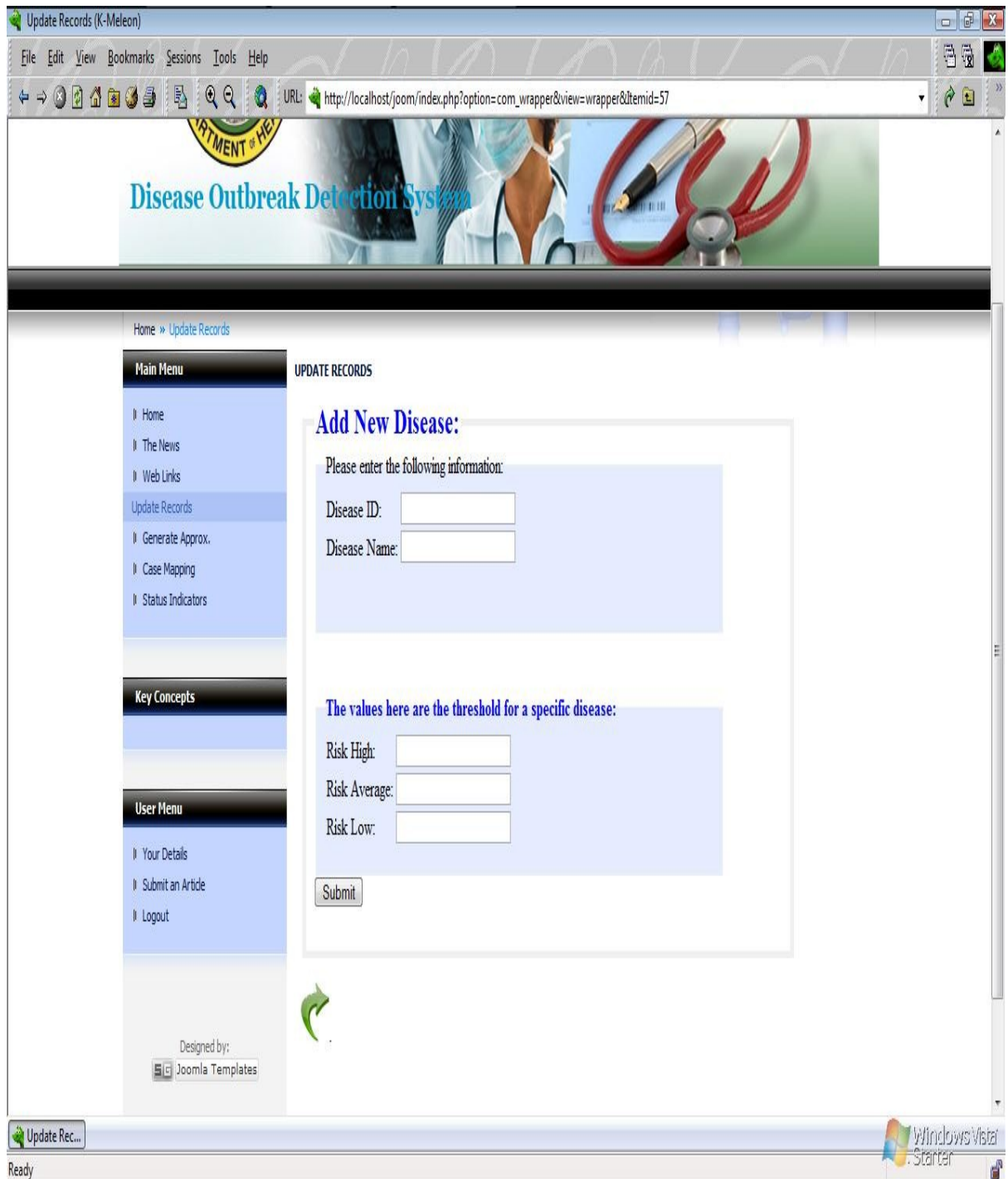
**FIGURE 16: NEC MEDICAL OFFICER PAGE, DODS**

The page which the NEC medical officer can view is shown in Figure 16. Here, several features are added and these are: Update Records, Generate Approx, Case Mapping, and Status Indicators. This function gives the user basis in assessing disease outbreak.



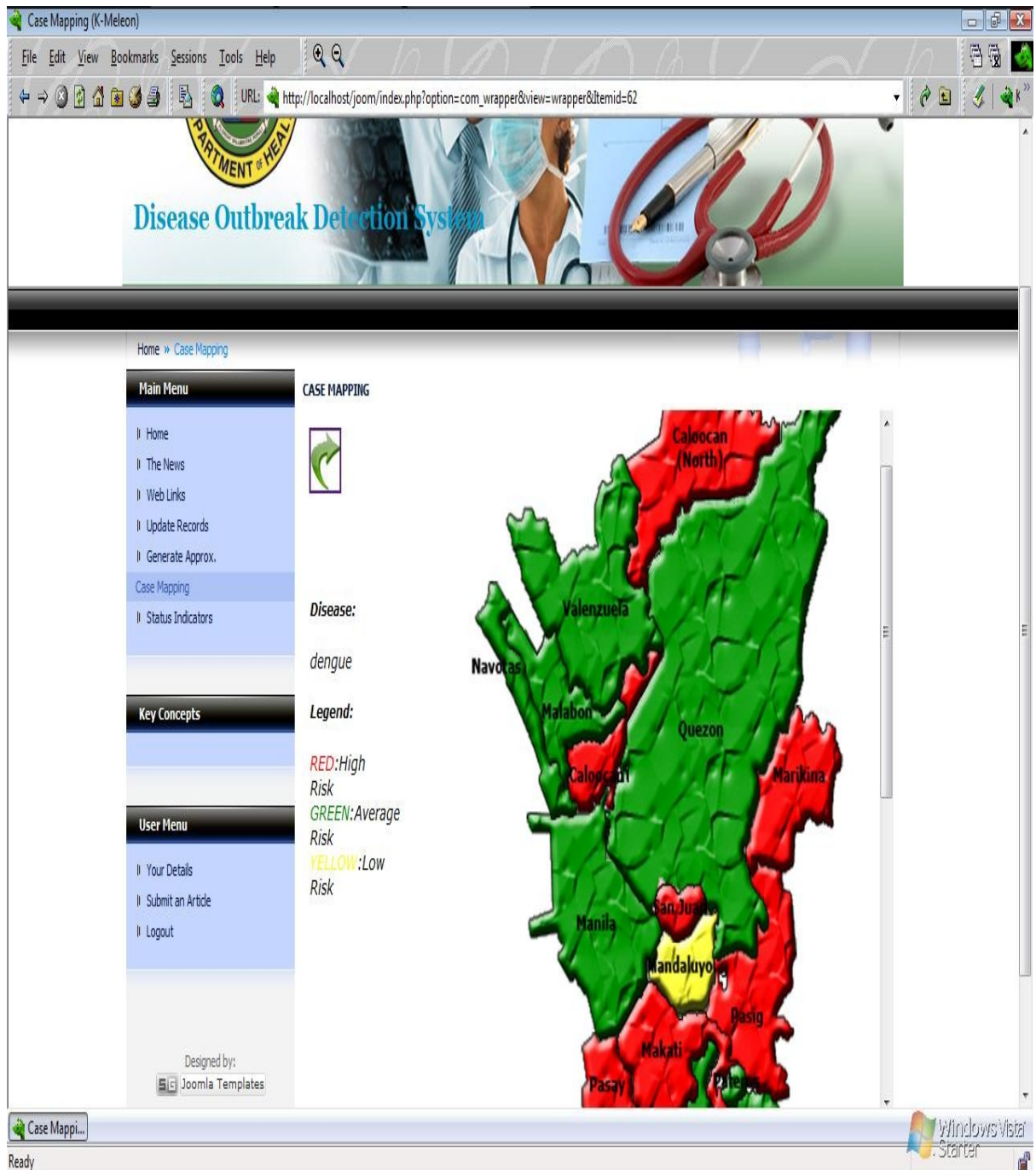
**FIGURE 17: UPDATE RECORD PAGE FOR NEC MEDICAL OFFICER, DODS**

Following the previous page, the Update Record Page shown in Figure 17 offers several functionalities to the user. These functionalities are: Add Disease record and Update Disease Record.



**FIGURE 18: ADD DISEASE RECORD PAGE, DODS**

Following the previous page, the Add Disease Record Page is shown in Figure 18. Here, the user is given the option to add a new disease record to the database.



**FIGURE 19: CASE MAPPING RESULT PAGE , DODS**

The Case Mapping result page is shown in Figure 19. Here, cities were colored according to their risk-level for a particular disease. This page helps users to easily identify cities that require immediate attention.

Generate Approx. (K-Meleon)

File Edit View Bookmarks Sessions Tools Help

URL: http://localhost/joom/index.php?option=com\_wrapper&view=wrapper&Itemid=60

## Disease Outbreak Detection System

Home » Generate Approx.

**Main Menu**

- || Home
- || The News
- || Web Links
- || Update Records
- Generate Approx.
- || Case Mapping
- || Status Indicators

**Key Concepts**

**User Menu**

- || Your Details
- || Submit an Article
- || Logout

Designed by:  
Joomla Templates

**GENERATE APPROX.**

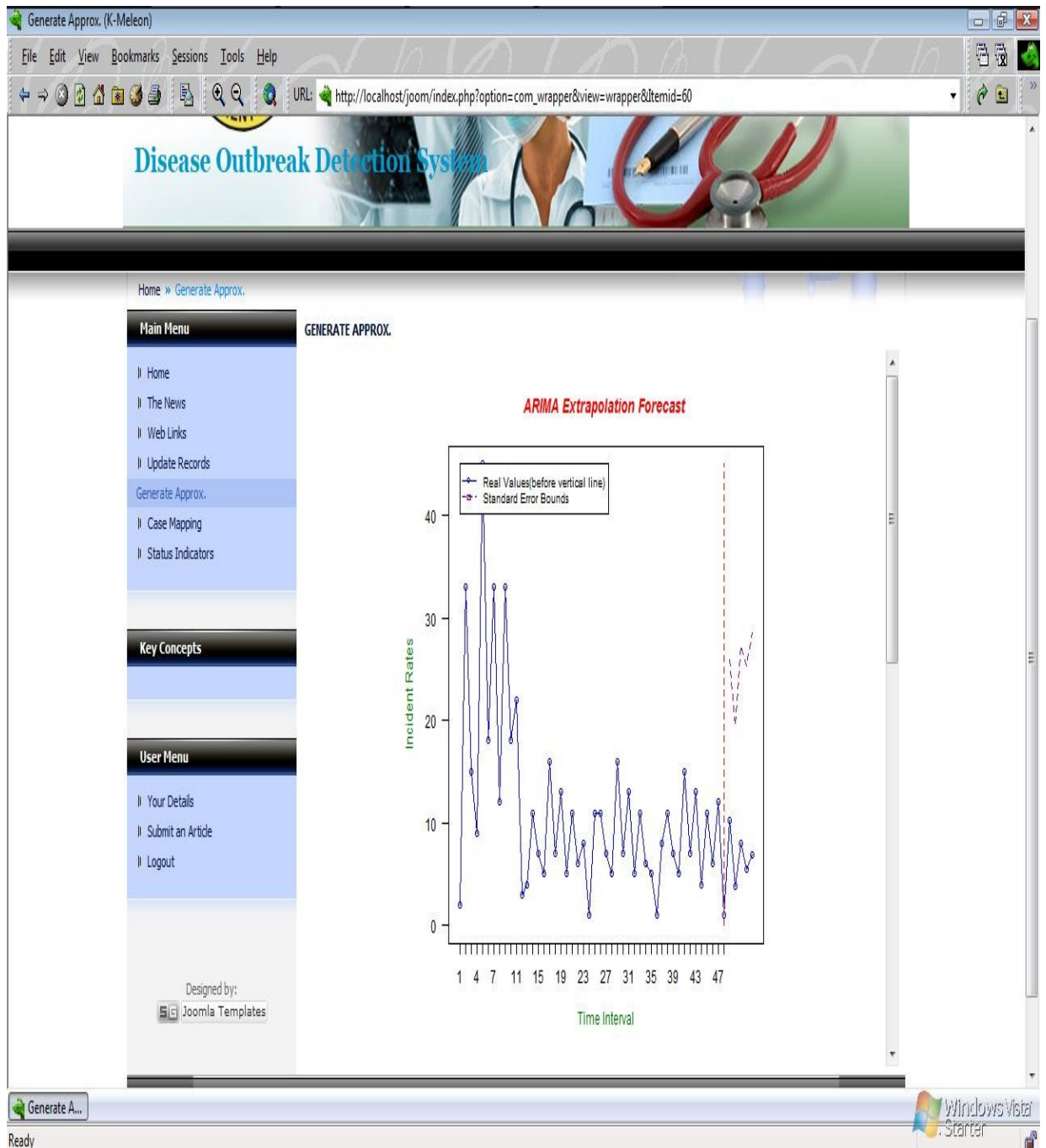
**ARIMA Extrapolation Forecast**

Time Interval	Incident Rate (Real Values)
1	2
2	33
3	15
4	9
5	40
6	18
7	33
8	12
9	33
10	22
11	18
12	3
13	11
14	5
15	16
16	11
17	13
18	6
19	13
20	11
21	11
22	11
23	11
24	11
25	11
26	11
27	11
28	11
29	11
30	11
31	11
32	11
33	11
34	11
35	11
36	11
37	11
38	11
39	11
40	11
41	11
42	11
43	11
44	11
45	11
46	11
47	11

Time Interval

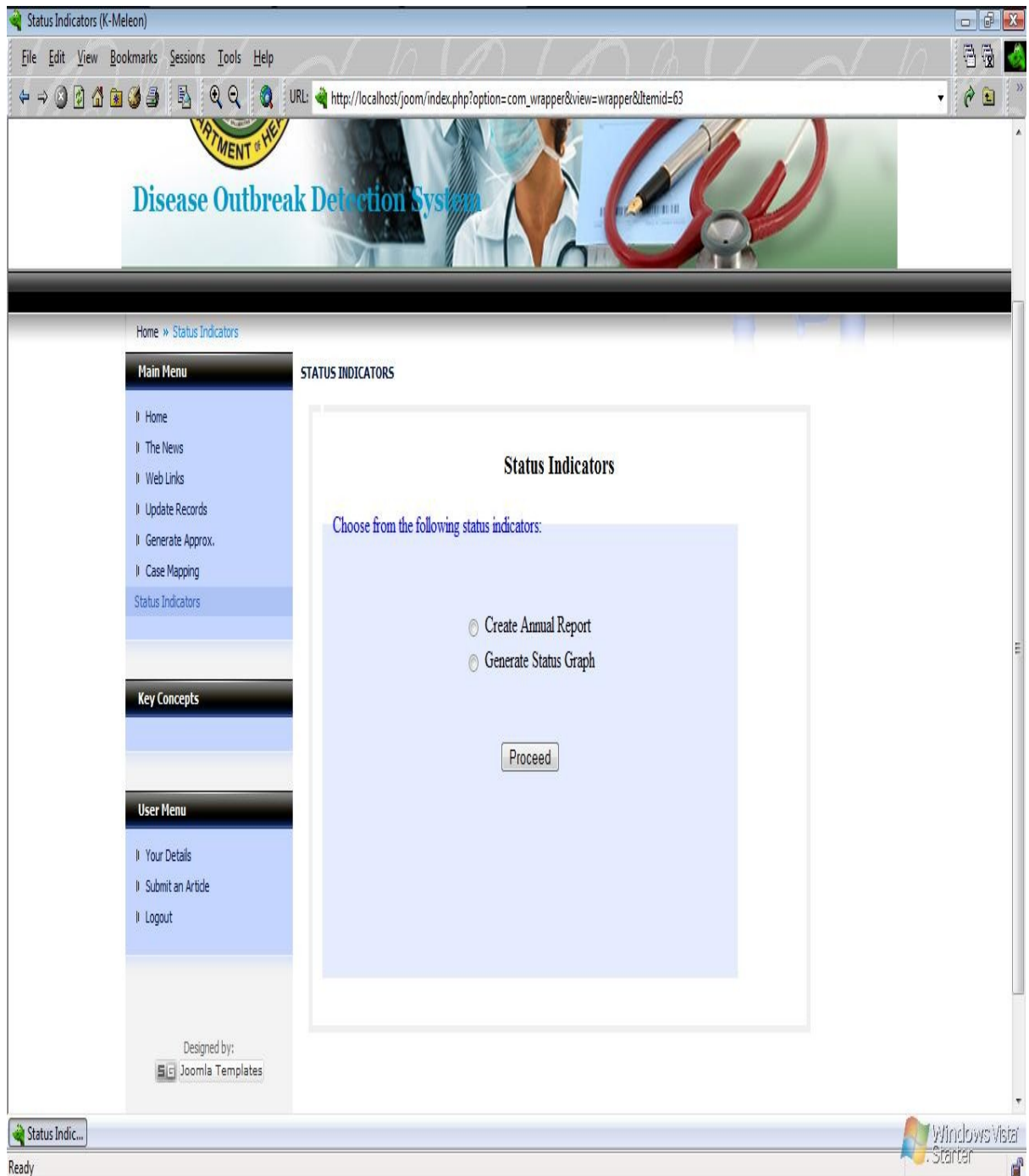
45

Windows Vista



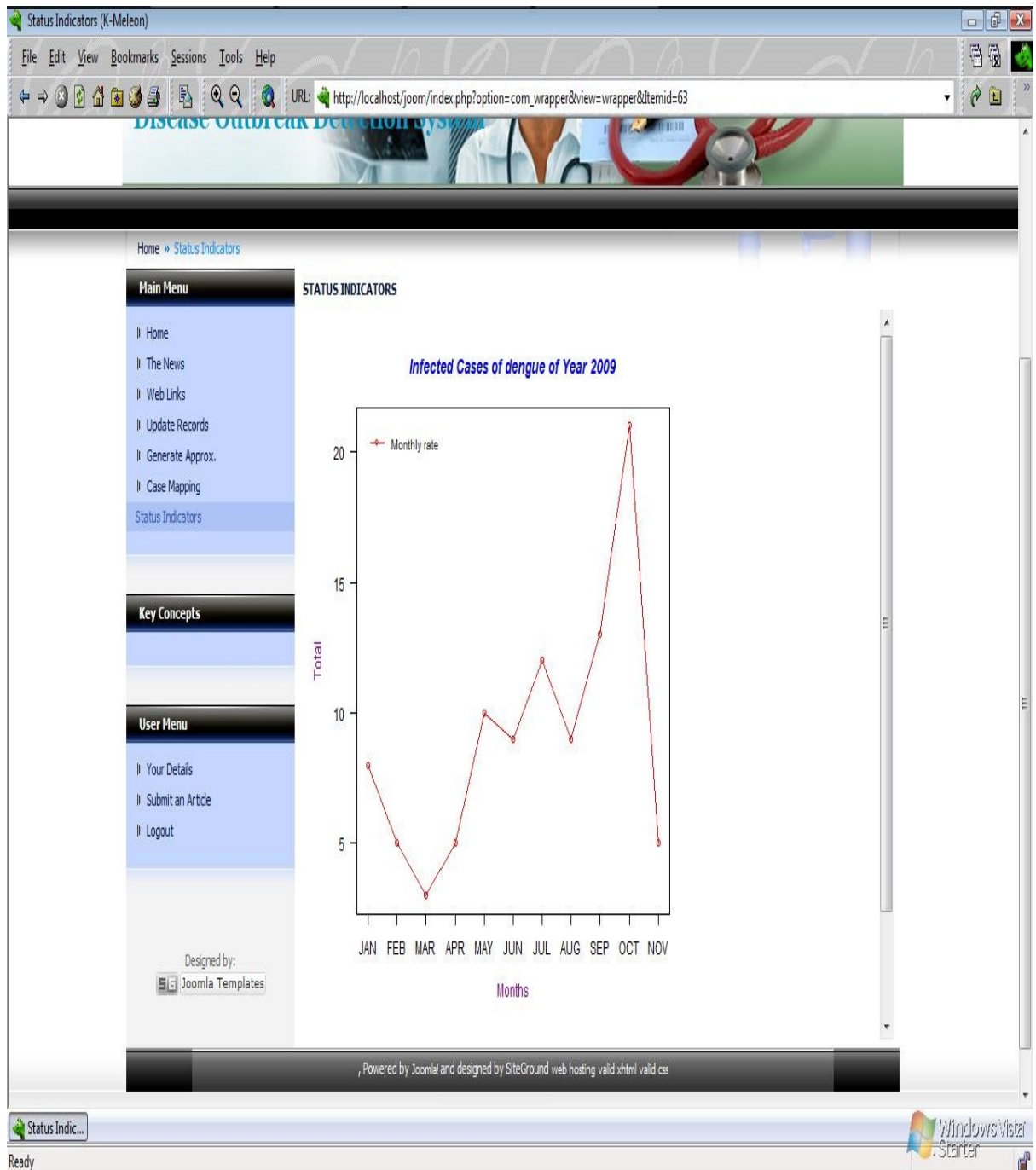
**FIGURE 20: GENERATE APPROX. RESULT PAGE , DODS**

The Generate Approx. result page is shown in Figure 20. Here, a short summary of the trend of the outbreak for a particular time interval is shown together with the prediction values and the standard error bounds.



**FIGURE 21: STATUS INDICATORS PAGE, DODS**

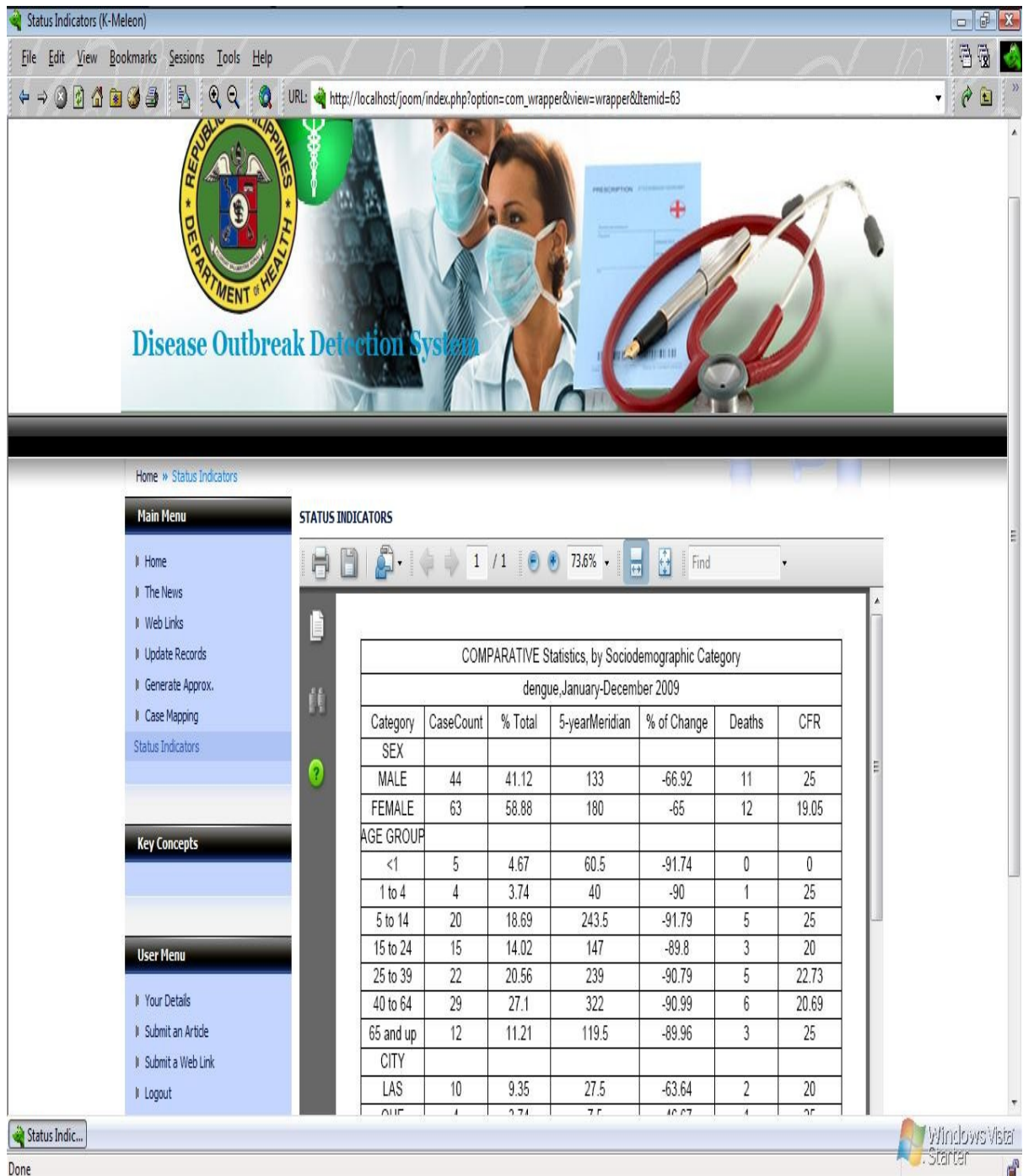
The Status Indicators page is shown in Figure 21. Here, the user is given options to what tool he/she will be using to assess the disease outbreak. And these options are: Create Annual Report and View Status Graph.



**FIGURE 22: STATUS GRAPH RESULT PAGE, DODS**

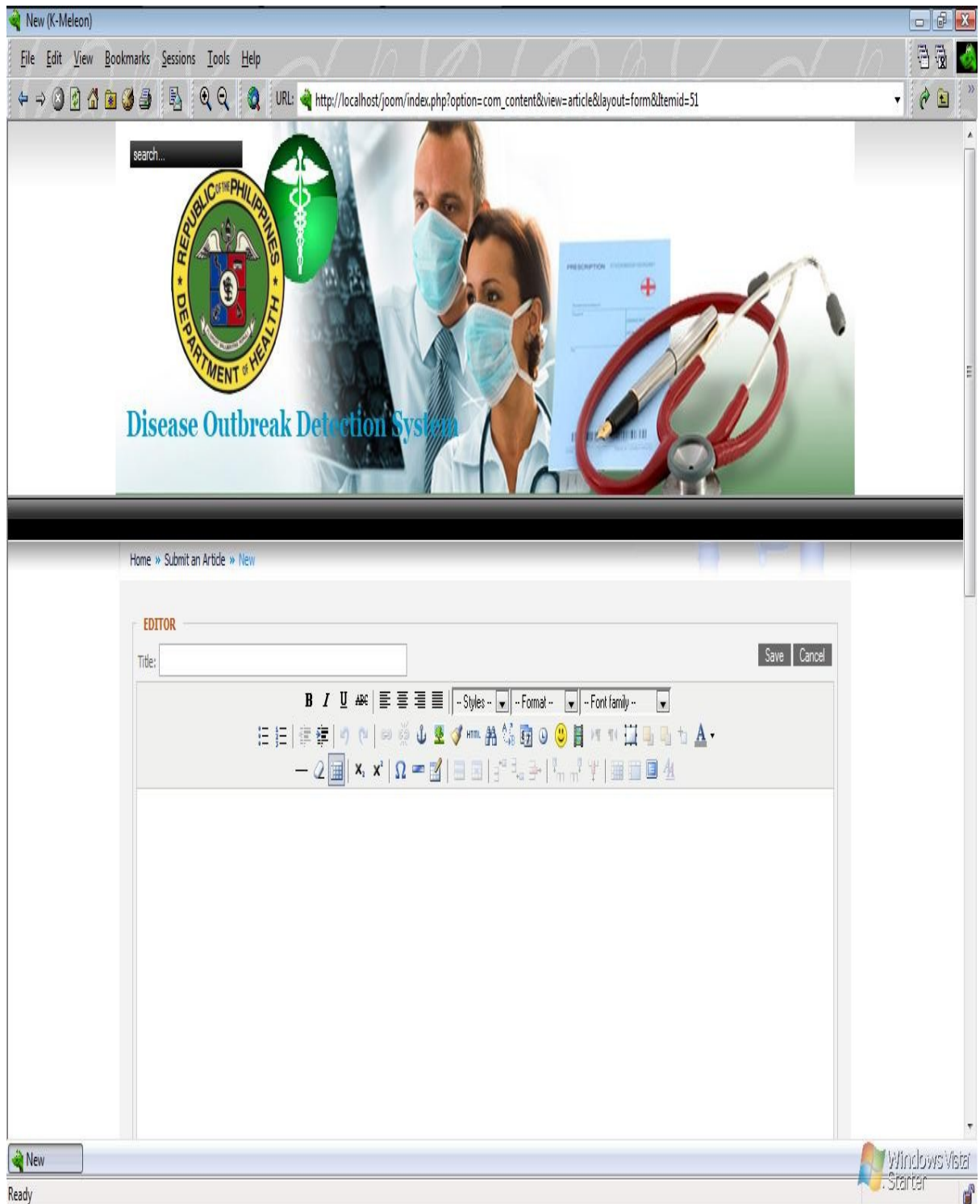
The Status Graph Result page is shown in Figure 22. This page displays the appropriate graph that will reflect the outbreak status of a particular disease. In the figure, we can see that the frequency of the case reported per month is displayed for easy analysis of the behaviour of the outbreak.





**FIGURE 23: ANNUAL REPORT PAGE, DODS**

The Annual Report page is shown in Figure 23. This page displays the summary report of a disease for a particular year. The report is subdivided into groups such as: Sex, Age group and City. The user can save a copy for archiving purposes.



**FIGURE 24: SUBMIT HEALTH NEWS PAGE, DODS**

The Submit Health News is shown in Figure 24. Here, the user can contribute an article regarding health issues to be displayed and viewed by all the users.

## **VI. DISCUSSION**

The disease outbreak detection system has two main users namely the local field health personnel and the NEC medical officer. The task of the local field health personnel is to input data they gathered from different sources such as hospitals in their respective cities. The system offers them the Update Records functionality which has Search Patient record, Add New case Record and Update Existing Case record. These functionalities enable the local field health personnel to manage the patient's data stored in the system. The Search patient records allow the user to provide fields that will limit the results giving them the option to narrow down the outputs. The Add New Cases allows the user to input data of patients into the system and this information will be used later on for disease outbreak analysis by the NEC medical officer. The Update Existing Case record allows the users to edit or delete patients' records that are already in the system to correct any errors if any.

For the NEC medical officer, several tools are provided that aims to assist the user in assessing the disease outbreak in a faster and with less errors. The functions that the system provides are as follows: Update Records, Generate Approx., status Indicators and Case Mapping. First is the Update Records. Unlike the one that the local field health personnel can use, this functionality offers the NEC medical officer to update the existing records of diseases that the system analyses. Options like Add New Disease Records and Update Existing Disease records are also given. Add New Disease records allow the user to input a new disease and to set its threshold to identify its risk-levels. In Update Disease records, list of diseases with their corresponding details are displayed and the options whether to edit or delete them. This gives the capability to the users to manage disease records. Second is the Generate Approx. feature. This feature enables the user to generate predicted values for a certain time interval given the incident cases reported from the past to serve as the basis in determining the trend of the disease outbreak. Here, the user is given the freedom to set the time interval the system will predict, the time unit whether its monthly or yearly, and the disease to analyze. For the output, several statistical identities are also displayed aside from the predicted values such as the estimated standard errors to further understand the behaviour of a disease. This feature aims to provide basis to the health personnel in their decision-making and planning of actions to control or prevent the outbreak. Next is the Status Indicators. This feature gives the user an option whether the analysis of disease will be in the form of annual reports or status graphs. Choosing Annual reports required the user to input

the year he/she wants to view and also the particular disease. The result page will display the report in a PDF format which allows the user to save a copy for archiving purposes. The generated report follows the standard and format of the reports created and used by NEC. For the status graphs, the user is required to input the particular disease to plot and the type of graph to be used. These functionalities help assess the current status of the disease outbreak within the National Capital Region. Lastly, the case mapping feature requires the user to input the particular disease to view. The result page will display a map of NCR and cities are colored differently according to its risk-level to a particular disease. High-risk cities are colored red, green for average risk levels and cities having low-risk level are colored yellow. This aims to provide easy and faster identification of cities that require more attention. The system also provides additional feature to the NEC medical officer and this feature is to Publish Health News. This allows the NEC medical officer to create health related articles or news that will reflect the status of an outbreak to inform other users like the online guests regarding an outbreak.

In general, Disease Outbreak Detection System provides useful tool for health personnel in analysing and understanding the behaviour and trend of an outbreak. The outputs generated using the system will reflect the current or even future status of the outbreak giving health personnel basis in their decision-making and in planning their actions to prevent and control an outbreak. And also, the system helps in providing status reports and analysis of an outbreak having less errors and in a much faster way.

## VII. CONCLUSION

The disease outbreak detection system aims to provide health personnel a number of tools that they can use in order to assess the behaviour of the disease in a faster and easier way. The results yielded by the system will serve as a basis for decision making and formulating preventive measures in order to control such outbreaks. Tools such as annual reports that summarize incident cases of a disease for particular categories such as age group and gender and also graphs showing how a disease outbreak behaves over a particular time interval are necessary to identify the trend and provide them a basis to predict how it will turn out ahead of time.

Autoregressive moving averages is the model that was used in this system. It is shown that this model can be used to predict the outcome of an outbreak using previous incident cases as a basis in analyzing and generating prediction values. We noticed that as the estimation goes further from the basis, the standard error increases. This effect is because the predicted values generated from the previous calculations will be used to generate other values, therefore the strength of the predicted values weakens as the time interval increases.

In conclusion, the disease outbreak detection system is a useful tool in assessing and understanding the behaviour of a disease hence giving the health personnel some basis in planning some actions to prevent and control the outbreak. Also, the system solves the problem of inconsistency and errors brought by consolidating data from local health personnel to the NEC. The system solves this because as the local field health personnel enter data in the system, it will be automatically directed to NEC and can be quickly used to generate an analysis of the outbreak.

## VIII. RECOMMENDATION

Disease Outbreak Detection System is limited only within the cities of National Capital region therefore it is recommend to apply the system in a much wider scope. It can be regional as for Luzon only or maybe it can be applied nationwide.

Also, the system uses Time Series analysis particularly the Autoregressive Moving Averages. The model is proven to be effective in generating prediction values and for the analysis of the trend and behaviour of a disease outbreak but other models or techniques are available to be used instead of the ARMA model.

Furthermore, the system uses a statistical software particularly the R software for its Time Series requirements. The software is chosen because of it being free and open source software. And this software provides several statistical tools including graphical and analytical features that can be used in many different areas. But this is not the only software that offers such features. There are a number of software available that can be used for other studies.

Lastly, the system focuses mainly on Communicable diseases namely Tuberculosis, Malaria, Leprosy and Schistosomiasis. It is recommended to include or limit the system to other kinds of disease. Other systems can be developed to analyze disease outbreaks specializing on other types of diseases.

## IX. BIBLIOGRAPHY

- [1] Domingo, Estrella V. "Administrative Data Key to Health Policies and Programs: Philippine Experience ".Inter-Secretariat Working Group-Health Statistics Meeting International Association for Official Statistics Conference.Shanghai China. 14-17 October 2008.
- [2] "NESSS Annual Report". Office of the Secretary, Department of Health.Republic of the Philippines.2001
- [3] Exaltacion, Lamberte. "Data Collection and Recording System: The Case of the Health Centers in the National Capital Region". DLSU-Social Development Research Center. SDRC TR 1993-01
- [4] <http://www.sciencedaily.com/releases/2005/02/050218130731.htm>
- [5] <http://www2.computer.org/portal/web/csdl/doi/10.1109/TKDE.2009.115>
- [6] Zhongguo Ji Sheng, Chong Xue Yu Ji Sheng ,Chong Bing Za Zhi."Study on the feasibility for ARIMA model application to predict malaria incidence in an unstable malaria area".PBMEd. 2007 Jun;25(3):232-6.
- [7] S. Promprou, M. Jaroensutasinee and K. Jaroensutasinee."Forecasting Dengue Haemorrhagic Fever Cases in Southern Thailand using ARIMA Models".School of Science, Walailak University, 222 Thaiburi, Thasala District, Nakhonsithammarat 80161, Thailand.
- [8] Akhtar, S. and Shafquat Rozi." An autoregressive integrated moving average model for short-term prediction of hepatitis C virus seropositivity among male volunteer blood donors in Karachi, Pakistan". World Journal of Gastroenterology. 2009 April 7; 15(13): page 1607-1612
- [9] Triedman, J., Perrott, M., Cohen, J. and J.P. Saul." Respiratory Sinus Arrhythmia: Time Domain Characterization using Autoregressive Moving Average Analysis".Boston USA.
- [10] Earnest, Arul, Chen, Mark, Ng Donald and Yee Sin Leo. "Using autoregressive integrated moving average (ARIMA) models to predict and monitor the number of beds occupied during a SARS outbreak in a tertiary hospital in Singapore".BioMed Central. Singapore.2005.
- [11] B.J. Ayeni and R. Pilat." Crude oil reserve estimation: An Application of the Autoregressive Integrated Moving Average (ARIMA) Model". Journal of Petroleum Science and Engineering. July 1, 1992.
- [12] Williams, B M, Durvasula, P K and Brown, D E." Urban Freeway Traffic Flow Prediction: Application of Seasonal Autoregressive Integrated Moving Average and Exponential Smoothing Models". Transportation Research Record.1998. p. 132-141

[13] Blood, Christopher and Edward O'Donnell." A Bivariate Autoregressive Integrated Moving Average Analysis of Combat Troop Casualty Rates". Storming Media. San Diego CA. Oct 93-Feb 94.

[14] N.P.P. Macciotta, A. Cappio-Borlino, and G. Pulina." Time Series Autoregressive Integrated Moving Average Modelling of Test-Day Milk Yields of Dairy Ewes". Journal of Dairy Science Vol. 83, No. 5, 2000.

[15] Peng Chen, Hongyong Yuan, Xueming Shu, "Forecasting Crime Using the ARIMA Model," Fuzzy Systems and Knowledge Discovery, Fourth International Conference on, vol. 5, pp. 627-630, 2008 Fifth International Conference on Fuzzy Systems and Knowledge Discovery, 2008.

[16] [http://www.ltrr.arizona.edu/~dmeko/notes\\_5.pdf](http://www.ltrr.arizona.edu/~dmeko/notes_5.pdf)

[17] "Time Series".<[http://en.wikipedia.org/wiki/Time\\_series](http://en.wikipedia.org/wiki/Time_series)>

[18] Shaddick, Gavin." Using R (with application in Time Series analysis)". January 2004. page 4.

[19] "CFR".<<http://en.wikipedia.org/wiki/CFR>>

[20] "Akaike's Information Criterion".<[http://en.wikipedia.org/wiki/Akaike\\_information\\_criterion](http://en.wikipedia.org/wiki/Akaike_information_criterion)>

[21] "Communicable Disease".<[http://en.wikipedia.org/wiki/Communicable\\_disease](http://en.wikipedia.org/wiki/Communicable_disease)>

[22] "Tuberculosis".<<http://en.wikipedia.org/wiki/Tuberculosis>>

[23] "Malaria".< <http://en.wikipedia.org/wiki/Malaria>>

[24] "Schistosomiasis".<<http://en.wikipedia.org/wiki/Schistosomiasis>>.

[25] "Leprosy".<<http://en.wikipedia.org/wiki/Leprosy>>

[26] [http://www.doh.gov.ph/nec/index.php?option=com\\_content&task=view&id=22&Itemid=36](http://www.doh.gov.ph/nec/index.php?option=com_content&task=view&id=22&Itemid=36)

[27] Domingo, Estrella. "Administrative Data Key to Health Policies and Programs: Philippine Experience". Inter-Secretariat Working Group-Health Statistics Meeting International Association for Official Statistics Conference. October 2008.

[28] <http://www.fpdf.org>

[29] <http://www.pradosoft.com>

[30] [http://en.wikipedia.org/wiki/Database\\_management\\_system](http://en.wikipedia.org/wiki/Database_management_system)

[31] [http://en.wikipedia.org/wiki/Information\\_systems](http://en.wikipedia.org/wiki/Information_systems)



**X. APPENDIX A: Sample Annual Report from NEC**

**Table 6. COMPARATIVE Statistics, by Sociodemographic Category and Region  
Dengue, January - December, 2008**

CATEGORY		Case count	% of Total	5-Year Median	%Change from 5-Year Median	Deaths	CFR (%)
Sex	Male	24573	52.7	17956	36.9	183	0.7
	Female	22087	47.3	13940	58.4	232	1.1
Age group (Years)	<1	1307	2.8	689	89.7	18	1.4
	1 to 4	6167	13.2	4930	25.1	112	1.8
	5 to 14	21529	46.1	16242	32.6	225	1.0
	15 to 24	11883	25.5	8320	42.8	38	0.3
	25 to 39	4180	8.9	2918	42.6	13	0.3
	40 to 64	1398	3.0	997	40.2	7	0.5
	65 & up	216	0.5	149	45.0	2	0.9
	Unknown						
Region	1	3374	7.2	1247	170.6	29	0.9
	2	1208	2.6	1404	-14.0	13	1.1
	3	7161	15.3	3567	100.8	58	0.8
	4A	4181	9.0	1429	192.6	31	0.7
	4B	272	0.6	93	192.5	4	1.5
	5	650	1.4	1205	-46.1	7	1.1
	6	2247	4.8	1718	30.8	20	0.9
	7	3425	7.3	2997	14.3	96	2.8
	8	940	2.0	1205	-22.0	11	1.2
	9	2172	4.7	933	132.8	8	0.4
	10	1075	2.3	2989	-64.0	14	1.3
	11	3173	6.8	2162	46.8	18	0.6
	12	1105	2.4	1326	-16.7	6	0.5
	ARMM	281	0.6	340	-23.2	3	1.1
CAR	598	1.3	1057	-43.4	2	0.3	
CARAGA	660	1.4	739	-10.7	5	0.8	
NCR	14158	30.3	5767	145.5	90	0.6	
Philippines		46660	100	33896	37.7	415	0.9

There were 46,660 dengue cases reported in 2008. Majority (52.7%) were male. The age group with the highest (46.1%) number of cases is the 5 years to 14 years age group. Most (30.3%) of the cases were from NCR. Four hundred fifteen died (CFR=0.9%)

**APPENDIX B: Sample Case Investigation Form**



Philippine Integrated Disease Surveillance and Response



Case Investigation Form

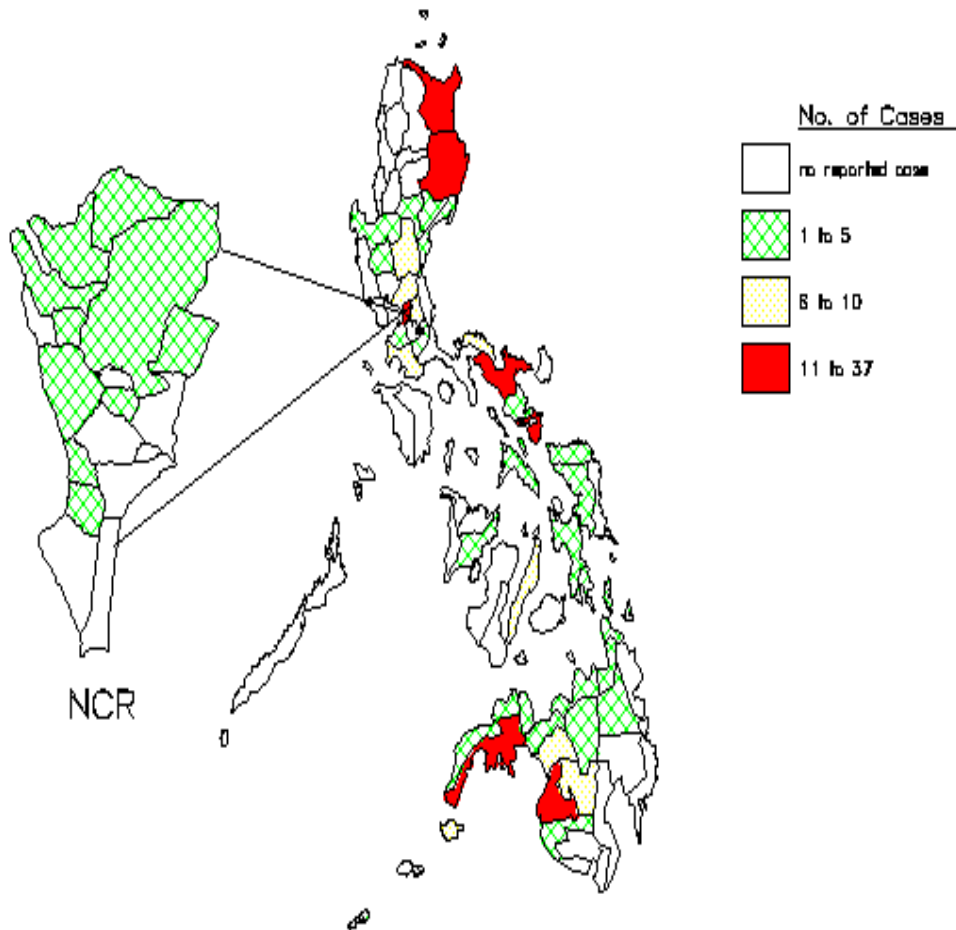
**Measles**

(ICD 10 Code: B05)

Name of DRU:		Type: <input type="checkbox"/> RHU <input type="checkbox"/> CHO <input type="checkbox"/> Gov't Hospital <input type="checkbox"/> Private Hospital <input type="checkbox"/> Clinic <input type="checkbox"/> Gov't Laboratory <input type="checkbox"/> Private Laboratory <input type="checkbox"/> Airport/Seaport					
Address:							
I. PATIENT INFORMATION:	Patient Number:	Patient's First Name		Middle Name	Last Name		
	Complete Address:						
District:		ILHZ:	Sex: <input type="checkbox"/> Male <input type="checkbox"/> Female	Date of Birth:	MM	DD	YY
Patient Admitted? <input type="checkbox"/> Yes <input type="checkbox"/> No <input type="checkbox"/> Unknown		Date Admitted/ Seen/Consult	MM	DD	YY	Date Onset of illness	MM
Date of Report:		MM	DD	YY	Date of Investigation:	MM	DD
II. CLINICAL INFORMATION:				III. VITAMIN A AND VACCINATION HISTORY:			
Fever: <input type="checkbox"/> Y <input type="checkbox"/> N <input type="checkbox"/> U Date onset: ___/___/___ Rash: <input type="checkbox"/> Y <input type="checkbox"/> N <input type="checkbox"/> U Date onset: ___/___/___ Cough: <input type="checkbox"/> Y <input type="checkbox"/> N <input type="checkbox"/> U Runny nose/coryza: <input type="checkbox"/> Y <input type="checkbox"/> N <input type="checkbox"/> U Red eyes/conjunctivitis: <input type="checkbox"/> Y <input type="checkbox"/> N <input type="checkbox"/> U Other symptoms: _____ Are there any complications? <input type="checkbox"/> Y <input type="checkbox"/> N <input type="checkbox"/> U If YES, specify: _____				Was the patient given therapeutic Vitamin A during this illness?: <input type="checkbox"/> Y <input type="checkbox"/> N <input type="checkbox"/> U Patient received routine measles vaccination? <input type="checkbox"/> Y <input type="checkbox"/> N <input type="checkbox"/> U No. of doses received: _____ Date of last vaccination: ___/___/___ If NO, state the reasons: <input type="checkbox"/> Mother was busy <input type="checkbox"/> Child was sick <input type="checkbox"/> Forgot the schedule <input type="checkbox"/> No vaccine available <input type="checkbox"/> Against belief <input type="checkbox"/> Not eligible for vaccination <input type="checkbox"/> Medical contraindication <input type="checkbox"/> Fear of side effects Other reasons, specify: _____ Patient received vaccination during special campaigns? <input type="checkbox"/> Y <input type="checkbox"/> N <input type="checkbox"/> U			
IV. EXPOSURE HISTORY:							
Is there a history of travel to an area with known measles transmission 7-18 days prior to the appearance of rash? <input type="checkbox"/> Y <input type="checkbox"/> N <input type="checkbox"/> U Where did exposure probably occur? <input type="checkbox"/> Day care <input type="checkbox"/> Home/ dormitory <input type="checkbox"/> School <input type="checkbox"/> Health Care Facilities <input type="checkbox"/> Community <input type="checkbox"/> Unknown <input type="checkbox"/> Other, specify _____ Was there contact with a laboratory confirmed Measles case 7-18 days prior to rash onset? <input type="checkbox"/> Y <input type="checkbox"/> N <input type="checkbox"/> U Are there other measles cases in the community? <input type="checkbox"/> Y <input type="checkbox"/> N <input type="checkbox"/> U							
V. LABORATORY TESTS:							
Specimen	Collected?	If YES, date taken	Date sent to RITM	Date received RITM	Measles IgM Result	Rubella IgM Result	Date result
Serum	<input type="checkbox"/> Y <input type="checkbox"/> N	___/___/___	___/___/___	___/___/___			___/___/___
Dried blood	<input type="checkbox"/> Y <input type="checkbox"/> N	___/___/___	___/___/___	___/___/___			___/___/___
NP swab	<input type="checkbox"/> Y <input type="checkbox"/> N	___/___/___	___/___/___	___/___/___			___/___/___
Urine	<input type="checkbox"/> Y <input type="checkbox"/> N	___/___/___	___/___/___	___/___/___			___/___/___
NP aspirate	<input type="checkbox"/> Y <input type="checkbox"/> N	___/___/___	___/___/___	___/___/___			___/___/___
Throat swab	<input type="checkbox"/> Y <input type="checkbox"/> N	___/___/___	___/___/___	___/___/___			___/___/___
VI. CLASSIFICATION AND OUTCOME:							
CASE CLASSIFICATION			OUTCOME			FINAL DIAGNOSIS	
<input type="checkbox"/> Laboratory Confirmed <input type="checkbox"/> Epidemiologically-linked <input type="checkbox"/> Clinically-confirmed <input type="checkbox"/> Discarded Case			<input type="checkbox"/> Alive <input type="checkbox"/> Died Date died: ___/___/___ <input type="checkbox"/> Unknown				

## APPENDIX C: Sample Case Mapping

### Neonatal Tetanus Cases by Province Philippines, 2001



## APPENDIX D: Source codes

### caseMap.css

```
#Manila{
  position: absolute;
  top: 278px;
  right: 455px;
  padding-right: 0px;
  width: 10px;
  height: 5px;
  vertical-align: top;
  text-align: center;

  padding-bottom: 0px;
  z-index: 10;
}

#Pasay{
  position: absolute;
  top: 420px;
  right: 405px;
  padding-right: 0px;
  width: 10px;
  height: 5px;
  vertical-align: top;
  text-align: center;

  padding-bottom: 0px;
  z-index: 10;
}

#Taguig{
  position: absolute;
  top: 465px;
  right: 295px;
  padding-right: 0px;
  width: 10px;
  height: 5px;
  vertical-align: top;
  text-align: center;

  padding-bottom: 0px;
  z-index: 10;
}

#Pateros{
  position: absolute;
  top: 455px;
  right: 215px;
  padding-right: 0px;
  width: 10px;
  height: 5px;
  vertical-align: top;
  text-align: center;

  padding-bottom: 0px;
  z-index: 10;
}

#Paranaque{
  position: absolute;
  top: 512px;
  right: 395px;
  padding-right: 0px;
  width: 10px;
  height: 5px;
  vertical-align: top;
  text-align: center;

  padding-bottom: 0px;
  z-index: 10;
}

padding-bottom: 0px;
z-index: 10;
}

#LasPinas{
  position: absolute;
  top: 600px;
  right: 432px;
  padding-right: 0px;
  width: 10px;
  height: 5px;
  vertical-align: top;
  text-align: center;

  padding-bottom: 0px;
  z-index: 10;
}

#Muntinlupa{
  position: absolute;
  top: 655px;
  right: 358px;
  padding-right: 0px;
  width: 10px;
  height: 5px;
  vertical-align: top;
  text-align: center;

  padding-bottom: 0px;
  z-index: 10;
}

#Mandaluyong{
  position: absolute;
  top: 325px;
  right: 332px;
  padding-right: 0px;
  width: 10px;
  height: 5px;
  vertical-align: top;
  text-align: center;

  padding-bottom: 0px;
  z-index: 10;
}

#Makati{
  position: absolute;
  top: 383px;
  right: 358px;
  padding-right: 0px;
  width: 10px;
  height: 5px;
  vertical-align: top;
  text-align: center;

  padding-bottom: 0px;
  z-index: 10;
}

#Pasig{
  position: absolute;
  top: 328px;
  right: 254px;
  padding-right: 0px;
  width: 10px;
  height: 5px;
  vertical-align: top;
  text-align: center;

  padding-bottom: 0px;
  z-index: 10;
}
```

```

padding-bottom: 0px;
z-index: 10;
}
#SanJuan{
position: absolute;
top:300px;
right:330px;
padding-right: 0px;
width: 10px;
height: 5px;
vertical-align: top;
text-align: center;

padding-bottom: 0px;
z-index: 10;
}
#Marikina{
position: absolute;
top:193px;
right:193px;
padding-right: 0px;
width: 10px;
height: 5px;
vertical-align: top;
text-align: center;

padding-bottom: 0px;
z-index: 10;
}
#Quezon{
position: absolute;
top:1px;
right:346px;
padding-right: 0px;
width: 10px;
height: 5px;
vertical-align: top;
text-align: center;

padding-bottom: 0px;
z-index: 10;
}
#Caloocan{
position: absolute;
top:160px;
right:425px;
padding-right: 0px;
width: 10px;
height: 5px;
vertical-align: top;
text-align: center;

padding-bottom: 0px;
z-index: 10;
}
#Caloocan_North{
position: absolute;
top:-5px;
right:330px;
padding-right: 0px;
width: 10px;
height: 5px;
vertical-align: top;
text-align: center;

padding-bottom: 0px;
z-index: 10;
}
#Malabon{
position: absolute;
top:145px;
right:495px;
padding-right: 0px;
width: 10px;
height: 5px;
vertical-align: top;
text-align: center;

padding-bottom: 0px;
z-index: 10;
}
#Navotas{
position: absolute;
top:138px;
right:532px;
padding-right: 0px;
width: 10px;
height: 5px;
vertical-align: top;
text-align: center;

padding-bottom: 0px;
z-index: 10;
}
#Valenzuela{
position: absolute;
top:58px;
right:475px;
padding-right: 0px;
width: 10px;
height: 5px;
vertical-align: top;
text-align: center;

padding-bottom: 0px;
z-index: 10;
}
#legend{
position: absolute;
top:150px;
right:700px;
padding-right: 0px;
width: 10px;
height: 5px;
vertical-align: top;
text-align: center;
font-family: Verdana;
font-style: italic;
font-size:14px;

padding-bottom: 0px;
z-index: 10;
}
body,td,th {

font-family: Verdana;

font-style: italic;
font-size:18px;
}

```

## addCase.page

```
<com:THead>
<title>Add New Disease</title>
<style>
.addcase
{
    width: 500px;
    height: 350px;
    border: 5px solid ButtonFace;
    overflow: auto;
        background-color: #E6ECFF;
        font-family: Verdana;
        font-size: 10px;
}
.add
{
    width: 600px;
    height: 400px;
    border: 5px solid ButtonFace;
    overflow: auto;
        font-family: Verdana;
        font-size: 12px;
}
</style>
</com:THead>
<com:TForm>
<fieldset class="add">
<legend><font color="blue"><h3>Add New Case
Report</h3></font></legend>
<fieldset class="addcase">
<legend>Fields marked with an asterisk (*) are
required</legend>
<table >
<tr >
<td >
Patient Number:
</td >
<td >
<com:TTextBox ID="patientnumber" />*
<com:TRequiredFieldValidator
    ValidationGroup="Group3"
    Display="Dynamic"
    ControlToValidate="patientnumber"
    FocusOnError="true"
    Text="Field required." />
<com:TCustomValidator
    ControlToValidate="patientnumber"
    Display="Dynamic"
    ValidationGroup="Group3"
    OnServerValidate="checkRecord"
    FocusOnError="true"
    ErrorMessage="The case record already exist." />
</td >
</tr >
<tr >
<td >
Age:
</td >
<td >
<com:TTextBox ID="age" />*

```

```
<com:TRequiredFieldValidator
    ValidationGroup="Group3"
    Display="Dynamic"
    ControlToValidate="age"
    FocusOnError="true"
    Text="Field required." />
<com:TDataTypeValidator
    ValidationGroup="Group3"
    ControlToValidate="age"
    DataType="Integer"
    FocusOnError="true"
    Display="Dynamic"
    Text="You must enter an integer." />
</td >
</tr >
<tr >
<td >
Address:
</td >
<td >
<com:TTextBox ID="address" />
</td >
</tr >
<td >
City:
</td >
<td >
<com:TDropDownList ID="city"
OnSelectedIndexChanged="populateSentinel"
AutoPostBack="true">
    <com:TListItem Value="CITY" Text="CITY" />
    <com:TListItem Value="LP" Text="LAS PINAS" />
    <com:TListItem Value="MNL" Text="MANILA" />
    <com:TListItem Value="PSG" Text="PASIG" />
    <com:TListItem Value="PAS" Text="PASAY" />
    <com:TListItem Value="QUE" Text="QUEZON" />
    <com:TListItem Value="TAG" Text="TAGUIG" />
    <com:TListItem Value="MAL" Text="MALABON" />
    <com:TListItem Value="MAR" Text="MARIKINA" />
    <com:TListItem Value="PAT" Text="PATEROS" />
    <com:TListItem Value="VAL"
Text="VALENZUELA" />
    <com:TListItem Value="MAKATI"
Text="MAKATI" />
    <com:TListItem Value="SAN" Text="SAN JUAN" />
    <com:TListItem Value="NAVOTAS"
Text="NAVOTAS" />
    <com:TListItem Value="PQUE"
Text="PARANAQUE" />
    <com:TListItem Value="MAN"
Text="MANDALUYONG" />
    <com:TListItem Value="MUN"
Text="MUNTINLUPA" />
</com:TDropDownList>*
<com:TRequiredFieldValidator
    ControlToValidate="city"
    ErrorMessage="You must make a selection other than
the first option"
    InitialValue="CITY"
    FocusOnError="true"
    Display="Dynamic"
    ValidationGroup="Group3"
    />
</td >
</tr >
<tr >

```

```

<td>
Sentinel ID:
</td>
<td>
<com:TDropDownList ID="sentinel"
  AutoPostBack="true"
  DataTextField="name"
  DataValueField="id"
  OnSelectedIndexChanged="selectionChanged"
  />*
<com:TRequiredFieldValidator
  ControlToValidate="sentinel"
  ErrorMessage="You must make a selection other than
the first option"
  InitialValue="SENTINEL"
  FocusOnError="true"
  Display="Dynamic"
  ValidationGroup="Group3"
  />
<com:TMultiView ID="MultiView">
<com:TView ID="View1">
<com:TLabel
  Text="Input here:"
  ForeColor="black"
  ID="Label2"
  ForControl="newSentinel"
  />
<com:TTextBox ID="newSentinel" Text=" "
Width="50px" />
</com:TView>
<com:TView ID="View2">
</com:TView>

</com:TMultiView>
</td>
</tr>
<tr>
<td>
Gender:
</td>
<td>
<com:TRadioButton
  ID="Male"
  GroupName="RadioGroup"
  Text="Male"
  />
<com:TRadioButton
  ID="Female"
  GroupName="RadioGroup"
  Text="Female"
  />*
<com:TCustomValidator
  ControlToValidate="Female"
  Display="Dynamic"
  ValidationGroup="Group3"
  OnServerValidate="checkRadio"
  FocusOnError="true"
  ErrorMessage="Field required." />
</td>
</tr>
<tr>
<td>
Disease:
</td>
<td>
<com:TDropDownList ID="disease"
  AutoPostBack="false"

```

```

  DataTextField="name"
  DataValueField="id"
  />*
<com:TRequiredFieldValidator
  ControlToValidate="disease"
  ErrorMessage="You must make a selection other than
the first option"
  InitialValue="DISEASE"
  FocusOnError="true"
  Display="Dynamic"
  ValidationGroup="Group3"
  />
</td>
</tr>
<tr>
<td>
Condition:
</td>
<td>
<com:TDropDownList ID="condition"
AutoPostBack="false">
  <com:TListItem Value="CONDITION"
Text="CONDITION" />
  <com:TListItem Value="INFECTED"
Text="INFECTED" />
  <com:TListItem Value="CURED" Text="CURED" />
  <com:TListItem Value="DECEASED"
Text="DECEASED" />
</com:TDropDownList>*
<com:TRequiredFieldValidator
  ControlToValidate="condition"
  ErrorMessage="You must make a selection other than
the first option"
  InitialValue="CONDITION"
  FocusOnError="true"
  Display="Dynamic"
  ValidationGroup="Group3"
  />
</td>
</tr>
<tr>
<td>
<com:TButton Text="Submit" ID="Submit"
ValidationGroup="Group3" OnClick="saveEntry"/>
</td>
</tr>
</table>
</fieldset>
</fieldset>
</com:TForm>
<com:THyperLink
  NavigateUrl="http://localhost/DiseaseOutbreak/update
Record2.php"
  ImageUrl=~/images/back.png%>
  Text="Back" />.

```

## addCase.php

```

<?php
class addCase extends TPage
{
  public function onLoad($param)
  {
    parent::onLoad($param);
  }
}

```

```

if(!$this->IsPostBack)
{
    $prompt=$ _GET['alert'];
    $sql = "SELECT * FROM `disease`";
";
    $disease=Disease::finder()->findAllBySql($sql);
    $data[0]=array('id'=>'DISEASE','name'=>'DISEASE');
    $index=1;
    foreach($disease as $row){
        $id=$row->Disease_id;
        $name=$row->name;
        $data[$index]=array('id'=>".Sid,'name'=>".
$name);
        $index++;
    }
    $this->disease->DataSource=$data;
    $this->disease->dataBind();
    if($prompt){
        echo
"<SCRIPT>alert(\"You have Successfully Added a New
Record!!\")</SCRIPT>";
    }
    $this->MultiView-
>ActiveView=$this->View2;
}

public function selectionChanged($sender,
$param){
    $indices=$this->sentinel->SelectedIndices;
    foreach($indices as $index){
        $item=$this->sentinel-
>Items[$index];
        $value=$item->Value;
    }

    if($value=="NOT"){
        $this->MultiView-
>ActiveView=$this->View1;
    }else{
        $this->MultiView-
>ActiveView=$this->View2;
    }
}

public function checkRecord($sender, $param)
{
    $id= $this->patientnumber->Text;
    $indices=$this->sentinel->SelectedIndices;
    foreach($indices as $index){
        $item=$this->sentinel-
>Items[$index];
        $sentinel=$item->Value;
    }
    $indices=$this->disease->SelectedIndices;
    foreach($indices as $index){
        $item=$this->disease-
>Items[$index];
        $disease=$item->Value;
    }
    $sql="SELECT * FROM `case_reports`
WHERE `patient_number`='$id' &&
`Sentinel_id`='$sentinel' && `Disease_id`='$disease'";
    $scases=CaseReports::finder()->findBySql($sql);
    $spatNumber=$scases->patient_number;

```

```

if($patNumber==$id){
    $param->IsValid=false;
}else{
    $param->IsValid = true;
}
}

protected function collectSelectionResult($input)
{
    $indices=$input->SelectedIndices;
    foreach($indices as $index){
        $item=$input->Items[$index];
        $value=$item->Value;
    }
    $sql = "SELECT DISTINCT
`Sentinel_id` FROM `case_reports` WHERE
`city`='$value' ";
    $scases=CaseReports::finder()-
>findAllBySql($sql);
    $data[0]=array('id'=>'SENTINEL','name'=>'SENTI
NEL ID');
    $data[1]=array('id'=>'NOT','name'=>'NOT IN THE
LIST');
    $index=2;
    foreach($scases as $row){
        $id=$row->Sentinel_id;
    }
    $data[$index]=array('id'=>".Sid,'name'=>".Sid);
    $index++;
}
    $this->sentinel->DataSource=$data;
    $this->sentinel->dataBind();
}

public function populateSentinel($sender,$param)
{
    $this->collectSelectionResult($sender);
}

public function checkRadio($sender, $param)
{
    if($this->Male->Checked){
        $param->IsValid =true;
    }else if($this->Female-
>Checked){
        $param->IsValid =true;
    }else{
        $param->IsValid =false;
    }
}

protected function saveEntry($sender,$param){
    if($this->IsValid) // when all validations
succeed
    {
        // populates a UserRecord object with user inputs
        $CaseReport=new CaseReports;
        $CaseReport->patient_number=$this-
>patientnumber->Text;
        $CaseReport->age=$this-
>age->Text;
        $CaseReport-
>address=$this->address->Text;
        if($this->Male->Checked){

```



```

        $selection.='M';
    }else{
        $selection.='F';
    }
        $CaseReport->gender=$selection;
        $indices=$this->city-
>SelectedIndices;
        foreach($indices as $index){
            $item=$this->city->Items[$index];
            $value=$item->Value;
        }
        $CaseReport->city=$value;
        $text=$this->newSentinel-
>Text;
        if($text==" "){
            $indices=$this->sentinel-
>SelectedIndices;
            foreach($indices as $index){
                $item=$this->sentinel-
>Items[$index];
                $value=$item->Value;
            }
            }else{
                $value=$text;
            }
            $CaseReport-
>Sentinel_id=$value;
            $indices=$this->disease-
>SelectedIndices;
            foreach($indices as $index){
                $item=$this->disease-
>Items[$index];
                $value=$item->Value;
            }
            $CaseReport->Disease_id=$value;
            $indices=$this->condition-
>SelectedIndices;
            foreach($indices as $index){
                $item=$this->condition-
>Items[$index];
                $value=$item->Value;
            }
            $CaseReport-
>condition=$value;

            $curdate=date("Y-m-d");
            $CaseReport-
>date_reported=$curdate;
            // saves to the database via Active Record
            mechanism
            $CaseReport->save();

            // redirect to same page
            $alert=true;
            $this->Response-
>redirect("index.php?
page=addCase&alert=".urlencode($alert));

        }
    }
}
?>

```

## addDisease.page

```

<head>
<title>Add New Disease</title>
<style>
.adddisease
{
    width: 500px;
    height: 105px;
    border: 1px solid ButtonFace;
    overflow: auto;
    background-color: #E6ECFF;
    font-family: Verdana;
    font-size: 10px;
}

.add
{
    width: 600px;
    height: 350px;
    border: 5px solid ButtonFace;
    overflow: auto;
    font-family: Verdana;
    font-size: 12px;
}
</style>
</head>
<com:TForm>

<fieldset class="add">

<legend><font color="blue"><h2>Add New
Disease:</h2></font></legend>
<fieldset class="adddisease">
<legend><font color="blue"><h4>Please enter the
following information:</h4></font></legend>
<table>
<tr>
<td>
<com:TLabel ForControl="Disease_id" Text="Disease
ID:" />
</td>
<td>
<com:TTextBox ID="Disease_id" />
<com:TRequiredFieldValidator
    ValidationGroup="Group3"
    Display="Dynamic"
    ControlToValidate="Disease_id"
    FocusOnError="true"
    Text="Field required." />
<com:TCustomValidator
    ControlToValidate="Disease_id"
    Display="Dynamic"
    ValidationGroup="Group3"
    OnServerValidate="checkDisease"
    FocusOnError="true"
    ErrorMessage="The disease already exist." />
</td>
</tr>
<tr>
<td>
<com:TLabel ForControl="Disease_name"
Text="Disease Name:" />
</td>
<td>
<com:TTextBox ID="Disease_name" />

```

```

</td>
</tr>
</table>
</fieldset>
<br><br>
<fieldset class="addisease">
<legend><font color="blue"><h4>The values here are
the threshold for a specific
disease:</h4></font></legend>
<table>
<tr>
<td>
<com:TLabel ForControl="rhigh" Text="Risk High:" />
</td>
<td>
<com:TTextBox ID="rhigh" />
<com:TRequiredFieldValidator
  ValidationGroup="Group3"
  Display="Dynamic"
  ControlToValidate="rhigh"
  FocusOnError="true"
  Text="Field required." />
<com:TDataTypeValidator
  ValidationGroup="Group3"
  ControlToValidate="rhigh"
  DataType="Integer"
  FocusOnError="true"
  Display="Dynamic"
  Text="You must enter an integer." />
</td>
</tr>
<tr>
<td>
<com:TLabel ForControl="rave" Text="Risk
Average:" />
</td>
<td>
<com:TTextBox ID="rave" />
<com:TRequiredFieldValidator
  ValidationGroup="Group3"
  Display="Dynamic"
  ControlToValidate="rave"
  FocusOnError="true"
  Text="Field required." />
<com:TDataTypeValidator
  ValidationGroup="Group3"
  ControlToValidate="rave"
  DataType="Integer"
  FocusOnError="true"
  Display="Dynamic"
  Text="You must enter an integer." />
</td>
</tr>
<tr>
<td>
<com:TLabel ForControl="rlow" Text="Risk Low:" />
</td>
<td>
<com:TTextBox ID="rlow" />
<com:TRequiredFieldValidator
  ValidationGroup="Group3"
  Display="Dynamic"
  ControlToValidate="rlow"
  FocusOnError="true"
  Text="Field required." />
<com:TDataTypeValidator
  ValidationGroup="Group3"

```

```

  ControlToValidate="rlow"
  DataType="Integer"
  FocusOnError="true"
  Display="Dynamic"
  Text="You must enter an integer." />
</td></tr>
</table>
</fieldset>
<com:TButton Text="Submit" ID="Submit"
ValidationGroup="Group3" OnClick="saveEntry"/>
</fieldset>
</com:TForm>
<com:THyperLink
  NavigateUrl="http://localhost/joom/updateRecord2.php"
  ImageUrl="~/images/back.png"
  Text="Back" />

```

## addDisease.php

```

<?php
class addDisease extends TPage
{
  public function onLoad($param)
  {
    parent::onLoad($param);
    if(!$this->IsPostBack)
    {
      $prompt=$_GET['alert'];
      if($prompt){
        echo
" <SCRIPT>alert("You have Successfully Added a New
Disease!!")</SCRIPT>";
      }
    }
  }

  public function checkDisease($sender, $param)
  {
    $id= $this->Disease_id->Text;
    $finder=Disease::finder();
    $disease = $finder->findByPk($id);
    $temp=$disease->Disease_id;

    if($id==$temp){
      $param->IsValid=false;
    }else{
      $param->IsValid = true;
    }
  }

  protected function saveEntry($sender,$param){
    if($this->IsValid) // when all validations
succeed
    {
      // populates a Disease object with user inputs
      $Disease=new Disease;
      $Disease->Disease_id=$this->Disease_id->Text;
      $Disease->name=$this-
>Disease_name->Text;

```

```

        $Disease->risk_high=$this->rhigh->Text;
        $Disease->risk_ave=$this->rave->Text;
        $Disease->risk_low=$this->rlow->Text;

        // saves to the database via Active Record mechanism
        $Disease->save();

        // redirects the browser to the same page
        $alert=true;
        $this->Response->redirect("index.php?page=addDisease&alert=".urlencode($alert));
    }
}
?>

```

## caseMap.page

```

<com:THead>
<style>
.selectView
{
    width: 400px;
    height: 100px;
    border: 1px solid ButtonFace;
    overflow: auto;
    background-color: #E6ECFF;
    font-family: Verdana;
    font-size: 12px;
}
.select
{
    width: 400px;
    height: 210px;
    border: 1px solid ButtonFace;
    overflow: auto;
    font-family: Verdana;
    font-size: 12px;
}
</style>
</com:THead>
<com:TForm>
<fieldset class="select">
<legend><h2>Generate Case Map</h2></legend>
<fieldset class="selectView">
<legend><font color="blue">Choose from the following diseases and view the cities colored differently based on their risk-levels.</font></legend><br><br>
<com:TDropDownList ID="disease"
    AutoPostBack="false"
    DataTextField="name"
    DataValueField="id"
/>
<com:TRequiredFieldValidator
    ControlToValidate="disease"
    ErrorMessage="You must make a selection other than the first option"

```

```

        InitialValue="DISEASE"
        FocusOnError="true"
        Display="Dynamic"
        ValidationGroup="casemap"
/>
</fieldset>
<br><br><com:TButton Text="Generate" ID="Create"
ValidationGroup="casemap"
OnClick="generateCasemap"/>
</fieldset>
</com:TForm>

```

## caseMap.php

```

<?php
class caseMap extends TPage
{
    public function onLoad($param)
    {
        if (!$this->IsPostBack){
            $sql = "SELECT * FROM `disease` ";
            $disease=Disease::finder()->findAllBySql($sql);
            $data[0]=array('id'=>'DISEASE','name'=>'DISEASE');
            $index=1;
            foreach($disease as $row){
                $id=$row->Disease_id;
                $name=$row->name;
                $data[$index]=array('id'=>".$id','name'=>".
$name);
                $index++;
            }
            //populates the selection list for
            disease
            $this->disease->DataSource=$data;
            $this->disease->dataBind();
        }

        public function generateCasemap($sender,
        $param){
            if($this->IsValid){
                $indices=$this->disease->SelectedIndices;
                foreach($indices as $index){
                    $item=$this->disease->Items[$index];
                    $disease_id=$item->Value;
                }

                $this->Response->redirect("generateCaseMap.php?
disease_id=".urlencode($disease_id));
            }
        }
    }
}
?>

```

## caseReports.php

```

<?php
/**
 * Auto generated by prado-cli.php on 2009-12-02
 10:03:07.
 */

```

```

class CaseReports extends TActiveRecord
{
    const TABLE='case_reports';

    public $case_number;
    public $patient_number;
    public $age;
    public $address;
    public $city;
    public $gender;
    public $Sentinel_id;
    public $Disease_id;
    public $condition;
    public $date_reported;

    public static function
finder($className=__CLASS__)
    {
        return parent::finder($className);
    }
}
?>

```

## computation.page

```

<com:THead>
<title>Add New Disease</title>
<style>
.selectView
{
    width: 500px;
    height: 300px;
    border: 1px solid ButtonFace;
    overflow: auto;
    background-color: #E6ECFF;
}
.selectVarTab
{
    width: 500px;
    height: 110px;
    border: 1px solid ButtonFace;
    overflow: auto;
    background-color: #E6ECFF;
}
.selectPACF
{
    width: 500px;
    height: 200px;
    border: 1px solid ButtonFace;
    overflow: auto;
    background-color: #E6ECFF;
}
.selectARMA
{
    width: 500px;

```

```

height: 300px;
border: 1px solid ButtonFace;
overflow: auto;
background-color: #E6ECFF;
}
.select
{
    width: 600px;
    height: 400px;
    border: 5px solid ButtonFace;
    font-family: Verdana;
    font-size: 12px;
    overflow: auto;
}
.select2
{
    width: 600px;
    height: 520px;
    border: 5px solid ButtonFace;
    font-family: Verdana;
    font-size: 12px;
    overflow: auto;
}
}
</style>
</com:THead>
<com:TForm>
<com:TMultiView ID="ComputationView">
<com:TView ID="IntroView">
<fieldset class="select">
<legend><h3><center>Generate
Approximation</h3></legend>
<fieldset class="selectView">
<legend><font color="blue">Choose from the following
computations:</font></legend><br><br>
<table align="center">
<tr>
<td>
<com:TImageButton
    imageUrl=<%~ images\compute.jpg %>
    Text="hello world"
    OnClick="directPACF"
/>
</td>
<td>
<com:TLabel Text="Perform PACF Computation" />
</td>
<tr>
<td>
<com:TImageButton
    imageUrl=<%~ images\compute.jpg %>
    Text="hello world"
    OnClick="directVariance"
/>
</td>
<td>
<com:TLabel Text="Generate Variance Reduction
Matrix" />
</td></tr>
<tr>

```

```

</td>
<com:TImageButton
  ImageUrl=<%~ images/status.jpg %>
  Text="hello world"
  OnClick="predictArima"
 />

</td>
<td>
<com:TLabel Text="Generate ARIMA forecast" />
</td></tr>
</table><br><br>
</fieldset>
</fieldset>
</com:TView>
<com:TView ID="VarianceMatrix">
<fieldset class="select">
<legend><h3><center>Generate Variance Reduction
Table</center></h3></legend>
<label>NOTE: This feature computes the Variance
Reduction Matrix that can be used to determine which
combination of seasonal and non-seasonal differencing is
able to reduce the variance most. In most cases, the
combination with the lowest variance corresponds to the
optimal degree of seasonal and non-seasonal differencing
that induces stationarity of the mean.
</label><br>
<fieldset class="selectVarTab">
<legend><font color="blue">Please fill up the necessary
information:</font></legend>
<label>Data to be Used:</label>
<table>
<tr>
<td>
<com:TLabel ForControl="disease1" Text="Disease:" />
</td>
<td>
<com:TDropDownList ID="disease1"
  AutoPostBack="false"
  DataTextField="name"
  DataValueField="id"
 />
<com:TRequiredFieldValidator
  ControlToValidate="disease1"
  ErrorMessage="You must make a selection other than
the first option"
  InitialValue="DISEASE"
  FocusOnError="true"
  Display="Dynamic"
  ValidationGroup="selectVariance"
 />
</td>
</tr>
</table>
<com:TLabel ForControl="Monthly" Text="Time
Unit:" />
</td>
<td>
<com:TRadioButton
  ID="Monthly"
  GroupName="RadioGroup"
 />
      <com:TLabel ForControl="Monthly"
Text="Monthly" />
<com:TRadioButton
  ID="Yearly"
      GroupName="RadioGroup"
 />
<com:TLabel ForControl="Yearly" Text="Yearly" />
<com:TCustomValidator
  ControlToValidate="Yearly"
  Display="Dynamic"
  ValidationGroup="selectVariance"
  OnServerValidate="checkRadio"
  FocusOnError="true"
  ErrorMessage="Please select one from the choices." />
</td>
</tr>
</table>
</fieldset>
<fieldset class="selectVarTab">
<legend>Parameters:</legend>
<table>
<tr>
<td>
<com:TLabel ForControl="period" Text="Seasonal
Period(?)" />
</td>
<td>
<com:TDropDownList ID="period"
  AutoPostBack="false">
  <com:TListItem Value="0" Text="0" />
  <com:TListItem Value="1" Text="1" />
  <com:TListItem Value="2" Text="2" />
  <com:TListItem Value="3" Text="3" />
  <com:TListItem Value="4" Text="4" />
  <com:TListItem Value="5" Text="5" />
  <com:TListItem Value="6" Text="6" />
  <com:TListItem Value="7" Text="7" />
  <com:TListItem Value="8" Text="8" />
  <com:TListItem Value="9" Text="9" />
  <com:TListItem Value="10" Text="10" />
</com:TDropDownList>
</td>
</tr>
</table><br>
<center><com:TButton Text="Generate"
ID="Compute" ValidationGroup="selectVariance"
OnClick="generateVariance"/></center>
</fieldset>
<com:THyperLink
  NavigateUrl="http://localhost/joom/dods/index.php?
page=computation"
  ImageUrl=<%~ images/back.png%>
  Text="Back" />
</fieldset>
</com:TView>
<com:TView ID="PACFView">
<fieldset class="select2">
<legend><h3><center>Partial Autocorrelation Function
Computation</center></h3></legend>
<label>NOTE: This features computes the
autocorrelation and partial autocorrelation function for
any univariate time series. The parameters lambda, d, D,
and seasonality are used to apply a Box-Cox
transformation and (non-)seasonal differencing in order
to induce stationarity of the time series. If lambda = 1, d
= 0, and D = 0 then no transformation/differencing is
applied before the (P)ACF is computed. DEFAULT time
lags= 0.
</label><br>
<fieldset class="selectVarTab">

```

```

<legend><font color="blue">Please fill up the necessary
information:</font></legend>
<label>Data to be Used:</label>
<table>
<tr>
<td>
<com:TLabel ForControl="disease2" Text="Disease:" />
</td>
<td>
<com:TDropDownList ID="disease2"
  AutoPostBack="false"
  DataTextField="name"
  DataValueField="id"
  />
<com:TRequiredFieldValidator
  ControlToValidate="disease2"
  ErrorMessage="You must make a selection other than
the first option"
  InitialValue="DISEASE"
  FocusOnError="true"
  Display="Dynamic"
  ValidationGroup="performPACF"
  />
</td>
</tr>
<tr>
<td>
<com:TLabel ForControl="Monthly2" Text="Time
Unit:" />
</td>
<td>
<com:TRadioButton
  ID="Monthly2"
  GroupName="RadioGroup"
  />
      <com:TLabel ForControl="Monthly2"
Text="Monthly" />
<com:TRadioButton
  ID="Yearly2"
  GroupName="RadioGroup"
  />
<com:TLabel ForControl="Yearly2" Text="Yearly" />
<com:TCustomValidator
  ControlToValidate="Yearly2"
  Display="Dynamic"
  ValidationGroup="performPACF"
  OnServerValidate="checkRadio2"
  FocusOnError="true"
  ErrorMessage="Please select one from the choices." />
</td>
</tr>
</table>
</fieldset>
<fieldset class="selectPACF">
<legend>Parameters:</legend>
<table>
<tr>
<td>
<com:TLabel ForControl="lags" Text="Number of Time
Lags:" />
</td>
<td>
<com:TDropDownList ID="lags"
AutoPostBack="false">
  <com:TListItem Value="Default"
Text="DEFAULT" />
  <com:TListItem Value="5" Text="5" />

```

```

  <com:TListItem Value="6" Text="6" />
  <com:TListItem Value="7" Text="7" />
  <com:TListItem Value="8" Text="8" />
  <com:TListItem Value="9" Text="9" />
  <com:TListItem Value="10" Text="10" />
  <com:TListItem Value="11" Text="11" />
  <com:TListItem Value="12" Text="12" />
  <com:TListItem Value="24" Text="24" />
  <com:TListItem Value="36" Text="36" />
  <com:TListItem Value="48" Text="48" />
  <com:TListItem Value="60" Text="60" />
</com:TDropDownList>
</td>
</tr>
<tr>
<td>
<com:TLabel ForControl="lambda" Text="Box-Cox
lambda transformation parameter (lambda):" />
</td>
<td>
<com:TDropDownList ID="lambda"
AutoPostBack="false">
  <com:TListItem Value="1" Text="1" />
  <com:TListItem Value="-2.0" Text="-2.0" />
  <com:TListItem Value="-1.9" Text="-1.9" />
  <com:TListItem Value="-1.8" Text="-1.8" />
  <com:TListItem Value="-1.7" Text="-1.7" />
  <com:TListItem Value="-1.6" Text="-1.6" />
  <com:TListItem Value="-1.5" Text="-1.5" />
  <com:TListItem Value="-1.4" Text="-1.4" />
  <com:TListItem Value="-1.3" Text="-1.3" />
  <com:TListItem Value="-1.2" Text="-1.2" />
  <com:TListItem Value="-1.1" Text="-1.1" />
  <com:TListItem Value="-1.0" Text="-1.0" />
  <com:TListItem Value="-0.9" Text="-0.9" />
  <com:TListItem Value="-0.8" Text="-0.8" />
  <com:TListItem Value="-0.7" Text="-0.7" />
  <com:TListItem Value="-0.6" Text="-0.6" />
  <com:TListItem Value="-0.5" Text="-0.5" />
  <com:TListItem Value="-0.4" Text="-0.4" />
  <com:TListItem Value="-0.3" Text="-0.3" />
  <com:TListItem Value="-0.2" Text="-0.2" />
  <com:TListItem Value="-0.1" Text="-0.1" />
  <com:TListItem Value="0.0" Text="0.0" />
  <com:TListItem Value="0.1" Text="0.1" />
  <com:TListItem Value="0.2" Text="0.2" />
  <com:TListItem Value="0.3" Text="0.3" />
  <com:TListItem Value="0.4" Text="0.4" />
  <com:TListItem Value="0.5" Text="0.5" />
  <com:TListItem Value="0.6" Text="0.6" />
  <com:TListItem Value="0.7" Text="0.7" />
  <com:TListItem Value="0.8" Text="0.8" />
  <com:TListItem Value="0.9" Text="0.9" />
  <com:TListItem Value="1.1" Text="1.1" />
  <com:TListItem Value="1.2" Text="1.2" />
  <com:TListItem Value="1.3" Text="1.3" />
  <com:TListItem Value="1.4" Text="1.4" />
  <com:TListItem Value="1.5" Text="1.5" />
  <com:TListItem Value="1.6" Text="1.6" />
  <com:TListItem Value="1.7" Text="1.7" />
  <com:TListItem Value="1.8" Text="1.8" />
  <com:TListItem Value="1.9" Text="1.9" />
  <com:TListItem Value="2.0" Text="2.0" />
</com:TDropDownList>
</td>
</tr>
</table>

```

```

<td>
<com:TLabel ForControl="nonseasonal" Text="Degree
of Non-Seasonal Differencing:(d)" />
</td>
<td>
<com:TDropDownList ID="nonseasonal"
AutoPostBack="false">
<com:TListItem Value="0" Text="0" />
<com:TListItem Value="1" Text="1" />
<com:TListItem Value="2" Text="2" />
</com:TDropDownList>
</td>
</tr>
<tr>
<td>
<com:TLabel ForControl="seasonal" Text="Degree of
Seasonal Differencing:(D)" />
</td>
<td>
<com:TDropDownList ID="seasonal"
AutoPostBack="false">
<com:TListItem Value="0" Text="0" />
<com:TListItem Value="1" Text="1" />
</com:TDropDownList>
</td>
</tr>
<tr>
<td>
<com:TLabel ForControl="seasonality"
Text="Seasonality" />
</td>
<td>
<com:TDropDownList ID="seasonality"
AutoPostBack="false">
<com:TListItem Value="12" Text="12" />
<com:TListItem Value="1" Text="1" />
<com:TListItem Value="2" Text="2" />
<com:TListItem Value="3" Text="3" />
<com:TListItem Value="4" Text="4" />
<com:TListItem Value="6" Text="6" />
</com:TDropDownList>
</td>
</tr>
<tr>
<td>
<com:TLabel ForControl="citype" Text="CI Type" />
</td>
<td>
<com:TDropDownList ID="citype"
AutoPostBack="false">
<com:TListItem Value="White Noise" Text="White
Noise" />
<com:TListItem Value="Ma" Text="Ma" />
</com:TDropDownList>
</td>
</tr>
<tr>
<td>
<com:TLabel ForControl="ci" Text="CI Interval:" />
</td>
<td>
<com:TTextBox ID="ci" Text=".95"/>
</td>
</tr>
</table><br>

```

```

<center><com:TButton Text="Generate" ID="Perform"
ValidationGroup="performPACF"
OnClick="generatePACF"/></center>
</fieldset>
<com:THyperLink
NavigateUrl="http://localhost/joom/dods/index.php?
page=computation"
ImageUrl="~/images/back.png%">
Text="Back" />
</fieldset>
</com:TView>
<com:TView ID="Arima">
<fieldset class="select2">
<legend><h3><center>Generate ARIMA
Prediction</center></h3></legend>
<fieldset class="selectVarTab">
<legend><font color="blue">Please fill up the following
information</font></legend>
<label>Data to be Used:</label>
<table>
<tr>
<td>
<com:TLabel ForControl="disease3" Text="Disease"/>
</td>
<td>
<com:TDropDownList ID="disease3"
AutoPostBack="false"
DataTextField="name"
DataValueField="id"
/>
<com:TRequiredFieldValidator
ControlToValidate="disease3"
ErrorMessage="You must make a selection other than
the first option"
InitialValue="DISEASE"
FocusOnError="true"
Display="Dynamic"
ValidationGroup="predictArima"
/>
</td>
</tr>
<tr>
<td>
<com:TLabel ForControl="Monthly3" Text="Time
Unit:" />
</td>
<td>
<com:TRadioButton
ID="Monthly3"
GroupName="RadioGroup"
/>
<com:TLabel ForControl="Monthly3"
Text="Monthly" />
<com:TRadioButton
ID="Yearly3"
GroupName="RadioGroup"
/>
<com:TLabel ForControl="Yearly3" Text="Yearly" />
<com:TCustomValidator
ControlToValidate="Yearly3"
Display="Dynamic"
ValidationGroup="predictArima"
OnServerValidate="checkRadio3"
FocusOnError="true"
ErrorMessage="Please select one from the choices." />
</td>
</tr>
</tr>

```

```

</table>
</fieldset>
<fieldset class="selectARMA">
<legend>Parameters:</legend>
<table>
<tr>
<td>
<com:TLabel ForControl="training" Text="Training
Period(?):" />
</td>
<td>
<com:TDropDownList ID="training"
AutoPostBack="false">
<com:TListItem Value="1" Text="1" />
<com:TListItem Value="2" Text="2" />
<com:TListItem Value="3" Text="3" />
<com:TListItem Value="4" Text="4" />
<com:TListItem Value="5" Text="5" />
<com:TListItem Value="6" Text="6" />
<com:TListItem Value="7" Text="7" />
<com:TListItem Value="8" Text="8" />
<com:TListItem Value="9" Text="9" />
<com:TListItem Value="10" Text="10" />
</com:TDropDownList>
</td>
</tr>
<tr>
<td>
<com:TLabel ForControl="nonseasonal2" Text="Degree
of Non-Seasonal Differencing:(d)" />
</td>
<td>
<com:TDropDownList ID="nonseasonal2"
AutoPostBack="false">
<com:TListItem Value="0" Text="0" />
<com:TListItem Value="1" Text="1" />
<com:TListItem Value="2" Text="2" />
</com:TDropDownList>
</td>
</tr>
<tr>
<td>
<com:TLabel ForControl="seasonal2" Text="Degree of
Seasonal Differencing:(D)" />
</td>
<td>
<com:TDropDownList ID="seasonal2"
AutoPostBack="false">
<com:TListItem Value="0" Text="0" />
<com:TListItem Value="1" Text="1" />
</com:TDropDownList>
</td>
</tr>
<tr>
<td>
<com:TLabel ForControl="seasonalperiod"
Text="Seasonal Period (s)" />
</td>
<td>
<com:TDropDownList ID="seasonalperiod"
AutoPostBack="false">
<com:TListItem Value="1" Text="1" />
<com:TListItem Value="2" Text="2" />
<com:TListItem Value="3" Text="3" />
<com:TListItem Value="4" Text="4" />
<com:TListItem Value="6" Text="6" />
<com:TListItem Value="12" Text="12" />
</com:TDropDownList>
</td>
</tr>
</table>
</fieldset>
</td>
</tr>
<tr>
<td>
<com:TDropDownList ID="ar" AutoPostBack="false">
<com:TListItem Value="0" Text="0" />
<com:TListItem Value="1" Text="1" />
<com:TListItem Value="2" Text="2" />
<com:TListItem Value="3" Text="3" />
</com:TDropDownList>
</td>
<tr>
<td>
<com:TLabel ForControl="ma" Text="MA (q) order" />
</td>
<td>
<com:TDropDownList ID="ma" AutoPostBack="false">
<com:TListItem Value="0" Text="0" />
<com:TListItem Value="1" Text="1" />
<com:TListItem Value="2" Text="2" />
</com:TDropDownList>
</td>
</tr>
<tr>
<td>
<com:TLabel ForControl="sar" Text="SAR (P) order" />
</td>
<td>
<com:TDropDownList ID="sar" AutoPostBack="false">
<com:TListItem Value="0" Text="0" />
<com:TListItem Value="1" Text="1" />
<com:TListItem Value="2" Text="2" />
<com:TListItem Value="3" Text="3" />
</com:TDropDownList>
</td>
</tr>
<tr>
<td>
<com:TLabel ForControl="sma" Text="SMA(Q)
order" />
</td>
<td>
<com:TDropDownList ID="sma"
AutoPostBack="false">
<com:TListItem Value="0" Text="0" />
<com:TListItem Value="1" Text="1" />
</com:TDropDownList>
</td>
</tr>
<tr>
<td>
<com:TLabel ForControl="mean" Text="Include
Mean?" />
</td>
<td>
<com:TDropDownList ID="mean"
AutoPostBack="false">
<com:TListItem Value="FALSE" Text="FALSE" />
<com:TListItem Value="TRUE" Text="TRUE" />
</com:TDropDownList>
</td>
</tr>
</table>

```



```

<tr>
<td>
<com:TLabel Text="Show Solutions:"
ForControl="showComputation" />
</td>
<td>
<com:TCheckBox
  AutoPostBack="false"
  ID="showComputation"
  />
</td>
</tr>
</table>
<center><com:TButton Text="Predict" ID="Predict"
ValidationGroup="predictArima"
OnClick="generateArima"/></center>
</fieldset>
<com:THyperLink
  NavigateUrl="http://localhost/joom/dods/index.php?
page=computation"
  ImageUrl=~/images/back.png%>
  Text="Back" />.
</fieldset>
</com:TView>
</com:TMultiView>
</com:TForm>

```

## computation.php

```

<?php
class computation extends TPage
{
  public function onLoad($param)
  {
    if (!$this->IsPostBack)
      $this->ComputationView-
>ActiveView=$this->IntroView;
      $sql = "SELECT * FROM
`disease` ";
      $disease=Disease::finder()->findAllBySql($sql);
      $data[0]=array('id'=>'DISEASE','name'=>'DISEA
SE');
      $index=1;
      foreach($disease as $row){
        $id=$row->Disease_id;
        $name=$row->name;
        $data[$index]=array('id'=>".$id,'name'=>".
$name);
        $index++;
      }
      $this->disease1->DataSource=$data;
      $this->disease1->dataBind();
      $this->disease2->DataSource=$data;
      $this->disease2->dataBind();
      $this->disease3->DataSource=$data;
      $this->disease3->dataBind();
    }

    public function checkRadio($sender, $param)
    {
      if($this->Yearly->Checked){
        $param->IsValid =true;

```

```

      }else if($this->Monthly-
>Checked){
        $param->IsValid =true;
      }else{
        $param->IsValid =false;
      }
    }
    public function
checkRadio2($sender, $param)
    {
      if($this->Yearly2->Checked){
        $param->IsValid =true;
      }else if($this->Monthly2-
>Checked){
        $param->IsValid =true;
      }else{
        $param->IsValid =false;
      }
    }
    public function
checkRadio3($sender, $param)
    {
      if($this->Yearly3->Checked){
        $param->IsValid =true;
      }else if($this->Monthly3-
>Checked){
        $param->IsValid =true;
      }else{
        $param->IsValid =false;
      }
    }
    public function directVariance($sender,
$param)
    {
      $this->ComputationView->ActiveView=$this-
>VarianceMatrix;
    }
    public function predictArima($sender,$param)
    {
      $this->ComputationView->ActiveView=$this-
>Arima;
    }
    public function directPACF($sender,$param)
    {
      $this->ComputationView->ActiveView=$this-
>PACFView;
    }
    public function generateVariance($sender,
$param)
    {
      if($this->IsValid){
        if($this->Yearly->Checked){
          $time="yearly";
        }else{

```

```

$time="monthly";
    }
    $indices=$this->disease1-
>SelectedIndices;
    foreach($indices as $index){
        $item=$this->disease1-
>Items[$index];
        $disease_id=$item->Value;
    }
    $indices=$this->period-
>SelectedIndices;
    foreach($indices as $index){
        $item=$this->period-
>Items[$index];
        $seasonality=$item->Value;
    }
    $this->Response-
>redirect("performVarMat.php?
unit=".urlencode($time)."&disease_id=".urlencode($dise
ase_id)."&par1=".urlencode($seasonality));
    }
}

public function generatePACF($sender,
$param)
{
    if($this->IsValid){
        if($this->Yearly2->Checked){
            $time="yearly";
        } else {
            $time="monthly";
        }
    }

    $indices=$this->disease2-
>SelectedIndices;
    foreach($indices as $index){
        $item=$this->disease2-
>Items[$index];
        $disease_id=$item->Value;
    }

    $indices=$this->lags-
>SelectedIndices;
    foreach($indices as $index){
        $item=$this->lags->Items[$index];
        $lag=$item->Value;
    }

    $indices=$this->nonseasonal-
>SelectedIndices;
    foreach($indices as $index){
        $item=$this->nonseasonal-
>Items[$index];
        $nondegree=$item->Value;
    }

    $indices=$this->seasonal-
>SelectedIndices;
    foreach($indices as $index){
        $item=$this->seasonal-
>Items[$index];
        $seasonaldegree=$item->Value;
    }

```

```

$indices=$this->seasonality-
>SelectedIndices;
    foreach($indices as $index){
        $item=$this->seasonality-
>Items[$index];
        $seasonality=$item->Value;
    }

    $indices=$this->citype-
>SelectedIndices;
    foreach($indices as $index){
        $item=$this->citype-
>Items[$index];
        $citype=$item->Value;
    }

    $indices=$this->lambda-
>SelectedIndices;
    foreach($indices as $index){
        $item=$this->lambda-
>Items[$index];
        $lambda=$item->Value;
    }

    $ci=$this->ci->Text;
    $this->Response-
>redirect("performPACF.php?
unit=".urlencode($time)."&disease_id=".urlencode($dise
ase_id)."&par1=".urlencode($lag)."&par2=".urlencode($
lambda)."&par3=".urlencode($nondegree)."&par4=".urle
ncode($seasonaldegree)."&par5=".urlencode($seasonalit
y)."&par6=".urlencode($citype)."&par7=".urlencode($ci
));
    }
}

public function generateArima($sender,
$param)
{
    if($this->IsValid){
        if($this->Yearly3->Checked){
            $time="yearly";
        } else {
            $time="monthly";
        }
    }

    $indices=$this->disease3-
>SelectedIndices;
    foreach($indices as $index){
        $item=$this->disease3-
>Items[$index];
        $disease_id=$item->Value;
    }

    $indices=$this->training-
>SelectedIndices;
    foreach($indices as $index){
        $item=$this->training-
>Items[$index];
        $training=$item->Value;
    }

    $indices=$this->nonseasonal2-
>SelectedIndices;
    foreach($indices as $index){

```

```

        $item=$this->nonseasonal2-
>Items[$index];
        $nondegree=$item->Value;
    }

    $indices=$this->seasonal2-
>SelectedIndices;
    foreach($indices as $index){
        $item=$this->seasonal2-
>Items[$index];
        $seasonaldegree=$item->Value;
    }

    $indices=$this->seasonalperiod-
>SelectedIndices;
    foreach($indices as $index){
        $item=$this->seasonalperiod-
>Items[$index];
        $seasonalperiod=$item->Value;
    }

    $indices=$this->ar->SelectedIndices;
    foreach($indices as $index){
        $item=$this->ar->Items[$index];
        $sar=$item->Value;
    }

    $indices=$this->ma-
>SelectedIndices;
    foreach($indices as $index){
        $item=$this->ma->Items[$index];
        $sma=$item->Value;
    }

    $indices=$this->sar-
>SelectedIndices;
    foreach($indices as $index){
        $item=$this->sar->Items[$index];
        $sar=$item->Value;
    }

    $indices=$this->sma-
>SelectedIndices;
    foreach($indices as $index){
        $item=$this->sma->Items[$index];
        $sma=$item->Value;
    }

    $indices=$this->mean-
>SelectedIndices;
    foreach($indices as $index){
        $item=$this->mean->Items[$index];
        $mean=$item->Value;
    }

    if($this->showComputation-
>Checked){
        $show=true;
    }else{
        $show=false;
    }

    $this->Response-
>redirect("predictARIMA.php?
unit=".urlencode($time)."&disease_id=".urlencode($disease_id)."&par1=".urlencode($training)."&par3=".urlencode($nondegree)."&par4=".urlencode($seasonaldegree)."

```

```

&par5=".urlencode($seasonalperiod)."&par6=".urlencode($ar)."&par7=".urlencode($sma)."&par8=".urlencode($sar)."&par9=".urlencode($sma)."&par10=".urlencode($mean)."&showSolutions=".urlencode($show));

```

```

    }
}
}
?>

```

## connect.php

```

<?
$gdods_db_link = mysql_connect ("localhost", "richard", "buendia");

if (!$gdods_db_link)
{
    echo "Error connecting to the database!<br>" .
mysql_error();
    $error_msg = "[ $gNow_mysql ]
[ $PHP_SELF ]\n\t$query\n\t" . mysql_error();
    exit(1);
}

if (!mysql_select_db ("dods",$gdods_db_link))
{
    echo "Error selecting the database!<br>" .
mysql_error();
    $error_msg = "[ $gNow_mysql ]
[ $PHP_SELF ]\n\t$query\n\t" . mysql_error();
    exit(1);
}

?>

```

## Disease.php

```

<?php
/**
 * Auto generated by prado-cli.php on 2009-12-02
 10:07:20.
 */
class Disease extends TActiveRecord
{
    const TABLE='disease';

    public $Disease_id;
    public $name;
    public $total;
    public $risk_high;
    public $risk_ave;
    public $risk_low;

    public static function
finder($className=__CLASS__)
    {
        return parent::finder($className);
    }
}
?>

```

## export.php

```
<?php
require('fpdf/fpdf.php');

class PDF extends FPDF
{
//Load data
function LoadData($file,$pdf,$header)
{
//Read file lines

$lines=file($file);
$ddata=array();
$index=0;
foreach($lines as $line)

if($index==0){

$ddata[0]=explode(':',chop($line));
$temp[$index]=$ddata[0];

//Column widths
$w=array(40,25,25,35,20,20,25);
//Header

for($i=0;$i<count($header);$i++)

$index++;
} else{

$ddata[$index]=explode(':',chop($line));
$temp[$index]=$ddata[$index];
$index++;
}

return $temp;
}

function ImprovedTable2($header,$data,$stringDisease,
$temp,$temp2,$temp3)
{
$this->Cell(190,7,"COMPARATIVE Statistics, by
Sociodemographic Category",1,1,'C');
$this->Cell(190,7,$stringDisease,1,1,'C');
//Column widths
$w=array(35,25,25,35,30,20,20);
//Header

for($i=0;$i<count($header);$i++)

$this-
>Cell($w[$i],7,$header[$i],1,0,'C');
$this->Ln();
//Data
$index=0;
foreach($data as $row)
//$this->Cell(0,6,"",1,1,'C');
if($index==0){
```

```
$this->Cell($w[0],6,$row[0],1,0,'C');
$this->Cell($w[1],6,$row[1],1,0,'C');
$this->Cell($w[2],6,$row[2],1,0,'C');
$this->Cell($w[3],6,$row[3],1,0,'C');
$this->Cell($w[4],6,
($row[4],1,0,'C');
$this->Cell($w[5],6,$row[5],1,0,'C');
$this->Cell($w[6],6,$row[6],1,0,'C');

$index++;
} else {
$this->Cell(0,6,"",0,1,'C');
$this->Cell($w[0],6,$row[0],1,0,'C');
$this->Cell($w[1],6,$row[1],1,0,'C');
$this->Cell($w[2],6,$row[2],1,0,'C');
$this->Cell($w[3],6,$row[3],1,0,'C');
$this->Cell($w[4],6,
($row[4],1,0,'C');
$this->Cell($w[5],6,$row[5],1,0,'C');
$this->Cell($w[6],6,$row[6],1,0,'C');
$index++;
}

//Closure line
$this->Cell(10,6,"",0,1,'C');
$this->Cell(190,5,$temp,0,1,'L');
$this->Cell(190,5,$temp2,0,1,'L');
$this->Cell(190,5,$temp3,0,1,'L');
$this->Cell(array_sum($w),0,"",1,'T');
}
}

function exportPDF($stringDisease,$temp,
$temp2,$temp3){

$pdf=new FPDF();

$pdf=new PDF();
//Column titles
$header=array('Category','CaseCount','% Total','5-
yearMeridian','% of Change','Deaths','CFR');
$pdf->SetFont('Arial','',12);
$pdf->AddPage();
//Data loading
$ddata=$pdf->LoadData('fpdf/annualReport.txt',$pdf,
$header);
$pdf->ImprovedTable2($header,$data,$stringDisease,
$temp,$temp2,$temp3);

$pdf->Output();
}
?>
```

## generateCaseMap.php

```
<html>
<head>
<title>Case Mapping</title>
<link href="caseMap.css" rel="stylesheet"
type="text/css">
</head>
<body>
```

```

<?php
require_once('getInfectedCount.php');
require_once('connect.php');
$disease_id=$_GET['disease_id'];
$query=mysql_query("SELECT name FROM `disease`
WHERE `Disease_id`='$disease_id'");
$result=mysql_fetch_array($query);

echo "<div id='legend'>";
echo "<h5>Disease:</h5> $result[0]<br>";
echo "<h5>Legend:</h5>";
echo "<font color='red'>RED</font>:High Risk<br>";
echo "<font color='green'>GREEN</font>:Average
Risk<br>";
echo "<font color='yellow'>YELLOW</font>:Low
Risk";
echo "</div>";
$risk=getInfectedCount('MNL', $disease_id);
$url="maps/MAN_.$risk.png";

echo "<div id='Manila'><img src='$url'></div>";
$risk=getInfectedCount('PAS', $disease_id);
$url="maps/PAS_.$risk.png";
echo "<div id='Pasay'><img src='$url'></div>";
$risk=getInfectedCount('TAG', $disease_id);
$url="maps/TAG_.$risk.png";
echo "<div id='Taguig'><img src='$url'></div>";
$risk=getInfectedCount('MND', $disease_id);
$url="maps/MND_.$risk.png";
echo "<div id='Mandaluyong'><img src='$url'></div>";
$risk=getInfectedCount('PQUE', $disease_id);
$url="maps/PAR_.$risk.png";
echo "<div id='Paranaque'><img src='$url'></div>";
$risk=getInfectedCount('LP', $disease_id);
$url="maps/LAS_.$risk.png";
echo "<div id='LasPinas'><img src='$url'></div>";
$risk=getInfectedCount('MUN', $disease_id);
$url="maps/MUN_.$risk.png";
echo "<div id='Muntinlupa'><img src='$url'></div>";
$risk=getInfectedCount('MAN', $disease_id);
$url="maps/MND_.$risk.png";
echo "<div id='Mandaluyong'><img src='$url'></div>";
$risk=getInfectedCount('PSG', $disease_id);
$url="maps/PSG_.$risk.png";
echo "<div id='Pasig'><img src='$url'></div>";
$risk=getInfectedCount('MAR', $disease_id);
$url="maps/MAR_.$risk.png";
echo "<div id='Marikina'><img src='$url'></div>";
$risk=getInfectedCount('QUE', $disease_id);
$url="maps/QUE_.$risk.png";
echo "<div id='Quezon'><img src='$url'></div>";
$risk=getInfectedCount('SAN', $disease_id);
$url="maps/SAN_.$risk.png";
echo "<div id='SanJuan'><img src='$url'></div>";
$risk=getInfectedCount('CAL', $disease_id);
$url="maps/CAL_.$risk.png";
echo "<div id='Caloocan'><img src='$url'></div>";
$url="maps/CALN_.$risk.png";
echo "<div id='Caloocan_North'><img
src='$url'></div>";
$risk=getInfectedCount('MAL', $disease_id);
$url="maps/MAL_.$risk.png";
echo "<div id='Malabon'><img src='$url'></div>";
$risk=getInfectedCount('NAV', $disease_id);
$url="maps/NAV_.$risk.png";
echo "<div id='Navotas'><img src='$url'></div>";
$risk=getInfectedCount('VAL', $disease_id);

```

```

$url="maps/VAL_.$risk.png";
echo "<div id='Valenzuela'><img src='$url'></div>";
$risk=getInfectedCount('MAK', $disease_id);
$url="maps/MAK_.$risk.png";
echo "<div id='Makati'><img src='$url'></div>";
$risk=getInfectedCount('PAT', $disease_id);
$url="maps/PAT_.$risk.png";
echo "<div id='Pateros'><img src='$url'></div>";
echo "<br><br><a href='index.php?
page=caseMap'><img src='back.png'></a>";
?>
</body>
</html>

```

## generateStatus.php

```

<?php
ini_set( "display_errors", 0);
require_once('getCount.php');
require_once('getMeridian.php');
require_once('exportPDF.php');
require_once('connect.php');
$status=$_GET['status'];
if($status=='report'){
/*****begin of annual report generation*****/
set_time_limit(200);
$disease_id=$_GET['disease_id'];
$year=$_GET['year'];
$myFile = "fpdf/annualReport.txt";
$fh = fopen($myFile, 'w') or die("can't open file");
$query=mysql_query("SELECT name FROM `disease`
WHERE `Disease_id`='$disease_id'");
$result=mysql_fetch_array($query);
$disease=$result[0];
$stringDisease="$disease,January-December $year";
$temp=getGenderCount($disease_id,$year,$fh,$disease);
$temp2=getAgeCount($disease_id,$year,$fh);
//summary report that will be created below the table
$temp3=getCityCount($disease_id,$year,$fh);
fclose($fh);
exportPDF($stringDisease,$temp,$temp2,$temp3);
/*****end of annual report generation *****/
echo "<br><br><a href='statusIndicators.php'><img
src='back.png'>&nbsp;&nbsp;&nbsp;<font color='black' size=3
face='Courier New'>BACK</font></a>";
} else {
/*****here is where the status graphs are
generated*****/
set_time_limit(120);
$graphtype=$_GET['graph'];
$disease_id=$_GET['disease_id'];
$curyear=$_GET['year'];
$query=mysql_query("SELECT DISTINCT
year( `date_reported` ), month( `date_reported` ) FROM
`case_reports` WHERE `Disease_id`='$disease_id'
ORDER BY `date_reported` ASC ");
$stringData="Date,Cases\r\n";
$myFile = "R-2.10.0/bin/testFile.txt";
$fh = fopen($myFile, 'w') or die("can't open file");
fwrite($fh, $stringData);
while($date = mysql_fetch_array($query)){

```

```

$year=$date[0];
if($year==$curyear){
$month=$date[1];
$count=0;
$query2=mysql_query("SELECT * FROM
`case_reports` WHERE `Disease_id`='$disease_id'");
while($row = mysql_fetch_array($query2)){
    $recordYear=$row['date_reported'];
    $query3=mysql_query("SELECT
year('$recordYear'),month('$recordYear') FROM
`case_reports`");
    $date2=mysql_fetch_array($query3);
    $year2=$date2[0];
    $month2=$date2[1];
    if($year==$year2&&$month==$month2){
        $count++;
    }
}
$stringData="$year-$month,$count\r\n";
fwrite($fh, $stringData);
}

}
fclose($fh);
$queryDisease=mysql_query("SELECT name FROM
`disease` WHERE `Disease_id`='$disease_id'");
$resultDisease=mysql_fetch_array($queryDisease);
$diseaseName=$resultDisease[0];

$myFile = "R-2.10.0/bin/graphCommand.txt";
$fh = fopen($myFile, 'w') or die("can't open file");
fwrite($fh,
"file=read.table('testFile.txt',header=T,sep=',')\r\n");
if($graphtype=="line"){
fwrite($fh, "max_y<-max(file"."$"."Cases)\r\n");
fwrite($fh, "png('plot.png')\r\n");
fwrite($fh,
"plot(file"."$"."Cases,type='o',col='red',axes=FALSE,ann
=FALSE)\r\n");
fwrite($fh, "title(main='Infected Cases of $diseaseName
of Year $curyear,col.main='blue',font.main=4)\r\n");
fwrite($fh,
"title(ylab='Total',col.lab=rgb(0.5,0,0.5))\r\n");
fwrite($fh,
"title(xlab='Months',col.lab=rgb(0.5,0,0.5))\r\n");
fwrite($fh,
"axis(1,at=1:12,lab=c('JAN','FEB','MAR','APR','MAY','J
UN','JUL','AUG','SEP','OCT','NOV','DEC'))\r\n");
fwrite($fh, "axis(2,las=1,at=5*0:max_y)\r\n");
fwrite($fh, "box()\r\n");
fwrite($fh, "legend(1,max_y,c('Monthly
rate'),cex=0.8,col=c('red'),pch=(21:22),lty=1:2,bty='n')\r
n");
} else if($graphtype=="log"){
fwrite($fh, "png('plot.png')\r\n");
fwrite($fh, "h <- hist(file"."$"."Cases, plot=F,
breaks=c(seq(0,max(file"."$"."Cases)+1, .1))\r\n");
fwrite($fh, "plot(h$counts, log='xy', pch=20,
col='blue',main='Log-normal distribution of
$diseaseName of Year $curyear,xlab='Value',
ylab='Frequency')\r\n");
} else if($graphtype=="bar"){
fwrite($fh, "png('plot.png')\r\n");
fwrite($fh, "barplot(file"."$"."Cases,border='blue',
density=c(10,20,30,40,50,60,70,80,90,100,110,120))\r\n"
);
}

```

```

fwrite($fh, "title(main='Infected Cases of $diseaseName
of Year $curyear,col.main='blue',font.main=4)\r\n");
fwrite($fh,
"title(ylab='Total',col.lab=rgb(0.5,0,0.5))\r\n");
fwrite($fh,
"title(xlab='Months',col.lab=rgb(0.5,0,0.5))\r\n");
fwrite($fh,
"axis(1,at=1:12,lab=c('JAN','FEB','MAR','APR','MAY','J
UN','JUL','AUG','SEP','OCT','NOV','DEC'))\r\n");
} else {
fwrite($fh, "png('plot.png')\r\n");
fwrite($fh, "max_x<-max(file"."$"."Cases)\r\n");
fwrite($fh, "brk <-
c(0,10,20,30,40,50,60,70,80,90,100)\r\n");
fwrite($fh, "hist(file"."$"."Cases,
col=heat.colors(length(brk)),
breaks=brk,xlim=c(0,max_x), right=F, main='Probability
Density for $diseaseName of Year $curyear',las=1,
cex.axis=0.8, freq=F,xlab='Cases',ylab='Density')\r\n");
}
fwrite($fh, "dev.off()\r\n");
fclose($fh);
chdir('C:\xampp\htdocs\joom\dods\R-2.10.0\bin');
exec("R CMD BATCH graphCommand.txt");
chdir('C:\xampp\htdocs\joom\dods');
echo "<img src='R-2.10.0/bin/plot.png' ><br><br>";
echo "<br><br><a
href='http://localhost/joom/dods/index.php?
page=statusIndicators'><img
src='/protected/pages/images/back.png'></a>";
}
?>

```

## getCount.php

```

<?
function getGenderCount($disease_id,$year,$fh,
$disease){
$sql=mysql_query("SELECT * FROM `case_reports`
WHERE `Disease_id`='$disease_id'");
$total=0;
$totalMale=0;
$totalFemale=0;
$totalDeathMale=0;
$totalDeathFemale=0;

while($query=mysql_fetch_array($sql)){
    $recordYear=$query['date_reported'];
    $query3=mysql_query("SELECT year('$recordYear')
FROM `case_reports`");
    $date2=mysql_fetch_array($query3);
    $year2=$date2[0];
    if($year2==$year){
        $total++;
        if($query['gender']=='M'){
            $totalMale++;
        }
    }
} else {
    $totalFemale++;
}

if($query['condition']=='DECEASED'){
    $totalDeathMale++;
}

if($query['condition']=='DECEASED'){
    $totalDeathFemale++;
}
}

```

```

    }
}
}
$meridianMale=getGenderMeridian('M',$year,
$disease_id);
$meridianFemale=getGenderMeridian('F',$year,
$disease_id);
$percentChangeMale=round(((($totalMale-
$meridianMale)/$meridianMale)*100,2);
$percentChangeFemale=round(((($totalFemale-
$meridianFemale)/$meridianFemale)*100,2);

$CFRMale=round((($totalDeathMale/$totalMale)*100,2);
$CFRFemale=round((($totalDeathFemale/
$totalFemale)*100,2);
$percentMale=round((($totalMale/$total)*100,2);
$percentFemale=round((($totalFemale/$total)*100,2);
$stringData="SEX;;;;;\r\n";
fwrite($fh, $stringData);
$stringData="MALE;$totalMale;$percentMale;
$meridianMale;$percentChangeMale;$totalDeathMale;
$CFRMale\r\n";
fwrite($fh, $stringData);
$stringData="FEMALE;$totalFemale;$percentFemale;
$meridianFemale;$percentChangeFemale;
$totalDeathFemale;$CFRFemale\r\n";
fwrite($fh, $stringData);
//
$stringData="_____
\r\n";
//fwrite($fh, $stringData);

if($totalMale>$totalFemale){
$result1=$percentMale;
$result2="Male";
}else{
$result1=$percentFemale;
$result2="Female";
}

$stringData="$total cases of $disease were reported in
$year.Majority ($result1%) of the cases were $result2.
\r\n";
return $stringData;
}

function getAgeCount($disease_id,$year,$fh){

$sql=mysql_query("SELECT * FROM `case_reports`
WHERE `Disease_id`='$disease_id'");
$total=0;
$totalAge1=0;
$totalAge2=0;
$totalAge3=0;
$totalAge4=0;
$totalAge5=0;
$totalAge6=0;
$totalAge7=0;

$totalDeathAge1=0;
$totalDeathAge2=0;
$totalDeathAge3=0;
$totalDeathAge4=0;
$totalDeathAge5=0;

```

```

$totalDeathAge6=0;
$totalDeathAge7=0;
$MaxTotal=0;
$MaxGroup="";
while($query=mysql_fetch_array($sql)){
$recordYear=$query['date_reported'];
$query3=mysql_query("SELECT year('$recordYear')
FROM `case_reports`");
$date2=mysql_fetch_array($query3);
$year2=$date2[0];
if($year2==$year){
$total++;
$age=$query['age'];
if($age<1){
$totalAge1++;
if($totalAge1>$MaxTotal){
$MaxTotal=$totalAge1;
$MaxGroup=" <1 ";
}
}
if($query['condition']=='DECEASED'){
$totalDeathAge1++;
}
}else if($age<=4){
$totalAge2++;
if($totalAge2>$MaxTotal)
{
$MaxTotal=$totalAge2;
$MaxGroup=" 1 to 4 ";
}
}
if($query['condition']=='DECEASED'){
$totalDeathAge2++;
}
}else if($age<=14){
$totalAge3++;
if($totalAge3>$MaxTotal)
{
$MaxTotal=$totalAge3;
$MaxGroup=" 5 to 14 ";
}
}
if($query['condition']=='DECEASED'){
$totalDeathAge3++;
}
}else if($age<=24){
$totalAge4++;
}
if($totalAge4>$MaxTotal){
$MaxTotal=$totalAge4;
$MaxGroup=" 15 to 24 ";
}
}
if($query['condition']=='DECEASED'){
$totalDeathAge4++;
}
}else if($age<=39){
$totalAge5++;
if($totalAge5>$MaxTotal)
{
$MaxTotal=$totalAge5;
$MaxGroup=" 25 to 39 ";
}
}
if($query['condition']=='DECEASED'){
$totalDeathAge5++;
}
}
}

```

```

    }else if($age<=64){
        $totalAge6++;
    }
    if($totalAge6>$MaxTotal){
        $MaxTotal=$totalAge6;
        $MaxGroup=" 40 to 64 ";
    }
    if($query['condition']!="DECEASED"){
        $totalDeathAge6++;
    }
    }else{
        $totalAge7++;
    }
    if($totalAge7>$MaxTotal){
        $MaxTotal=$totalAge7;
        $MaxGroup=" 65 and up ";
    }
    if($query['condition']!="DECEASED"){
        $totalDeathAge7++;
    }
    }
}
}
$percentAge1=round(($totalAge1/$total)*100,2);
$percentAge2=round(($totalAge2/$total)*100,2);
$percentAge3=round(($totalAge3/$total)*100,2);
$percentAge4=round(($totalAge4/$total)*100,2);
$percentAge5=round(($totalAge5/$total)*100,2);
$percentAge6=round(($totalAge6/$total)*100,2);
$percentAge7=round(($totalAge7/$total)*100,2);
$CFRAge1=round(($totalDeathAge1/$totalAge1)*100,2);
;
$CFRAge2=round(($totalDeathAge2/$totalAge2)*100,2);
;
$CFRAge3=round(($totalDeathAge3/$totalAge3)*100,2);
;
$CFRAge4=round(($totalDeathAge4/$totalAge4)*100,2);
;
$CFRAge5=round(($totalDeathAge5/$totalAge5)*100,2);
;
$CFRAge6=round(($totalDeathAge6/$totalAge6)*100,2);
;
$CFRAge7=round(($totalDeathAge7/$totalAge7)*100,2);
;
$percentMax=round(($MaxTotal/$total)*100,2);
$meridianAge1=getAgeMeridian(0,1,$year,$disease_id);
$meridianAge2=getAgeMeridian(1,5,$year,$disease_id);
$meridianAge3=getAgeMeridian(5,15,$year,
$disease_id);
$meridianAge4=getAgeMeridian(15,25,$year,
$disease_id);
$meridianAge5=getAgeMeridian(25,40,$year,
$disease_id);
$meridianAge6=getAgeMeridian(40,65,$year,
$disease_id);
$meridianAge7=getAgeMeridian(65,200,$year,
$disease_id);
$percentMeridianAge1=round(((($totalAge1-
$meridianAge1)/$meridianAge1)*100,2);
$percentMeridianAge2=round(((($totalAge2-
$meridianAge2)/$meridianAge2)*100,2);
$percentMeridianAge3=round(((($totalAge3-
$meridianAge3)/$meridianAge3)*100,2);

```

```

$percentMeridianAge4=round(((($totalAge4-
$meridianAge4)/$meridianAge4)*100,2);
$percentMeridianAge5=round(((($totalAge5-
$meridianAge5)/$meridianAge5)*100,2);
$percentMeridianAge6=round(((($totalAge6-
$meridianAge6)/$meridianAge6)*100,2);
$percentMeridianAge7=round(((($totalAge7-
$meridianAge7)/$meridianAge7)*100,2);

$stringData="AGE GROUP;;;;;\r\n";
fwrite($fh,$stringData);
$stringData="<1;$totalAge1;$percentAge1;$meridianAge1;$percentMeridianAge1;$totalDeathAge1;$CFRAge1\r\n";
fwrite($fh,$stringData);
$stringData="1 to
4;$totalAge2;$percentAge2;$meridianAge2;$percentMeridianAge2;$totalDeathAge2;$CFRAge2\r\n";
fwrite($fh,$stringData);
$stringData="5 to
14;$totalAge3;$percentAge3;$meridianAge3;$percentMeridianAge3;$totalDeathAge3;$CFRAge3\r\n";
fwrite($fh,$stringData);
$stringData="15 to
24;$totalAge4;$percentAge4;$meridianAge4;$percentMeridianAge4;$totalDeathAge4;$CFRAge4\r\n";
fwrite($fh,$stringData);
$stringData="25 to
39;$totalAge5;$percentAge5;$meridianAge5;$percentMeridianAge5;$totalDeathAge5;$CFRAge5\r\n";
fwrite($fh,$stringData);
$stringData="40 to
64;$totalAge6;$percentAge6;$meridianAge6;$percentMeridianAge6;$totalDeathAge6;$CFRAge6\r\n";
fwrite($fh,$stringData);
$stringData="65 and up;
$totalAge7;$percentAge7;$meridianAge7;$percentMeridianAge7;$totalDeathAge7;$CFRAge7\r\n";
fwrite($fh,$stringData);
$stringData="The age group with the
highest($percentMax%) number of cases where the
$MaxGroup years age group. \r\n";
return $stringData;
}

```

```

function getCityCount($disease_id,$year,$fh){
    $sql=mysql_query("SELECT * FROM `case_reports`
    WHERE `Disease_id`='$disease_id'");
    $total=0;
    $totalLas=0;
    $totalPar=0;
    $totalPsg=0;
    $totalPas=0;
    $totalQue=0;
    $totalMnla=0;
    $totalMan=0;
    $totalMar=0;
    $totalMak=0;
    $totalMal=0;
    $totalCal=0;
    $totalVal=0;
    $totalNav=0;
    $totalTag=0;
    $totalSan=0;
    $totalPat=0;

```

```

    $totalDeathLas=0;

```



```

$TotalDeathPar=0;
$TotalDeathPsg=0;
$TotalDeathPas=0;
$TotalDeathQue=0;
$TotalDeathMnla=0;
$TotalDeathMan=0;
$TotalDeathMar=0;
$TotalDeathMak=0;
$TotalDeathMal=0;
$TotalDeathMun=0;
$TotalDeathCal=0;
$TotalDeathVal=0;
$TotalDeathNav=0;
$TotalDeathTag=0;
$TotalDeathSan=0;
$TotalDeathPat=0;
$TotalDeath=0;
$MaxTotal=0;

while($query=mysql_fetch_array($sql)){
    $recordYear=$query['date_reported'];
    $query3=mysql_query("SELECT year('$recordYear')
FROM `case_reports`");
    $date2=mysql_fetch_array($query3);
    $year2=$date2[0];
    if($year2==$year){
        $total++;
        $city=$query['city'];
        if($city=='LP'){
            $totalLas++;
            if($totalLas>$MaxTotal){
                $MaxTotal=$totalLas;
                $MaxGroup="Las Pinas ";
            }
        }
        if($query['condition']=='DECEASED'){
            $totalDeathLas++;
            $totalDeath++;
        }
        }else if($city=='MNLA'){
            $totalMnla++;
            if($totalMnla>$MaxTotal){
                $MaxTotal=$totalMnla;
                $MaxGroup="Manila ";
            }
        }
        if($query['condition']=='DECEASED'){
            $totalDeathMnla++;
            $totalDeath++;
        }
        }else if($city=='QUE'){
            $totalQue++;
            if($totalQue>$MaxTotal){
                $MaxTotal=$totalQue;
                $MaxGroup="Quezon ";
            }
        }
        if($query['condition']=='DECEASED'){
            $totalDeathQue++;
            $totalDeath++;
        }
        }else if($city=='PAT'){
            $totalPat++;
            if($totalPat>$MaxTotal){
                $MaxTotal=$totalPat;
            }
        }
    }
}

$MaxGroup="Pateros ";
}
if($query['condition']=='DECEASED'){
    $totalDeathPat++;
    $totalDeath++;
}
}else if($city=='PQUE'){
    $totalPar++;
    if($totalPar>$MaxTotal){
        $MaxTotal=$totalPar;
        $MaxGroup="Paranaque ";
    }
}
if($query['condition']=='DECEASED'){
    $totalDeathPar++;
    $totalDeath++;
}
}else if($city=='PSG'){
    $totalPsg++;
    if($totalPsg>$MaxTotal){
        $MaxTotal=$totalPsg;
        $MaxGroup="Pasig ";
    }
}
if($query['condition']=='DECEASED'){
    $totalDeathPsg++;
    $totalDeath++;
}
}else if($city=='PAS'){
    $totalPas++;
    if($totalPas>$MaxTotal){
        $MaxTotal=$totalPas;
        $MaxGroup="Pasay ";
    }
}
if($query['condition']=='DECEASED'){
    $totalDeathPas++;
    $totalDeath++;
}
}else if($city=='MAN'){
    $totalMan++;
    if($totalMan>$MaxTotal){
        $MaxTotal=$totalMan;
        $MaxGroup="Mandaluyong ";
    }
}
if($query['condition']=='DECEASED'){
    $totalDeathMan++;
    $totalDeath++;
}
}else if($city=='MAR'){
    $totalMar++;
    if($totalMar>$MaxTotal){
        $MaxTotal=$totalMar;
        $MaxGroup="Marikina ";
    }
}
if($query['condition']=='DECEASED'){
    $totalDeathMar++;
    $totalDeath++;
}
}
}

```

```

} else if($city=='MAK'){
$totalMak++;
if($totalMak>$MaxTotal){
    $MaxTotal=$totalMak;
    $MaxGroup="Makati ";
}
}

if($query['condition']=='DECEASED'){
    $totalDeathMak++;
    $totalDeath++;
}

} else if($city=='MAL'){
$totalMal++;
if($totalMal>$MaxTotal){
    $MaxTotal=$totalMal;
    $MaxGroup="Malabon ";
}
}

if($query['condition']=='DECEASED'){
    $totalDeathMal++;
    $totalDeath++;
}

} else if($city=='CAL'){
$totalCal++;
if($totalCal>$MaxTotal){
    $MaxTotal=$totalCal;
    $MaxGroup="Caloocan ";
}
}

if($query['condition']=='DECEASED'){
    $totalDeathCal++;
    $totalDeath++;
}

} else if($city=='NAV'){
$totalNav++;
if($totalNav>$MaxTotal){
    $MaxTotal=$totalNav;
    $MaxGroup="Navotas ";
}
}

if($query['condition']=='DECEASED'){
    $totalDeathNav++;
    $totalDeath++;
}

} else if($city=='VAL'){
$totalVal++;
if($totalVal>$MaxTotal){
    $MaxTotal=$totalVal;
    $MaxGroup="Valenzuela
";
}
}

if($query['condition']=='DECEASED'){
    $totalDeathVal++;
    $totalDeath++;
}

} else if($city=='SAN'){
$totalSan++;
if($totalSan>$MaxTotal){
    $MaxTotal=$totalSan;
    $MaxGroup="San Juan ";
}
}

if($query['condition']=='DECEASED'){
    $totalDeathSan++;
    $totalDeath++;
}

} else if($city=='MUN'){
$totalMun++;
if($totalMun>$MaxTotal){
    $MaxTotal=$totalMun;
    $MaxGroup="Muntinlupa
";
}
}

if($query['condition']=='DECEASED'){
    $totalDeathMun++;
    $totalDeath++;
}

} else{
    $totalTag++;
    if($totalTag>$MaxTotal){
        $MaxTotal=$totalTag;
        $MaxGroup="Taguig ";
    }
}

if($query['condition']=='DECEASED'){
    $totalDeathTag++;
    $totalDeath++;
}
}
}

}

$percentLas=round(($totalLas/$total)*100,2);
$percentQue=round(($totalQue/$total)*100,2);
$percentMnla=round(($totalMnla/$total)*100,2);
$percentMan=round(($totalMan/$total)*100,2);
$percentPas=round(($totalPas/$total)*100,2);
$percentPar=round(($totalPar/$total)*100,2);
$percentTag=round(($totalTag/$total)*100,2);
$percentPat=round(($totalPat/$total)*100,2);
$percentCal=round(($totalCal/$total)*100,2);
$percentNav=round(($totalNav/$total)*100,2);
$percentVal=round(($totalVal/$total)*100,2);
$percentMun=round(($totalMun/$total)*100,2);
$percentSan=round(($totalSan/$total)*100,2);
$percentMar=round(($totalMar/$total)*100,2);
$percentMak=round(($totalMak/$total)*100,2);
$percentMal=round(($totalMal/$total)*100,2);
$CFRLas=round(($totalDeathLas/$totalLas)*100,2);
$CFRQue=round(($totalDeathQue/$totalQue)*100,2);
$CFRMnla=round(($totalDeathMnla/
$totalMnla)*100,2);
$CFRMan=round(($totalDeathMan/$totalMan)*100,2);
$CFRPas=round(($totalDeathPas/$totalPas)*100,2);
$CFRPar=round(($totalDeathPar/$totalPar)*100,2);
$CFRTag=round(($totalDeathTag/$totalTag)*100,2);
$CFRPat=round(($totalDeathPat/$totalPat)*100,2);
$CFRCal=round(($totalDeathCal/$totalCal)*100,2);
$CFRNav=round(($totalDeathNav/$totalNav)*100,2);
$CFRVal=round(($totalDeathVal/$totalVal)*100,2);
$CFRMun=round(($totalDeathMun/$totalMun)*100,2);
$CFRSan=round(($totalDeathSan/$totalSan)*100,2);
$CFRMar=round(($totalDeathMar/$totalMar)*100,2);
$CFRMak=round(($totalDeathMak/$totalMak)*100,2);
$CFRMal=round(($totalDeathMal/$totalMal)*100,2);
$meridianLas=getCityMeridian('LP',$year,$disease_id);

```

```

$meridianQue=getCityMeridian('QUE',$year,
$disease_id);
$meridianMnla=getCityMeridian('MnLA',$year,
$disease_id);
$meridianMan=getCityMeridian('MAN',$year,
$disease_id);
$meridianPas=getCityMeridian('PAS',$year,
$disease_id);
$meridianPar=getCityMeridian('PQUE',$year,
$disease_id);
$meridianTag=getCityMeridian('TAG',$year,
$disease_id);
$meridianPat=getCityMeridian('PAT',$year,$disease_id);
$meridianCal=getCityMeridian('CAL',$year,
$disease_id);
$meridianNav=getCityMeridian('NAV',$year,
$disease_id);
$meridianVal=getCityMeridian('VAL',$year,
$disease_id);
$meridianMun=getCityMeridian('MUN',$year,
$disease_id);
$meridianSan=getCityMeridian('SAN',$year,
$disease_id);
$meridianMar=getCityMeridian('MAR',$year,
$disease_id);
$meridianMak=getCityMeridian('MAK',$year,
$disease_id);
$meridianMal=getCityMeridian('MAL',$year,
$disease_id);
$percentMeridianLas=round(((TotalLas-$meridianLas)/
$meridianLas)*100,2);
$percentMeridianQue=round(((TotalQue-
$meridianQue)/$meridianQue)*100,2);
$percentMeridianMnla=round(((TotalMnla-
$meridianMnla)/$meridianMnla)*100,2);
$percentMeridianMan=round(((TotalMan-
$meridianMan)/$meridianMan)*100,2);
$percentMeridianPas=round(((TotalPas-$meridianPas)/
$meridianPas)*100,2);
$percentMeridianPar=round(((TotalPar-$meridianPar)/
$meridianPar)*100,2);
$percentMeridianTag=round(((TotalTag-$meridianTag)/
$meridianTag)*100,2);
$percentMeridianPat=round(((TotalPat-$meridianPat)/
$meridianPat)*100,2);
$percentMeridianCal=round(((TotalCal-$meridianCal)/
$meridianCal)*100,2);
$percentMeridianNav=round(((TotalNav-
$meridianNav)/$meridianNav)*100,2);
$percentMeridianVal=round(((TotalVal-$meridianVal)/
$meridianVal)*100,2);
$percentMeridianMun=round(((TotalLas-$meridianLas)/
$meridianLas)*100,2);
$percentMeridianSan=round(((TotalSan-$meridianSan)/
$meridianSan)*100,2);
$percentMeridianMar=round(((TotalMar-
$meridianMar)/$meridianMar)*100,2);
$percentMeridianMak=round(((TotalMak-
$meridianMak)/$meridianMak)*100,2);
$percentMeridianMal=round(((TotalMal-$meridianMal)/
$meridianMal)*100,2);
$percentMax=round((($MaxTotal/$total)*100,2);
$CFRTtotal=round((($totalDeath/$total)*100,2);
$stringData="CITY;;;;;\r\n";
fwrite($fh,$stringData);

```

```

$stringData="CALOOCAN;$totalCal;$percentCal;
$meridianCal;$percentMeridianCal;$totalDeathCal;
$CFRCAL\r\n";
fwrite($fh,$stringData);
$stringData="LASPINAS;$totalLas;$percentLas;
$meridianLas;$percentMeridianLas;$totalDeathLas;
$CFRLAS\r\n";
fwrite($fh,$stringData);
$stringData="MAKATI;$totalMak;$percentMak;
$meridianMak;$percentMeridianMak;$totalDeathMak;
$CFRMak\r\n";
fwrite($fh,$stringData);
$stringData="MALABON;$totalMal;$percentMal;
$meridianMal;$percentMeridianMal;$totalDeathMal;
$CFRMAL\r\n";
fwrite($fh,$stringData);
$stringData="MANDALUYONG;$totalMan;
$percentMan;$meridianMan;$percentMeridianMan;
$totalDeathMan;$CFRMAN\r\n";
fwrite($fh,$stringData);
$stringData="MANILA;$totalMnla;$percentMnla;
$meridianMnla;$percentMeridianMnla;$totalDeathMnla;
$CFRMnla\r\n";
fwrite($fh,$stringData);
$stringData="MARIKINA;$totalMar;$percentMar;
$meridianMar;$percentMeridianMar;$totalDeathMar;
$CFRMAR\r\n";
fwrite($fh,$stringData);
$stringData="MUNTINLUPA;$totalMun;$percentMun;
$meridianMun;$percentMeridianMun;$totalDeathMun;
$CFRMUN\r\n";
fwrite($fh,$stringData);
$stringData="NAVOTAS;$totalNav;$percentNav;
$meridianNav;$percentMeridianNav;$totalDeathNav;
$CFRNav\r\n";
fwrite($fh,$stringData);
$stringData="PARANAQUE;$totalPar;$percentPar;
$meridianPar;$percentMeridianPar;$totalDeathPar;
$CFRPAR\r\n";
fwrite($fh,$stringData);
$stringData="PASAY;$totalPas;$percentPas;
$meridianPas;$percentMeridianPas;$totalDeathPas;
$CFRPAS\r\n";
fwrite($fh,$stringData);
$stringData="PATEROS;$totalPat;$percentPat;
$meridianPat;$percentMeridianPat;$totalDeathPat;
$CFRPAT\r\n";
fwrite($fh,$stringData);
$stringData="QUEZON;$totalQue;$percentQue;
$meridianQue;$percentMeridianQue;$totalDeathQue;
$CFRQUE\r\n";
fwrite($fh,$stringData);
$stringData="SANJUAN;$totalSan;$percentSan;
$meridianSan;$percentMeridianSan;$totalDeathSan;
$CFRSAN\r\n";
fwrite($fh,$stringData);
$stringData="TAGUIG;$totalTag;$percentTag;
$meridianTag;$percentMeridianTag;$totalDeathTag;
$CFRTAG\r\n";
fwrite($fh,$stringData);
$stringData="VALENZUELA;$totalVal;$percentVal;
$meridianVal;$percentMeridianVal;$totalDeathVal;
$CFRVAL\r\n";
fwrite($fh,$stringData);
$stringData="NCR;$total;100;;;;$totalDeath;
$CFRTOTAL\r\n";
fwrite($fh,$stringData);

```

```

$stringData="$MaxGroup reported the
highest($percentMax%) number of cases.$totalDeath
died(CFR=$CFRTotal%.)\r\n";
return $stringData;
}
?>

```

## getInfectedCount.php

```

<?php
require_once('connect.php');

function getInfectedCount($city,$disease_id){
$query=mysql_query("SELECT count(*) FROM
`case_reports` WHERE `Disease_id`='$disease_id' &&
`city`='$city' && `condition`='INFECTED'");
$result=mysql_fetch_array($query);
$count=$result[0];
$query=mysql_query("SELECT
risk_high,risk_ave,risk_low FROM `disease` WHERE
`Disease_id`='$disease_id' ");
$result=mysql_fetch_array($query);
$risk_high=$result[0];
$risk_ave=$result[1];
$risk_low=$result[2];
if($count<$risk_low){
$risk="LOW";
}else if($count<$risk_ave){
$risk="AVE";
}else{
$risk="HIGH";
}
return $risk;
}
?>

```

## getMeridian.php

```

<?php
require('getYear.php');
function getGenderMeridian($gender,$year,$disease_id)
{
$index=0;
$sum=0;
for($i=1;$i<6;$i++){
$years[$index]=$year-$i;
$index++;
}
for($l=0;$l<5;$l++){
$sql=mysql_query("SELECT * FROM `case_reports`
WHERE `Disease_id`='$disease_id'");
$total=0;

while($query=mysql_fetch_array($sql)){
$recordYear=$query['date_reported'];
$query3=mysql_query("SELECT year('$recordYear')
FROM `case_reports`");
$date2=mysql_fetch_array($query3);
$year2=$date2[0];

if($year2==$years[$l]){
$temp=$query['gender'];
if($temp==$gender){
$total++;
}
}
}
}
}

```

```

}
}
$sum=$sum+$total;
}

$meridian=round(($sum/2),2);

return $meridian;
}

function getAgeMeridian($lowerbound,$upperbound,
$year,$disease_id){
$index=0;
$sum=0;
for($i=1;$i<6;$i++){
$years[$index]=$year-$i;
$index++;
}

for($l=0;$l<5;$l++){

$sqlAge=mysql_query("SELECT * FROM
`case_reports` WHERE `Disease_id`='$disease_id'");

while($query1=mysql_fetch_array($sqlAge)){

$recordYear=$query1['date_reported'];
$query3=mysql_query("SELECT year('$recordYear')
FROM `case_reports` LIMIT 1");
$date2=mysql_fetch_array($query3);
$year2=$date2[0];
$age=$query1['age'];
if($year2==$years[$l]){
if(($age>=$lowerbound) && ($age<$upperbound)){
$total++;
}
}
}
}
$sum=$sum+$total;
}

$meridian=round(($sum/2),2);
return $meridian;
}

function getCityMeridian($city,$year,$disease_id){
set_time_limit(120);
$index=0;
$sum=0;
for($i=1;$i<6;$i++){
$years[$index]=$year-$i;
$index++;
}
for($l=0;$l<5;$l++){

$sqlCity=mysql_query("SELECT * FROM
`case_reports` WHERE `Disease_id`='$disease_id'");
$total=0;

```

```

while($query2=mysql_fetch_array($sqlCity)){
    $stempYear=$query2['date_reported'];

    $query3=mysql_query("SELECT year('$stempYear')
FROM `case_reports` LIMIT 1");
    $date2=mysql_fetch_array($query3);
    $year2=$date2[0];

    $stempCity=$query2['city'];
    if($year2==$years[$i]){
        if($stempCity==$city){
            $total++;
        }
    }
}
$sum=$sum+$total;
}

$meridian=round(($sum/2),2);
return $meridian;

}
?>

```

## getYear.php

```

<?php
function getYear($date){
    $sql=mysql_query("SELECT year('$date') FROM
disease LIMIT 1");
    $result=mysql_fetch_array($sql);
    return $result[0];
}
?>

```

## performPACF.php

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title>Approximation Result</title>
<link href="style2.css" rel="stylesheet" type="text/css">
</head>

<body>
<?php
set_time_limit(600);
require_once('connect.php');
$time_unit=$_GET['unit'];
$disease_id=$_GET['disease_id'];
$par1=$_GET['par1'];
$par2=$_GET['par2'];
$par3=$_GET['par3'];
$par4=$_GET['par4'];
$par5=$_GET['par5'];
$par6=$_GET['par6'];
$par7=$_GET['par7'];
if($time_unit==""||$disease_id==""){

```

```

echo "<font color='red'>ERROR: Missing field values !!
<br></font>";
include("genPACF.php");
}else{
if($time_unit=="yearly"){
$query=mysql_query("SELECT DISTINCT
year( `date_reported` ) FROM `case_reports` WHERE
`Disease_id`='$disease_id' ORDER BY `date_reported`
ASC");
//chdir('C:\xampp\htdocs\joom\R-2.10.0\bin');
$stringData="Date,Cases\r\n";
$myFile = "R-2.10.0/bin/testFile.txt";
$fh = fopen($myFile, 'w') or die("can't open file");
fwrite($fh, $stringData);
while($date = mysql_fetch_array($query)){
$year=$date[0];
$count=0;
$query2=mysql_query("SELECT * FROM
`case_reports` WHERE `Disease_id`='$disease_id'
ORDER BY `date_reported` ASC ");
while($row = mysql_fetch_array($query2)){
$recordYear=$row['date_reported'];
$query3=mysql_query("SELECT year('$recordYear')
FROM `case_reports`");
$date2=mysql_fetch_array($query3);
$year2=$date2[0];
if($year==$year2){
$count++;
}
}
$stringData="$year,$count\r\n";
fwrite($fh, $stringData);
//}
}
fclose($fh);

}else{
$query=mysql_query("SELECT DISTINCT
year( `date_reported` ), month( `date_reported` ) FROM
`case_reports` WHERE `Disease_id`='$disease_id'
ORDER BY `date_reported` ASC");
$stringData="Date,Cases\r\n";
$myFile = "R-2.10.0/bin/testFile.txt";
$fh = fopen($myFile, 'w') or die("can't open file");
fwrite($fh, $stringData);
while($date = mysql_fetch_array($query)){
$year=$date[0];
$month=$date[1];
$count=0;
$query2=mysql_query("SELECT * FROM
`case_reports` WHERE `Disease_id`='$disease_id'");
while($row = mysql_fetch_array($query2)){
$recordYear=$row['date_reported'];
$query3=mysql_query("SELECT
year('$recordYear'),month('$recordYear') FROM
`case_reports`");
$date2=mysql_fetch_array($query3);
$year2=$date2[0];
$month2=$date2[1];
if($year==$year2&&$month==$month2){
$count++;
}
}
$stringData="$year-$month,$count\r\n";

```

```

fwrite($fh, $stringData);
}
fclose($fh);

}
$myFile = "R-2.10.0/bin/result.txt";
$fh = fopen($myFile, 'w+') or die("can't open file");
fclose($fh);
$myFile = "R-2.10.0/bin/rCommand.txt";
$fh = fopen($myFile, 'w') or die("can't open file");
fwrite($fh,
"data=read.table('testFile.txt',header=T,sep=',')\r\n");
fwrite($fh, "x<-c(data.".$".Cases)\r\n");
fwrite($fh, "par7 = '$par7'\r\n");
fwrite($fh, "par6 = '$par6'\r\n");
fwrite($fh, "par5 = '$par5'\r\n");
fwrite($fh, "par4 = '$par4'\r\n");
fwrite($fh, "par3 = '$par3'\r\n");
fwrite($fh, "par2 = '$par2'\r\n");
fwrite($fh, "par1 = '$par1'\r\n");
fwrite($fh, "if(par1 == 'Default') {\r\n");
fwrite($fh, "par1 = 10*log10(length(x))\r\n");
fwrite($fh, "} else {\r\n");
fwrite($fh, "par1 <- as.numeric(par1);\r\n");
fwrite($fh, "par2 <- as.numeric(par2);\r\n");
fwrite($fh, "par3 <- as.numeric(par3);\r\n");
fwrite($fh, "par4 <- as.numeric(par4);\r\n");
fwrite($fh, "par5 <- as.numeric(par5);\r\n");
fwrite($fh, "if (par6 == 'White Noise') par6 <- 'white' else
par6 <- 'ma'\r\n");
fwrite($fh, "par7 <- as.numeric(par7)\r\n");
fwrite($fh, "if (par2 == 0) {\r\n");
fwrite($fh, "x <- log(x)\r\n");
fwrite($fh, "} else {\r\n");
fwrite($fh, "x <- (x ^ par2 - 1) / par2\r\n");
fwrite($fh, "if (par3 > 0) x <-
diff(x,lag=1,difference=par3)\r\n");
fwrite($fh, "if (par4 > 0) x <-
diff(x,lag=par5,difference=par4)\r\n");
fwrite($fh, "png('pic1.png')\r\n");
fwrite($fh, "racf <- acf(x, par1, main='Autocorrelation',
xlab='time lag',
ylab='ACF', ci.type=par6, ci=par7,
sub=paste('(lambda=',par2,',d=',par3,', D=',par4,', CI=',
par7,', CI type=',par6,',sep='))\r\n");
fwrite($fh, "dev.off()\r\n");
fwrite($fh, "png('pic2.png')\r\n");
fwrite($fh, "rpacf <- pacf(x,par1,main='Partial
Autocorrelation',xlab='lags',ylab='PACF')\r\n");
fwrite($fh, "dev.off()\r\n");
fclose($fh);
chdir("C:\xampp\htdocs\joomla\dods\R-2.10.0\bin");
exec("R CMD BATCH rCommand.txt");
chdir("C:\xampp\htdocs\joomla\dods");
echo "<center><img src='R-2.10.0/bin/pic1.png'><br>";
echo "<img src='R-2.10.0/bin/pic2.png'></center>";
echo "<br><br><a
href='http://localhost/joomla/dods/index.php?
page=computation'><img
src='/protected/pages/images/back.png'></a>";
}
?>
</body>

```

## performVarMat.php

```

<html>
<head>
</head>

<body>
<?php
set_time_limit(600);
require_once('connect.php');
$time_unit=$_GET['unit'];
$disease_id=$_GET['disease_id'];
$par1=$_GET['par1'];
if($time_unit=="yearly"){
$query=mysql_query("SELECT DISTINCT
year( `date_reported` ) FROM `case_reports` WHERE
`Disease_id`='$disease_id' ORDER BY `date_reported`
ASC");
$stringData="Date,Cases\r\n";
$myFile = "R-2.10.0/bin/testFile.txt";
$fh = fopen($myFile, 'w') or die("can't open file");
fwrite($fh, $stringData);
while($date = mysql_fetch_array($query)){
$year=$date[0];
$count=0;
$query2=mysql_query("SELECT * FROM
`case_reports` WHERE `Disease_id`='$disease_id'
ORDER BY `date_reported` ASC");
while($row = mysql_fetch_array($query2)){
$recordYear=$row['date_reported'];
$query3=mysql_query("SELECT year('$recordYear')
FROM `case_reports`");
$date2=mysql_fetch_array($query3);
$year2=$date2[0];
if($year==$year2){
$count++;
}
}
$stringData="$year,$count\r\n";
fwrite($fh, $stringData);
}
fclose($fh);

}else{
$query=mysql_query("SELECT DISTINCT
year( `date_reported` ), month( `date_reported` ) FROM
`case_reports` WHERE `Disease_id`='$disease_id'
ORDER BY `date_reported` ASC");
$stringData="Date,Cases\r\n";
$myFile = "R-2.10.0/bin/testFile.txt";
$fh = fopen($myFile, 'w') or die("can't open file");
fwrite($fh, $stringData);
while($date = mysql_fetch_array($query)){
$year=$date[0];
$month=$date[1];
$count=0;
$query2=mysql_query("SELECT * FROM
`case_reports` WHERE `Disease_id`='$disease_id'");
while($row = mysql_fetch_array($query2)){
$recordYear=$row['date_reported'];
$query3=mysql_query("SELECT
year('$recordYear'),month('$recordYear') FROM
`case_reports`");
$date2=mysql_fetch_array($query3);

```



```

$par4=$_GET['par4'];
$par5=$_GET['par5'];
$par6=$_GET['par6'];
$par7=$_GET['par7'];
$par8=$_GET['par8'];
$par9=$_GET['par9'];
$par10=$_GET['par10'];
$showCalculation=$_GET['showSolutions'];
if($time_unit=="|||$disease_id==""){
echo "<font color='red'>ERROR: Missing field values !!
<br></font>";
include("genApprox2.php");
}else{
if($time_unit=="yearly"){
$query=mysql_query("SELECT DISTINCT
year( `date_reported` ) FROM `case_reports` WHERE
`Disease_id`='$disease_id' ORDER BY `date_reported`
ASC");
$stringData="Date,Cases\r\n";
$myFile = "R-2.10.0/bin/testFile.txt";
$fh = fopen($myFile, 'w') or die("can't open file");
fwrite($fh, $stringData);
while($date = mysql_fetch_array($query)){
$year=$date[0];
$count=0;
$query2=mysql_query("SELECT * FROM
`case_reports` WHERE `Disease_id`='$disease_id'
ORDER BY `date_reported` ASC ");
while($row = mysql_fetch_array($query2)){
$recordYear=$row['date_reported'];
$query3=mysql_query("SELECT year('$recordYear')
FROM `case_reports`");
$date2=mysql_fetch_array($query3);
$year2=$date2[0];
if($year==$year2){
$count++;
}
}
$stringData="$year,$count\r\n";
fwrite($fh, $stringData);
}
fclose($fh);

}else{

$query=mysql_query("SELECT DISTINCT
year( `date_reported` ), month( `date_reported` ) FROM
`case_reports` WHERE `Disease_id`='$disease_id'
ORDER BY `date_reported` ASC");
$stringData="Date,Cases\r\n";
$myFile = "R-2.10.0/bin/testFile.txt";
$fh = fopen($myFile, 'w') or die("can't open file");
fwrite($fh, $stringData);
while($date = mysql_fetch_array($query)){
$year=$date[0];
$month=$date[1];
$count=0;
$query2=mysql_query("SELECT * FROM
`case_reports` WHERE `Disease_id`='$disease_id'");
while($row = mysql_fetch_array($query2)){
$recordYear=$row['date_reported'];
$query3=mysql_query("SELECT
year('$recordYear'),month('$recordYear') FROM
`case_reports`");
$date2=mysql_fetch_array($query3);
$year2=$date2[0];
$month2=$date2[1];
if($year==$year2&&$month==$month2){
$count++;
}
}
$stringData="$year-$month,$count\r\n";
fwrite($fh, $stringData);
}
fclose($fh);

}

$myFile = "R-2.10.0/bin/result.txt";
$fh = fopen($myFile, 'w+') or die("can't open file");
fclose($fh);
$myFile = "R-2.10.0/bin/rCommand.txt";
$fh = fopen($myFile, 'w') or die("can't open file");
fwrite($fh,
"data=read.table('testFile.txt',header=T,sep=',')\r\n");
fwrite($fh, "source('result.txt')\r\n");
fwrite($fh, "sink('result.txt')\r\n");
fwrite($fh, "x <-data"."$"."Cases\r\n");
fwrite($fh, "par10 = $par10\r\n");
fwrite($fh, "par9 = $par9\r\n");
fwrite($fh, "par8 = $par8\r\n");
fwrite($fh, "par7 = $par7\r\n");
fwrite($fh, "par6 = $par6\r\n");
fwrite($fh, "par5 = $par5\r\n");
fwrite($fh, "par4 = $par4\r\n");
fwrite($fh, "par3 = $par3\r\n");
fwrite($fh, "par2 = $par2\r\n");
fwrite($fh, "par1 = $par1\r\n");
fwrite($fh,"par1 <- as.numeric(par1)\r\n");
fwrite($fh,"par2 <- as.numeric(par2)\r\n");
fwrite($fh,"par3 <- as.numeric(par3)\r\n");
fwrite($fh,"par4 <- as.numeric(par4)\r\n");
fwrite($fh,"par5 <- as.numeric(par5)\r\n");
fwrite($fh,"par6 <- as.numeric(par6)\r\n");
fwrite($fh,"par7 <- as.numeric(par7)\r\n");
fwrite($fh,"par8 <- as.numeric(par8)\r\n");
fwrite($fh,"par9 <- as.numeric(par9)\r\n");
fwrite($fh,"par10 <- $par10\r\n");
fwrite($fh,"lx <- length(x)\r\n");
fwrite($fh,"first <- lx - 2*par1\r\n");
fwrite($fh,"nx <- lx - par1\r\n");
fwrite($fh,"nx1 <- nx + 1\r\n");
fwrite($fh,"fx <- lx - nx\r\n");
fwrite($fh,"if (fx < 1) {}");
fwrite($fh,"fx <- par5\r\n");
fwrite($fh,"nx1 <- lx + fx - 1\r\n");
fwrite($fh,"first <- lx - 2*fx\r\n");
fwrite($fh,"}\r\n");
fwrite($fh,"first <- 1\r\n");
fwrite($fh,"if (fx < 3) fx <- round(lx/10,0)\r\n");
fwrite($fh,"(arima.out <- arima(data"."$"."Cases,
order=c(par6,par3,par7),
seasonal=list(order=c(par8,par4,par9), period=par5),
include.mean=par10, method='ML', transform.pars =
TRUE,))\r\n");
fwrite($fh, "(forecast <- predict(arima.out,fx))\r\n");
fwrite($fh,"predVal<-c(forecast"."$"."pred)\r\n");
fwrite($fh,"values<-c(data"."$"."Cases,predVal)\r\n");
fwrite($fh,"L = forecast"."$"."pred -
2*forecast"."$"."se\r\n");

```



```

fwrite($fh,"U = forecast"."$"."pred +
2*forecast"."$"."se\r\n");
fwrite($fh,"g_range<-
range(0,data"."$"."Cases,predVal)\r\n");
fwrite($fh,"l<-length(data"."$"."Cases)\r\n");
fwrite($fh,"first <- lx - 2*fx\r\n");
fwrite($fh,"maxValues<-max(values)\r\n");
fwrite($fh,"png('approxImage.png')\r\n");
fwrite($fh,"plot(values, type='o', col='blue',
ylim=g_range,axes=FALSE, ann=FALSE)\r\n");
fwrite($fh,"lines(lx,maxValues,col='red',pch=22,lty=2,ty
pe='h')\r\n");
fwrite($fh,"lines(U, col=rgb(0.5,0,0.5),
lty='dashed')\r\n");
fwrite($fh,"lines(L, col=rgb(0.5,0,0.5), lty='dashed')
\r\n");
fwrite($fh,"axis(1, at=1:lr)\r\n");
fwrite($fh,"axis(2, las=1, at=10*0:g_range[2])\r\n");
fwrite($fh,"title(xlab='Time Interval',
col.lab=rgb(0,0.5,0))\r\n");
fwrite($fh,"title(ylab='Incident Rates',
col.lab=rgb(0,0.5,0))\r\n");
fwrite($fh,"title(main='ARIMA Extrapolation Forecast ',
col.main='red', font.main=4)\r\n");
fwrite($fh,"legend(1, g_range[2], c('Real Values(before
vertical line)','Standard Error Bounds'),
cex=0.8,col=c('blue',rgb(0.5,0,0.5)), pch=21:22,
lty=1:2)\r\n");
fwrite($fh,"box()\r\n");
fwrite($fh,"dev.off()\r\n");
fclose($fh);
chdir("C:\xampp\htdocs\joom\dods\R-2.10.0\bin");
exec("R CMD BATCH rCommand.txt");
chdir("C:\xampp\htdocs\joom\dods");
echo "<center><img src='R-
2.10.0/bin/approxImage.png'></center>";
chdir("C:\xampp\htdocs\joom\R-2.10.0\bin");
if($showCalculation=='1'){
$myFile = "result.txt";
$fh = fopen($myFile, 'r');
echo "<table align='center'>";
echo "<tr>";
echo "<td align='center'>";
echo "<h2>- = VALUES = -<h2>";
echo "</td>";
echo "</tr>";
echo "<tr>";
echo "<td align='center'>";
echo "$"."pred = prediction values<br>";
echo "$"."se = estimated standard errors<br>";
echo "</td>";
echo "</tr>";
echo "<tr>";
echo "<td align='center'>";
while(!feof($fh)){
$theData = fgets($fh);
echo "".$theData."<br>";
}
fclose($fh);
echo "</td>";
echo "</tr>";
echo "</table>";
}
echo "<br><br><a href='performCalc.html'><img
src='back.png'></font></a>";
}
?>

```

```
</body>
```

## searchRecord.page

```

<com:THead>
<title>Add New Disease</title>
<style>
.searchrecord
{
width: 500px;
height: 310px;
border: 5px solid ButtonFace;
overflow: auto;
background-color: #E6ECFF;
font-family: Verdana;
font-size: 10px;
}
.search
{
width: 600px;
height: 400px;
border: 5px solid ButtonFace;
overflow: auto;
font-family: Verdana;
font-size: 12px;
}
</style>
</com:THead>
<fieldset class="search">
<legend><font color="blue"><h3>Search Case
Report</h3></font></legend>
<fieldset class="searchrecord">
<legend>Please enter necessary information:</legend>
<label>Note: To view all the reports, leave all
information blank.</label>
<com:TForm>
<table>
<tr>
<td>
Patient Number:
</td>
<td>
<com:TTextBox ID="patientnumber" />
</td>
<tr>
<td>
Age:
</td>
<td>
<com:TTextBox ID="age" />
</td>
<tr>
<td>
Address:
</td>
<td>
<com:TTextBox ID="address" />
</td>
<tr>
<td>
</td>
<td>
</td>

```

```

City:
</td>
<td>
<com:TDropDownList ID="city"
AutoPostBack="false">
  <com:TListItem Value="" Text=" " />
  <com:TListItem Value="LP" Text="LAS PINAS" />
  <com:TListItem Value="MNLA" Text="MANILA" />
  <com:TListItem Value="PSG" Text="PASIG" />
  <com:TListItem Value="PAS" Text="PASAY" />
  <com:TListItem Value="QUE" Text="QUEZON" />
  <com:TListItem Value="TAG" Text="TAGUIG" />
  <com:TListItem Value="MAL" Text="MALABON" />
  <com:TListItem Value="MAR" Text="MARIKINA" />
  <com:TListItem Value="PAT" Text="PATEROS" />
  <com:TListItem Value="VAL"
Text="VALENZUELA" />
  <com:TListItem Value="MAKATI"
Text="MAKATI" />
  <com:TListItem Value="SAN" Text="SAN JUAN" />
  <com:TListItem Value="NAVOTAS"
Text="NAVOTAS" />
  <com:TListItem Value="PAR"
Text="PARANAQUE" />
  <com:TListItem Value="MAN"
Text="MANDALUYONG" />
  <com:TListItem Value="MUN"
Text="MUNTINLUPA" />
</com:TDropDownList>
</td>
</tr>
<tr>
<td>
Sentinel ID:
</td>
<td>
<com:TTextBox ID="sentinel" />
</td>
</tr>
<tr>
<td>
Gender:
</td>
<td>
<com:TRadioButton
  ID="Male"
  GroupName="RadioGroup"
  Text="Male"
  />
<com:TRadioButton
  ID="Female"
  GroupName="RadioGroup"
  Text="Female"
  />
</td>
</tr>
<tr>
<td>
Disease:
</td>
<td>
<com:TDropDownList ID="disease"
  AutoPostBack="false"
  DataTextField="name"
  DataValueField="id"
  />
</td>

```

```

</tr>
<tr>
<td>
Condition:
</td>
<td>
<com:TDropDownList ID="condition"
AutoPostBack="false">
  <com:TListItem Value=" " Text=" " />
  <com:TListItem Value="INFECTED"
Text="INFECTED" />
  <com:TListItem Value="CURED" Text="CURED" />
  <com:TListItem Value="DECEASED"
Text="DECEASED" />
</com:TDropDownList>
</td>
</tr>
<tr>
<td>
Date Reported:
</td>
<td>
<com:TDatePicker Mode="ImageButton" ID="date"
  DateFormat="yyyy-MM-dd"
  />
</td>
</tr>
<tr>
<td>
<com:TButton Text="View"  OnClick="searchEntry"/>
</td>
</tr>
</table>
</fieldset>
</fieldset>
<com:THyperLink
  NavigateUrl="http://localhost/joom/updateRecord2.ph
p"
  ImageUrl=<%~images/back.png%>
  Text="Back" />.
</com:TForm>

```

## searchRecord.php

```

<?php
class searchRecord extends TPage
{
  public function onLoad($param)
  {
    if (!$this->IsPostBack){
      $sql = "SELECT * FROM
`disease` ";
      $disease=Disease::finder()->findAllBySql($sql);
      $data[0]=array('id'=>' ', 'name'=>' ');
      $index=1;
      foreach($disease as $row){
        $id=$row->Disease_id;
        $name=$row->name;
        $data[$index]=array('id'=>"$id", 'name'=>"$
$name");
        $index++;
      }
      $this->disease->DataSource=$data;
      $this->disease->dataBind();
    }
  }
}

```

```

    }
}

public function searchEntry($sender,$param)
{
    $age=$this->age->Text;
    $patNumber=$this->patientnumber-
>Text;
    $address=$this->address->Text;
    $sen_id=$this->sentinel->Text;
    $indices=$this->disease-
>SelectedIndices;
    foreach($indices as $index){
    $item=$this->disease-
>Items[$index];
    $disease_id=$item->Value;
    }
    $indices=$this->condition-
>SelectedIndices;
    foreach($indices as $index){
    $item=$this->condition-
>Items[$index];
    $condition=$item->Value;
    }
    $indices=$this->city-
>SelectedIndices;
    foreach($indices as $index){
    $item=$this->city->Items[$index];
    $city=$item->Value;
    }
    $daterecorded=$this->date->Text;
    if($this->Male->Checked){
        $gender.='M';
    }else if($this->Female->Checked){
        $gender.='F';
    }else {
        $gender.='';
    }

    if($city==""){
        $city="";
    }
    if($condition==""){
        $condition="";
    }
    if($disease_id==""){
        $disease_id="";
    }
    if($daterecorded==""){
        $daterecorded="";
    }
    $status=$_GET['status'];
    if($status=="Search"){
        $this->Response->redirect("index.php?
page=searchResult&pat_number=".urlencode($patNum
ber)."&age=".urlencode($age)."&address=".urlencode($ad
dress)."&sentinel_id=".urlencode($sen_id)."&disease="
urlencode($disease_id)."&sex=".urlencode($gender)."&d
aterecorded=".urlencode($daterecorded)."&condition="
urlencode($condition)."&city=".urlencode($city));
    }else {
        $this->Response->redirect("index.php?
page=updateRecord&pat_number=".urlencode($patNum
ber)."&age=".urlencode($age)."&address=".urlencode($a
ddress)."&sentinel_id=".urlencode($sen_id)."&disease="
urlencode($disease_id)."&sex=".urlencode($gender)."&

```

```

daterecorded=".urlencode($daterecorded)."&condition="
urlencode($condition)."&city=".urlencode($city));
    }
}

public function changePage($sender,$param)
{
    $this->DataGrid->CurrentPageIndex=$param-
>NewPageIndex;
    $this->DataGrid->DataSource=$this->Data;
    $this->DataGrid->dataBind();
}

public function pagerCreated($sender,$param)
{
    $param->Pager->Controls->insertAt(0,'Page: ');
}
?>

```

## searchResult.page

```

<com:TForm>
<h1>Case Reports</h1>
<com:TDataGrid
    Width="700px"
    CellPadding="2"
    ID="DataGrid"
    DataKeyField="ID"
    AllowPaging="true"
    PageSize="10"
    PagerStyle.Mode="Numeric"
    PagerStyle.HorizontalAlign="Left"
    AutoGenerateColumns="false"
    HeaderStyle.BackColor="black"
    HeaderStyle.ForeColor="white"
    ItemStyle.BackColor="#BFCFFF"
    ItemStyle.Font.Italic="true"
    AlternatingItemStyle.BackColor="#E6ECFF"
    EditItemStyle.BackColor="lightyellow"
    OnPageIndexChanged="changePage"
    OnPagerCreated="pagerCreated"
>

<com:TBoundColumn
    ID="ID"
    HeaderText="Case Number"
    HeaderStyle.Width="50px"
    DataField="ID"
/>

<com:TBoundColumn
    ID="patnumber"
    HeaderText="Patient Number"
    HeaderStyle.Width="150px"
    DataField="patnumber"
/>

<com:TBoundColumn
    ID="age"
    HeaderText="Age"
    HeaderStyle.Width="50px"
    DataField="age"
/>

<com:TBoundColumn
    ID="gender"

```

```

HeaderText="Gender"
HeaderStyle.Width="50px"
DataField="gender"
/>

<com:TDropDownListColumn
ID="city"
HeaderText="City"
DataTextField="city"
>
<com:TListItem Value="LP"/>
<com:TListItem Value="MNLA"/>
<com:TListItem Value="PSG"/>
<com:TListItem Value="PAS"/>
<com:TListItem Value="QUE"/>
<com:TListItem Value="TAG"/>
<com:TListItem Value="MAL"/>
<com:TListItem Value="MAR"/>
<com:TListItem Value="PAT"/>
<com:TListItem Value="VAL"/>
<com:TListItem Value="MAK"/>
<com:TListItem Value="SAN"/>
<com:TListItem Value="NAV"/>
<com:TListItem Value="PAR"/>
<com:TListItem Value="MAN"/>
<com:TListItem Value="MUN"/>
</com:TDropDownListColumn>

<com:TBoundColumn
ID="sen_id"
HeaderText="Sentinel ID"
HeaderStyle.Width="150px"
DataField="sen_id"
/>
<com:TBoundColumn
ID="disease"
HeaderText="Disease"
HeaderStyle.Width="150px"
DataField="disease"
/>
<com:TDropDownListColumn
ID="condition"
HeaderText="Status"
DataTextField="condition"
>
<com:TListItem Value="INFECTED" />
<com:TListItem Value="CURED" />
<com:TListItem Value="DECEASED" />
</com:TDropDownListColumn>
<com:TBoundColumn
ID="date"
HeaderText="Date Reported"
HeaderStyle.Width="150px"
DataField="date"
/>
</com:TDataGrid>
</com:TForm>
<com:THyperLink
NavigateUrl="http://localhost/joom/dods/index.php?
page=searchRecord&status=Search"
ImageUrl=~/images/back.png%>
Text="Back" />

```

## searchResult.php

```

<?php
class searchResult extends TPage
{
    private $_data=null;

    protected function getData()
    {
        if($this->_data===null)
            $this->loadData();
        return $this->_data;
    }

    protected function loadData()
    {
        // We use viewstate keep track of data.
        // In real applications, data should come from
        database using an SQL SELECT statement.
        // In the following tabular data, field 'ISBN' is the
        primary key.
        // All update and delete operations should come with
        an 'id' value in order to go through.
        if(($this->_data=$this-
>getViewState('Data',null))===null)
        {
            $patNumber=$_GET['pat_number'];
            $age=$_GET['age'];
            $address=$_GET['address'];
            $city=$_GET['city'];
            $disease_id=$_GET['disease'];
            $condition=$_GET['condition'];
            $sen_id=$_GET['sentinel_id'];
            $gender=$_GET['sex'];
            $daterecorded=$_GET['daterecorded'];

            if($address==""){
                if($age==""){
                    $query = "SELECT *
FROM `case_reports` where `gender` LIKE '%$gender%'
&& `patient_number` LIKE '%$patNumber%' &&
`Sentinel_id` LIKE '%$sen_id%' && `city` LIKE '%
$city%' && `condition` LIKE '%$condition%' && `age`
LIKE '%$age%' && `Disease_id` LIKE '%$disease_id%'
&& `date_reported` LIKE '%$daterecorded%';

                }else{
                    $query = "SELECT *
FROM `case_reports` where `gender` LIKE '%$gender%'
&& `patient_number` LIKE '%$patNumber%' &&
`Sentinel_id` LIKE '%$sen_id%' && `city` LIKE '%
$city%' && `condition` LIKE '%$condition%' && `age`
= '$age' && `Disease_id` LIKE '%$disease_id%' &&
`date_reported` LIKE '%$daterecorded%';

                }
            }else{
                if($age==""){
                    $query = "SELECT *
FROM `case_reports` where `gender` LIKE '%$gender%'
&& `patient_number` LIKE '%$patNumber%' &&

```

```

`Sentinel_id` LIKE '%$sen_id%' && `city` LIKE '%
$city%' && `condition` LIKE '%$condition%' && `age`
LIKE '%$age%' && `Disease_id` LIKE '%$disease_id%'
&& `date_reported` LIKE '%$daterecorded%";

    }else{
        $query = "SELECT *
FROM `case_reports` where `gender` LIKE '%$gender%'
&& `patient_number` LIKE '%$patNumber%' &&
`Sentinel_id` LIKE '%$sen_id%' && `city` LIKE '%
$city%' && `condition` LIKE '%$condition%' && `age`
= '$age' && `Disease_id` LIKE '%$disease_id%' &&
`date_reported` LIKE '%$daterecorded%";

    }
}
$case=CaseReports::finder()-
>findAllBySql($query);
$index=0;
foreach($case as $row){
    $id=$row->case_number;
    $patNumber=$row-
>patient_number;
    $age=$row->age;
    $gender=$row->gender;
    $city=$row->city;
    $sen_id=$row-
>Sentinel_id;
    $disease=$row-
>Disease_id;
    $condition=$row->condition;
    $date=$row-
>date_reported;
    $data[$index]=array('ID'=>". $id,'patnumber'=>".
$patNumber,'age'=>". $age,'disease'=>".
$disease,'condition'=>". $condition,'date'=>".
$date,'sen_id'=>". $sen_id,'gender'=>". $gender,'city'=>".
$city);
    $index++;
}
if($index==0){
    echo "<h3>Status:</h3><h4><font
color='red'>Case Record Not Found!!</font></h4>";
}
$this->_data=$data;
$this->saveData();
}
}

protected function saveData()
{
    $this->setViewState('Data',$this->_data);
}

protected function updateBook($isbn,$patnumber,
$age,$gender,$city,$disease,$sen,$condition,$date,
$indx)
{
    $finder=CaseReports::finder();
    // In real applications, data should be saved to
    database using an SQL UPDATE statement
    if($this->_data===null)
        $this->loadData();
    $updateRow=null;
    foreach($this->_data as $index=>$row)
        if($row['ID']===$isbn)
            $temp=$row['ID'];
            $case = $finder-
>findByPk($temp);
            $case->patient_number=$patnumber;
            $case->age=$age;
            $case->gender=$gender;
            $case->city=$city;
            $case-
>Disease_id=$disease;
            $case->Sentinel_id=$sen;
            $case-
>condition=$condition;
            $case-
>date_reported=$date;
            $case->save(); //update it.
            $updateRow=&$this->_data[$index];
            if($updateRow!==null)
            {
                $updateRow['patnumber']=$patnumber;
                $updateRow['gender']=$gender;
                $updateRow['city']=$city;
                $updateRow['sen_id']=$sen;
                $updateRow['condition']=$condition;
                $updateRow['date']=$date;
                $updateRow['disease']=$disease;
                $updateRow['age']=TPropertyValue::ensureInteg
er($age);
                $this->saveData();
            }
}

protected function deleteRecord($isbn,$tmp)
{
    $finder = CaseReports::finder();
    // In real applications, data should be saved to
    database using an SQL DELETE statement
    if($this->_data===null)
        $this->loadData();
    $deleteIndex=-1;
    foreach($this->_data as $index=>$row)
        if($row['ID']===$isbn)
            $temp=$row['ID'];
            //$finder->deleteByPk($temp);
            $deleteIndex=$tmp;
}

if($deleteIndex>=0)
{
    unset($this->_data[$tmp]);
    $this->saveData();
}
}

public function onLoad($param)
{
    parent::onLoad($param);
    if(!$this->IsPostBack)
    {
        $this->DataGrid->DataSource=$this->Data;
    }
}

```

```

        $this->DataGrid->dataBind();
    }
}

public function itemCreated($sender,$param)
{
    $item=$param->Item;
    if($item->ItemType==='EditItem')
    {
        // set column width of textboxes
        //$item->ID->TextBox->Columns=5;
        $item->patnumber->TextBox->Columns=10;
        $item->age->TextBox-
>Columns=3;
        $item->gender->TextBox-
>Columns=3;
        $item->sen_id->TextBox-
>Columns=5;
        $item->disease->TextBox-
>Columns=5;
    }
    if($item->ItemType==='Item' || $item-
>ItemType==='AlternatingItem' || $item-
>ItemType==='EditItem')
    {
        // add an alert dialog to delete buttons
        $item->DeleteColumn->Button->Attributes-
>onclick='if(!confirm("Are you sure?")) return false;';
    }
}

public function editItem($sender,$param)
{
    $this->DataGrid->EditItemIndex=$param->Item-
>ItemIndex;
    $this->DataGrid->DataSource=$this->Data;
    $this->DataGrid->dataBind();
}

public function saveItem($sender,$param)
{
    $index=$param->Item->ItemIndex;
    $item=$param->Item;
    $this->updateBook(
        $this->DataGrid->DataKeys[$item->ItemIndex],
// ISBN
        $item->patnumber->TextBox->Text, //
title
        $item->age->TextBox->Text, // publisher
        $item->gender->TextBox->Text,
        $item->city->DropDownList->SelectedValue,
// price
        $item->disease->TextBox->Text, // instock
        $item->sen_id->TextBox-
>Text,
        $item->condition-
>DropDownList->SelectedValue,
        $item->date->TextBox-
>Text,
        $index
    );
    $this->DataGrid->EditItemIndex=-1;
    $this->DataGrid->DataSource=$this->Data;
    $this->DataGrid->dataBind();
}

public function cancelItem($sender,$param)

```

```

    {
        $this->DataGrid->EditItemIndex=-1;
        $this->DataGrid->DataSource=$this->Data;
        $this->DataGrid->dataBind();
    }

    public function deleteItem($sender,$param)
    {
        $index=$param->Item->ItemIndex;
        echo "B4: $index";
        $this->deleteRecord($this-
>DataGrid->DataKeys[$param->Item->ItemIndex],
$index);
        $this->DataGrid->EditItemIndex=-1;
        $this->DataGrid->DataSource=$this-
>Data;
        $this->DataGrid->dataBind();
    }

    public function changePage($sender,$param)
    {
        $this->DataGrid->CurrentPageIndex=$param-
>NewPageIndex;
        $this->DataGrid->DataSource=$this->Data;
        $this->DataGrid->dataBind();
    }

    public function pagerCreated($sender,$param)
    {
        $param->Pager->Controls->insertAt(0,'Page: ');
    }
}
?>

```

## statusIndicators.page

```

<com:THed>
<title>Add New Disease</title>
<style>
.selectView
{
    width: 500px;
    height: 300px;
    border: 1px solid ButtonFace;
    overflow: auto;
    background-color: #E6ECFF;
    font-family: Verdana;
    font-size: 10px;
}
.select
{
    width: 600px;
    height: 400px;
    border: 5px solid ButtonFace;
    overflow: auto;
    font-family: Verdana;
    font-size: 12px;
}
</style>
</com:THed>

```

```

<com:TForm>
<com:TMultiView ID="statusView">
<com:TView ID="IntroView">
<fieldset class="select">
<legend><h3><center>Status
Indicators</center></h3></legend>
<fieldset class="selectView">
<legend><font color="blue"><h4>Choose from the
following status
indicators:</h4></font></legend><br><br>
<table align="center">
<tr>
<td>
<com:TRadioButton
  ID="Report"
  GroupName="RadioGroup"
  />
</td>
<td>
<com:TLabel ForControl="Report" Text="Create Annual
Report" />
</td>
</tr>
<tr>
<td>
<com:TRadioButton
  ID="Graph"
  GroupName="RadioGroup"/>
</td>
<td>
<com:TLabel ForControl="Graph" Text="Generate
Status Graph" />
</td>
</tr>
<tr>
<td>
<com:TCustomValidator
  ControlToValidate="Report"
  Display="Dynamic"
  ValidationGroup="selectStatus"
  OnServerValidate="checkRadio"
  FocusOnError="true"
  ErrorMessage="Please select one from the choices." />
</td>
</tr>
</table><br><br>
<center>
<com:TButton Text="Proceed"
ValidationGroup="selectStatus" ID="Proceed"
OnClick="selectView"/>
</center>
</fieldset>
</fieldset>
</com:TView>
<com:TView ID="ReportView">
<fieldset class="select">
<legend><h3><center>Status
Indicators</center></h3></legend>
<fieldset class="selectView">
<legend><font color="blue">Create Annual
Report</font></legend><br><br>
<table>
<tr>
<td>
<com:TLabel ForControl="year1" Text="Year:" />
</td><td>
<com:TDropDownList ID="year1"
  AutoPostBack="false"
  DataTextField="name"
  DataValueField="id"
  />
</td>
</tr>
<tr>
<td>
<com:TLabel ForControl="disease1" Text="Disease:" />
</td><td>
<com:TDropDownList ID="disease1"
  AutoPostBack="false"
  DataTextField="name"
  DataValueField="id"
  />
</td>
</tr>
</table><br><br>
<center><com:TButton Text="Create" ID="Create"
ValidationGroup="select"
OnClick="createReport"/></center><br><br>
</fieldset>
<com:THyperLink
  NavigateUrl="http://localhost/joom/dods/index.php?
page=statusIndicators"
  ImageUrl="~/images/back.png%">
  Text="Back" />
</com:TView>
<com:TView ID="GraphView">
<fieldset class="select">
<legend><h3><center>Status
Indicators</center></h3></legend>
<fieldset class="selectView">
<legend><font color="blue">Generate Status
Graph</font></legend><br><br>
<table>
<tr>
<td>
<com:TLabel ForControl="year2" Text="Year:" />
</td><td>
<com:TDropDownList ID="year2"
  AutoPostBack="false"
  DataTextField="name"
  DataValueField="id"
  />
</td>
</tr>
<tr>
<td>
<com:TLabel ForControl="year2"
  ControlToValidate="year2"
  ErrorMessage="You must make a selection other than
the first option"
  InitialValue="YEAR"
  />
</td>
</tr>

```

```

        FocusOnError="true"
        Display="Dynamic"
        ValidationGroup="chooseGraph"
    />
</td></tr>
<tr>
<td>
<com:TLabel ForControl="disease2" Text="Disease:" />
</td></td>
<com:TDropDownList ID="disease2"
    AutoPostBack="false"
    DataTextField="name"
    DataValueField="id"
    />
<com:TRequiredFieldValidator
    ControlToValidate="disease2"
    ErrorMessage="You must make a selection other than
the first option"
    InitialValue="DISEASE"
    FocusOnError="true"
    Display="Dynamic"
    ValidationGroup="chooseGraph"
    />
</td>
</tr>
<tr>
<td>
<com:TLabel ForControl="graph" Text="Type of
Graph:" />
</td></td>
<com:TDropDownList ID="graph"
AutoPostBack="false">
    <com:TListItem Value="GRAPH" Text="GRAPH" />
    <com:TListItem Value="bar" Text="BarGraph" />
    <com:TListItem Value="density" Text="Density
Probability" />
    <com:TListItem Value="log" Text="Log
Distribution" />
    <com:TListItem Value="line" Text="Line Graph" />
</com:TDropDownList>
<com:TRequiredFieldValidator
    ControlToValidate="graph"
    ErrorMessage="You must make a selection other than
the first option"
    InitialValue="GRAPH"
    FocusOnError="true"
    Display="Dynamic"
    ValidationGroup="chooseGraph"
    />
</td></tr>
</table><br><br>
<center><com:TButton Text="Generate"
ID="Generate" ValidationGroup="chooseGraph"
OnClick="generateGraph"/></center><br><br>
</fieldset>
<com:THyperLink
    NavigateUrl="http://localhost/joom/dods/index.php?
page=statusIndicators"
    ImageUrl="~/images/back.png%"
    Text="Back" />
</fieldset>
</com:TView>
</com:TMultiView>
</com:TForm>

```

## statusIndicators.php

```

<?php
require_once('connect.php');
class statusIndicators extends TPage
{
    public function onLoad($param)
    {
        if (!$this->IsPostBack)
            $this->statusView->ActiveView=$this->IntroView;

            $sql = "SELECT * FROM
`disease` ";
            $disease=Disease::finder()->findAllBySql($sql);
            $data[0]=array('id'=>'DISEASE','name'=>'DISEA
SE');

                $index=1;
                foreach($disease as $row){
                    $id=$row->Disease_id;
                    $name=$row->name;
                    $data[$index]=array('id'=>".$id,'name'=>".
$name);
                        $index++;
                    }
                    $this->disease1->DataSource=$data;
                    $this->disease1->dataBind();
                    $this->disease2->DataSource=$data;
                    $this->disease2->dataBind();
                    $search =
mysql_query("SELECT DISTINCT
year( `date_reported` ) FROM `case_reports` ");

                    $data2[0]=array('id'=>'YEAR','name'=>'YEAR');
                    $index=1;
                    while($row = mysql_fetch_array($search)){
                        $year=$row[0];
                        $data2[$index]=array('id'=>".$year,'name'=>".
$year);
                            $index++;
                        }
                            $this->year1->DataSource=$data2;
                            $this->year1->dataBind();
                            $this->year2->DataSource=$data2;
                            $this->year2->dataBind();
                    }

                public function checkRadio($sender, $param)
                {
                    if($this->Report->Checked){
                        $param->IsValid =true;
                    }else if($this->Graph->Checked){
                        $param->IsValid =true;
                    }else{
                        $param->IsValid =false;
                    }
                }

                public function selectView($sender,$param)
                {
                    if($this->IsValid) // when all validations
succeed

```



```

    {
    if($this->Report->Checked){
        $this->statusView->ActiveView=$this-
>ReportView;
    }else{
        $this->statusView->ActiveView=$this-
>GraphView;
    }
    }
}

    public function createReport($sender,$param)
    {
        $indices=$this->disease1->SelectedIndices;
        foreach($indices as $index){
            $item=$this->disease1-
>Items[$index];
            $disease=$item->Value;
        }

        $indices=$this->year1-
>SelectedIndices;
        foreach($indices as $index){
            $item=$this->year1->Items[$index];
            $year=$item->Value;
        }

        $status="report";
        $this->Response-
>redirect("generateStatus.php?
year=".urlencode($year)."&status=".urlencode($status)."
&disease_id=".urlencode($disease));
    }

    public function generateGraph($sender,
$param)
    {
        $indices=$this->disease2->SelectedIndices;
        foreach($indices as $index){
            $item=$this->disease2-
>Items[$index];
            $disease=$item->Value;
        }

        $indices=$this->year2-
>SelectedIndices;
        foreach($indices as $index){
            $item=$this->year2->Items[$index];
            $year=$item->Value;
        }

        $indices=$this->graph-
>SelectedIndices;
        foreach($indices as $index){
            $item=$this->graph->Items[$index];
            $graph=$item->Value;
        }

        $status="graph";
        $this->Response-
>redirect("generateStatus.php?
year=".urlencode($year)."&status=".urlencode($status)."
&disease_id=".urlencode($disease)."&graph=".urlencode
($graph));
    }
}

```

```

}
?>

```

## updateDisease.page

```

<com:TForm>
<h1>Diseases</h1>

<com:TDataGrid
Width="700px"
CellPadding="2"
ID="DataGrid"
DataKeyField="ID"
AutoGenerateColumns="false"
HeaderStyle.BackColor="black"
HeaderStyle.ForeColor="white"
ItemStyle.BackColor="#BFCFFF"
ItemStyle.Font.Italic="true"
AlternatingItemStyle.BackColor="#E6ECFF"
EditItemStyle.BackColor="lightyellow"
OnItemCreated="itemCreated"
OnEditCommand="editItem"
OnUpdateCommand="saveItem"
OnCancelCommand="cancelItem"
OnDeleteCommand="deleteItem"
>

<com:TBoundColumn
ID="ID"
HeaderText="ICD 10 Code"
HeaderStyle.Width="50px"
DataField="ID"
/>
<com:TBoundColumn
ID="name"
HeaderText="Disease Name"
HeaderStyle.Width="100px"
DataField="name"
/>
<com:TBoundColumn
ID="rhigh"
HeaderText="Risk High"
HeaderStyle.Width="50px"
DataField="rhigh"
/>
<com:TBoundColumn
ID="rave"
HeaderText="Risk Average"
HeaderStyle.Width="50px"
DataField="rave"
/>
<com:TBoundColumn
ID="rlow"
HeaderText="Risk Low"
HeaderStyle.Width="50px"
DataField="rlow"
/>
<com:TEditCommandColumn
HeaderText="Edit"
HeaderStyle.Width="50px"
UpdateText="Save"
ItemStyle.HorizontalAlignment="Center"
ItemStyle.Font.Italic="false"
/>

```

```

<com:TButtonColumn
  ID="DeleteColumn"
  HeaderText="Delete"
  HeaderStyle.Width="50px"
  ItemStyle.HorizontalAlign="Center"
  ItemStyle.Font.Italic="false"
  Text="Delete"
  CommandName="delete"
/>
</com:TDataGrid>
</com:TForm>
<com:THyperLink
  NavigateUrl="http://localhost/joom/updateRecord2.php"
  ImageUrl="~/images/back.png%">
  Text="Back" />

```

## updateDisease.php

```

<?php
require_once('connect.php');
class updateDisease extends TPage
{
  private $_data=null;

  protected function getData()
  {
    if($this->_data===null)
      $this->loadData();
    return $this->_data;
  }

  protected function loadData()
  {
    if(($this->_data=$this->getViewState('Data',null))===null)
    {
      $sql = "SELECT * FROM `disease`";
      $disease=Disease::finder()->findAllBySql($sql);
      $index=0;
      foreach($disease as $row){
        $id=$row->Disease_id;
        $name=$row->name;
        $rhigh=$row->risk_high;
        $rave=$row->risk_ave;
        $rlow=$row->risk_low;
        $data[$index]=array('ID'=>".$id','name'=>".
$name,'rhigh'=>".$rhigh,'rave'=>".$rave,'rlow'=>".
$rlow);
        $index++;
      }

      $this->_data=$data;
      $this->saveData();
    }
  }

  protected function saveData()
  {
    $this->setViewState('Data',$this->_data);
  }
}

```

```

protected function updateDisease($isbn,$id,$name,
$rhig,$rave,$rlow,$in)
{
  $finder=Disease::finder();
  if($this->_data===null)
    $this->loadData();
  $updateRow=null;
  //update database
  $disease = $finder->findByPk($isbn);
  $disease->name=$name;
  $disease->risk_high=TPropertyValue::ensureInteger($rhig);
  $disease->risk_ave=TPropertyValue::ensureInteger($rave);
  $disease->risk_low=TPropertyValue::ensureInteger($rlow);
  $disease->save(); //update
  it.
  $query=mysql_query("UPDATE
`dods`.`disease` SET `Disease_id` = '$id' WHERE
`Disease_id` = '$isbn' LIMIT 1 ;");
  $updateRow=&$this->_data[$in];
  if($updateRow!==null)
  {
    $updateRow['ID']=$id;
    $updateRow['name']=$name;
    $updateRow['rhigh']=TPropertyValue::ensureInteger($rhig);
    $updateRow['rave']=TPropertyValue::ensureInteger($rave);
    $updateRow['rlow']=TPropertyValue::ensureInteger($rlow);
    $this->saveData();
  }
}

protected function deleteDisease($isbn,$index)
{
  $finder = Disease::finder();
  if($this->_data===null)
    $this->loadData();
  $deleteIndex=-1;
  //delete in database
  $finder->deleteByPk($isbn);
  $deleteIndex=$index;

  if($deleteIndex>=0)
  {
    unset($this->_data[$deleteIndex]);
    $this->saveData();
  }
}

public function onLoad($param)
{
  parent::onLoad($param);
  if(!$this->IsPostBack)
  {
    $this->DataGrid->DataSource=$this->Data;
    $this->DataGrid->dataBind();
  }
}

```

```

}

public function itemCreated($sender,$param)
{
    $item=$param->Item;
    if($item->ItemType==='EditItem')
    {
        // set column width of textboxes

        $item->name->TextBox->Columns=20;
        $item->rhigh->TextBox-
>Columns=5;
        $item->rave->TextBox-
>Columns=5;
        $item->rlow->TextBox-
>Columns=5;
    }
    if($item->ItemType==='Item' || $item-
>ItemType==='AlternatingItem' || $item-
>ItemType==='EditItem')
    {
        // add an alert dialog to delete buttons
        $item->DeleteColumn->Button->Attributes-
>onclick='if(!confirm('\Are you sure you want to delete
this Disease?')) return false;';
    }
}

public function editItem($sender,$param)
{
    $this->DataGrid->EditItemIndex=$param->Item-
>ItemIndex;
    $this->DataGrid->DataSource=$this-
>Data;
    $this->DataGrid->dataBind();
}

public function saveItem($sender,$param)
{
    $in=$param->Item->ItemIndex;
    $item=$param->Item;
    $this->updateDisease(
        $this->DataGrid->DataKeys[$item->ItemIndex],
// ID
        $item->ID->TextBox-
>Text, //ID
        $item->name->TextBox-
>Text, // name
        $item->rhigh->TextBox->Text, //
risk_high
        $item->rave->TextBox->Text, //
risk_average
        $item->rlow->TextBox->Text, // risk_low
        $in
    );
    $this->DataGrid->EditItemIndex=-1;
    $this->DataGrid->DataSource=$this->Data;
    $this->DataGrid->dataBind();
}

public function cancelItem($sender,$param)
{
    $this->DataGrid->EditItemIndex=-1;
    $this->DataGrid->DataSource=$this->Data;
    $this->DataGrid->dataBind();
}

```

```

public function deleteItem($sender,$param)
{
    $index=$param->Item->ItemIndex;
    $this->deleteDisease($this->DataGrid-
>DataKeys[$param->Item->ItemIndex],$index);
    $this->DataGrid->EditItemIndex=-1;
    $this->DataGrid->DataSource=$this-
>Data;
    $this->DataGrid->dataBind();
}
}
?>

```

## updateRecord.page

```

<h1>Case Reports</h1>
<com:TForm>
<com:TDataGrid
    Width="700px"
    CellPadding="2"
    ID="DataGrid"
    DataKeyField="ID"
    AllowPaging="true"
    PageSize="10"
    PagerStyle.Mode="Numeric"
    PagerStyle.HorizontalAlign="Left"
    AutoGenerateColumns="false"
    HeaderStyle.BackColor="black"
    HeaderStyle.ForeColor="white"
    ItemStyle.BackColor="#BFCFFF"
    ItemStyle.Font.Italic="true"
    AlternatingItemStyle.BackColor="#E6ECFF"
    EditItemStyle.BackColor="lightyellow"
    OnItemCreated="itemCreated"
    OnEditCommand="editItem"
    OnUpdateCommand="saveItem"
    OnCancelCommand="cancelItem"
    OnDeleteCommand="deleteItem"
    OnPageIndexChanged="changePage"
    OnPagerCreated="pagerCreated"
>
<com:TBoundColumn
    ID="ID"
    HeaderText="Case Number"
    HeaderStyle.Width="50px"
    DataField="ID"
/>
<com:TBoundColumn
    ID="patnumber"
    HeaderText="Patient Number"
    HeaderStyle.Width="50px"
    DataField="patnumber"
/>
<com:TBoundColumn
    ID="age"
    HeaderText="Age"
    HeaderStyle.Width="50px"
    DataField="age"
/>
<com:TBoundColumn
    ID="gender"
    HeaderText="Gender"
    HeaderStyle.Width="50px"
    DataField="gender"

```

```

/>
    <com:TDropDownListColumn
    ID="city"
    HeaderText="City"
    DataTextField="city"
    >
<com:TListItem Value="LP"/>
<com:TListItem Value="MNLA"/>
<com:TListItem Value="PSG"/>
<com:TListItem Value="PAS"/>
<com:TListItem Value="QUE"/>
<com:TListItem Value="TAG"/>
<com:TListItem Value="MAL"/>
<com:TListItem Value="MAR"/>
<com:TListItem Value="PAT"/>
<com:TListItem Value="VAL"/>
<com:TListItem Value="MAK"/>
<com:TListItem Value="SAN"/>
<com:TListItem Value="NAV"/>
<com:TListItem Value="PQUE"/>
<com:TListItem Value="MAN"/>
<com:TListItem Value="MUN"/>
</com:TDropDownListColumn>

<com:TBoundColumn
    ID="sen_id"
    HeaderText="Sentinel ID"
    HeaderStyle.Width="50px"
    DataField="sen_id"
    />
    <com:TBoundColumn
    ID="disease"
    HeaderText="Disease"
    HeaderStyle.Width="50px"
    DataField="disease"
    />

    <com:TDropDownListColumn
    ID="condition"
    HeaderText="Status"
    DataTextField="condition"
    >
    <com:TListItem Value="INFECTED" />
    <com:TListItem Value="CURED" />
    <com:TListItem Value="DECEASED" />
</com:TDropDownListColumn>
    <com:TBoundColumn
    ID="date"
    HeaderText="Date Reported"
    HeaderStyle.Width="50px"
    DataField="date"
    />
<com:TEditCommandColumn
    HeaderText="Edit"
    HeaderStyle.Width="50px"
    UpdateText="Save"
    ItemStyle.HorizontalAlign="Center"
    ItemStyle.Font.Italic="false"

    />
<com:TButtonColumn
    ID="DeleteColumn"
    HeaderText="Delete"
    HeaderStyle.Width="50px"
    ItemStyle.HorizontalAlign="Center"

```

```

    ItemStyle.Font.Italic="false"
        Text="Delete"
    CommandName="delete"
    />
</com:TDataGrid>
</com:TForm>
<com:THyperLink
    NavigateUrl="http://localhost/joom/dods/index.php?
page=searchRecord"
    ImageUrl="~/images/back.png%">
    Text="Back" />.

```

## updateRecord.php

```

<?php
class updateRecord extends TPage
{
    private $_data=null;

    protected function getData()
    {
        if($this->_data===null)
            $this->loadData();
        return $this->_data;
    }

    protected function loadData()
    {
        //load data from the database
        if(($this->_data=$this-
>getViewState('Data',null))===null)
        {
            $patNumber=$_GET['pat_number'];
            $age=$_GET['age'];
            $address=$_GET['address'];
            $city=$_GET['city'];
            $disease_id=$_GET['disease'];
            $condition=$_GET['condition'];
            $sen_id=$_GET['sentinel_id'];
            $gender=$_GET['sex'];
            $daterecorded=$_GET['daterecorded'];

            if($address==""){
                if($age==""){
                    $query = "SELECT *
FROM `case_reports` where `gender` LIKE '%$gender%'
&& `patient_number` LIKE '%$patNumber%' &&
`Sentinel_id` LIKE '%$sen_id%' && `city` LIKE '%
$city%' && `condition` LIKE '%$condition%' && `age`
LIKE '%$age%' && `Disease_id` LIKE '%$disease_id%'
&& `date_reported` LIKE '%$daterecorded%'";

                }else{
                    $query = "SELECT *
FROM `case_reports` where `gender` LIKE '%$gender%'
&& `patient_number` LIKE '%$patNumber%' &&
`Sentinel_id` LIKE '%$sen_id%' && `city` LIKE '%
$city%' && `condition` LIKE '%$condition%' && `age`
= '$age' && `Disease_id` LIKE '%$disease_id%' &&
`date_reported` LIKE '%$daterecorded%'";

                }
            }
        }
    }
}

```

```

    }else{
        if($age==""){
            $query = "SELECT *
FROM `case_reports` where `gender` LIKE '%$gender%'
&& `patient_number` LIKE '%$patNumber%' &&
`Sentinel_id` LIKE '%$sen_id%' && `city` LIKE '%
$city%' && `condition` LIKE '%$condition%' && `age`
LIKE '%$age%' && `Disease_id` LIKE '%$disease_id%'
&& `date_reported` LIKE '%$daterecorded%";
        }else{
            $query = "SELECT *
FROM `case_reports` where `gender` LIKE '%$gender%'
&& `patient_number` LIKE '%$patNumber%' &&
`Sentinel_id` LIKE '%$sen_id%' && `city` LIKE '%
$city%' && `condition` LIKE '%$condition%' && `age`
= '$age' && `Disease_id` LIKE '%$disease_id%' &&
`date_reported` LIKE '%$daterecorded%";
        }
    }

    $case=CaseReports::finder()-
>findAllBySql($query);
    $index=0;
    foreach($case as $row){
        $id=$row->case_number;
        $patNumber=$row-
>patient_number;
        $age=$row->age;
        $gender=$row->gender;
        $city=$row->city;
        $sen_id=$row-
>Sentinel_id;
        $disease=$row-
>Disease_id;

        $condition=$row->condition;
        $date=$row-
>date_reported;
        $data[$index]=array('ID'=>".$id,'patnumber'=>".
$patNumber,'age'=>".$age,'disease'=>".
$disease,'condition'=>". $condition,'date'=>".
$date,'sen_id'=>". $sen_id,'gender'=>". $gender,'city'=>".
$city);
        $index++;
    }

    if($index==0){
        echo "<h3>Status:</h3><h4><font
color='red'>Case Record Not Found!!</font></h4>";
    }

    $this->_data=$data;
    $this->saveData();
}

protected function saveData()
{
    $this->setViewState('Data',$this->_data);
}

```

```

protected function updateCase($isbn,$patnumber,$age,
$gender,$city,$disease,$sen,$condition,$date,$indx)
{
    if($this->_data===null)
        $this->loadData();
    $updateRow=null;
    $finder=CaseReports::finder();
    foreach($this->_data as $index=>$row){
        if($row['ID']===$isbn){
            $case = $finder-
>findByPk($isbn);
            $case->patient_number=$patnumber;
            $case-
>age=TPROPERTYVALUE::ensureInteger($age);
            $case->gender=$gender;
            $case->city=$city;
            $case-
>Disease_id=$disease;
            $case->Sentinel_id=$sen;
            $case-
>condition=$condition;
            $case-
>date_reported=$date;
            $case->save(); //update it.
        }
    }
    $updateRow=&$this->_data[$indx];
    if($updateRow!==null)
    {
        $updateRow['ID']=$isbn;
        $updateRow['patnumber']=$patnumber;

        $updateRow['gender']=$gender;
        $updateRow['city']=$city;

        $updateRow['sen_id']=$sen;

        $updateRow['condition']=$condition;
        $updateRow['date']=$date;

        $updateRow['disease']=$disease;
        $updateRow['age']=TPROPERTYVALUE::ensureInteg
er($age);
        $this->saveData();
    }
}

protected function deleteRecord($isbn,$tmp)
{
    $finder = CaseReports::finder();

    if($this->_data===null)
        $this->loadData();
    $deleteIndex=-1;
    //delete in database
    $finder->deleteByPk($isbn);
    $deleteIndex=$tmp;

    if($deleteIndex>=0)
    {
        unset($this->_data[$deleteIndex]);
        $this->saveData();
    }
}

```

```

    }
}

public function onLoad($param)
{
    parent::onLoad($param);
    if(!$this->IsPostBack)
    {
        $this->DataGrid->DataSource=$this->Data;
        $this->DataGrid->dataBind();
    }
}

public function itemCreated($sender,$param)
{
    $item=$param->Item;
    if($item->ItemType==='EditItem')
    {
        // set column width of textboxes
        $item->ID->TextBox->Columns=3;
        $item->patnumber->TextBox->Columns=3;
        $item->age->TextBox-
>Columns=3;
        $item->gender->TextBox-
>Columns=3;
        $item->sen_id->TextBox-
>Columns=5;
        $item->disease->TextBox-
>Columns=5;
        $item->date->TextBox-
>Columns=8;
    }
    if($item->ItemType==='Item' || $item-
>ItemType==='AlternatingItem' || $item-
>ItemType==='EditItem')
    {
        // add an alert dialog to delete buttons
        $item->DeleteColumn->Button->Attributes-
>onclick='if(!confirm('\Are you sure you want to delete
this Record?\')) return false;';
    }
}

public function editItem($sender,$param)
{
    $this->DataGrid->EditItemIndex=$param->Item-
>ItemIndex;
    $this->DataGrid->DataSource=$this->Data;
    $this->DataGrid->dataBind();
}

public function saveItem($sender,$param)
{
    $index=$param->Item->ItemIndex;
    $item=$param->Item;
    $this->updateCase(
        $this->DataGrid->DataKeys[$param->Item-
>ItemIndex], // case number
        $item->patnumber->TextBox->Text, //
patient number
        $item->age->TextBox->Text, // age
        $item->gender->TextBox->Text, //gender

```

```

        $item->city->DropDownList->SelectedValue,
        // city
        $item->disease->TextBox->Text, //
Disease_id
        $item->sen_id->TextBox-
>Text, //Sentinel_id
        $item->condition-
>DropDownList->SelectedValue, //condition
        $item->date->TextBox-
>Text, //date_reported
        $index
    );
    $this->DataGrid->EditItemIndex=-1;
    $this->DataGrid->DataSource=$this-
>Data;
    $this->DataGrid->dataBind();
}

public function cancelItem($sender,$param)
{
    $this->DataGrid->EditItemIndex=-1;
    $this->DataGrid->DataSource=$this->Data;
    $this->DataGrid->dataBind();
}

public function deleteItem($sender,$param)
{
    $index=$param->Item->ItemIndex;
    $this->deleteRecord($this-
>DataGrid->DataKeys[$param->Item->ItemIndex],
    $index);
    $this->DataGrid->EditItemIndex=-1;
    $this->DataGrid->DataSource=$this-
>Data;
    $this->DataGrid->dataBind();
}

public function changePage($sender,$param)
{
    $this->DataGrid->CurrentPageIndex=$param-
>NewPageIndex;
    $this->DataGrid->DataSource=$this->Data;
    $this->DataGrid->dataBind();
}

public function pagerCreated($sender,$param)
{
    $param->Pager->Controls->insertAt(0,'Page: ');
}
}
?>

```

## updateRecord2.php

```

<html>
<head>
<title>Update Records</title>
</head>
<body>
<?php
ini_set( "display_errors", 0);
define( '_JEXEC', 1 );

define('JPATH_BASE', dirname(__FILE__));

```

```

define( 'DS', DIRECTORY_SEPARATOR );

require_once ( JPATH_BASE
.DS.'includes'.DS.'defines.php' );
require_once ( JPATH_BASE
.DS.'includes'.DS.'framework.php' );

require('libraries/joomla/factory.php');

// initialize the application
$mainframe =& JFactory::getApplication('site');
$mainframe->initialise();
$user =& JFactory::getUser();
$user_id = $user->get('id');
$user_type=$user->get('usertype');

echo "<table width='200' border='0' cellpadding='10'>";
if($user_type=="Super Administrator"){
echo "<tr>";
echo "<td><a href='dods/index.php?
page=addDisease'><img src='editIcon.png' width='93'
height='48' alt='add' /></a></td>";
echo " <td>Add Disease Record</td>";
echo "</tr>";
echo "<tr>";
echo "<td><a href='dods/index.php?
page=updateDisease'><img src='editIcon.png' width='93'
height='48' alt='update' /></a></td>";
echo " <td>Update Disease Record</td>";
echo "</tr>";
}else{
echo " <tr>";
echo "<td><a href='dods/index.php?
page=searchRecord&status=Search'><img
src='search.png' width='93' height='48' alt='search'
/></a></td>";
echo " <td>Search Patient</td>";
echo "</tr>";
echo "<tr>";
echo "<td><a href='dods/index.php?
page=addCase'><img src='editIcon.png' width='93'
height='48' alt='add' /></a></td>";
echo "<td>Add Record</td>";
echo "</tr>";
echo " <tr>";
echo "<td><a href='dods/index.php?
page=searchRecord&status=Record'><img
src='editIcon.png' width='93' height='48' alt='add'
/></a></td>";
echo " <td>Update Case Record</td>";
echo "</tr>";
}
echo "</table>";
?>
</body>

```

</html>

## XI. ACKNOWLEDGEMENT

Finally I reached this point. This is the end of my journey in college and the start of a new chapter in my life. This is the result of my hard work, dedication and perseverance. But I know that I cannot do this without the people that helped and guided me throughout the years.

First and foremost, I dedicate this triumphant moment of mine to God. He proved to me that He is always by my side in any obstacles I encountered and still continues to provide me the courage to push through and never give up. He showers me his love everyday and for that I am very grateful.

I also dedicate this moment to my ever supportive family especially to my parents. The most wonderful and loving parents that I could ever have. Who never give up with me and always reminds me that no matter what, I should finished my studies because that is the most important thing that they can give me. Despite of the problems that we altogether faced, they stay right beside me to put me back right on the track again. I love them so much and I know they too love me as well. I cannot ask for more because God gave me a wonderful family that I have right now.

I also want to offer my deepest gratitude to all my teachers for devoting their time to share their wisdom to us. To my very dedicated and supportive SP adviser Sir Solano who was so patient with me throughout my SP time. I am grateful for giving his spare time to me when I seek for some advice regarding my SP and for preparing me for my presentations. I am very fortunate that he was my adviser for I cannot picture myself under other's supervision (hehehehe...). My life at UP became a very learning experience to me because of the best teachers I had.

Who am I to forget all the friends that I had during my college years. They are the ones whom I share almost my entire stay at school. Throughout the struggles that I met, there will always at least one friend that I can share it with. Projects, machine problems, assignments and other school stuff are the common topics of our discussions. But we are not that nerdy type, except for a few☺, we also share some experiences that a typical person in an early adulthood stage does. We always hang out at the mall to eat and to do our favourite past time, videoke. We regularly go to a particular place at Robinson and because of that, I feel that the personnel in charge of the machines already recognized us. I am very happy that I met this wonderful group of people with whom I share some common interests (the Irregz) and to other people (CompSci people , DevStud people and others) which made my college life this awesome.



I also want to thank all UPM people, the staff and personnel, Ate Eden for letting me borrow some resources in RH114. I also want to thank kuya who is in charge of the Xerox machines, to the ladies of the cafeteria, to the librarian, to people I met when I became a student assistant, to the OCS staff, to the people in charge of the laboratories, to the staff of each department and other people I fail to mention. I offer my gratitude to you.

I don't know how to express my emotions but I am more than grateful to share this moment with you. I hope that this will be the beginning of a more exciting life for me. And I hope that a number of moments like this will come to me in the future. I know that we will meet again someday and perhaps when that day comes, it will be like nothing had changed. I will never forget all the lessons that this school had taught me and I will use those to develop myself and become a successful individual and a productive citizen of this country. Again, thank you very very very much and may God Bless us all.