A CLOUD-BASED PROTOTYPE IMPLEMENTATION OF A DISEASE OUTBREAK NOTIFICATION SYSTEM

Farag Azzedin, Salahadin Mohammed, Jaweed Yazdani, Taher Ahmed Ghaleb and Mustafa Ghaleb

King Fahd University of Petroleum and Minerals
Dhahran 31261, Saudi Arabia

ABSTRACT

This paper describes the design, prototype implementation and performance characteristics of a Disease Outbreak Notification System (DONS). The prototype was implemented in a hybrid cloud environment as an online/real-time system. It detects potential outbreaks of both listed and unknown diseases. It uses data mining techniques to choose the correct algorithm to detect outbreaks of unknown diseases. Our experiments showed that the proposed system has very high accuracy rate in choosing the correct detection algorithm. To our best knowledge, DONS is the first of its kind to detect outbreaks of unknown diseases using data mining techniques.

KEYWORDS

Disease Outbreak Notification System, Saudi Arabia, Prototype, Outbreak

1. INTRODUCTION

Outbreaks such as MERS, SARS [1, 2], the threat of bio-terrorism (e.g. anthrax) [3] and Mad Cow diseases (BSE) [4] as well as the recent different strains of "bird flu" or Influenza (i.e., H1N1 and H5N1) [5] are the most intriguing and complex phenomena that confront scientists in the field of microbiology, virology and epidemiology [6, 7, 8]. The ability of these viruses to mutate and evolve is one of the acute mysteries that puzzle health officials, who are trying to find out the root cause of worldwide pandemics since the late 1880s. Pandemics occur when small changes in the virus over a long period of time eventually “shift” the virus into a whole new subtype, leaving the human population with no time to develop a new immunity.

Unlike bacteria, viruses are sub-microscopic and do not have a cellular structure [9]. Their essential component is genetic material—either DNA (Deoxyribonucleic Acid) or RNA (Ribonucleic Acid)—that allows them to take control of a host cell. Viruses reproduce by invading a host cell and directing it to produce more viruses that eventually burst out of the cell, killing it in the process.

Therefore, a tremendous and an unforeseen threat could mark the start of a global outbreak given the above mentioned scenarios, namely (a) the ability of the virus to shift into a whole new subtype, (b) the time shortage of the human population to develop a new immunity, (c) the limitation in terms of the effect of immunization, and (d) the viruses’ ability to change to a form that is highly infectious for humans and spreads easily from person to person. Furthermore, as outlined by World Health Organization (WHO) and the World Organization for Animal Health (OIE) [10, 11], the international standards, guidelines and recommendations in an event of an outbreak state that member countries are obliged to notify within 24 hours epidemiological

DOI : 10.5121/ijcsea.2015.5202

information with regards to occurrence/reoccurrence of listed notifiable diseases, the occurrence of a new strain of a listed disease, a significant change in the epidemiology of a listed disease, or the detection of an emerging disease.

The disease outbreak reports within specific time limits are required to be sent on the presence or the evolution of the listed diseases and their strains. It is apparent that the increasing threat of disease outbreak highlights the need to provide timely and accurate information to public health professionals across many jurisdictional and organizational boundaries. Also, the increasing frequency of biological crises, both accidental and intentional, further illustrates that Disease Outbreak Notification System (DONS) needs to be in place to meet the challenges facing today's society. Such DONS should prevent, prepare, and respond to an outbreak having the potential to affect humans and/or animals. The surveillance and management roles and responsibilities should be identified for a unified approach that considers humans, domestic animals, and wildlife.

The importance of such a system to Kingdom of Saudi Arabia (KSA) is tremendous. It is well known that the KSA is a vital hub for two major events: (a) around 2.5 million pilgrims from more than 160 countries take part in the Hajj in the holy city of Makkah every year during a very short time spanning only four weeks, and (b) around six million Muslims perform Umrah every year. These two events make KSA a fertile place for outbreaks as people fly into KSA from overseas every year. As such, a contagious disease outbreak overseas such as MERS, H1N1 or H5N1, whether natural or due to bioterrorism can spread long before an epidemic is recognized. This will not only jeopardize people's health but also impact worldwide because pilgrims and those performing religious duties will eventually return to their countries and can spread the disease worldwide.

This paper, as stated previously, focuses on the design and prototype implementation of a cloud-based DONS. The rest of the paper is organized as follows. Section 2 describes the feature of the system. Section 3 provides implementation detail of the prototype system implemented. Section 4 lists the performance characteristics and results from the prototype system implemented. Finally, Section 5 outlines the conclusions reached from our work.

2. THE PROTOTYPE CLOUD-BASED SYSTEM

In this section, we describe the features included in the prototype implementation of the Cloud-based DONS. This system is designed to detect potential disease outbreaks and notify preregistered experts about the outbreak. In order to come up with a sound design for DONS, our effort was concentrated on the functionalities and features of a typical and generic DONS. Using the taxonomy and critical features generated in that effort, we mapped those features and functionalities to the DONS design. Based on the taxonomy of various DONS implementation worldwide that we studied, we have identified the critical features that our DONS should have. These features are shown in Tables 1 and 2.

Table 1: Core Features of the DONS

	Collection & Analysis						**Core Functions**							
	Collection				Analysis		Case-Level		Group-Level			Notifications		
System Name	Stakeholders	Data Sources	Collection Strategy	Data Formats	Computation Detection Algorithms	Statistical Analysis	Detection	Confirm & Register	Signal Extraction	Interpretation	Reporting	Communication Procedures	Target Entities	Output/Message Formats
Prototype Cloud-based DONS	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	✓

Table 2: Additional features of the existing DONS

	System Features														**Support Functions & System Interfaces**						
	Attributes														Support Functions					Networking& Partnerships	
System Name	Completeness	Timeliness	Usefulness	Real-time	Coverage	Portability	Sensitivity	Availability	Accessibility	Usability	Interface Design	Mobility	Flexibility	Specificity	Standards	Training	Communication	Monitoring & Evaluation	System Administration	Experts & Institutions	Coordination	Feedback
Prototype Cloud-based DONS	✓	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	×	✓

3. DONS PROTOTYPE IMPLEMENTATION

A prototype was implemented as a proof-of-concept (PoC). The prototype was a hybrid implementation using a multi-tier architecture spread across physical hosts and the private research cloud infrastructure at KFUPM. The prototype implementation was thoroughly tested for functional and technical performance with a considerably large dataset of diseases and cases. Integration testing between the various modules of DONS was done as well. The prototype DONS is deployed on the KFUPM Cloud Infrastructure service called KLOUD. KFUPM cloud service offers servers and storage as per required specifications from a wide range of available infrastructure templates.

The detection algorithms used in the DONS were selected and adapted after a thorough study of all algorithms from selected DONS systems. We have used five efficient algorithms that are used in CASE system [16]. These five algorithms can detect isolated cases of known diseases and their potential outbreaks. Our preference for these algorithms was based on the fact that the coverage of the list of known diseases included in DONS is handled by these detection algorithms. We

have also implemented an outbreak detection algorithm for unknown diseases using data mining techniques. Our implementation uses expert epidemiologists for consultation purposes to confirm outbreaks.

We have adopted a three-tier system architecture which supports features such as scalability, availability, manageability, and resource utilization. Three-tier architecture - consisting of the presentation tier, application or business logic tier and data tier - is an architectural deployment style that describes the separation of functionality into layers with each segment being a tier that can be located on a physically separate entity. They evolved through the component-oriented approach, generally using platform specific methods for communication instead of a message-based approach. This architecture has different usages with different applications. It can be used in web applications and distributed applications. The strength of this approach in particular is when using this architecture for geographically distributed systems. Since our platform is cloud-based and geographically spread across the Kingdom, we were motivated to use this three tier architecture for our prototype implementation.

In our implementation, the methodology used is as follows:

- The presentation tier is through the network using wired and wireless devices.
- The application or business logic tier is entirely hosted on virtual hardware on the cloud platform
- The data tier is hosted on physical servers

The application development tools in implementation of the DONS are JCreator, PHP, and Java. The database development tools used in the implementation of the DONS are MySQL, Oracle SQLPLUS, and Oracle SQL Developer.

The presentation tier is the topmost level of the application. The presentation layer provides the application's user interface. Typically, this involves the use of Graphical User Interface for smart client interaction, and Web based technologies for browser-based interaction. As shown in the Figure 1, the terminals for primary health centres and experts use the DONS browser-based application for data entry and decision making.

The application tier controls an application's functionality by performing detailed processing. The application or the business logic tier is where mission-critical business problems are solved. The components that make up this layer can exist on a server machine, to assist in resource sharing. These components can be used to enforce business rules, such as business algorithms and data rules, which are designed to keep the data structures consistent within either specific or multiple databases.

Because these middle-tier components are not tied to a specific client, they can be used by all applications and can be moved to different locations, as response time and other rules require. In Figure 1, the web servers and application server constitutes the logic tier. A cluster of web servers and applications servers can be used for load balancing and failover among the cluster nodes.
The data tier consisting of database servers is the actual DBMS access layer. It can be accessed through the business services layer and on occasion by the user services layer. Here information is stored and retrieved. This tier keeps data neutral and independent from application servers or business logic. Giving data its own tier also improves scalability and performance. In Fig. 2, the database servers and the shared storage constitute this tier.

The flow of data in the three tiered architecture is described next. In the presentation layer, the users access the DONS applications over the network through the web browser. The request is securely sent over the network to a firewall. Then the trusted requests from the firewall are

forwarded to load balancers. The firewall ensures that trust relationships between the presentation and application tiers are complied with. The trusted requests are sent to a DNS server for name resolution and to load balancers which are capable of distributing the load across the web servers and manage the network traffic.

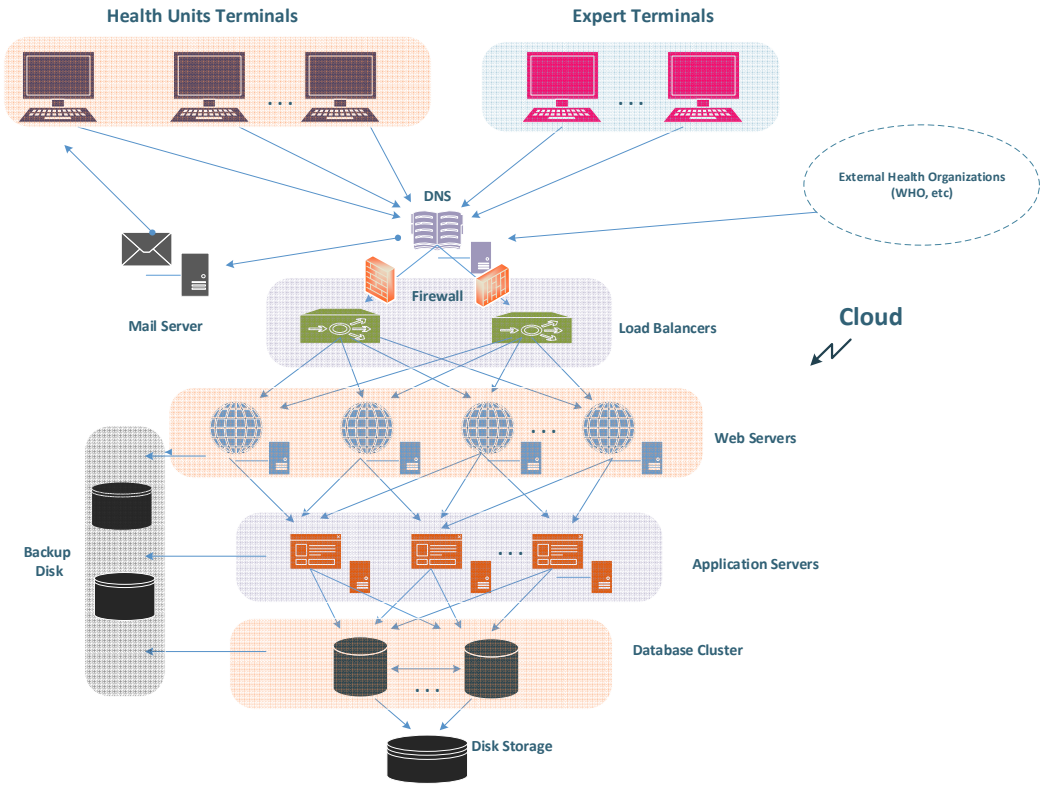


Figure 1 : DONS Prototype System Architecture

In the logic tier of the DONS, web and application servers are deployed to handle all user requests. While the user requests (http or https) are served by the Apache web servers, the application servers handle all the business logic processing and data processing. The data tier provides all the data needed for the logic tier through SQL queries using database specific protocol over TCP/IP. The database tables are maintained by insertion, updating and deleting of data. To avoid loss of data due to data corruption or system failure, the data backup of all the critical servers and databases is performed periodically in a separate storage location. The data retention policy is applied for timely recovery of data in case of any disaster.

3.1 PROTOTYPE APPLICATION TIER

The prototype implementation of the generic DONS architecture, described in the previous section, utilizes the KFUPM research cloud KLOUD and associated hardware and software tools. In this section, we describe all the entities, software and tools used in this tier. A prototype is shown in the
Figure 2 and contains the following entities: Health Unit 1, Health Unit 2, Eastern Province Health Department and MoH (Ministry of Heath), Database System, Experts and Users.
Health Unit 1 and Health Unit 2 are physical units with Windows 2008 Server operating system (DELL OptiPlex 9010 server). Eastern Province Health Department client is a KLOUD virtual machine with Red Hat Linux 6.4 as its operating system. Ministry of Health (MOH) is also a

KLOUD virtual machine with Windows Server 2008 server on it. The database server is again a physical server with Red Hat Linux 6.3 operating system. The Oracle client software was configured on all the servers and clients mentioned in the above figure in order to communicate with each other and to connect to the database server. The web services offered to the clients are configured on an Apache web server on MoH KLOUD virtual server for accessing the application tier of DONS.

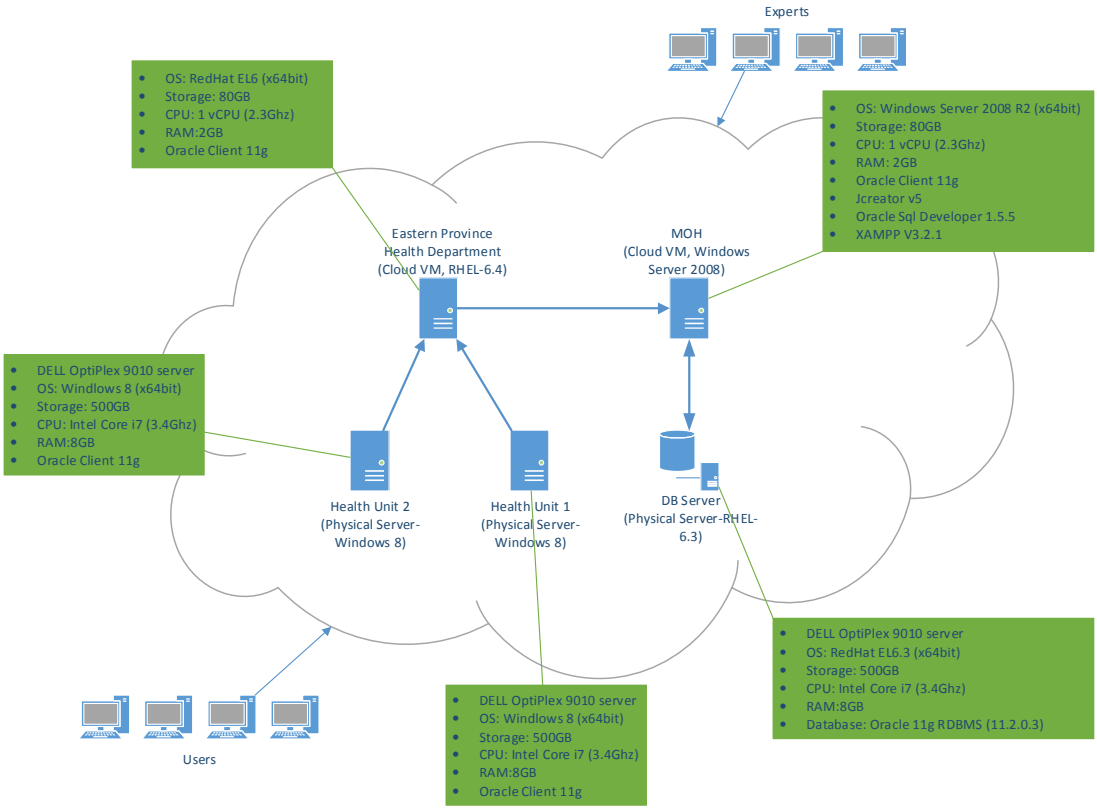


Figure 2 : Prototype System Architecture for the Application Tier

In many disease notification systems, the users had to be manually notified of any new outbreak. In the DONS prototype system, we have developed an application to automate this notification procedure.
Figure 3 shows the data flow of notification delivery system in DONS.

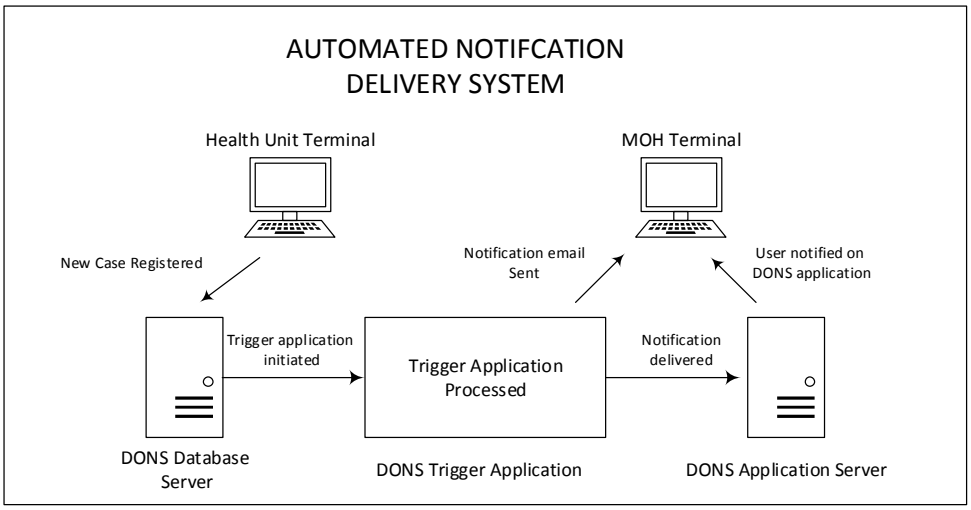


Figure 3 : Data flow in the Notification Delivery System in DONS

Whenever a new disease is registered by the Health Units in the DONS database, a trigger defined in the database runs the trigger application. This application is responsible for delivering the disease notification to the DONS application server over the network, which in turn notifies the appropriate/responsible user in MOH. The trigger application also sends the notification to the email box of the users. The technical details of this implementation of the data flow of notification delivery system in DONS are shown in Figure 4 .

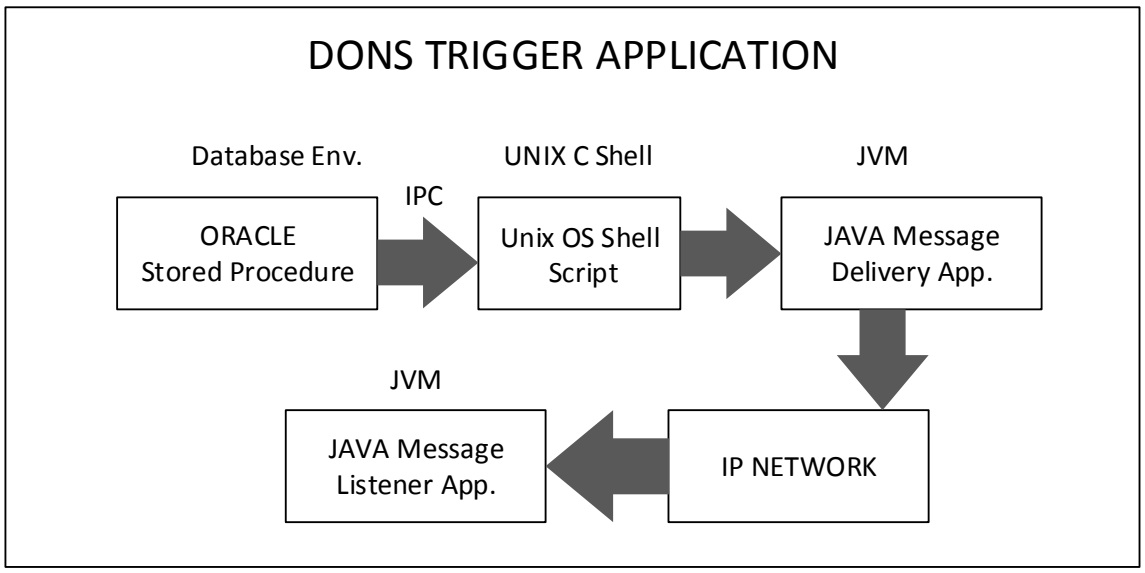


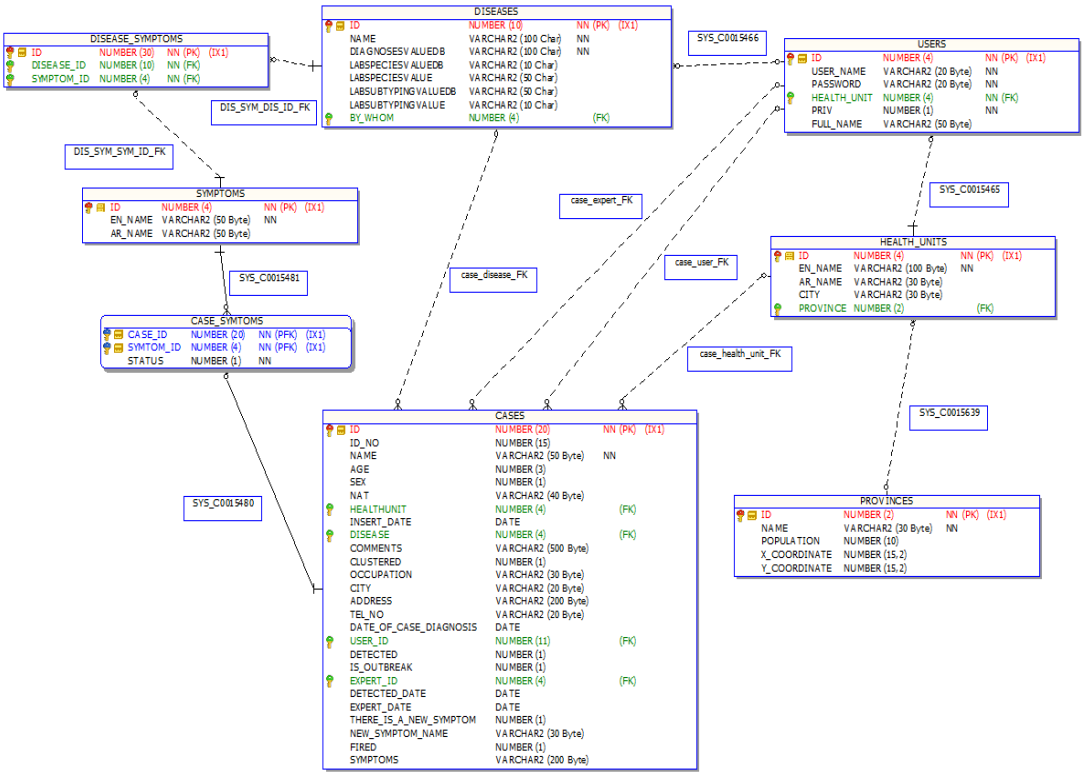
Figure 4 : Technical Implementation of the Notification Delivery System in DONS

The entire process implemented in the trigger application shown in Figure 4 is described here. A new disease case is registered in the database by inserting a row of data in a CASE table. A trigger is defined in the DONS database which executes a stored procedure whenever a row of data is inserted in the CASE table. The stored procedure from the database environment calls a shell script present in the operating system (OS) environment through an Inter Process Communication (IPC).

This shell script is a batch program which can perform multiple tasks sequentially. The shell script runs a Java-based message delivery application within a JVM and also sends the notification e-mail to the end users. The message delivery application sends the notification message over the network to DONS application residing on the application server. At the application server, a message listener application continuously monitors an application port for notification message reception.

3.2 PROTOTYPE DATA TIER

The data tier implemented in the DONS prototype implementation went through two phases of development. Initially, the open database development platform MySQL was used in the prototype development. At a later stage, the entire database with all the associated objects such as the database schema, stored procedures, triggers etc., were migrated to the Oracle database platform. The database server is configured with Oracle Version 11gR2 relational DBMS and was created database using DBCA utility. The data from MySQL database was migrated to the Oracle database using SQL Developer utility offered by Oracle. The complete DONS database schema on the Oracle platform is shown in
Figure 5.



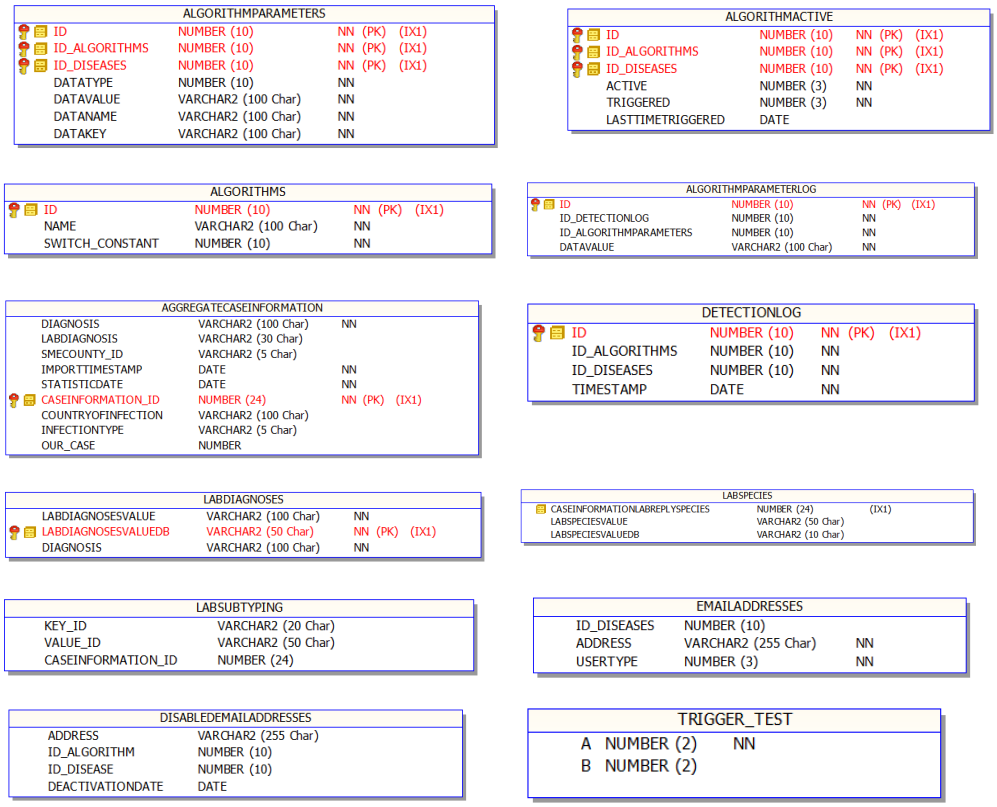


Figure 5 : Complete DONS Database schema on the Oracle platform

4. DONS PERFORMANCE CHARACTERISTICS & RESULTS

4.1 CONFUSION MATRIX

We performed a number of experiments to test the performance of the proposed system. Confusion matrix (also called the contingency table) is a table layout that demonstrates the performance of an algorithm, especially for classification problems [17]. As shown in Figure 4, table of confusion comprises of two columns and two rows. Columns and rows represent the instances of actual and predicted classes of values. Each cell in this table reports one of the following numbers: *true positives, false positives*, *true negatives,* and *false negatives*.

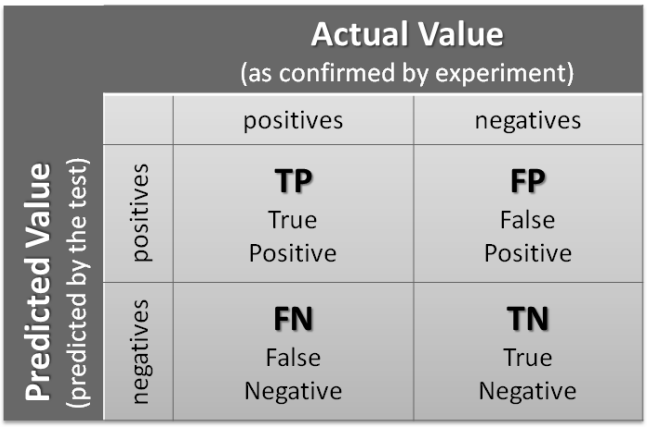


Figure 4: Confusion Matrix

In general, Positive means identified and Negative means rejected. On the other hand, True means correctly while False means incorrectly. Therefore:

- **T**rue **P**ositive: represents the cases that are correctly identified.
- **T**rue **N**egative: represents the cases that are correctly rejected.
- **F**alse **P**ositive: represents the cases that are incorrectly identified.
- **F**alse **N**egative: represents the cases that are incorrectly rejected.

4.2 PERFORMANCE METRICS

To measure the performance of a classifier, different common performance metrics can be used to identify major aspects of the performance. These metrics can be calculated using the values given in the confusion matrix were the major diagonal provides the correct decisions, since its numbers indicate the errors between the available classes [18].
In this paper, we only focus on three performance statistical metrics, naming Sensitivity, Specificity, and Accuracy.

4.2.1 Sensitivity

Sensitivity is a statistical metric that measures the fraction of actual positives that are correctly identified as such. Essentially, it complements the false negative rate. In other fields, it is called the recall rate or the true positive rate; and is calculated using the following equation:

Sensitivity = Positive Hits / Total Positives
= True Positives / (True Positives + False Negative)
= TP / (TP + FN)

4.2.2 Specificity

Specificity is a statistical metric that measures the fraction of negatives that are correctly identified as such. Essentially, it complements the false positive rate. It is sometimes called the true negative rate; and is calculated using the following equation:

Specificity = Negative Hits / Total Negatives
= True Negative / (True Negative + False Positives)
= TN / (TN + FP)

4.2.3 Accuracy

Accuracy is the fraction of correct predictions regardless of being positives or negatives. This metric may not be enough for measuring the performance, especially when negative cases are much higher that positive ones [18]. So, other metrics described above are needed to measure the performance from different aspects. Accuracy is calculated using the following equation:

Accuracy = Total Hits / Number of Entries in the dataset
= (True Positives + True Negative) / (Positives + Negatives)
= (TP + TN) / (P + N)

4.3 DONS PERFORMANCE RESULTS

In order to measure the performance of DONS, we have prepared our dataset to contain 1700 cases, each with unknown disease (say U). Each of these unknown diseases have a set of *S* symptoms that is similar to other known disease's (say K) symptoms in the database, such that: S(U) = S(K) – (1, 2 or 3).

We accomplished such experiment by cloning 1700 cases of 17 different diseases, 100 cases per each disease. Then, we altered the systems in three stages. The first stage, we altered only one symptom per case to make its disease unknown, in way that (for example) a case with 10 symptoms will be with 9 symptoms, and so on. After that, we applied the clustering algorithm to match each unknown disease with the nearest known disease according to their symptoms. Eventually, was have created the confusion matrix for each disease, and then we have calculated the sensitivity, specificity, and accuracy metrics accordingly.

The second and third phases were performed similarly but by modifying 2 and 3 symptoms, respectively. All confusion matrices, their respective calculated performance metrics along with the average results are demonstrated in Table 3.

Table 3: Confusion Matrices of DONS Performance (17 diseases)

Disease Name	Altering 1 symptom	Altering 2 symptoms	Altering 3 symptoms
Botulism	Predicted / Actual: TP 100, FP 0, FN 0, TN 1600	Predicted / Actual: TP 100, FP 14, FN 0, TN 1586	Predicted / Actual: TP 100, FP 61, FN 0, TN 1539
True positive rate (TPR, Sensitivity, Recall)	1	1	1
True negative rate (TNR, Specificity, SPC)	1	0.99125	0.96188
Accuracy (ACC)	1	0.99176	0.96412
Viral hemorrhagic fever (excl dengue fever and nephropthia epidemica)	Predicted / Actual: TP 100, FP 0, FN 0, TN 1600	Predicted / Actual: TP 100, FP 5, FN 0, TN 1595	Predicted / Actual: TP 100, FP 31, FN 0, TN 1569
True positive rate (TPR, Sensitivity, Recall)	1	1	0.88
True negative rate (TNR, Specificity, SPC)	1	0.99688	0.98063
Accuracy (ACC)	1	0.99706	0.97471

Extended Spectrum Beta-lactamase(ESBL)		Actual			Actual			Actual	
	Predicted	TP 100	FP 0	Predicted	TP 86	FP 0	Predicted	TP 67	FP 0
		FN 0	TN 1600		FN 14	TN 1600		FN 33	TN 1600
True positive rate (TPR, Sensitivity, Recall)		1			0.86			0.67	
True negative rate (TNR, Specificity, SPC)		1			1			1	
Accuracy (ACC)		1			0.99176			0.98059	

Yellow fever		Actual			Actual			Actual	
	Predicted	TP 100	FP 0	Predicted	TP 80	FP 23	Predicted	TP 40	FP 114
		FN 0	TN 1600		FN 20	TN 1577		FN 60	TN 1486
True positive rate (TPR, Sensitivity, Recall)		1			0.8			0.4	
True negative rate (TNR, Specificity, SPC)		1			0.98563			0.92875	
Accuracy (ACC)		1			0.97471			0.89765	

Hepatitis B		Actual			Actual			Actual	
	Predicted	TP 92	FP 0	Predicted	TP 68	FP 0	Predicted	TP 60	FP 0
		FN 8	TN 1600		FN 32	TN 1600		FN 40	TN 1600
True positive rate (TPR, Sensitivity, Recall)		0.92			0.68			0.6	
True negative rate (TNR, Specificity, SPC)		1			1			1	
Accuracy (ACC)		0.9953			0.98118			0.97647	

Hepatitis E		Actual			Actual			Actual	
	Predicted	TP 100	FP 8	Predicted	TP 100	FP 16	Predicted	TP 60	FP 9
		FN 0	TN 1592		FN 0	TN 1584		FN 40	TN 1591
True positive rate (TPR, Sensitivity, Recall)		1			1			0.6	
True negative rate (TNR, Specificity, SPC)		0.995			0.99			0.99438	
Accuracy (ACC)		0.9953			0.99059			0.97118	

HIV infection		Actual			Actual			Actual	
	Predicted	TP 100	FP 0	Predicted	TP 100	FP 0	Predicted	TP 49	FP 0
		FN 0	TN 1600		FN 0	TN 1600		FN 51	TN 1600
True positive rate (TPR, Sensitivity, Recall)		1			1			0.49	
True negative rate (TNR, Specificity, SPC)		1			1			1	
Accuracy (ACC)		1			1			0.97	

Atypical mycobacteria (Infection with)		Actual			Actual			Actual	
	Predicted	TP 100	FP 0	Predicted	TP 95	FP 0	Predicted	TP 76	FP 0
		FN 0	TN 1600		FN 5	TN 1600		FN 24	TN 1600
True positive rate (TPR, Sensitivity, Recall)	1			0.95			0.76		
True negative rate (TNR, Specificity, SPC)	1			1			1		
Accuracy (ACC)	1			0.99706			0.98588		

Amoebiasis		Actual			Actual			Actual	
	Predicted	TP 100	FP 0	Predicted	TP 100	FP 0	Predicted	TP 100	FP 0
		FN 0	TN 1600		FN 0	TN 1600		FN 0	TN 1600
True positive rate (TPR, Sensitivity, Recall)	1			1			1		
True negative rate (TNR, Specificity, SPC)	1			1			1		
Accuracy (ACC)	1			1			1		

Pneumococcal infection (Invasive infection)		Actual			Actual			Actual	
	Predicted	TP 100	FP 0	Predicted	TP 100	FP 39	Predicted	TP 100	FP 244
		FN 0	TN 1600		FN 0	TN 1561		FN 0	TN 1356
True positive rate (TPR, Sensitivity, Recall)	1			1			1		
True negative rate (TNR, Specificity, SPC)	1			0.97563			0.8475		
Accuracy (ACC)	1			0.97706			0.85647		

Legionellosis		Actual			Actual			Actual	
	Predicted	TP 100	FP 0	Predicted	TP 77	FP 0	Predicted	TP 16	FP 0
		FN 0	TN 1600		FN 23	TN 1600		FN 84	TN 1600
True positive rate (TPR, Sensitivity, Recall)	1			0.77			0.16		
True negative rate (TNR, Specificity, SPC)	1			1			1		
Accuracy (ACC)	1			0.98647			0.95059		

Leptospirosis		Actual			Actual			Actual	
	Predicted	TP 100	FP 0	Predicted	TP 100	FP 0	Predicted	TP 59	FP 0
		FN 0	TN 1600		FN 0	TN 1600		FN 41	TN 1600
True positive rate (TPR, Sensitivity, Recall)	1			1			0.59		
True negative rate (TNR, Specificity, SPC)	1			1			1		
Accuracy (ACC)	1			1			0.97588		

Malaria		Actual			Actual			Actual	
	Predicted	TP 100	FP 0	Predicted	TP 100	FP 0	Predicted	TP 75	FP 0
		FN 0	TN 1600		FN 0	TN 1600		FN 25	TN 1600
True positive rate (TPR, Sensitivity, Recall)	1			1			0.75		
True negative rate (TNR, Specificity, SPC)	1			1			1		
Accuracy (ACC)	1			1			0.98529		

Polio		Actual			Actual			Actual	
	Predicted	TP 100	FP 0	Predicted	TP 78	FP 16	Predicted	TP 38	FP 6
		FN 0	TN 1600		FN 22	TN 1584		FN 62	TN 1594
True positive rate (TPR, Sensitivity, Recall)	1			0.78			0.38		
True negative rate (TNR, Specificity, SPC)	1			0.99			0.99625		
Accuracy (ACC)	1			0.97765			0.96		

Q fever		Actual			Actual			Actual	
	Predicted	TP 100	FP 0	Predicted	TP 100	FP 0	Predicted	TP 81	FP 29
		FN 0	TN 1600		FN 0	TN 1600		FN 19	TN 1571
True positive rate (TPR, Sensitivity, Recall)	1			1			0.81		
True negative rate (TNR, Specificity, SPC)	1			1			0.98188		
Accuracy (ACC)	1			1			0.97176		

Syphilis		Actual			Actual			Actual	
	Predicted	TP 100	FP 0	Predicted	TP 88	FP 22	Predicted	TP 62	FP 39
		FN 0	TN 1600		FN 12	TN 1578		FN 38	TN 1561
True positive rate (TPR, Sensitivity, Recall)	1			0.88			0.62		
True negative rate (TNR, Specificity, SPC)	1			0.98625			0.97563		
Accuracy (ACC)	1			0.98			0.95471		

Diphtheria		Actual			Actual			Actual	
	Predicted	TP 100	FP 0	Predicted	TP 93	FP 0	Predicted	TP 39	FP 57
		FN 0	TN 1600		FN 7	TN 1600		FN 61	TN 1543
True positive rate (TPR, Sensitivity, Recall)	1			0.93			0.39		
True negative rate (TNR, Specificity, SPC)	1			1			0.96438		
Accuracy (ACC)	1			0.99588			0.93059		

Average	Altering 1 symptom	Altering 2 symptoms	Altering 3 symptoms
True positive rate (TPR, Sensitivity, Recall)	0.99529412	0.92059	0.65294
True negative rate (TNR, Specificity, SPC)	0.99970588	0.99504	0.97831
Accuracy (ACC)	0.99944637	0.99066	0.95917

4.4 DONS Results & Analysis

In the previous section, we have presented confusion matrices, their respective calculated performance metrics along with the average accuracy results. We have observed that in our implementation approach of the detection algorithm for unknown diseases, the average accuracy achieved was 0.99944, 0.99066, 0.95917 with alteration of 1, 2 or 3 symptoms respectively. To our best knowledge, our implementation for detecting unknown diseases is a unique effort to measure accuracy as presented in this section. We could not find another implementation of a similar approach to compare our results. However, considering the high accuracy levels achieved in our approach, we believe that the proposed system van be used with high confidence.

5. Conclusions

This paper describes the design and development of Cloud-based Disease Outbreak Notification System (DONS). A prototype implementation in a hybrid cloud environment was completed to validate the design of the system. The main function of DONS is to warn for potential outbreaks. The system notifies experts of potential disease outbreaks of both pre-listed diseases and totally unknown diseases. The system only accepts cases from pre-registered sources and is designed to share information about disease outbreaks with international systems. We have presented the entire system along with the performance metrics. Our approach produced high accurate results in detecting unknown diseases.

Acknowledgment

The authors would like to acknowledge the support provided by the Deanship of Scientific Research at King Fahd University of Petroleum & Minerals (KFUPM). This project is funded by King Abdulaziz City for Science and Technology (KACST) under the National Science, Technology, and Innovation Plan (project number 11-INF1657-04).

References

[1] C. Leung, M. Ho, and A. Kiss, “Homelessness and the response to emerging infectious disease outbreaks: lessons from SARS,” J. Urban Heal., vol. 85, no. 3, pp. 402–410, 2008.
[2] M. Muller and S. Richardson, “Early diagnosis of SARS: lessons from the Toronto SARS outbreak,” Eur. J. Clin. Microbiol. Infect. Dis., vol. 25, no. 4, pp. 230–237, 2006.
[3] M. Donohoe, “Internists, epidemics, outbreaks, and bioterrorist attacks,” J. Gen. Intern. Med., vol. 22, no. 9, p. 1380, 2007.
[4] W. Slenczka, “Mad cow disease.,” Emerg. Infect. Dis., vol. 7, no. 3 Suppl, p. 605, 2001.
[5] N. Cohen, J. Morita, D. Plate, and R. Jones, “Control of an outbreak due to an adamantane-resistant strain of influenza A (H3N2) in a chronic care facility,” Infection, 2008.
[6] Guidelines for Outbreak Prevention , Control and Management in Acute Care and Facility Living Sites. Alberta Health Services, 2013, pp. 1–54.
[7] G. Cruz-Pacheco, L. Esteva, and C. Vargas, “Seasonality and outbreaks in West Nile virus infection.,” Bull. Math. Biol., vol. 71, no. 6, pp. 1378–93, Aug. 2009.

[8] D. Tam, S. Lee, and S. Lee, "Impact of SARS on avian influenza preparedness in healthcare workers," Infection, vol. 3, no. 5, pp. 320–325, 2007.
[9] R. M. Roop II, J. M. Gaines, E. S. Anderson, C. C. Caswell, and D. W. Martin, "Survival of the fittest: how Brucella strains adapt to their intracellular niche in the host," Med. Microbiol. Immunol., vol. 198, no. 4, pp. 221–238, 2009.
[10] "WHO | Influenza at the Human-Animal Interface (HAI)." [Online]. Available: http://www.who.int/influenza/human_animal_interface/en/. [Accessed: 11-Feb-2014].
[11] "Prevention and control of animal diseases worldwide. Economic analysis- Prevention versus outbreak costs. Final Report. Part 1, September 2007." [Online]. Available: http://www.oie.int/fileadmin/Home/eng/Support_to_OIE_Members/docs/pdf/OIE_-_Cost-Benefit_Analysis__Part_I_.pdf.
[12] "Guidelines for epidemiological surveillance and preventive measures for infectious diseases." [Online]. Available: http://qh.gov.sa/uploads/files/.
[13] "Health Electronic Surveillance Network." [Online]. Available: http://www.moh.gov.sa/Hesn/Pages/default.aspx.
[14] "IBM, Saudi Health Ministry rolls out cloud-based system.," May-2013. [Online]. Available: http://www.arabnews.com/news/450055.
[15] "HESN FAQ LEAFLET.," Mar-2013. [Online]. Available: http://www.moh.gov.sa/Hesn/Versions/Documents/HESN\ FAQ\ LEAFLET\ -\ ARABIC.pdf.
[16] B. Cakici, K. Hebing, M. Grünewald, P. Saretok, and A. Hulth, "CASE: a framework for computer supported outbreak detection," BMC Med. Inform. Decis. Mak., vol. 10, no. 1, p. 14, 2010.
[17] S. V. Stehman, "Selecting and interpreting measures of thematic classification accuracy," Remote sensing of Environment, vol. 62, no. 1, p. 77-89, 1997.
[18] T. Fawcett , "An introduction to ROC analysis," Pattern recognition letters, vol. 27, no. 8, p. 861-874, 2006.
[19] M. Kubat, R. C. Holte, and S. Matwin, "Machine learning for the detection of oil spills in satellite radar images," Machine learning, vol. 30, no. 2-3, p. 195-215, 1998.

AUTHORS

Farag Azzedin is an associate professor at the Department of Information and Computer Science, King Fahd University of Petroleum and Minerals (KFUPM), Dhahran, Saudi Arabia. He received his BSc degree in Computer Science from the University of Victoria, Canada. He received an MSc degree as well as a PhD degree in Computer Science from the Computer Science Department at the University of Manitoba, Canada.

Salahadin Mohammed is an assistant professor at the Department of Information and Computer Science, King Fahd University of Petroleum and Minerals (KFUPM). He received his BS and MS degrees in Computer Science from KFUPM. He received a PhD degree in Computer Science from the Computer Science Department at the Monash University, Melbourne, Australia.

Jaweed Yazdani is a faculty member at the Department of Information and Computer Science and the manager of Administrative Information Systems (ADIS) at King Fahd University of Petroleum and Minerals (KFUPM), Dhahran, Saudi Arabia. He received his MS degree and BS degrees in Computer Science from KFUPM.

Taher Ahmed Ghaleb earned his BS in computer science at Taiz University, Taiz, Yemen in 2008. After that, he worked as a teaching assistant at the same university in the period between 2009 and 2012. Currently, he is pursuing his MS degree in Information & Computer Science at King Fahd University of Petroleum & Minerals (KFUPM), Saudi Arabia.

Mustafa Ghaleb earned his BS in computer science at King Khalid University (KKU), Abha, KSA in 2007. At present, he is pursuing his MS degree in Information & Computer Sciences at King Fahd University of Petroleum & Minerals (KFUPM).