ISSN 1352-9404

UWS UNIVERSITY OF THE WEST of SCOTLAND

School of Engineering and Computing

Computing and Information Systems Journal

Vol 21, No 2, 2017

Edited by Abel Usoro

www.uws.ac.uk

Notes for Authors:

Articles for publication should be submitted on disk or via e-mail, in Microsoft Word for Windows format. A style template can be found at http://cis.uws.ac.uk/crow-ci0/articles/style.doc. Please observe the following guidelines:

1. The Normal paragraph style is Times New Roman 11 point, justified, with 6 pt space before. Avoid inserting extra empty paragraphs. Avoid the use of additional spaces between words or after a full stop. Use full stops after initials and precede subsequent initials by a space. Use the standard heading styles Heading 1, Heading 2 etc for top-level and subsidiary headings within the article.

2. Compose figures in an enclosing invisible frame or rectangle, with captions and any text on the figure provided in embedded textboxes, and all including the enclosing rectangle grouped into one object, defined to have "Top and Bottom" text wrapping. Try to avoid using anything based on the Normal paragraph style in embedded text boxes.

3. Equations should be embedded in the text, possibly as paragraphs in their own right. They should not be treated as figures.

4. All entries in the reference section should be cited in the text. List references at the end of the paper, alphabetically by the first letter of the first author's last name. References should be identified in the text of the paper by typing the corresponding name and year in parentheses: letters a, b, etc can be added to the year if several references by the same author for the same year are cited. If a page number is set it should be set as (author, year, page number). DO NOT NUMBER REFERENCES; they must be alphabetical and unnumbered. References should have a half-inch hanging indent as shown below and there should be no additional lines between references. Book titles and names of journals should be printed in italics, not underlined. Examples:

Author, A., Colleague, C., and Friend, F. (1995a). Title of paper, *Journal*, **3** (1): 25-50.

Author, A. and Student, S. (1995b), Another paper, in *Title of Book or Conference*, (E. Editor, ed.), XYZ Press, Place of Publication: 451-457.

Author, A. (1995c). *Title of Book*, XYZ Press, Place of Publication.

Subscriptions:

Computing and Information Systems is issued normally three times per year, in February, May, and October. It is available at an annual subscription of £50, or £20 per issue. Contact the School of Engineering and Computing, University of the West of Scotland, Paisley PA1 2BE. Telephone: (+44) 141 848 3959; Fax (+44) 141 848 3542.

A Prototype System for Mining Frequent Citizens' Demand Patterns from E-Government Databases

Adewale Opeoluwa Ogunde
Department of Computer Science,
Redeemer's University
P.M.B. 230, Ede, Osun State, Nigeria
ogundea@run,edu.ng

Chukwuebuka Odukwe
Department of Computer Science,
Redeemer's University
P.M.B. 230, Ede, Osun State, Nigeria
odukwe3276@run,edu.ng

ABSTRACT

Purpose: The objective of this study is to propose a prototype system for collecting, analyzing, understanding and mining citizens pressing needs and demands from e-government databases.

Design/Methodology/Approach: Common citizens demands were obtained from published literature and used to populate a database which was mined with the Apriori algorithm to obtain the most frequent citizens demands.

Findings: Mining a collection of citizens' pressing demands data from e-government databases is very feasible.

Research limitations/implications: The developed system is a prototype designed and tested with a randomized synthetic dataset. The prototype can greatly assist the government to make informed decisions regarding the needs of the citizens.

Practical implications: This will serve as a framework to the government and interested organizations for deploying such an important system that will provide a reliable link between the government and the citizens. The results should provide improved governance and also enhance citizens' satisfaction.

Originality/value: For any government to succeed, citizens' relationship management and satisfaction must be given a top priority. The developed prototype is practicable, achievable and promises great benefits to any interested government, researchers and system developers.

Keywords: *Apriori algorithm, Citizens demands, Data mining, E-government, National security*

Paper Type: Research Paper

1. Introduction

Electronic government (e-government), the ability of government to provide access to services and information twenty-four hours a day, seven days a week, is an emerging force today. Government is turning attention and resources to providing information and services on-line, exploring digital democracy, and using technology for economic development. As a result, government services have been revolutionized in this digital age. In this new age, good government is accessible government, which gives immediate access to pertinent information and it is fast, cheap and efficient.

One key factor that contributes to the success of e-government in any country is the citizens' participation. Citizens' participation is very crucial as it enables those in government at all levels to know exactly the pressing needs and demands of the citizens, which are sometimes politicised by the ruling class. According to the United Nations Economic and Social Council (ECOSOC, 2015), the politicization of participation, lack of trust between government and civil society, and superficial engagement with citizens can also act as a hindrance to citizen-based monitoring of government. Governments are often engaged in programmes and projects that do not really benefit the people. Occasionally some of these national programmes may benefit some but not others as each area of the country has specific needs at any point in time. Over the years, research has shown that when citizens' basic needs are not met, there is usually a tendency for high insecurity in the system. Civil unrest, violence, kidnapping, riots, corruption, armed robbery and the likes usually becomes the order of the day. Over the years in Nigeria, there has been a growing disconnect between the people and government.

Governments, whether military or civilian, have not tried to bridge this chasm, thus creating misunderstanding, mistrust and resentment (Onifade et al., 2013). E-government therefore has paved way for any serious government to positively maximize its impact and results by taking advantage of this modern day technology and most especially by allowing citizens' interaction and input in e-government.

Countries like Nigeria with six major geo-political zones, thirty six states and seven hundred and seventy four local government areas need more than a government-to-citizens (G2C) approach to succeed but rather a combination of G2C and Citizens-to-Government (C2G). Citizens Relationship Management (CiRM) is needed for any government to succeed in terms of planning, providing services and performance management (Latha et al., 2013).

The main purpose of CiRM is to understand the needs of different citizen groups and to provide respective services for each group accordingly. In this way, many governments are actively promoting the use of Information and Communication Technology (ICT). The main issues of this approach are "how e-governments can manage effectively" and "be more citizen-oriented". In other words, ICT can be strategically significant in promoting e-government effectiveness through understanding the citizen requirements. In this domain, the application of data mining tools seems to be useful. Appropriate data mining tools, which are good at extracting and identifying useful information and knowledge from enormous customer databases, are one of the best supporting tools to obtain a deep understanding of citizen's characteristics and needs.

In this paper, a citizen communication system is created and the data is stored in a Relational Database Management System. The data is mined using association rule data mining technique to determine patterns in the citizens' demands. The aim of this paper is therefore to suggest the use of such e-government services and to present a conceptual model for collecting, analyzing, understanding and managing citizens pressing needs and demands.

2. Literature Review

This paper examines citizen interaction with e-government. Much of the existing work on the development of e-government has explored it from a supply-side perspective, such as evidence presented from surveys of what governments offer online. The demand side explanation, which is relatively unexplored, examines citizen interaction with e-government and is the focus of this work. Latha et al. (2013) worked on citizen relationship management with data mining techniques. They used clustering and association rules on the data of the urban service management system to find the subjects that cause complaint and the factors that affect the rate of satisfaction in India. The researchers made it possible to understand the impact of factors such as time and responsible units on the rate of satisfaction. The researchers work focused on a small part of the country and assumed that the complaints all come to a central database. In this paper, we will consider the whole country, Nigeria as our study location and it will be impracticable to assume a central database for this type of problem considering the political and geographical inclination of the nation. People in each state have their own peculiar complaints and there would definitely be need for a system that can mine associations between these local (state) databases apart from the global patterns that would be generated from the central (national database).

According to Rao (2014), the government organizations can use enormous data for discovering hidden patterns, previously unknown relationships, extract meaningful information and trends for decision making. The data mining techniques can help in not only to detect fraud and security threats, but also it can be used for measuring influence facts like citizens' behavior, desire and need. With the increasing awareness among citizens about their rights and the resultant increase in expectations from the government to perform and deliver, the whole paradigm of governance has changed. Government, today, is expected to be transparent in its dealings, accountable for its activities and faster in its responses. This has made the use of ICT tools such as data mining imperative in any agenda drawn towards achieving good governance (Sangeetha and Rao, (2015).

More information on E-government, its benefits and web of interrelationships can be found in Sangeetha and Rao (2015), Rao (2014), Monga (2008), Subhash (2004) and Ndou (2004). Further literature on e-government portals are also found in Capek and Ritschelova (2006). Citizen requirements for E-Government are a very key part of government and at its heart lays the desire to change the way people, businesses, companies, tourists etc. all interact with government (Cook, 2000; Wang et al., 2005; Carter and Belanger, 2005; Yonazi et al., 2008).

Nigeria as a country has set for itself the goal of becoming one of the top 20 economies by the year 2020 and e-government is essential to achieving this goal as it makes the public sector more efficient (Olufemi, 2012). The applications of e-Government in Nigeria include issuance of national passport, driver's license, industry license etc. It is also applied in the registration of voters, payment of tax and delivery of educational services. Although the implementation of e-Government has begun in Nigeria, there is little evidence or research to suggest that a clear framework for the adoption of e-Government is being followed. Also, existing research is yet to address the task of knowledge discovery from the e-government databases.

In this paper, data mining was used to discover the most frequent citizens' demands from e-government databases. Data mining is the dissection or analysis of (frequently extensive) observational data sets to discover unsuspected connections or relationships and to condense the data in novel ways that are both reasonable and helpful to the data manager (Han et al., 2001). According to Agarwal et al. (2010), the use of efficient data mining techniques may surely enhance government decision making capabilities. Association rule mining is one of the most important data mining techniques. The basic algorithm for association rule mining is Apriori Algorithm (Rana and Mann, 2013). Apriori algorithm was used to mine the e-government databases containing citizens' demands.

3. Description of the Prototype System

3.1. Data Gathering and Description

Synthetic data was used in this work. The data consisted of records of attributes such as citizen location, citizen demands and suggested solutions. Distinctive analysis was carried out on the data in order to uncover hidden and vital patterns which would incredibly help the administration in settling on educated choices (informed decisions) on the most proficient method to take care of issues and fulfill its citizens' demands.

3.2 Data Pre-processing

The synthetic data were manually inserted into the system. The data consisted of the citizen's location, citizen's demand and the suggested solutions. The demands were given codes to assist in the analysis of data. This is shown in table 1.

Table 1: Description of data used

S/N	Demands	Codes
1	SECURITY	STY
2	HEALTH	HTH
3	EDUCATION	EDU
4	FINANCE	FIN
5	SCIENCE AND TECHNOLOGY	SCT
6	AGRICULTURE	AGR
7	POWER	POW
8	SPORTS	SPT
9	JUSTICE	JUS
10	WORKS	WRK

3.3. System Design

The system design characterizes the structure of the system. It includes the analysis and determination of the data processing prerequisites of the government and the designing of systems to satisfy them. The system comprises of forms and databases. The demand form is the fundamental structure that empowers a user to enter requests into the database, update and view records in the database. The government can thus query the system and view the frequent items in the database that pertains to any location of the country (e.g. frequent demands from the thirty six states of Nigeria). This was attained by making an online structure utilizing "python" and "php" to associate it to a MySQL database. The frequent items in the database were found by applying the Apriori algorithm on the database.

The system was therefore designed to perform the following functions:

- Accept data from the user through a form that ensures the input of relevant information.
- Store the data collected in a database for record keeping and reliable data mining results.

- Mine the data as requested by the administration officer.
- Allow modification to the mined result (prune) using the Apriori algorithm parameter (support).

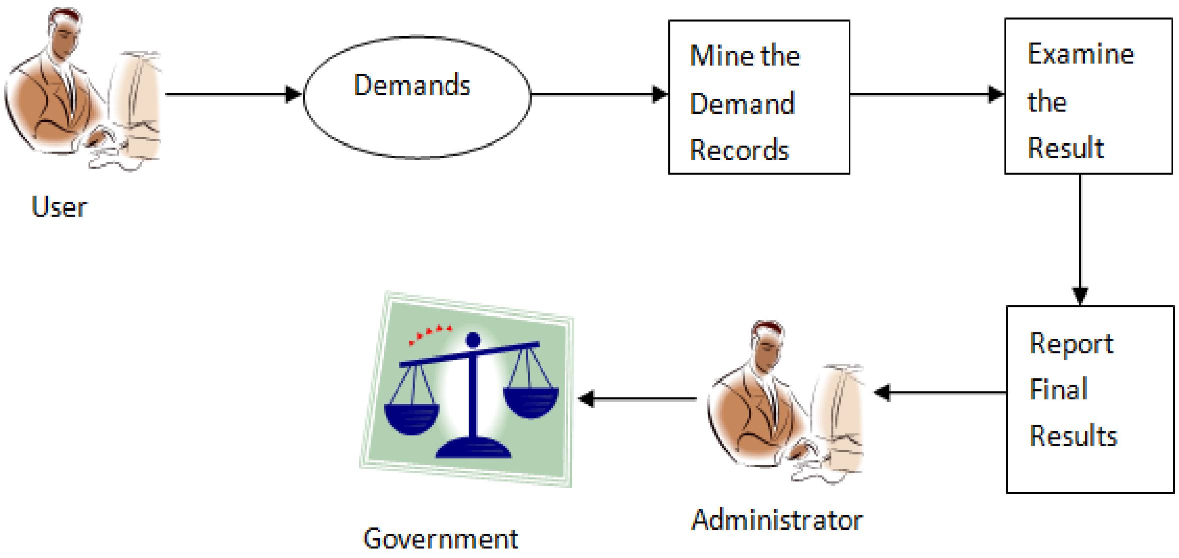


Figure 1: Functional Architecture of the Data Mining Process

The functional architecture for the mining process is given below in Figure 1.

The System Pseudo-code is shown below:

```
Step 1:  Provide citizens access to government through portals
Step 2:  Dissemination of government vital information and programmes
Step 3:  Feedback from citizens on needs, challenges and government appraisal
Step 4:  Feedbacks (complaints/needs) should be classified
Step 5:  Each class to fall under one category as item
Step 6:  Items from one person to form a transaction
Step 7:  All complaints to form the transaction database
Step 8:  Transaction database to be mined with Apriori Algorithm
Step 9:  Frequent associations between items to form the association rules
Step 10: Rules with strong confidence to form the major concerns of citizens
Step 11: Government attends to frequent citizens' demands
Step 12: As citizens' needs are met, violence is reduced, national security is guaranteed
```

3.4. *Database Design*

The database design shows the structure of the database in the system. The database contains the object table. There are two major tables in the system and they are:

- **The Demands Records:** This table consists of all the demands posted on the site. The table consists of the following fields:
    - Citizen ID
    - Citizen Location
    - Demands
    - Suggested Solution

- **The Itemset Records:** This table contains records of all frequent itemsets generated from the demand records table. This table is composed of the following fields:

- ItemsetID
- Itemset
- Support
- Confidence
- Rule

Tables 2 and 3 describe the tables.

tbItemsetRecords: This table contains records of frequent itemset generated from the database. The table structure is shown in Table 2.

tbDemandsRecords: This contains a record of demands entered into the database. The Demands Records table structure is shown in Table 3.

Table 2: The Frequent Item Records table structure

FIELD	**TYPE**	**CONSTRAINT**	**DESCRIPTION**
ITEMSETID	INT	NOT NULL	Contains the value that uniquely identifies each itemset generated and serves as the primary key
ITEMSET	NVARCHAR	NOT NULL	Contains the itemset that specifies the min_support count
SUPPORT	NVARCHAR	NOT NULL	Contains the support of the itemset
CONFIDENCE	NVARCHAR	NOT NULL	Contains the confidence of the itemset
RULE	NVARCHAR	NOT NULL	Contains the rule generated

Table 3: The Demands Records table structure

FIELD	**TYPE**	**CONSTRAINT**	**DESCRIPTION**
CITIZEN ID	INT	NOT NULL	Contains the value that uniquely identifies the citizen
CITIZEN LOCATION	NVARCHAR	NOT NULL	Contains the location of the citizen
CITIZEN DEMANDS	NVARCHAR	NOT NULL	Contains the demands of the citizen
SUGGESTED SOLUTION	NVARCHAR	NOT NULL	Contains the citizen's suggestion on how to solve the problem

4. Implementation and Results

The designed model was developed into a prototype system that could be deployed by any interested government. It has an interface that allows the user to input their demands. It also allows the government to view the most frequent citizens' demands. It has a data mining capacity to extract the frequent patterns of citizens' demands to help the government focus on the most pressing needs of their citizens.

4.1. *System Login*

The login page as shown in Figure 2, grants access to the system.

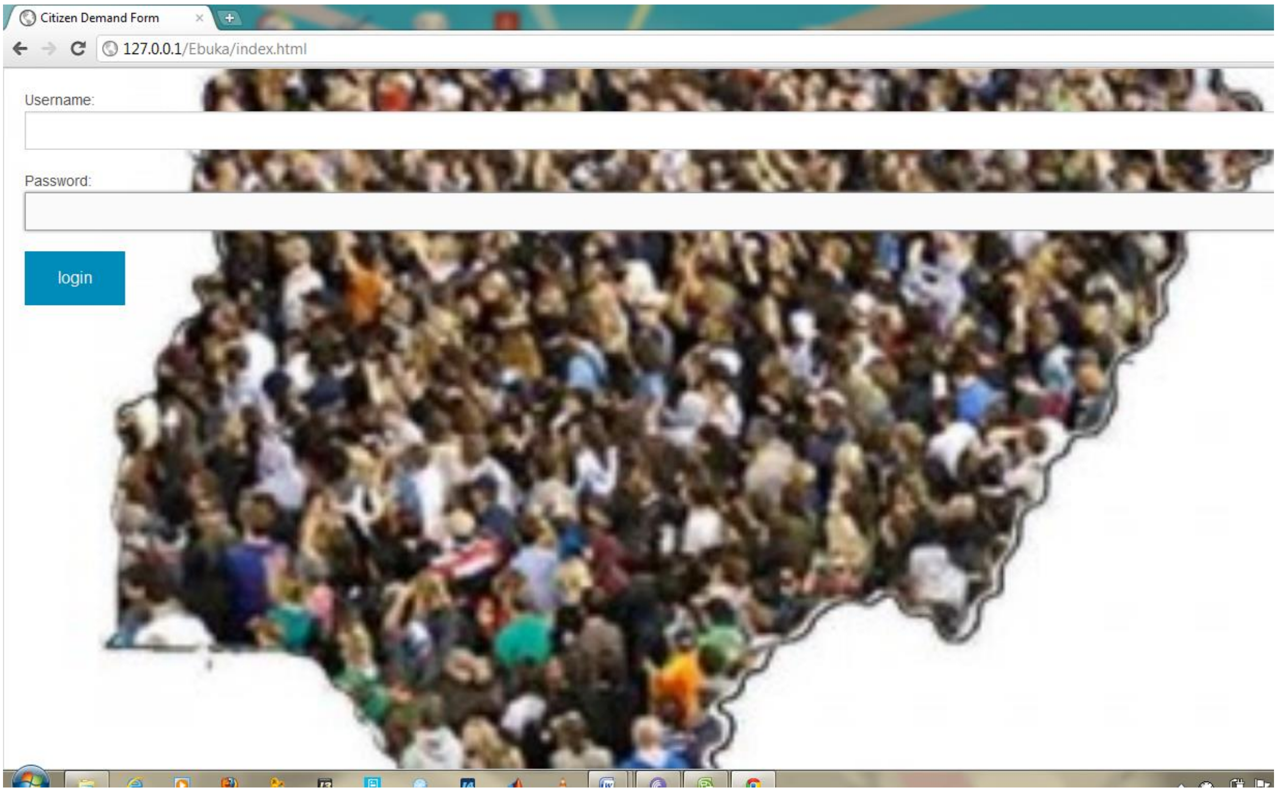


Figure 2: Citizen Demand Form

The user login grants access to the demand form and the admin login grants access to the results of mining the demands the citizens have submitted. It consists of the following:

Username- this represents a unique username that differentiates a normal or regular user from the administrator.

Password- this is a unique phrase of not more than 50 characters to allow access to the demand form (for the user) and the results (for the administrator).

4.2. *Citizens Demand Form*

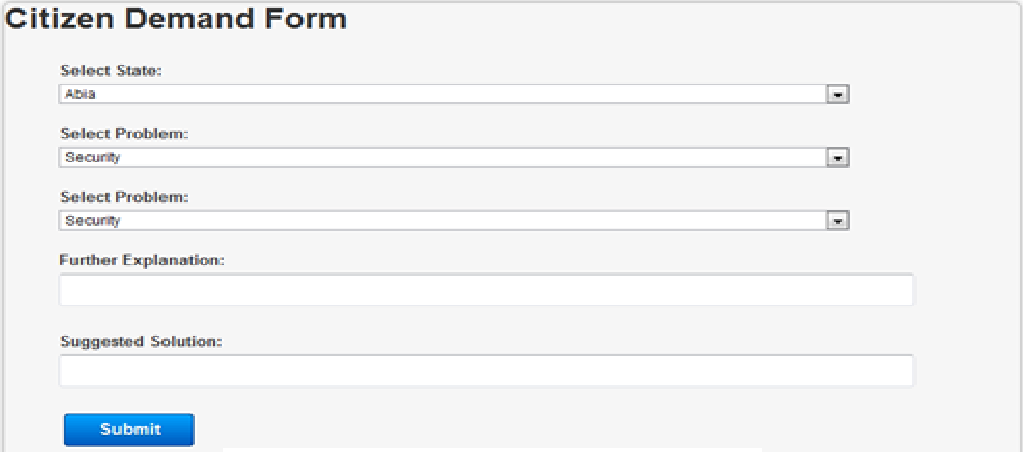
When a user logs in, he is immediately sent to the web page containing the demand form (figure 3). The demand form contains fields such as State, Problems, Further Explanation and Suggested Solutions.

Select State- this is a drop down menu showing all the thirty six (36) states of Nigeria.

Select Problem- this is a drop down menu showing a list of possible problems. There are ten (10) problems listed here. The user is allowed to select at most two problems.

Further Explanation- this allows the user to explain the problem or problems he has selected.

Suggested Solution- this allows the user to provide his opinion on how the problems should be solved.



Citizen Demand Form

Select State:
Abia

Select Problem:
Security

Select Problem:
Security

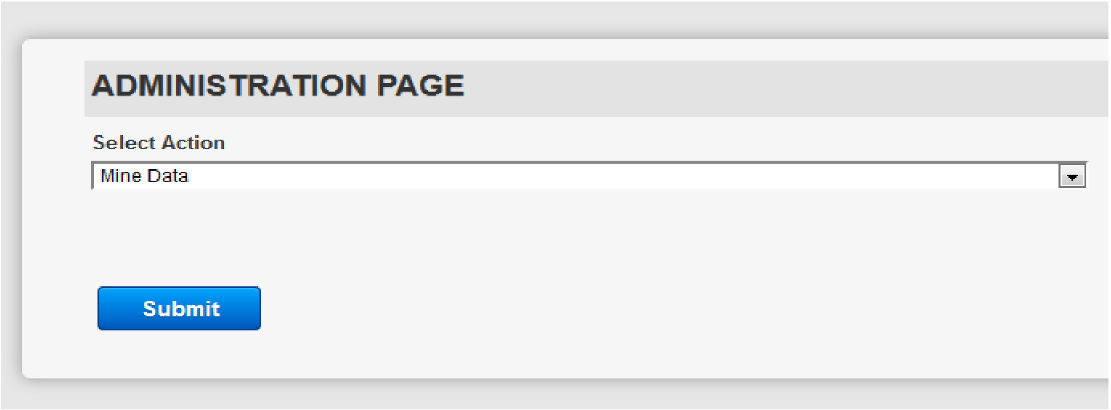
Further Explanation:

Suggested Solution:

Figure 3: Citizens Demand Form

4.3 *Administration Page*

This shows the result of data mining from the demand form. It showed the state and the most frequent demands of citizens in that state. It can only be accessed by the System Administrator. Figure 4 shows the administration page. The administrator can perform many actions such as mine the data, view results and update the database from the administrator page.



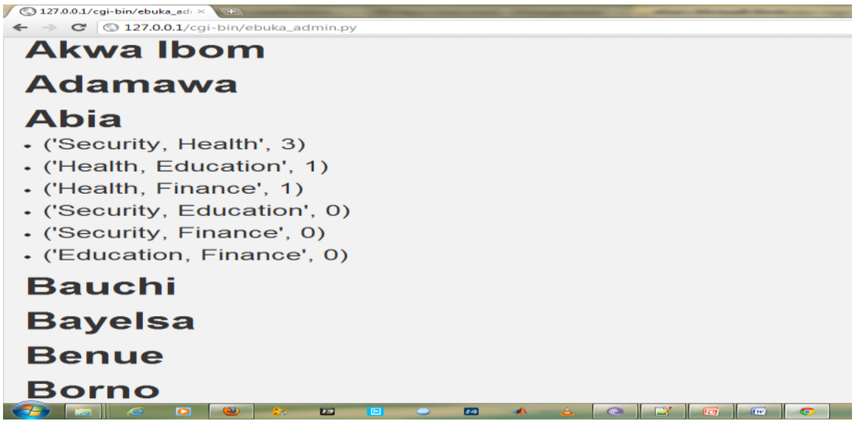
ADMINISTRATION PAGE

Select Action
Mine Data

Figure 4: Administration Page

4.3. *Testing and Results*

For the purpose of testing, the developed system was tested by populating the created database with random citizens' demand for all thirty-six states in Nigeria after which the mining action was invoked. The result of selecting the mined data, that is, the frequent citizens demands are shown in Figure 5. It shows some states and the frequent citizens' demand in that particular state.



```
Akwa Ibom
Adamawa
Abia
• ('Security, Health', 3)
• ('Health, Education', 1)
• ('Health, Finance', 1)
• ('Security, Education', 0)
• ('Security, Finance', 0)
• ('Education, Finance', 0)
Bauchi
Bayelsa
Benue
Borno
```

Figure 5: Mined Result of Frequent Citizens Demands

4.4. *Discussion*

The results in Figures 4 and 5 showed the most frequent demands in the various states using the test data. It also showed the most frequent pair of demands to further help the government to understand the needs of its people. If certain pairs of demands have high occurrences, the government will be able to prioritize those sectors and ensure that they are treated first before they can begin resolving problems of other sectors.

5. Conclusion and Future Works

A study on the development of a framework for mining frequent citizens' demands from e-government database was conducted. There is an urgent need for this type of system most especially in this age of severe unrest and agitations among citizens of many nations of the world, which could be attributed to neglect and possible marginalization by the government of such countries. Synthetic data of the citizens' location, demands and suggestions were inputted into the system. Analysis was done on the data collected to determine the most frequent demands. The Apriori algorithm was used to generate frequent item sets in the data. Mining the citizens' demands database revealed the most frequent demands of citizens based on their locations. This paper had carefully brought into focus the importance of knowing the demands of citizens so as to achieve an effective and efficient government. The proposed prototype can greatly assist the government to make informed decisions regarding the needs of the citizens. The possibility of developing the real mining system for frequent citizen's demands is something practicable and achievable. This work serves as a framework to the government and interested organizations for deploying such an important system that will provide a reliable link between the government and the citizens. This system has a potential benefit of providing improved governance and also enhancing citizens' satisfaction. Finally, a system for mining citizens' pressing demands is a feasible idea. The concept carries huge potential and can resolve some problems concerning citizens' management and governance. Future works will consider increasing the number of possible demands (this paper was limited to ten demands). It will also allow users to select their local government areas and factor in demands at the lowest level of government. A mobile version of the system will also be developed.

References

Agarwal S., Singh N., and Pandey G.N. (2010). Implementation of Data Mining and Data Warehousing In *E-Governance, International Journal of Computer Applications*, Vol. 9 No.4, pp 18-22.

Capek, J., and Ritschelova, I. (2006). Regional E-Government – some problems with data sharing. *European Regional Science Association.* Retrieved on 13th April 2016 from http://ideas.repec.org/p/wiw/ wiwrsa/ ersa06p361. html**.**

Carter, L. and Belanger, F. (2005). The Utilization of E-Government Services: Citizen Trust, Innovation and Acceptance Factors. *Information Systems Journal*, Vol 15, pp 5-25.

Cook, M.E. (2000). What Citizens Want From EGovernment: *Current Practice Research*. Centre for Technology in Government, University at Albany/SUNY, Albany, N.Y.

ECOSOC (2015), Citizen-based Monitoring of Development Cooperation to Support Implementation of the 2030 Agenda, *Development Cooperation Forum Policy Briefs*, No. 9, pp 1-11, www.un.org/ecosoc/dcf

Han, D., Mannila, H., Smyth, P. (2001). *Principles of Data Mining*. Cambridge, MA: The MIT Press; 2001.

Latha, P., Madhuri R., Prasad Rao, K., Vijaya Bharathi M. (2013). Effectual Citizen Relationship Management with Data Mining Techniques. *GMR Institute of Technology,* No 2, pp 158-161.

Monga, A. (2008). E-government in India: Opportunities and challenges, *JOAAG*, No. 2, Vol. 3, pp 52-61.

Ndou, V. (2004). E-government For Developing Countries: Opportunities and Challenges. *Department of Business Administration, University of Shkoder,* Vol 18, pp 1-24.

Olufemi, F. (2012). Electronic Governance: Myth or Opportunity for Nigerian Public Administration? *International Journal of Academic Research in Business and Social Sciences*, Vol. 2, No. 9, ISSN: 2222-6990, pp 122-140.

Rana, M. and Mann, P.S. (2013). A Review of Association Rule Mining with Genetic

Algorithm. *International Journal for Science and Emerging Technologies with Latest Trends*, Vol. 8, No. 1, 2013, pp 14-23.

Rao V. R. (2014). A Framework for e-Government Data Mining Applications (eGDMA) for Effective Citizen Services - An Indian Perspective, *International Journal of Computer Science and Information Technology Research*, Vol. 2, Issue 4, pp 209-225.

Sangeetha G. and Rao L. M. (2015), A Review on Contribution of Data mining in e-Governance Framework, *International Journal of Engineering Research and General Science* Vol. 3, Issue 2, pp 68-75.

Subhash, B. (2004). *E-government From Vision to Implementation.* New Delhi: Sage Publications.

Wang, L., Bretschneider, S. and Gant, J. (2005). Evaluating web-based e-government services with citizen-centric approach, Proceedings of the 38th Hawaii International Conference on System Science, pp 3-6.

Yonazi, J., Sol, H. and Boonstra, A. (2008). Developing a Framework for Assessing Adoptability of Citizen-Focused eGovernment Initiatives in Developing Countries: The Case of Tanzania; Exploratory Phase Results, Proceedings of the 8th European Conference on eGovernment 10-11 July 2008, Ecole Polytechnique, Lausanne, Switzerland.

About the Authors

Adewale O. Ogunde, Ph.D. is a Senior Lecturer in the Department of Computer Science, Redeemer's University, Nigeria. He holds a Ph.D. degree in Computer Science. His research interests include: knowledge discovery and data mining, machine learning, artificial intelligence, decision support systems, E-government, E-learning, and Information systems amongst others. He has published many articles both in international and local peer reviewed journals. He has also presented research papers at various international and local conferences, also contributed chapter to a book. He is a full member of many IT related professional bodies e.g. Nigeria Computer Society (NCS), Computer Professional Registration Council of Nigeria (CPN) and IAENG.

Chukwuebuka Odukwe, is a graduate of the department of computer science, Redeemer's University. His research interests include: E-government, data mining, distributed databases and information systems.

Analysis of Human Factors in Cyber Security: A Case Study of Anonymous Attack on Hbgary

Benjamin Aruwa Gyunka
Directorate of Information and Communication Technology
National Open University of Nigeria (NOUN)
Abuja, Nigeria
gyunkson@gmail.com

Abikoye Oluwakemi Christiana
Department of Computer Science
University of Ilorin
Ilorin, Nigeria
kemi_adeoye@yahoo.com

ABSTRACT

Purpose: This paper critically analyses the human factors or behaviours as major threats to cyber security. Focus is placed on the usual roles played by both the attackers and defenders (the targets of the attacker) in cyber threats' pervasiveness and the potential impacts of such actions on critical security infrastructures.

Design/Methodology/Approach: To enable an effective and practical analysis, the Anonymous attack against HBGary Federal (A security firm in the United State of America) was taken as a case study to reveal the huge damaging impacts of human errors and attitudes against the security of organizations and individuals.

Findings: The findings revealed that the powerful security firm was compromised and overtaken through simple SQL injection techniques and a very crafty social engineering attack which succeeded because of sheer personnel negligence and unwitting utterances. The damage caused by the attack was enormous and it includes the exposure of very sensitive and personal data, complete shutdown of the website, loss of backup data and personnel character deformations. The research also found that damaging human factors results from ignorance or illiteracy to basic security practices, carelessness and sometimes sabotage by disgruntled employees from within and these vulnerabilities have become prime target for exploitation by attackers through social engineering attacks. Social engineering was also discovered to be the leading attack technique adopted by attackers within the cyber space in recent years.

Practical Implications: The paper concludes by advocating assiduous training and cyber security awareness programmes for workforces and the implementations and maintenance of basic security culture and policies as a panacea for social engineering cyber attacks against individuals and organizations.

Originality: Lots of work has been done and many still on-going in the field of social engineering attacks and human factors, but this study is the first to adopt an approach of a practical case study to critically analyze the effects of human factors on cyber security.

Keywords: *The Anonymous; HBGary Federal; Uniform Resource Location (URL); Content Management System (CMS); SQL Injection; Cross-site Scripting (XXS); Social Engineering; Cyber Security; Information Security*

Paper Type: *Research Paper*

1 Introduction

Humans have been found to be truly the weakest link of security (Mitnick, Simon, & L., 2011) and (GBC-DELL Survey, 2015). The psychology of human workforce is being viewed as a critical factor that poses serious cyber-attacks risks to all users (Ranjeev & Lawless, 2015). Human cyber security behaviours has created serious vulnerabilities which attackers exploits using social engineering attack techniques and findings revealed that human factors are responsible for 95% of all security incidences (IBM, 2015). Human threats to critical infrastructures and services come mostly from careless work behaviours and ignorance of basic cyber security practices which include irregular software patching to get rid of bugs, installations of malicious software, careless communication of sensitive information and connection to insecure internet networks or Wi-Fi (Aziz, 2013) and (James, 2015). They also include poor attitudes to web applications usage and database management which opens door to cross-site scripting (XXS) and SQL Injection vulnerabilities (Stuttard & Marcus, 2011). Attackers these days find it interestingly easier to begin their attacks by the exploitation of human ignorance, weakness and selfish interests to gain an open entrance for a mega attack. People are now inadvertently deceived to either initiate or even carry out the attacks by themselves without the attacker necessarily introducing an external event or involving very expensive technical exploit kits. Human factor is an insider threat against security either through disgruntled employees seeking to cause pains or through social engineering which appeals to personnel's instincts and attackers would rather take advantage of these vulnerabilities, where available, than engaging other exploits against technical security devices (James, 2015), (Warwick, 2016) and (CeBIT Australia, 2017).

Research has shown that it is not good enough to have all the state-of-the-art security software and hardware properly installed and running in an organization if the human factor to cyber security is neglected (Nate L. , 2016), and (James, 2015). Firewalls, Intrusion Detection Systems, Antimalware and many authentication mechanisms such as time-based tokens or biometric smart devices, are usually installed to protect against external threats but cannot protect against threats from within, caused by ignorant and careless personnel (Mitnick, Simon, & L., 2011) or by disgruntled employees aiding external attacker (Blythe, 2013). Cyber attackers would rather now want to exploit the vulnerable human factors through simple tricks than to spend much time and resources trying to gain access by breaking through the different strong technical security systems. This paper seeks to practically analyze the impacts of human factors to critical security infrastructures. The attack of the Anonymous Hacktivist group against HBGary Federal, a US based security firm, was taken as a case study to analyze the different phases of cyber attacks against human cyber security behaviours. The different phases include the analysis of defender(s) vulnerabilities (target of attack – the human factors), the analysis of the attackers' tricks and techniques, and finally, the analysis of the resulting damages. The paper concludes with suggestive techniques for preventing against such exploitations.

2 Social Engineering

Social engineering is a non-technical method of cyber-attacks which absolutely depends on human psychology and mostly involves deceiving people into breaching standard security practices (Nate, 2016). Researches have shown that social engineering attacks are the top most threats against information security (Warwick, 2016) and (Nate, 2016). The whole technique of social engineering attacks is completely anchored on the principle and art of deception, making people do things that they would ordinarily not want to do for a complete stranger (Mitnick et al, 2011). Thus, victims of this attack techniques are usually persuaded to willingly open wide their security door ways to unknown persons (Ranjeev & Lawless, 2015) or are tricked to do things like giving out sensitive information or documents, disabling critical security systems, transferring money to unknown persons' accounts and many other devastating things (Warwick, 2016). Sometimes they are tricked to believe that the order they are obeying is coming from a superior, colleague, or partner sitting somewhere (Mitnick, Simon, & L., 2011). Often times, what they are persuaded to do are highly regrettable, causing irreversible damages.

Common approaches or attack vectors adopted in social engineering attacks include engaging people through fake emails, social media, voice calls, mobile apps, or through direct physical contact with the defendant (target of the attacker). Social engineering attacks, or attacks against human psychology and instincts, may come in the forms of phishing, malware attacks, pretexting, baiting, quid pro quo and tailgating (David, 2015). Phishing scams and malware infections have be found to be the most adopted forms of social engineering attacks (GBC-DELL Survey, 2015) as indicated in Figure 1. Anyone that falls victim of social engineering attack would normally become the enabler of the bigger attack or might even unknowingly be used to directly complete the full-scale attack.

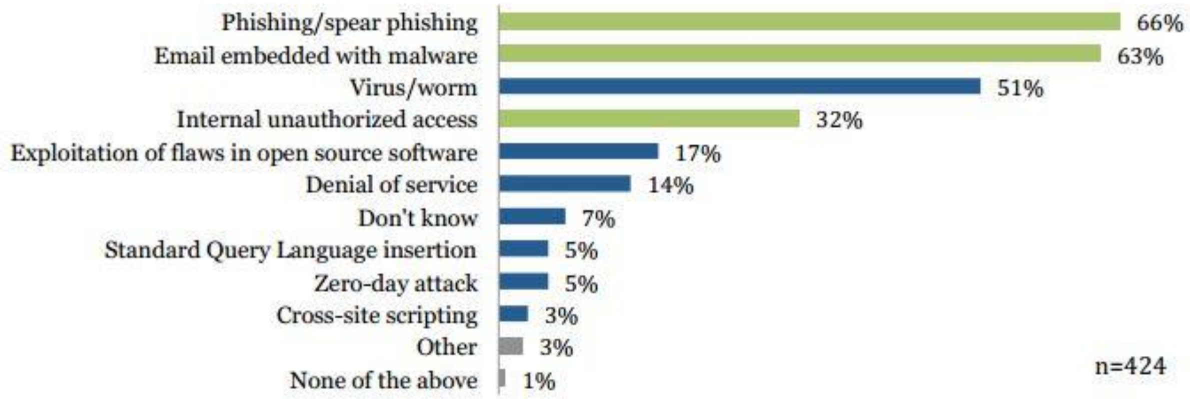


Figure 1: Significant Cyber Threats (GBC-DELL Survey, 2015)

This study takes a deep delve into some practical applications of social engineering attacks and its requisite consequences and prevention. The attack of the Anonymous Hacking group against HBGary Federal security firm was adopted as a case study for a critical analysis of this attack technique. The study begins by critically looking into the different services offered by HBGary and where they failed. A brief about the Anonymous group was also discussed; the different attack techniques deployed, the resulting damage, ways of preventing similar attacks on businesses, and the lessons learned form the core of this study.

3 The Defender – Hbgary Federal

HBGary was a well-known technology security company with offices in Washington D. C., California, Sacramento, and Bethesda, Maryland. The Security Firm was founded by Greg Hoglund in the year 2003. The company entered into a Security Innovation Alliance with McAfee in the year 2008. The Establishment was an affiliation between HBGary Federal and HBGary Inc, both being very distinct entities. HBGary Federal had one mega web server which could be accessed through a Web link, *www.hbgaryfederal.com*, and they also had one major Support Linux Machine which could be accessed through the link, *support.hbgary.com*. The Linux Machine contained most of the employees shell accounts, which they could access using SSH. Greg Hoglund also operated another website called Rootkit.com which was hosted by another Linux machine. All the email services of HBGary Federal were being managed by Google Apps. The National Security Agency (NSA) and Interpol had maintained a frequent contact with HBGary companies and HBGary also had been working with McAfee which is a well known security firm too (Peter, 2011).

HBGary Federal, being an information security firm, specializes in design and distributions, through sales, of the state-of-the-art tools for computer forensics and malware analysis to the United State government and other private Institutions (Peter, 2011) and (Krebs, 2011). Their services also included technical consultancy and supports. The support covers areas such as the implementation and deployment of intrusion detection systems, designing secure networks, performing vulnerability assessment and penetration testing of systems and software. The United State Government and some Strong Private Organizations were some of the strong patronisers and customers of the services of HBGary Federal.

4 The Attacker – Anonymous

The Anonymous is a group of hacktivists which comprises of people from different backgrounds, diverse professional experiences and different age groups. This involves professional office employees, software developers, IT technicians, and even students. The membership of the group are found scattered in different countries of the world, a few amongst them includes the United State, The United Kingdom, Germany, Netherlands, Italy, and Australia. The hacktivist group mostly adopt cyber attack as their main campaign medium to show their displeasures and grievances against any government policies or any Organization that might have crossed their ways. The group was allegedly founded in the year 2003.

Figure 2: The Faceless Group – Anonymous (Peter, 2011).

A few amongst many other exploits perpetrated by the Hacktivist Group includes the bringing down of PayPalblog.com, MasterCard.com and Visa.com (Nate & Technica, 2011). The attacks against these Companies were done to punish the financial companies for their involvement in shutting down WikiLeaks from the internet. Anonymous attacked these websites using Distributed Denial of Service (DDoS) attacks through a modified version of the Low Orbit Ion Cannon (LOIC) load-testing tool.

5 Human Factors Vulnerabilities Analysis

HBGary was operating a content driven website whose data was stored in an SQL database. As it is with every thriving business, there was always a constant need for updating the contents of the website by correcting, adding or removing some information from the database. To make the administration of the website easier, HBGary Federal deployed a Content Management System (CMS) in the organization. Although this approach was a good idea, but best practice would have been for them to implement an off-the-shelf Content Management System which would have enabled them the ability to directly monitor and control the system, but they rather chose a custom CMS from a third-party developer. Third party's applications do not always have good reputations as they mostly have issues with malware and wrong coding (Rahul, Venkiteswaran, Anoop, & Soumya, 2014), so the CMS deployed by HBGary had serious coding flaws which made it highly vulnerable to cyber attacks. Although the CMS had bugs, HBGary was negligent and careless about the CMS. They could have exercised their own expertise as security experts in finding and fixing (debugging) the bugs and also setting up and configuring bug tracking devices to track security vulnerabilities of the software, but they failed to do any of this. HBGary was completely blind to this dangerous flaw, allowing the CMS to become highly vulnerable to SQL injection attacks.

The Security Firm, HBGary Federal, was also guilty of poor password management. The senior executives of the Firm, CEO Aaron Barr and COO Ted Vera, became too busy about their work that they forgot and neglected simple and standard information security practices especially in the areas of password policies and management. They became an extreme bad example to be emulated in this regard. Both top Officers had extremely weak passwords with each comprising of only six lower case letters and two numbers. As though that was not bad enough, they also maintained the same passwords across platforms and applications. That is, the same password was used to login into their twitter accounts, email accounts, Linkedln, and SSH. This practice subjected them to a security single point of failure (failure at one point implies failure at all points). The most disturbing part of it was that Aaron had the administrative right over the Google App that hosted the entire company's emails and while Ted had a user privilege in the Linux SSH account. Password misuse and negligence alone had exposed the Company to serious security threats.

In managing the SSH access to the Firms Server, the authority also carelessly ignored the principles and policies governing safe SSH connections. It did not come into their minds to remember that password authentication was not the best security verification practice for any SSH connection, so they continued to use only passwords to gain access via SSH to the Support Linux Machine. They could have included the hard-to-crack cryptographic encryption methods in the system which would provide each user with a secret key which must be kept private and with a public key that is associated with the user account. If these were put in place, the SSH would have then made use of both keys to authenticate the different users. The Firm adopted MD5 for their password encryption in a very weak way. Another serious security loophole entertained by HBGary Federal was inadequate software patching. Little or no attention was given to regularly patching the Linux Support Machine. This also exposed the Machine's Operating System and its system libraries to privilege escalation exploitation attack vulnerabilities.

Finally, there was serious lack of proper information dissemination within and outside the Company. They were very careless at releasing very sensitive information without minding who is listening. This attitude exposed the Corporation to the subtle danger of social engineering attacks. The Anonymous shows up mostly through cyber attacks, so they have been associated with majority of cybercrime in the world. Because of their activities, this Group became a prime suspect to the United State Government and this has set them on the list of the FBI for continuous investigation to uncover the identities of its members (Nate & Technica, 2011). The CEO, Aaron Barr, was too outright and straight, without caution, when he publicly announced the Firm's collaboration with the FBI (Federal Bureau of Investigation) against the Anonymous group. He revealed that the Firm had gotten some essential information about the identities and activities of some cardinal members of the Anonymous group, expressing his readiness to sell this information out to the FBI for further actions against the group. The method he claimed to have used in getting these essential details was emails monitoring, and using of fake names for Facebook and IRC chat. His action presented him as having a boast on the strength of the Firm and their victory over the Anonymous group (Nate & Technica, 2011). This pronouncement was regrettably a dangerous move that invited the wrath of the hacktivist group, Anonymous, against HBGary Federal. Without hesitation, the Anonymous reacted immediately against Aaron's moves by attacking HBGary Federal between the 5th and 6th of February 2011. The attack lasted for a period of 24 hours only.

6. Analysis of Attackers' Techniques And Tricks

Anonymous started by exploiting the vulnerability found in the Content Management System (CMS). They injected some SQL queries into the Firm's web server database. The coding of CMS are meant to enable it identify what details it should allow to be retrieved from a database system based on the receipt of a particular query or URL (Uniform Resource Location). The CMS is required to match the received query against the records in the database, render the collected content which may include an HTML, and then countless web pages can be created within seconds to display the required results. A typical CMS would usually have a web 'front-end' which allows the editing of database records through the web by the respective users. The SQL query injected by the Anonymous made use of the URL, **http://www.hbgaryfederal.com/pages.php?pageNav=2&page=27**. Two parameters included in the query to manipulate the CMS are pageNav=2 and page=27. Given that the CMS had bugs already in its code, it became easily tricked to misinterpret the query with these parameters, thus providing the hackers with open access to the database of the web server that hosted the Firm's very sensitive data. They completely took over the database from the CMS. Some details retrieved from the database include usernames, email addresses, and password hashes of privileged users who had the administrative right to make any required changes to the CMS. The vita data found on this server provided the attackers with more information that aided their invasion further.

One good property of the CMS was its ability to store only the hashed password of the users in the database which could be very difficult to break into plain text. Fortunately for the attackers, the hash was only a single one-way hashing that was done using MD5 hashing function without applying salting and iterative hashing methods. Taking advantage of the weak hashing procedure, the attackers deployed rainbow table cracking technique to crack the downloaded hashed passwords. Iterative hashing involve the process of having the output of a hash function re-hashed again repeatedly for several times (Sjoerd, 2016) and (Dunkelman & Eli, 2006), while salting technique involve adding a small amount of random data to the password before it is hashed (Sjoerd, 2016) and (Patel, Patel, & Virparia, 2013). If these hashing techniques were adopted, it would have become either very difficult or nearly impossible for the passwords to be cracked by the attackers. It suffices to say hbgaryfederal.com would have survived the rainbow password cracking attacks despite the loophole found with the MD5 hashing functions if they probably had adopted the best password protection policy (Daniel, 2015) and (SANS, 2014).

Rainbow table attacks commonly succeed against two kinds of password patterns; this include password of eight character length which compromises a mixture of lower case letters and numbers only, and a those of one to twelve character length which are made up of upper case letters only and anything outside these lengths, it becomes extremely difficult for the rainbow tables to generate (Avi, 2016) and (Coding Horror, 2007). Although CEO Aaron Barr and COO Ted Vera were expected to know better, given that they owned administrative rights to different systems, they both were still very careless to use password combinations of only six lower case letters and two numbers. Another huge mistake made by these executives was the reuse of same password on different platforms and applications including even the Support Linux Machine, *support.hbgary.com.* The attackers took advantage of this weakness and were able to easily attack the Linux Machine using Ted Vera's password. Unfortunately, the Linux Machine had some software vulnerabilities due to inadequate patching, so the attackers deployed privilege escalation exploits to gain root privilege and had total control over the machine from where they extracted gigabytes of backups and research data.

The password for Aaron Barr was used by the attackers to gain administrative access into the Google App that controls the entire Company's emails. Greg Hoglund, the founder and owner of rootkit.com, had his e-mail account also listed there, so the attackers accessed his email and were able to retrieve two additional passwords from there which were '88j4bb3rw0cky88' and '88Scr3am3r88' which could give them the root access to the server hosting rootkit.com, but they also found out that Jussi Jaakonaho (Chief Security Specialist) of Nokia had a root access to the machine too. Despite the details retrieved, it was still impossible for them to break into Greg's machine by direct SSH using root account (username & password), they would need to first login with a non-root privilege user account. The root account details could not be used to access the server from outside of the firewall and so they sought for ways to retrieve Greg's common user account details (username and password) (Keir, 2011). They resorted to social engineering attack using email (Peter, 2011) against Jussi Jaakonaho from whom they were able to get all the details they needed to complete their task. To implement the social engineering attack, the attackers disguised as Greg Hoglund by using his email account to send mails to Jussi Jaakonaho. The email conversations between the attackers and Jussi are as follows (Peter, 2011):

From: Greg
To: Jussi
Subject: need to ssh into rootkit
im in europe and need to ssh into the server. can you drop open up
firewall and allow ssh through port 59022 or something vague?
and is our root password still 88j4bb3rw0cky88 or did we change to
88Scr3am3r88 ?
thanks

From: Jussi
To: Greg
Subject: Re: need to ssh into rootkit
hi, do you have public ip? or should i just drop fw?
and it is w0cky - tho no remote root access allowed

From: Greg
To: Jussi
Subject: Re: need to ssh into rootkit
no i dont have the public ip with me at the moment because im ready
for a small meeting and im in a rush.
if anything just reset my password to changeme123 and give me public
ip and ill ssh in and reset my pw.

From: Jussi
To: Greg
Subject: Re: need to ssh into rootkit
ok,
it should now accept from anywhere to 47152 as ssh. i am doing
testing so that it works for sure.
your password is changeme123

i am online so just shoot me if you need something.

in europe, but not in finland? :-)

_jussi

From: Greg
To: Jussi
Subject: Re: need to ssh into rootkit
if i can squeeze out time maybe we can catch up.. ill be in germany
for a little bit.

anyway I can't ssh into rootkit. you sure the ips still
65.74.181.141?

thanks

From: Jussi
To: Greg
Subject: Re: need to ssh into rootkit
does it work now?

From: Greg
To: Jussi
Subject: Re: need to ssh into rootkit
yes jussi thanks

did you reset the user greg or?

From: Jussi
To: Greg
Subject: Re: need to ssh into rootkit
nope. your account is named as hoglund

From: Greg
To: Jussi
Subject: Re: need to ssh into rootkit
yup im logged in thanks ill email you in a few, im backed up

thanks

With the information gathered and access gained, the attacker succeeded in bringing down Rookit.com too so easily because the Server hosting it also had similar vulnerability as that of HBGary Federal; it did not use key authentications.

6.1 The Impact of Human Factor on Critical Infrastructure

The HBGary website was completely compromised, over sixty thousand (60,000) Company emails were downloaded and exposed on The Pirate Bay site (Chester, 2011). The Company's backup files were completely deleted by the Anonymous. The Group also retrieved and publicly displayed the documents HBGary Federal boasted about earlier to sell to FBI for everyone to see. They also retrieved and exposed users' database from Rootkit.com and all the email addresses and passwords hashes for everyone who had ever registered on the website. Aaron Barr's private and confidential credentials which include his private mails, home address, social security number and cell phone number were all exposed to the public. The greatest damage was on the Integrity, Reliability, Confidentiality and finally the Availability of the Company. The mistakes were completely irreversible resulting to a total shutdown of the security Firm, HBGary Federal, putting them out of business.

7. How To Prevent Similar Attacks On Businesses

Staff trainings on standard security principles and policies must be taken very seriously in every Organization in order to combat social engineering attacks (GBC-DELL Survey, 2015). This will be an essential tireless and continuous cybersecurity literacy and awareness training for the workforce. It is worth spending resources on keeping the security and risks management knowledge of workers updated all the time as this can reduce an organization's cyber security breaches by 70% (Pittsburgh, 2015). Proper policy must be put in place with the right password hashing techniques especially the use of iterative hashing and salting. A regular vulnerability testing of website must be carried out to look for security holes in order to cover them up. Public and private key encryptions and authentication techniques should be deployed for protecting the server when it comes to authentications. Systems and software patching should be done on regular basis. Vulnerability assessment must be done on all the information infrastructures deployed in the network. The practice of password reuse on different platforms should never be encouraged. Social engineering is a very subtle attack, thus personnel should always verify any requested task before agreeing to release very important details.

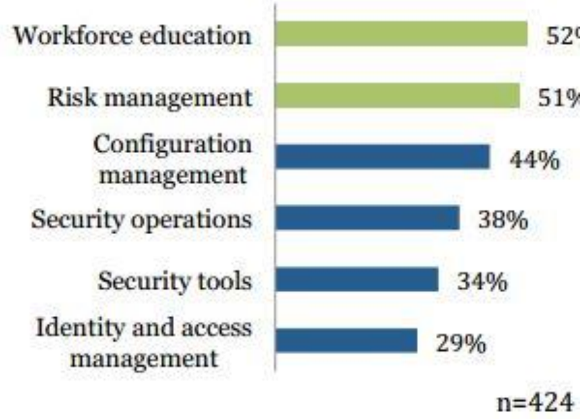


Figure 3: Cyber Defense Elements in Need of Significant Improvement (GBC-DELL Survey, 2015)

8. Conclusion

The case study analysed in this paper suggest that attackers will not usually attack from areas that are considered to be of great security strength, but would rather focus their attention on the very weak and neglected points of security, especially the human factor. Human factor was the greatest weakness that brought down HBGary Federal. They were too busy rendering security services to their clients that they failed to maintain positive attitudes in securing their own IT infrastructures. Challenging and going after the Anonymous group was the only little step needed to expose their massive negligence and vulnerable infrastructures. The little things they neglected became their biggest problems; no one would have expected such from an established security Firm like HBGary. The fall of HBGary is a clear indication that the bad guys are always a step ahead in their calculations, and they see tiny security lapses that are usually oblivious to security experts**.** Hence, this is a huge lesson to be learned by every individual, corporation and security professional, to stay equipped and well informed about standard security practices, maintaining positive security behaviour always. It is therefore very imperative that great security culture demands that nothing, however simple or irrelevant in appearance, should be treated casually when it pertains to security. Finally, it is now expedient that keeping a healthy cybersecurity work behaviour, cyber hygiene, and organizational planning is as core to information security as firewalls and anti-malware.

References

Avi, K. (2016). *The Dictionary Attack and the Rainbow-Table Attack on Password Protected Systems.* Purdue: Purdue University.

Aziz, A. (2013). The evolution of cyber attacks and next generation threat protection. *RSA Conference.*

Blythe, J. (2013). Cyber security in the workplace: Understanding and promoting behaviour change. *Proceedings of CHItaly 2013 Doctoral Consortium 1065,* (pp. 92-101).

Brian, D. (2010). *Determining the Role of the IA/Security Engineer.* SAN Institute.

CeBIT Australia. (2017, January 17). *The human factor in cyber security*. Retrieved January 26, 2017, from CelBit: http://blog.cebit.com.au/the-human-factor-in-cyber-security

Chester, W. (2011, February 7). *HBGary Federal hacked and exposed by Anonymous*. (naked security) Retrieved January 23, 2017, from https://nakedsecurity.sophos.com/2011/02/07/hbgary-federal-hacked-and-exposed-by-anonymous/

Coding Horror. (2007, September 8). *Rainbow Hash Cracking*. (Coding Horror: programming and human factors) Retrieved January 25, 2017, from https://blog.codinghorror.com/rainbow-hash-cracking/

Daniel, H. (2015, January 20). *Best Practices for Workplace Passwords*. (Software Advice) Retrieved January 25, 2017, from http://www.softwareadvice.com/security/industryview/password-workplace-report-2015/

Danish, J., & Hassan, Z. (2011). Cloud Computing Security. *International Journal of Engineering Science and Technology, 3*(4), 3478 - 3483.

David, B. (2015, March 23). *5 Social Engineering Attacks to Watch Out For*. (Tripwire) Retrieved January 25, 2017, from Tripwire: https://www.tripwire.com/state-of-security/security-awareness/5-social-engineering-attacks-to-watch-out-for/

Dunkelman, O., & Eli, B. (2006). A framework for iterative hash functions: Haifa. *2nd NIST Cryptographich Hash Workshop, 22.*

GBC-DELL Survey. (2015). *The Human Factor at the Core of Federal Cybersecurity.* Government Business Council.

IBM. (2015). *IBM Security Services 2014 Cyber Security Intelligence Index.* IBM Global Technology Services.

James, M. S. (2015). *10 Things Security Experts Wish End Users Knew.* Global Knowledge.

Keir, T. (2011, March 7). *8 Security Tips from the HBGary Hack*. Retrieved January 25, 2017, from PCWorld: http://www.pcworld.com/article/221504/8_security_tips_to_learn_from_the_hbgary_hack.html

Krebs, B. (2011, February 7). *HBGary Federal Hacked by Anonymous*. (krebsonsecurity) Retrieved January 23, 2017, from https://krebsonsecurity.com/2011/02/hbgary-federal-hacked-by-anonymous/

Mitnick, K. D., Simon, & L., W. (2011). *The art of deception: controlling the human element of security.* Indiana: John Wiley & Sons.

Nate, A., & Technica, a. (2011, February 10). *How One Man Tracked Down Anonymous — And Paid a Heavy Price*. Retrieved January 25, 2017, from WIRED: https://www.wired.com/2011/02/anonymous/

Nate, L. (2016, October 11). *What is Social Engineering? Defining and Avoiding Common Social Engineering Threats*. (Digital Guardian) Retrieved January 24, 2017, from https://digitalguardian.com/blog/what-social-engineering-defining-and-avoiding-common-social-engineering-threats

Patel, P. N., Patel, J. K., & Virparia, P. V. (2013). A Cryptography Application using Salt Hash Technique. *International Journal of Application or Innovation in Engineering & Management (IJAIEM), 2*(6), 1-4.

Peter, B. (2011, February 16). *Anonymous speaks: the inside story of the HBGary hack*. Retrieved January 25, 2017, from arsTechnica: http://arstechnica.com/tech-policy/2011/02/anonymous-speaks-the-inside-story-of-the-hbgary-hack/3/

Pittsburgh, P. A. (2015, January 13). *Security Awareness and Training Measurably Reduces Cyber Security Risk*. Retrieved January 28, 2017, from Wombat Security Technologies: https://www.wombatsecurity.com/press-releases/research-confirms-security-awareness-and-training-reduces-cyber-security-risk

Rahul, R., Venkiteswaran, R., Anoop, J. B., & Soumya, K. D. (2014). Android Malware Attacks and Countermeasures: Current and Future Directions. *Control, Instrumentation, Communication and Computational Technologies (ICCICCT), 2014 International Conference on IEEE,* (pp. 137-143).

Ranjeev, M., & Lawless, W. F. (2015). The human factor in cybersecurity and the Role for AI. *2015 AAAI Spring Symposium* (pp. 39-43). Springer.

Saeed, S., Saman, A., & Norafida, I. (2013). Main human factors affecting information system security. *Interdisciplinary Journal of Contemporary Research In Business, 5*(7), 329-354.

SANS. (2014). *Password Protection Policy.* London: SANS Institute.

Sjoerd, L. (2016, May 25). *Requirements for iterative password hashing.* Retrieved January 25, 2017, from https://www.sjoerdlangkemper.nl/2016/05/25/iterative-password-hashing/

Stuttard, D., & Marcus, P. (2011). *The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws.* John Wiley & Sons.

Warwick, A. (2016, February 26). *Social engineering confirmed as top information security threat*. (Computer Weekly) Retrieved January 23, 2017, from http://www.computerweekly.com/news/4500273577/Social-engineering-confirmed-as-top-information-security-threat

Software Engineering Consideration in the Design and Implementation of Electronic Patients Information Management System (E-PIMS) for University of Calabar Teaching Hospital (UCTH)

Sylvester I. Ele
Department of Computer Science
University of Calabar, Nigeria
el_silver2@yahoo.com

W. A. Adesola
Department of Computer Science,
Cross River University of Technology
Calabar, Cross River State Nigeria
adebisiadesola@crutech.edu.ng

E. E. Umoh
Department of Computer Science
Cross River University of Technology
Calabar, Nigeria
enoimah@yahoo.com

Olatunji Alani Akinola
Department of Computer Science and ICT
Institute of Technology and Management (ITM)
Ugep, Nigeria
akinolaolatunji@gmail.com

ABSTRACT

Purpose: The purpose of this work is to apply software engineering principles and methodologies to design and implement an electronic patients` information management system (E-PIMS) with a secured database, which would provide efficiency and reliability; reduces healthcare cost; and provide the best control of patients' status based on their test.

Design/Methodology/Approach: The E-PIMS employed the Structured System Analysis and Design Methodology (SSADM) to design the system using the waterfall process model. UML (Unified Modeling Language) was used as a tool to model the system. HTML with cascading style sheet (CSS) was used to create the web pages; JavaScript was adopted as the programming language because it has an Application Programming Interface (API). PHP was used to design our forms and MYSQL was used as the database engine.

Findings: The study has been able to demonstrate that following the software engineering life-cycle, from inception up to deployment stage produces software-based systems that allow customers achieve business goals.

Research Limitation: The major limiting factor in this research work was time, materials and money which hindered the full implementation of the system.

Originality/Value: This research work is valuable because a system of this nature has never been deployed or used in any hospital in Calabar including the UCTH, and if fully implemented and deployed will greatly improve the services of the University of Calabar Teaching Hospital.

***Keywords**: SoftwareEngineering, Information System, ICT.*

***Paper Type**: Implementation*

1.0 Introduction

Patient information system has been discovered to be the key contributor to success in the prescription of drugs for major ailments like diabetes and others (Yerokun, Ekechukwu & Ihemelu, 2012).

In the last couple of years the rise in Information and Communication technologies (ICT) has a very big impact in services for community, especially in the healthcare sector. This technology has been used and deployed in Nigeria, by developing an electronic medical database to manage patients' records. Primarily,in any teaching hospital and University of Calabar Teaching Hospital (UCTH) in particular,three types of medical care services exist, which include in-patient (hospitals patients), out-patient (patients in clinics unit) and emergency unit. The in-patient hospitals are classified into two different categories: specialized hospitals (GITCenters, Cardiac Centers, Cancer Centers, etc.), and general hospitals. Hospitals are seen asindispensable in healthcare infrastructure, hence the need to improve the services in it. Our E-PIMSwill beuseful for management of patient health records, research, and archiving. In management, it could be used for UCTH Chief Medical Director (CMD) to see the performance of the Doctors, or statistical reporting. Besides the Doctors can have the patient history in details from his previous records with minimal time. Archiving and securing electronic patients` records is seen as a more reliable and trusted approachthan paper-based records (Laman, C., 2004).

The aim of this work is to apply software engineering principles and methodologies in the design and implementation of an electronic patients` information management system. Software engineering is the application of a systematic, disciplined, quantifiable approach to the development, operation and maintenance of software (Schwaber and Beedle, 2002). The use of software engineering methodologies help all along the software engineering life-cycle, from inception up to deployment stage.The purpose of software engineering is to develop software-based systems that let customers achieve business goals. The customer may be a hospital manager who needs patient-record software to be used by secretaries in doctors' offices; or, a manufacturing manager who needs software to coordinate multiple parallel production activities that feed into a final assembly stage. Software engineer must understand the customer's business needs and design software to help meet them (Marsic, 2012).Many contemporary software developers tend to ignore software engineering principles which is the main kernel in software design and development. In our E-PIMS, we followed the well-defined and structured sequence of stages in software engineering to develop the E-PIMS package.

1.1 Architecture of our Proposed E-PIMS

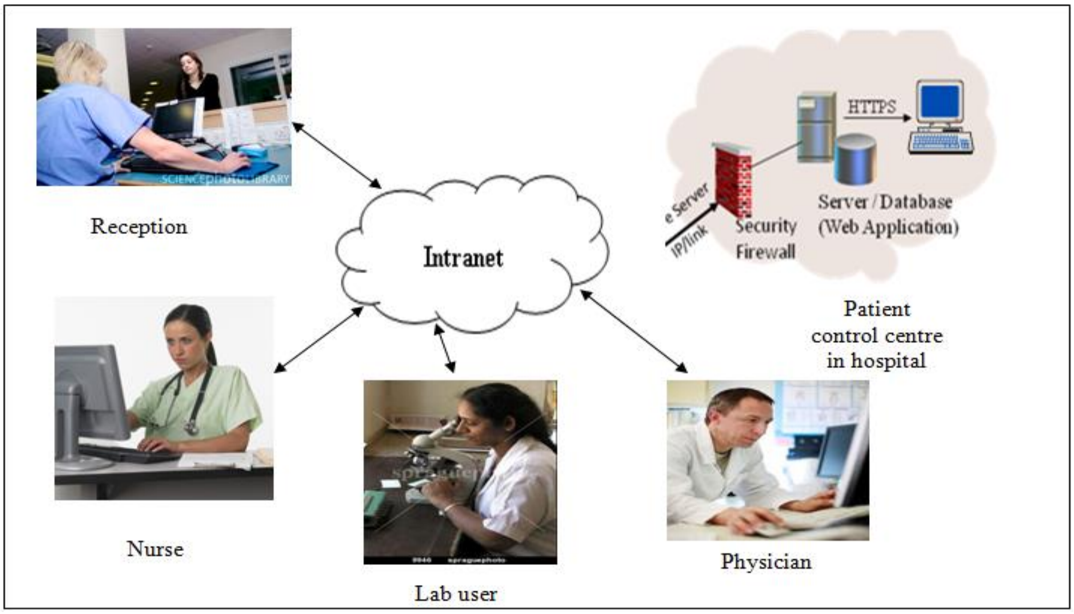


Figure 1.1: Architecture of our E-PIMS (Zaid, and Abd Alameer - unpublished)

2.0 System Design and Methodology

The principal objective of the Electronic Patient Information system Management System is to provide a solution to the manual methods of handling patient information in a hospital. The E-PIMS will help to improve data security and avoid data loss completely. The proposed system will minimize the time involved in locating patient information when needed. The methodology which will be used in the design of the E-PIMS and the analysis of system requirements will be discussed in this section. The system development lifecycle was divided into phases discussed below.

2.1 *Unstructured Interview*

To develop any system successfully is to understand its requirements. In E-PIMS, we conducted an interview withstakeholders such as doctors, nurse and, pharmacists,and statistics department for gathering rich information about system requirements to design and build the system.

2.2 *System Requirements Analysis*

Requirement is a service that the user desires the solution to perform or display. In this section we determined user expectations and conditions for the E-PIMS, and also took into account the possibly conflicting requirements of the various stakeholders, andanalyzed, documented, validated and managed the software requirements*(*Chemuturi, 2013).

The functional requirements for the E-PIMS are as follows:

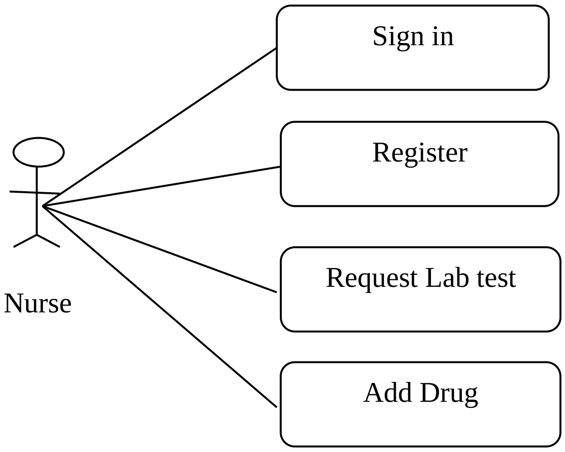
- New patient records can be added
- The system sends patient record to the nurse to add new patient records
- The system enables the nurse to search about a specific patient
- The system archives patient records electronically and centrally.
- The system enables the nurse to request a specific lab test.
- The system has the ability to send lab result to the nurse or the doctor.
- The system enables the nurse and the doctor to search and research
- The system enables the doctor to check patient history by searching with minimal time.

Non-functional requirements of the E-PIMS include updateable, security, compatibility, capacity, usability, maintainability and performance with database. The system for instance interacts with database, database searches, updates and retrieves the change to patient information on real time basis.

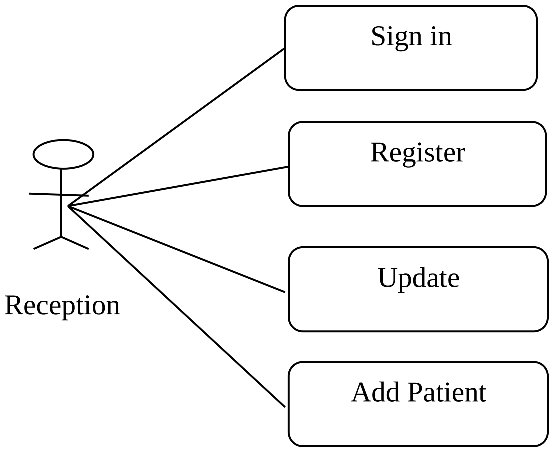
2.3 *System Design*

2.3.1 Use Case Diagram

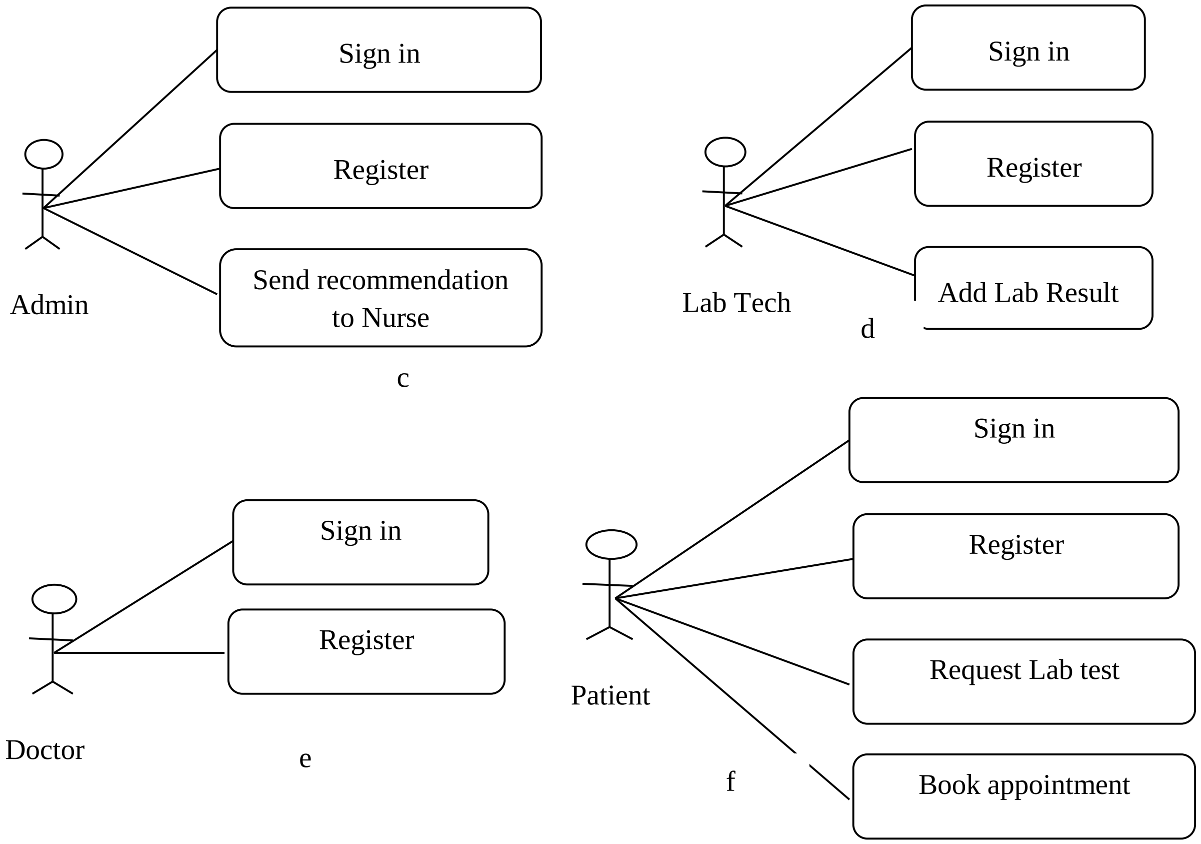
Use Case diagrams permit the definition of the system's boundary, and the relationship between the system and the environment (Marsic, 2012). There arefive users (Receptionist, Lab Technician, Nurse, Doctor and Administration) in the diagrams for the E-PIMS. Each one becomes an actor and has many functions in the system as shown below.



a



b



Figures 2.1a-f: Use Case Diagram of the E-PIMS

2.3.2 Class diagram

Class diagram is created simply by analyzing the class names and their operations of the interaction diagrams. The class diagram of our case-study system is shown in Figure 2.2. The objects that make up our classes in this system are Doctor_class, Nurse_class, Lab Tech_class, Receptionist_class, Admin_class and Patient_class.

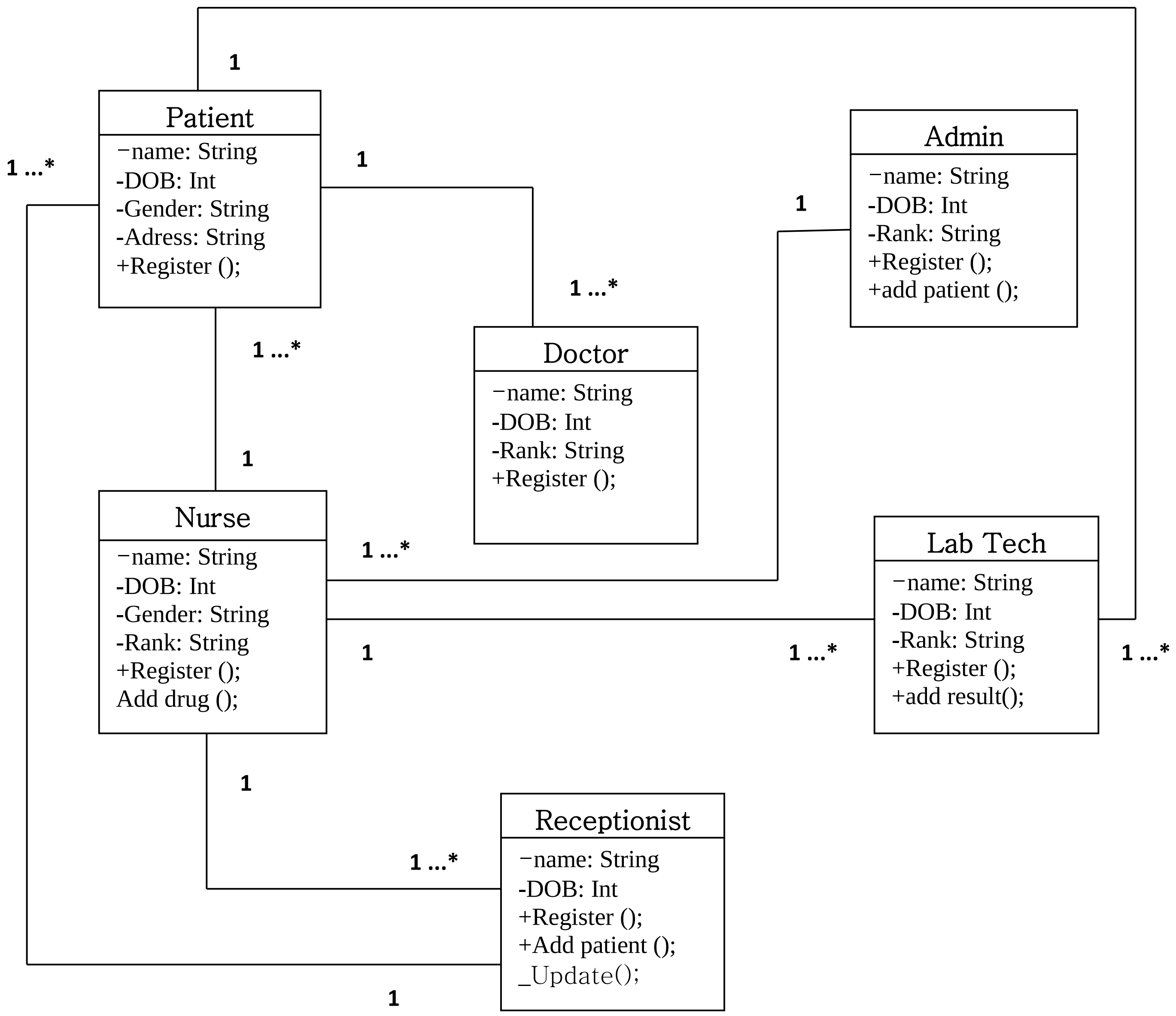


Figure 2.2: Class Diagram of the E-PIMS

2.3.3 *Database Design*

The various entities that make up the E-PIMS database has been identified. MySQL database was used for this system because of its simplicity and ease to use. The database consists of 6 tables which are described below.

Patient table: a table to fill patient personal data to be able to gain admission into the hospital for proper registration process for outpatients.

Table 2.1: Patient_table

FIELD NAME	TYPE	WIDTH	NULL	FOREIGNKEY
Firstname	Char	30	No	No
Middlename	Char	30	No	No
Surname	Char	30	No	No
Date of Birth	Date	20	No	No
Gender	Char	10	No	No
Status	Char	15	No	No
Disability	Varchar	20	No	No
Address	Varchar	50	No	No
Phone	Varchar	20	No	No
Local_government	Varchar	30	No	No
Country	Varchar	30	No	No
State	Varchar	30	No	No
Occupation	Varchar	20	No	No
Salary_Scale	Varchar	20	No	No
Next of kin	Varchar	50	No	No
Start_Time	Date	20	No	No
End_Time	Date	20	No	No
Remark	Varchar	50	No	No

Receptionist form: A table to fill employee’s personal data to be able to be gainfully employed into the hospital management staff list in the database.

Table 2.2: Reception_form

FIELD NAME	TYPE	WIDTH	NULL		FOREIGNKEY
Firstname	Char	30	No		No
Middlename	Char	30	No		No
Surname	Char	30	No		No
Employee_id	Varchar	15	No		No
Gender	Char	10	No		No
Status	Char	15	No		No
Age	Int	10	No		No
Address	Varchar	50	No		No
Phone	Varchar	20	No		No
Local_government	Varchar	30	No		No
Country	Varchar	30	No		No
State	Varchar	30	No		No
Occupation	Varchar	20	No		No
Salary_Scale	Varchar	20	No		No
Next of kin	Varchar	50	No		No
Start_Time	Date	20	No		No
End_Time	Date	20	No		No
Remark	Varchar	50	No		No

Table 2.3: Doctors_form

FIELD NAME	TYPE	WIDTH	NULL	FOREIGNKEY
Firstname	Char	30	No	No
Middlename	Char	30	No	No
Surname	Char	30	No	No
Doctor_id	Varchar	15	No	No
Gender	Char	10	No	No
Status	Char	15	No	No
Age	Int	10	No	No
Address	Varchar	50	No	No
Phone	Varchar	20	No	No
Local_government	Varchar	30	No	No
Country	Varchar	30	No	No
State	Varchar	30	No	No
Specialization	Varchar	20	No	No
Salary_Scale	Varchar	20	No	No
Next of kin	Varchar	50	No	No
Start_Time	Date	20	No	No
End_Time	Date	20	No	No
Notes	Varchar	50	No	No

Table 2.4: Nurse_form

FIELD NAME	TYPE	WIDTH	NULL	FOREIGNKEY
Firstname	Char	30	No	No
Middlename	Char	30	No	No
Surname	Char	30	No	No
Nurse_id	Varchar	15	No	No
Gender	Char	10	No	No
Status	Char	15	No	No
Age	Int	10	No	No
Address	Varchar	50	No	No
Phone	Varchar	20	No	No
Local_government	Varchar	30	No	No
Country	Varchar	30	No	No
State	Varchar	30	No	No
Job_description	Varchar	20	No	No
Notes	Varchar	50	No	No

Table 2.5: Admin table

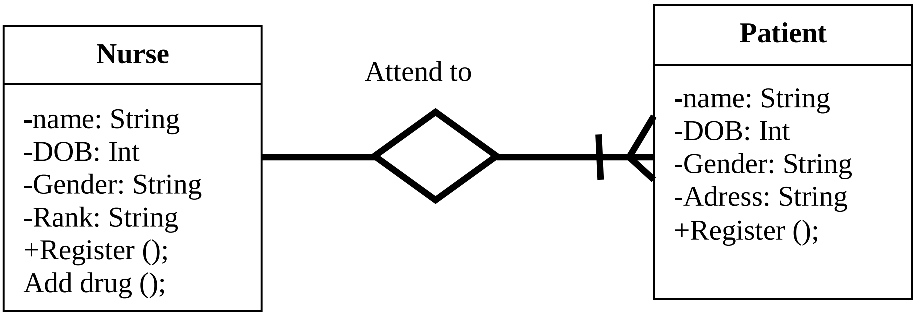
FIELD NAME	TYPE	WIDTH	NULL	FOREIGNKEY
Firstname	Char	30	No	No
Middlename	Char	30	No	No
Surname	Char	30	No	No
Username	Varchar	15	No	No
Password	Varchar	20	No	No

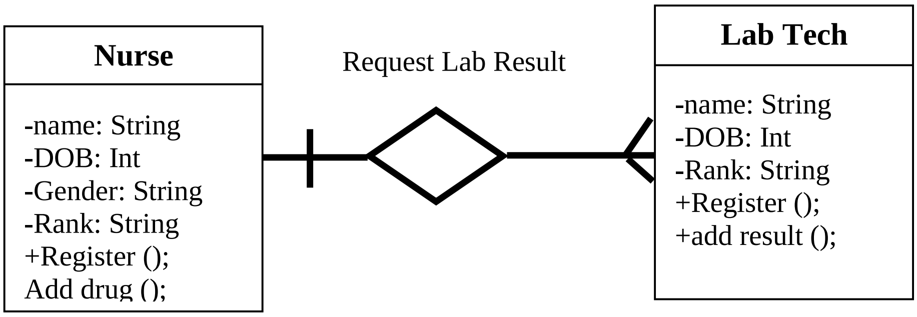
Table 2.4: Lab Tech_form

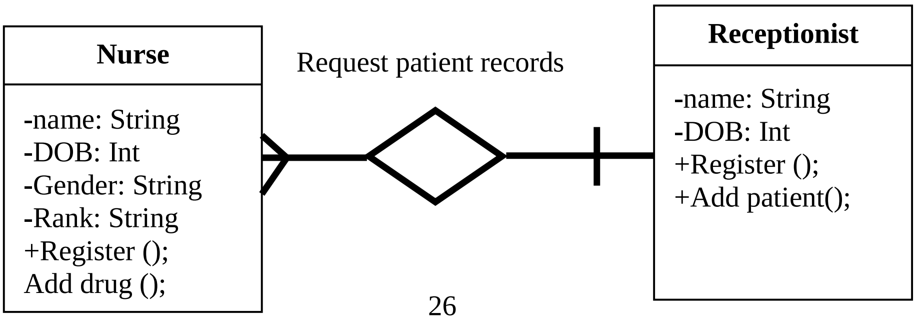
FIELD NAME	TYPE	WIDTH	NULL	FOREIGNKEY
Firstname	Char	30	No	No
Middlename	Char	30	No	No
Surname	Char	30	No	No
LTech_id	Varchar	15	No	No
Gender	Char	10	No	No
Status	Char	15	No	No
Age	Int	10	No	No
Address	Varchar	50	No	No
Phone	Varchar	20	No	No
Local_Government	Varchar	30	No	No
Country	Varchar	30	No	No
State	Varchar	30	No	No
Job_description	Varchar	20	No	No
Notes	Varchar	50	No	No

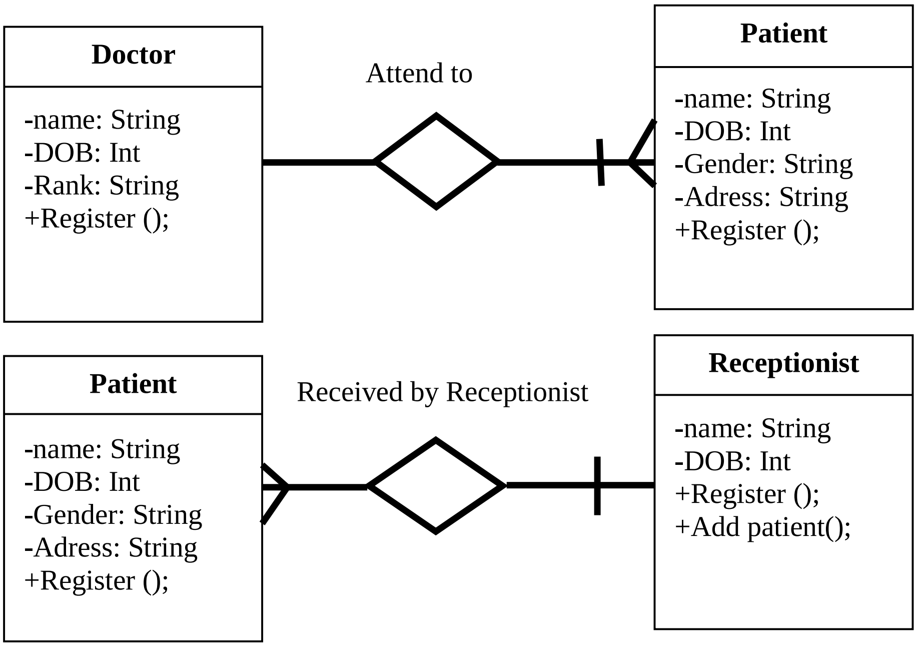
2.3.3.1 Relationship among Entities in the E-PIMS Database

Here we state the relationship that exists among entities in the database using the information engineering notation style to represent their relationships and cardinality.









3. System Implementation

The Prototype of the E-PIMS was implemented using HTML with cascading style sheet (CSS), JavaScript, PHP and MYSQ as the database engine.These programming languages were used based on the features and objectives of the system.

2.4 *System Testing*

System testing involves the various activities carried out to discover possible issues that might still be found in the designed system. System testing uncovers weaknesses that were not found in earlier testing normally. This can include system failure.

2.4.1 Unit Testing

The aim of this test is to separate each part of the program and show that each is working correctly. Each unit was tested separately before integrating them into modules to test the interfaces between modules.

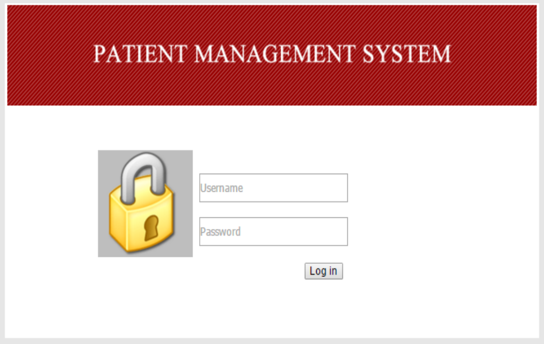
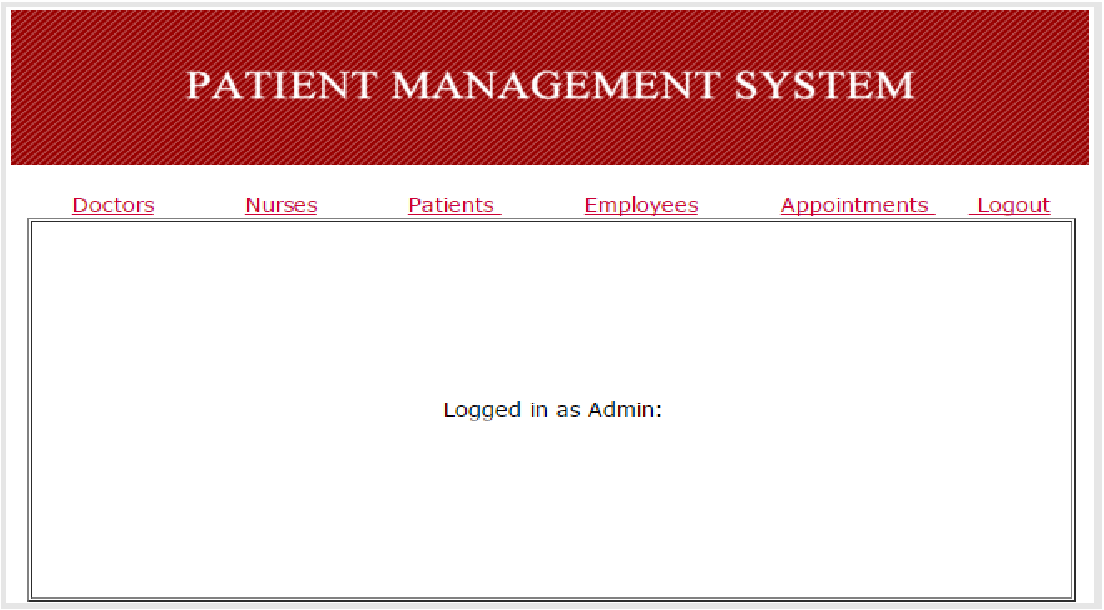
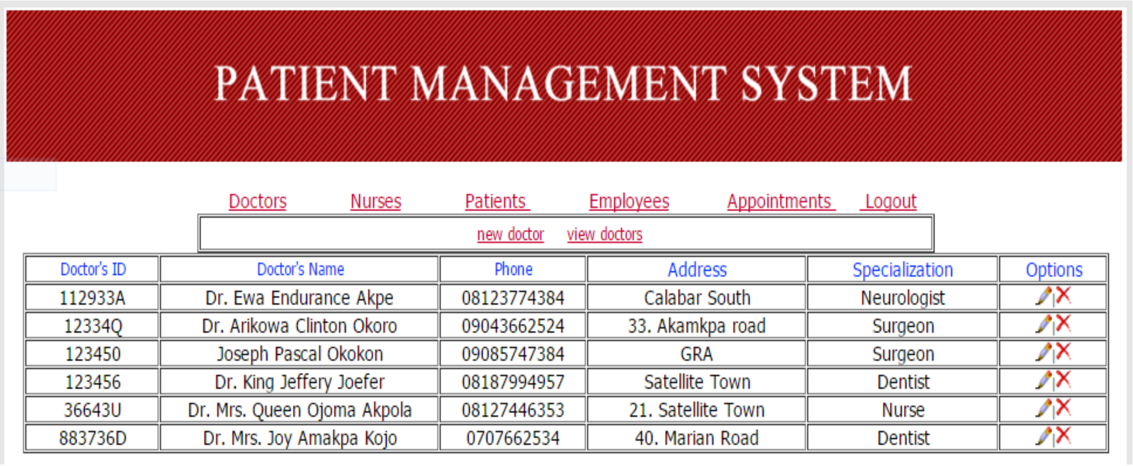


Fig. 3.1: Interface of the Login Page



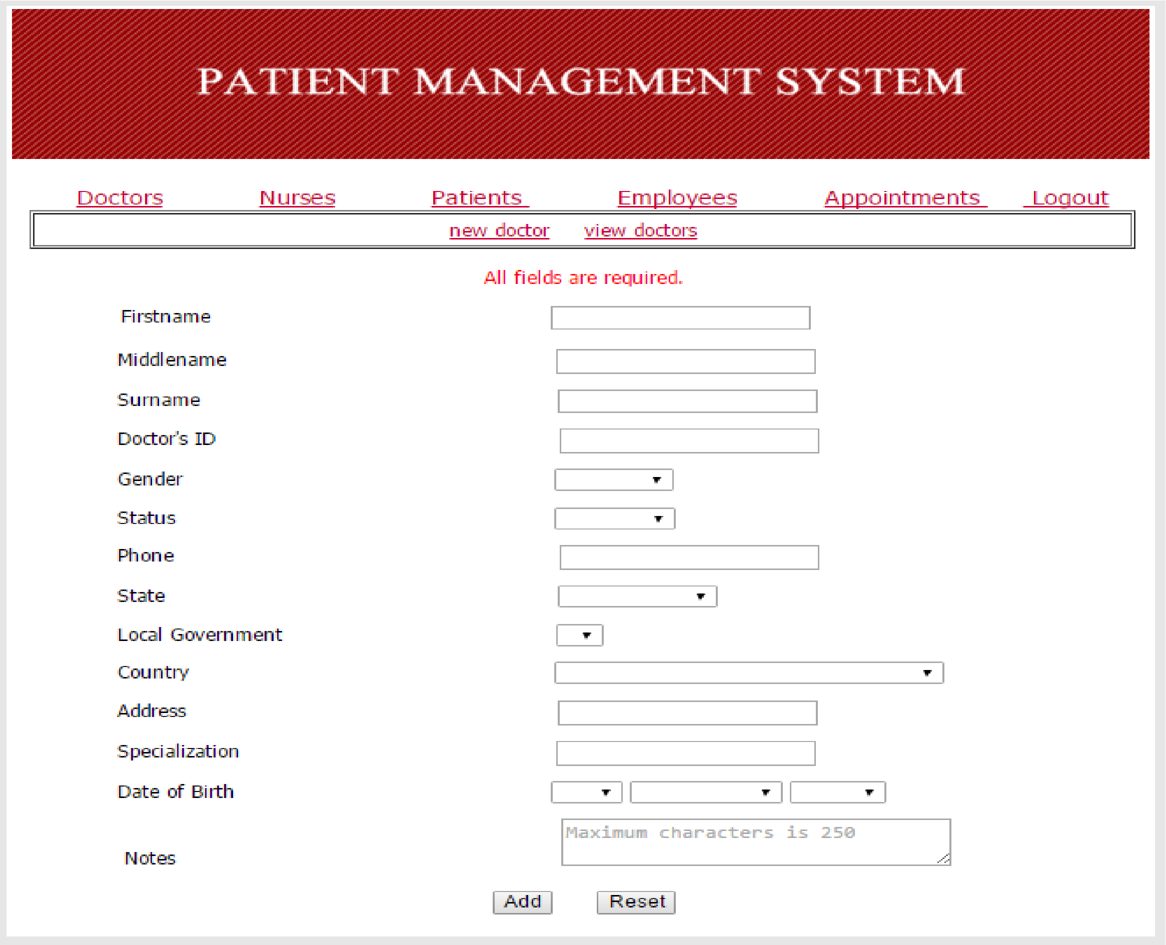
Logged in as Admin:

Figure.3.2: Interface of the Logged in Page



Doctor's ID	Doctor's Name	Phone	Address	Specialization	Options
112933A	Dr. Ewa Endurance Akpe	08123774384	Calabar South	Neurologist	✎✗
12334Q	Dr. Arikowa Clinton Okoro	09043662524	33. Akamkpa road	Surgeon	✎✗
123450	Joseph Pascal Okokon	09085747384	GRA	Surgeon	✎✗
123456	Dr. King Jeffery Joefer	08187994957	Satellite Town	Dentist	✎✗
36643U	Dr. Mrs. Queen Ojoma Akpola	08127446353	21. Satellite Town	Nurse	✎✗
883736D	Dr. Mrs. Joy Amakpa Kojo	0707662534	40. Marian Road	Dentist	✎✗

Figure 3.3: View doctors page



PATIENT MANAGEMENT SYSTEM

All fields are required.

Firstname

Middlename

Surname

Doctor's ID

Gender

Status

Phone

State

Local Government

Country

Address

Specialization

Date of Birth

Notes

Maximum characters is 250

Figure 3.4: Add doctor's form

3. Conclusion

In this research work, design and implementation of Electonric Patient Information Management System (E-PIMS), a centralized database contains the in-patient and out-patient recordswhich have been designed and the prototype of the proposed system implemented. The aim was to provide a reliable and efficient web-based healthcare services. The system would enhance the provision of services to patients by making their records available online for easy access by doctors and nurses to follow up cases with less effort, and would also make patient history readily available to health workers. Hospital directors and heads of departments can follow the physician work related to patients from diagnosis to treament and referal state. It is hoped that this system, when fully deployed would offer better customer service than those without proper software engineering approaches.

References

Chemuturi, M. (2013). *Requirements Engineering and Management for Software Development Project*, London: Springer.

Laman, C. (2004). Applying UML and Pattern: An Introduction to Object Oriented Analysis and Design and Interactive Development, 3ed. Addison Wesley Professional, USA.

Marsic, I. (2012). *Software Engineering*. Rutgers University, New Brunswick, New Jersey. http://www.ece.rutgers.edu/~marsic/books/SE/[Accessed on 28 – 10- 2015]

Yerokun, M., Ekechukwu, B. and Ihemelu, I. (2012). Automated Patient Information Systems for Federal Government Hospitals in Nigeria. *West African Journal of Industrial and Academic Research* (WAJIAR) Vol.5, No. 1, pp. 158 - 166, December 2012.

Schwaber, K. and M. Beedle (2002). *Agile Software Development with SCRUM*, NJ: Prentice-Hall.

Zaid H. N. and Abd Alameer, E. M. T. Electronic Patient Record Management System (EPRMS). Unpublished Research article, Kerbala University/College of Sciences.

About the authors

Ele, Sylvester Igbo is a lecturer in the Department of Computer Science, University of Calabar. Ele earned a B.Sc degree in computer Science in 2006 from Cross River University of Technology, Calabar. From 2008 to 2012 he garnered an M.Sc degree in Computer Science from the University of Nigeria Nsukka. Ele’s areas of interest are in Expert Systems, Databases, Cyber Security, Algorithms, Information Technology and software Engineering.

Adesola, W. A. isa lecturer in the Department of Computer Science, Cross River University of Technology, Calabar. Adesola holds a B.SC and MSc Degrees in Computer Science and an M.Sc and Ph.D degrees in Banking and Finance.

Umoh, E.E isa lecturer in the Department of Computer Science, Cross River University of Technology, Calabar. Umoh holds a B.Sc, MSc and a Ph.D Degrees in Computer Science.

Akinola, Olatunji Alani is a lecturer in the Department of Computer Science and ICTInstitute of Technology and Management (ITM)**,** Ugep, Nigeria. Akinola holds a B.Ed (Honour) in Computer Science and a Master Degree in Information Science (M.Inf.Sc).

Automating the data quality checks in health and demographic surveillance systems: Lessons from the Cross River HDSS, Nigeria

Iwara Arikpo*[,1,2], Ideba Mboto[1], Anthony Okoro[1], Martin Meremikwu[1,3]
[1]Cross River HDSS, Directorate of Research & Quality Assurance
[2]Department of Computer Science, [3]Department of Paediatrics
University of Calabar, Nigeria
*Corresponding Author: iiarikpo@gmail.com, iwara.arikpo@unical.edu.ng

ABSTRACT

Purpose: To investigate the challenge of replacing the manual data quality routines with valid electronic protocols for health and demographic surveillance systems (HDSS), using lessons from the in Cross River HDSS in Cross River State, Nigeria.

Methodology: Interviews and examination of household questionnaires. Then, a free software called Randomizer is used to generate random records which are matched with real records. In a second approach, an R Statistical Computing script was developed to extract 5% of the matched records either from an Excel spreadsheet or from a database. These household records are made available for quality assurance interviews.

Findings: The study showed major flaws with the manual procedures for data quality checks in health and demographic surveillance systems, and the impracticability of using the manual method in systems where data collection is predominantly electronic.

Research limitations: Repeat interviews by field supervisors are still performed with paper forms. Future research may therefore attempt to design solutions that can facilitate repeat interviews using mobile devices.

Practical implications: The automated solutions presented in this study fit into the information technology revolution and the design of health and demographic surveillance system sites to leverage the strengths of this technology.

Keywords: *Health and demographic surveillance system, information technology, update round, data quality checks, validation, database.*

Paper Type: *Research Paper*

1. Background

Health and Demographic Surveillance Systems (HDSS) sites are robust longitudinal data collection platforms operating under a global network called INDEPTH (International Network for the Demographic Evaluation of Populations and Their Health). The shared goal of health and demographic surveillance systems is to provide reliable longitudinal data on vital events such as, births, deaths, migrations and other socio-economic indicators in low-income and developing countries where routine vital health information is incomplete or near-absent (Ye, Wamukoya, Ezeh, Emina, & Sankoh, 2012; Byass, et al., 2002). This routine information collected during update rounds, guide government and relevant stakeholders on policy formulation & development, programme planning, health and social service delivery, allocation of resources, and monitoring and evaluation (Madeleine, YoonJoung, & Sandra, 2012), among others. Figure 1 shows the traditional events relationship in a health and demographic surveillance system.

The attainment of the goal of HDSSs lies on the quality of data generated from the research sites. Consequently, reliability, consistency and validity of data are sacrosanct in HDSS, in both the field operations and data systems. In order to accomplish these data quality standards, the INDEPTH Network, as a core policy, recommends standard quality control measures for all sites within the Network to follow in ensuring data quality. Some of these measures include spot checks, repeat interviews on 5% sample from the events monitored during a particular data collection cycle and Red Herring procedures (INDEPTH Network, 2006).

Over the years, across the various health and demographic surveillance system sites, data collection procedures had been predominantly paper-based, and so is the procedure for performing the mandatory data quality checks after each data collection cycle, known as an update round. However, the advent of mobile and ubiquitous computing propelled by the increasing reduction in cost of smartphones, PDAs (personal digital assistants), netbooks, tablet PCs and laptop computers has revolutionized the way data is collected and managed. HDSS sites are increasingly leveraging this information technology revolution. As a result, the older sites are migrating to paperless data collection systems, while newer HDSS sites are implementing complete paperless health and demographic surveillance systems (Arikpo, et al., 2013).

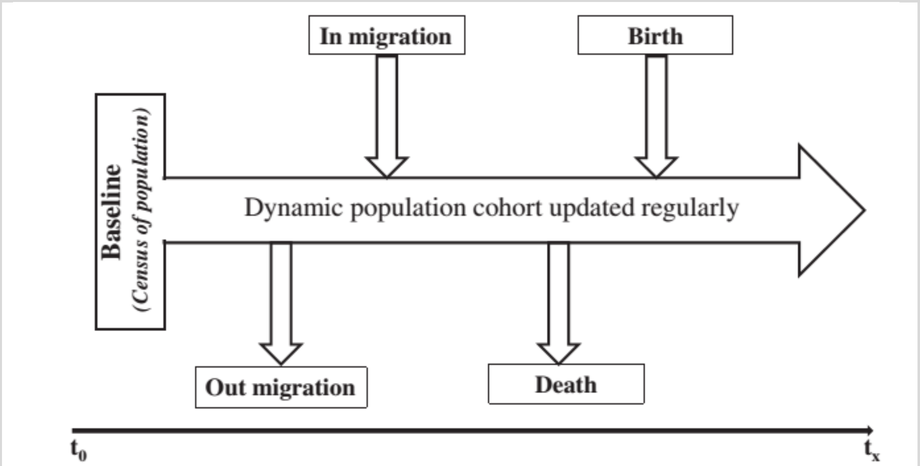


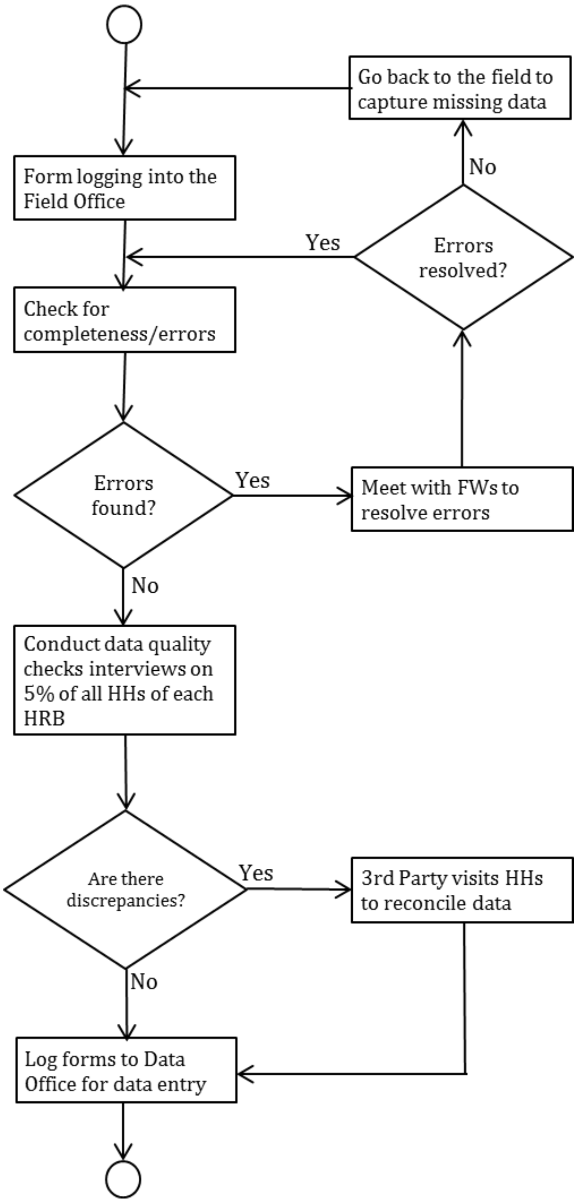
Figure 1: Traditional events relationships in a HDSS (Ye, et al., 2012)

In addition to the problem of migrating legacy data systems, this new approach comes with new challenges in terms of data collection and management, chief among which is, how to conduct the mandatory data quality checks in a paperless HDSS setting. In answer to this challenge, this paper presents the lessons learned from the automation of data quality checks in a paperless HDSS setting in line with the INDEPTH quality assurance standards. The paper begins with a description of the procedures for data quality checks in traditional paper-based health & demographic surveillance systems, highlighting the shortcomings inherent in these systems. This is followed by the methods, where the research setting and automated quality checks in paperless HDSS is described, followed by a discussion of the results, and then a conclusion of this study.

2. Data Quality Checks in Paper-based HDSS

In a typical paper-based HDSS (health & demographic surveillance systems) site, at the end of each update round (which is either 4-monthly or 6-monthly), the field workers (FWs) log in the various event forms [“*an event is a change in the state of an individual”* (INDEPTH Network, 2006)], such as Pregnancy Registration, Birth Registration, In- and Out-Migration, Verbal Autopsy, etc. Form logging is a process whereby each field worker brings his/her forms and the HRB (household registration book) to the Field Office (FO) for checks and submission. The data forms for each field worker are then checked to confirm that the number submitted tallies with the number indicated both in the Event Listing form and the HRB. Form logging is handled by a Filing Clerk in the Field Office, and serves as the first stage of the quality checking process.

A research assistant or equivalent, then checks each of the logged in forms for completeness and errors, vis-à-vis the recording of that event in the household registration book. If errors are found, the affected field worker is called in to explain the situation in a bid to resolving it, depending on the magnitude of the error. If the error cannot be resolved, the field worker goes back to the field to re-capture the missing or faulty data. If on the other hand, there were no errors found on the logged forms, the next stage is to conduct a quality checks interview on (minimum of) 5% of all households in each round of data collection. At this stage, the supervisor goes to the field with the randomly-selected event forms to re-interview these households using new data collection forms. At the end of this exercise, the supervisor returns to the field office with the forms, for a comparison with what was collected by the field workers. Each of the matched forms is compared with the re-interview forms for consistency. If no discrepancies are found, the original forms (by the field workers) are logged to the Data Office (DO) for data entry by data entry clerks (DECs). On the other hand, if discrepancies are established between the original data and the re-interview data, a third party, usually a research assistant visits the affected household to verify and reconcile the data before logging in to the Data Office. Figure 2 is a flowchart showing a high-level view of the logical steps involved in data quality checks in a typical paper-based health & demographic surveillance system.



Go back to the field to capture missing data
No
Form logging into the Field Office
Yes
Errors resolved?
Check for completeness/errors
Errors found?
Yes
Meet with FWs to resolve errors
No
Conduct data quality checks interviews on 5% of all HHs of each HRB
Are there discrepancies?
Yes
3rd Party visits HHs to reconcile data
No
Log forms to Data Office for data entry

Figure 2: Data quality checks in a Paper-based HDSS

2.1 Problems with data quality checks in the paper-based system

2.1.1 Reporting delays & costs on resources

Traditional data quality checks employed in paper-based HDSS sites is associated with higher costs in terms of time and resources than use of electronic protocols. The rigorous manual process of examining the data to discover inconsistencies and other anomalies takes a toll on human and material resources. The Data Quality Assurance Supervisors are subjected to long periods of waiting for the fieldworkers log forms into manual field and database registers. Data analysis and reporting activities become stalled quite often due to these delays in manual data quality procedures with frequent lost opportunities for timely utilization of useful information generated from the HDSS or nested studies.

2.1.2 Buck-passing Problem

In addition to delays, another problem is the begrudging of the relationship between field workers and data quality supervisors in paper-based data collection systems of health and demographic surveillance systems, because of the misconception by field workers that, the sole aim of data quality supervisors is to find fault with their work. This can generate some strained working relationships capable of eroding trust between field workers and data quality supervisors, thereby creating a problem of buck-passing. Field workers can question the 'relevance' of repeating interviews they had conducted, with conclusions that the supervisors lack trust in their capability and sincerity.

2.1.3 Problem with Red Herring Technique

In applying the "red herring technique" the quality assurance supervisor would deliberately add into a household query form, information that is known to be always true in that context overtime, but is not supposed to be collected from the particular individual or household in question. The idea is for the field worker to identify this false information about the individuals and report it as "non-existent" or "not required"; otherwise the field worker will have failed to detect the false information indicating that they did not physically visit the household to collect the missing/queried data. The central idea is to spot data falsifiers among the field workers (INDEPTH Network, 2008).

This technique, although very popular among HDSS sites, has faced some criticisms. For instance, while improving data quality through the red herring technique, drafting of the 'erroneous' information on paper form that already has recently collected data, is tasking and time consuming. Occasionally, these fictitious inclusions are easily spotted by the field workers even before going back to the field, because the data are written on paper forms that already have data, probably collected by them. Once the trick is detected, then the central idea behind the red herring technique will have been defeated, because field workers will deliberately return or fill the data as expected (Laney, et al., 2008).

Generally, these problems are commonly faced by data quality supervisors in paper-based settings, and they limit the robustness, timeliness and correctness of this approach to quality checks. The paper seeks to illustrate how some of these challenges can be eliminated or mitigated using the electronic data quality protocol.

3. Methods

3.1 Study Area

The Cross River health and demographic surveillance system (Cross River HDSS) operates two research cohorts located within the southern senatorial district of Cross River State in south-south Nigeria, with a combined population of 33,446 persons in 8,508 households (48% of which are rural dwellers). The first is a rural cohort located in the Akpabuyo Local Government Area (LGA) of the state and the second, an urban cohort located in Calabar-Municipal, the state capital. The rural and urban cohorts are delineated into 46 and 43 contiguous enumeration areas (EAs) respectively. Both cohorts are situated in the tropical rain forest belt of southern Nigeria, with an annual rainfall in the range of 2500mm to 3000mm and mean annual temperature of 30°C. English and Efik are the major languages spoken by inhabitants of the two LGAs. The Cross River HDSS was set up as a paperless HDSS. In other words, data collection, transmission and management is done with the use of information technology-enabled devices, instead of paper forms.

3.2 Data Quality Checks in Paperless HDSS

As noted earlier, in a paperless HDSS, events data are not collected with paper as is obtainable in traditional paper-based systems. Rather, data is collected and transmitted electronically from the field using mobile devices, such as smartphones, tablet PCs, PDAs, etc. (Kaneko, et al., 2012; Arikpo, et al., 2013). There is no manual form logging in both the Field Office and the Data Office. There is no separate data entry, since interviews are conducted with electronic devices carrying electronic data templates. As a first stage to data quality checks, the electronic templates running ODK Collect, used to capture the data, has inbuilt validation rules to check against incomplete or inconsistent data values at the point of data capture. In addition, the OpenHDS (Pasquale, Mitra, & Maire, n.d.) which is the core (open source) software for HDSS data management, comes with a host of other data validation tools on the side of the database, once the data drops onto the ODKAggregate Server (Open Data Kit, 2015). One of these tools is the

MirthConnect which checks to ensure that each transmitted data ties to a record in the database before the data is allowed to drop on the database server, thus facilitating interoperability among disparate systems (Mirth Corporation, 2015).

In addition to all these initial validation checks, the next layer of checks is to conduct repeat interviews (re-interviews) for a randomly-selected set of records after the data has dropped onto the database server. Selecting at least 5% of records from each submitted event is a process that must be carefully done to avoid bias in the selected samples. The best approach is to do a random selection using a method/formula that computes random records in a scientifically sound way. Figures 3 and 4 are flow charts describing data quality checks in a paperless HDSS.

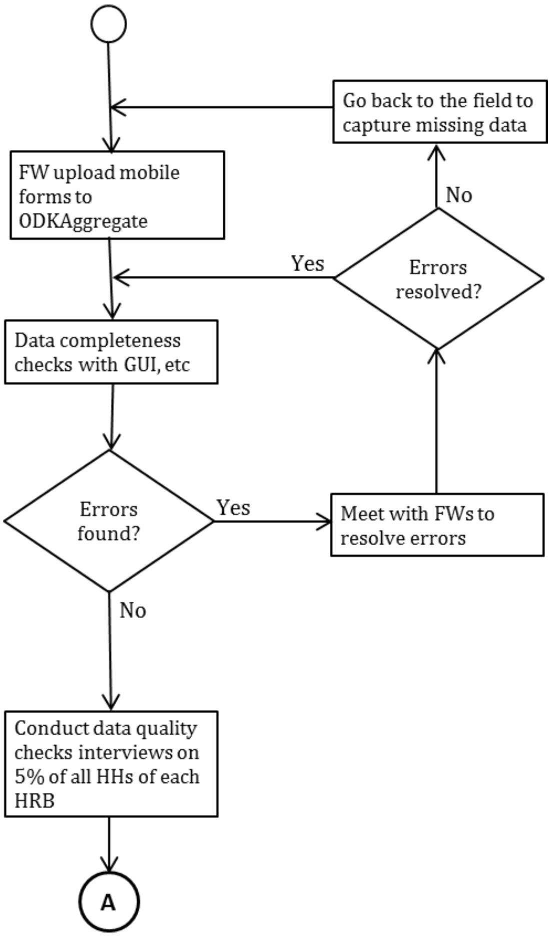

Go back to the field to capture missing data
FW upload mobile forms to ODKAggregate
No
Yes
Errors resolved?
Data completeness checks with GUI, etc
Errors found?
Yes
Meet with FWs to resolve errors
No
Conduct data quality checks interviews on 5% of all HHs of each HRB
A

Figure 3: Section of Flowchart for electronic quality checks

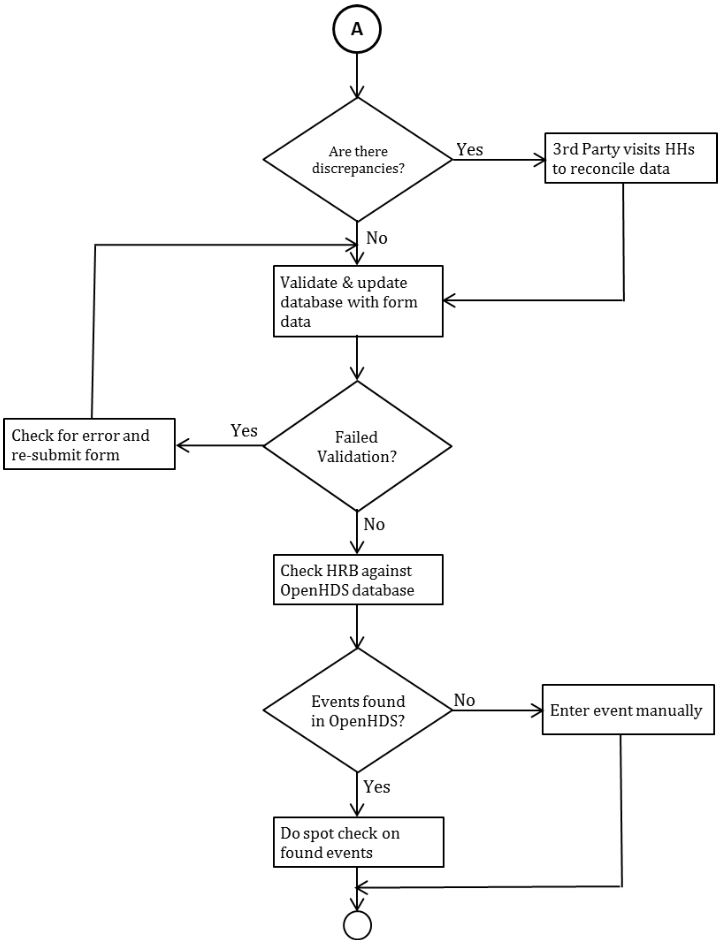

A
Are there discrepancies?
Yes
3rd Party visits HHs to reconcile data
No
Validate & update database with form data
Failed Validation?
Yes
Check for error and re-submit form
No
Check HRB against OpenHDS database
Events found in OpenHDS?
No
Enter event manually
Yes
Do spot check on found events

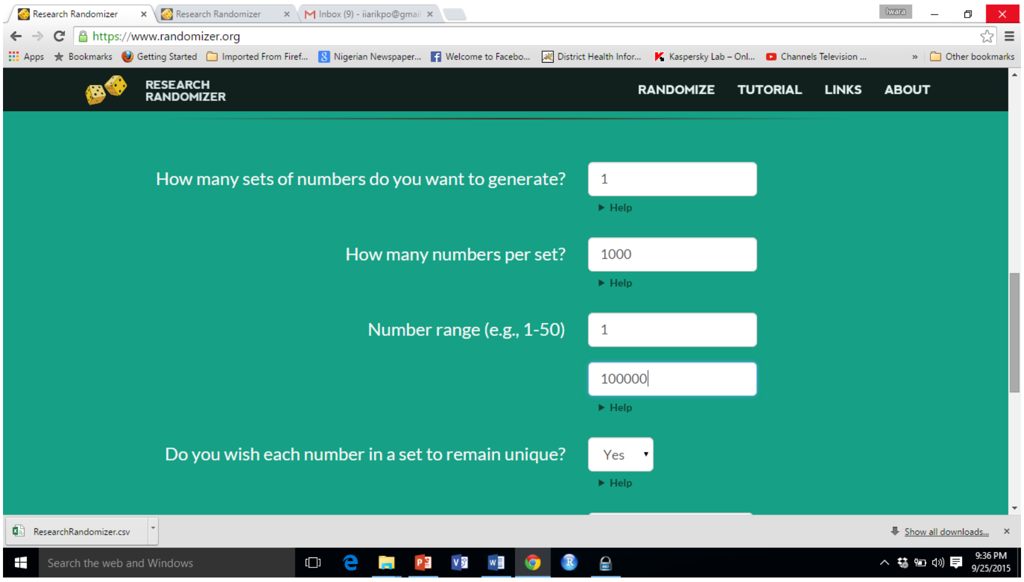
Figure 4: Second segment of Flowchart for electronic quality checks

3.2.1 Solution Option 1

At the Cross River HDSS, our first approach was the use of a free web-based application called 'Research Randomizer' (Social Psychology Network, 2015) for the computation of random records. This application generates the random records for quality assurance, from each event currently captured per update round and the households under surveillance. Apart from the households with events, quality assurance is also done on households without events during the round. This helps to validate the status of individual members in such households. The Research Randomizer accepts the following as inputs:

1) ***The number of records to be generated:*** To get this value, the data quality supervisor calculates the required number of records for random generation using calculator (in this case, 5% of all records for that event) and then inputs that value into the randomizer.

2) ***The range of values to generate the random numbers from:*** Based on the number of records for each event or households available, the supervisor enters the range within which the random numbers are generated. For instance, if the pregnancy observation event has 1000 records for that round, the range will be from 1 to 1000. Figures 5 – 7 outlays the random records generation process.



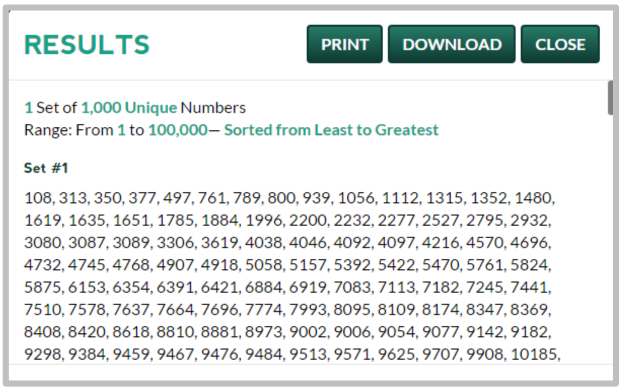
How many sets of numbers do you want to generate? 1

How many numbers per set? 1000

Number range (e.g., 1-50) 1 100000

Do you wish each number in a set to remain unique? Yes

Figure 5: Basic inputs for the Randomizer

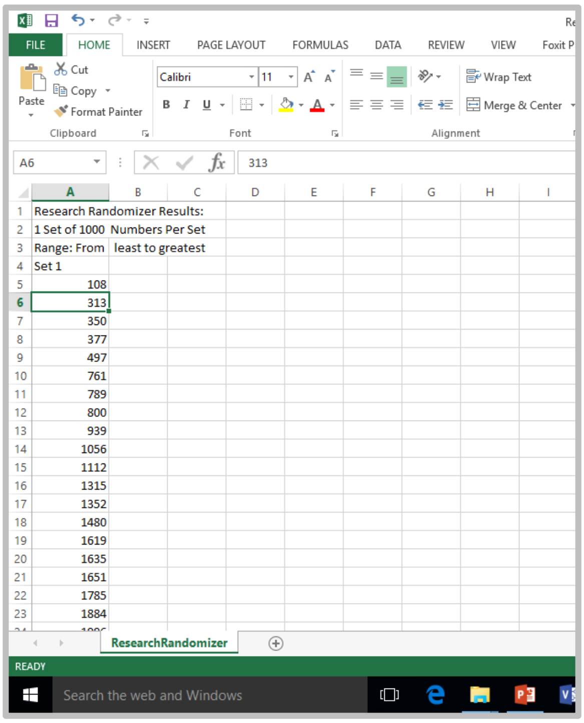


1 Set of 1,000 Unique Numbers
Range: From 1 to 100,000— Sorted from Least to Greatest

Set #1

108, 313, 350, 377, 497, 761, 789, 800, 939, 1056, 1112, 1315, 1352, 1480, 1619, 1635, 1651, 1785, 1884, 1996, 2200, 2232, 2277, 2527, 2795, 2932, 3080, 3087, 3089, 3306, 3619, 4038, 4046, 4092, 4097, 4216, 4570, 4696, 4732, 4745, 4768, 4907, 4918, 5058, 5157, 5392, 5422, 5470, 5761, 5824, 5875, 6153, 6354, 6391, 6421, 6884, 6919, 7083, 7113, 7182, 7245, 7441, 7510, 7578, 7637, 7664, 7696, 7774, 7993, 8095, 8109, 8174, 8347, 8369, 8408, 8420, 8618, 8810, 8881, 8973, 9002, 9006, 9054, 9077, 9142, 9182, 9298, 9384, 9459, 9467, 9476, 9484, 9513, 9571, 9625, 9707, 9908, 10185,

Figure 6: Results pane for the Randomizer



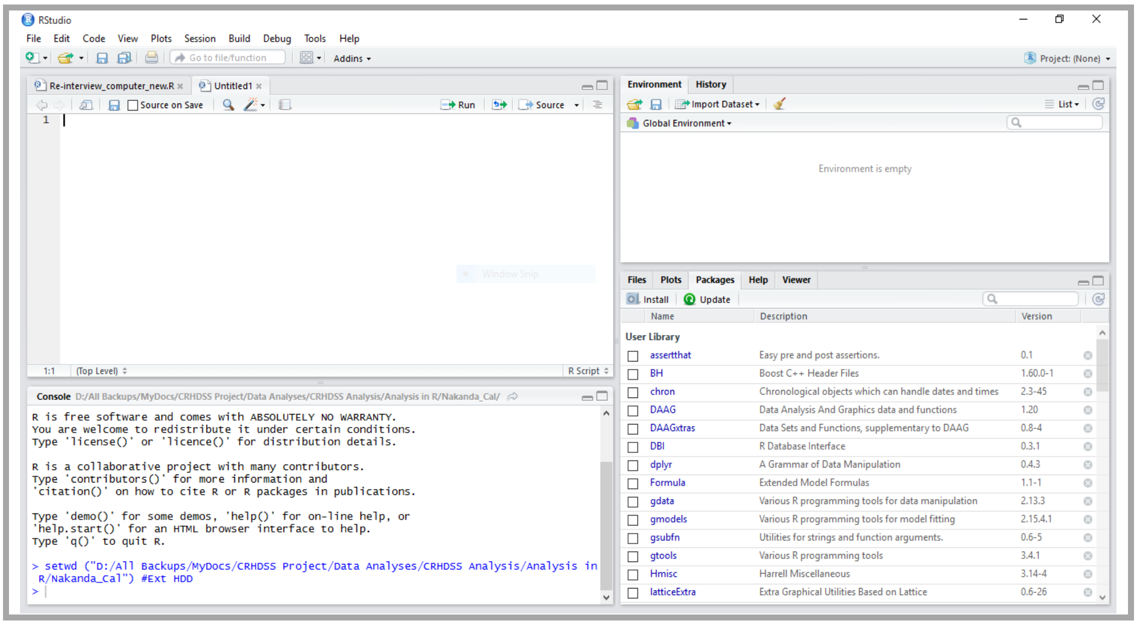
	A	B
1	Research Randomizer Results:	
2	1 Set of 1000	Numbers Per Set
3	Range: From	least to greatest
4	Set 1	
5	108	
6	313	
7	350	
8	377	
9	497	
10	761	
11	789	
12	800	
13	939	
14	1056	
15	1112	
16	1315	
17	1352	
18	1480	
19	1619	
20	1635	
21	1651	
22	1785	
23	1884	

Figure 7: Random numbers exported to Excel

With these inputs, the application generates the random numbers, provided there is active internet connection. Thereafter, the numbers so generated are matched against the record numbers of the data table for the event, which is already in Excel/CSV data formats. This matching is performed using Excel's LOOKUP formula. Each matching record is extracted from the original data file and made available for quality assurance repeat interview in the field. The repeat interview is done with paper forms at the present, and the data is then compared with what was electronically transmitted.

3.2.2 *Solution Option 2*

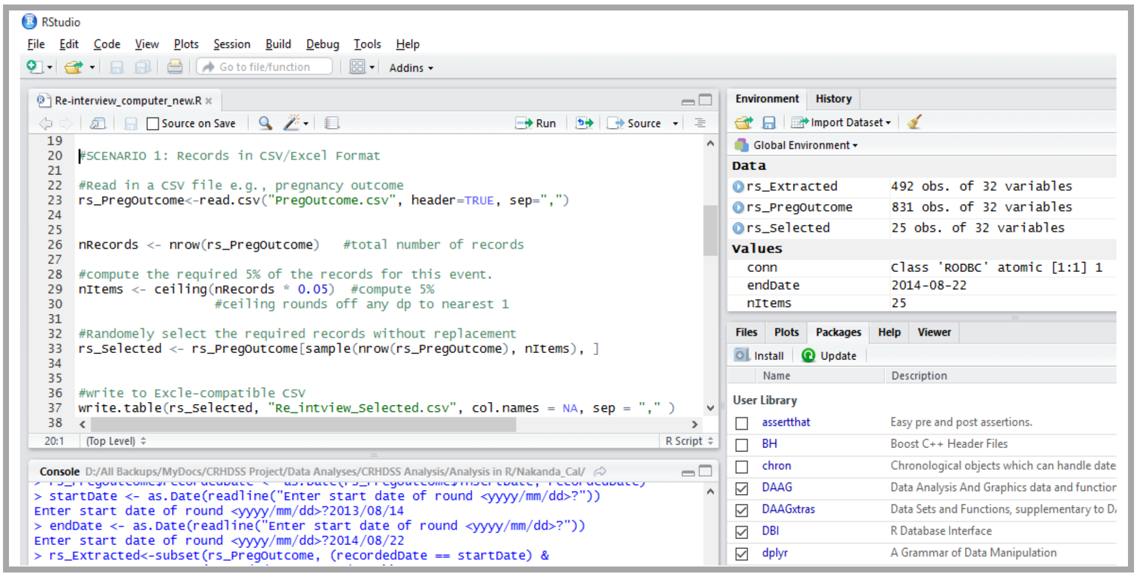
In our second approach, we wrote a software script using the R Statistical Computing software version 3.2.4 (The R Foundation, 2016). This script works in two ways, depending on the format of the data. Firstly, if the data is in Excel/CSV format, the software imports the data into R, computes the 5% of records, extracts the matching records from the result set, and then exports the results back to Excel/CSV formats for the repeat interviews. Figure 8 shows the R Studio programming window, an integrated programming environment (IDE) for R programming; while Figure 9 shows part of the R code for this first scenario.



```
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> setwd ("D:/All Backups/MyDocs/CRHDSS Project/Data Analyses/CRHDSS Analysis/Analysis in
 R/Nakanda_Cal") #Ext HDD
> 
```

	Name	Description	Version
User Library			
□	assertthat	Easy pre and post assertions.	0.1
□	BH	Boost C++ Header Files	1.60.0-1
□	chron	Chronological objects which can handle dates and times	2.3-45
□	DAAG	Data Analysis And Graphics data and functions	1.20
□	DAAGxtras	Data Sets and Functions, supplementary to DAAG	0.8-4
□	DBI	R Database Interface	0.3.1
□	dplyr	A Grammar of Data Manipulation	0.4.3
□	Formula	Extended Model Formulas	1.1-1
□	gdata	Various R programming tools for data manipulation	2.13.3
□	gmodels	Various R programming tools for model fitting	2.15.4.1
□	gsubfn	Utilities for strings and function arguments.	0.6-5
□	gtools	Various R programming tools	3.4.1
□	Hmisc	Harrell Miscellaneous	3.14-4
□	latticeExtra	Extra Graphical Utilities Based on Lattice	0.6-26

Figure 8: The R Studio IDE


RStudio
File Edit Code View Plots Session Build Debug Tools Help
Go to file/function
Addins
Re-interview_computer_new.R
Source on Save
Run
Source
20:1
(Top Level)
R Script
Environment
History
Import Dataset
Global Environment
Files
Plots
Packages
Help
Viewer
Install
Update
Console D:/All Backups/MyDocs/CRHDSS Project/Data Analyses/CRHDSS Analysis/Analysis in R/Nakanda_Cal/

```r

#SCENARIO 1: Records in CSV/Excel Format

#Read in a CSV file e.g., pregnancy outcome
rs_PregOutcome<-read.csv("PregOutcome.csv", header=TRUE, sep=",")


nRecords <- nrow(rs_PregOutcome)   #total number of records

#compute the required 5% of the records for this event.
nItems <- ceiling(nRecords * 0.05)  #compute 5%
                  #ceiling rounds off any dp to nearest 1

#Randomely select the required records without replacement
rs_Selected <- rs_PregOutcome[sample(nrow(rs_PregOutcome), nItems), ]


#write to Excle-compatible CSV
write.table(rs_Selected, "Re_intview_Selected.csv", col.names = NA, sep = "," )
```

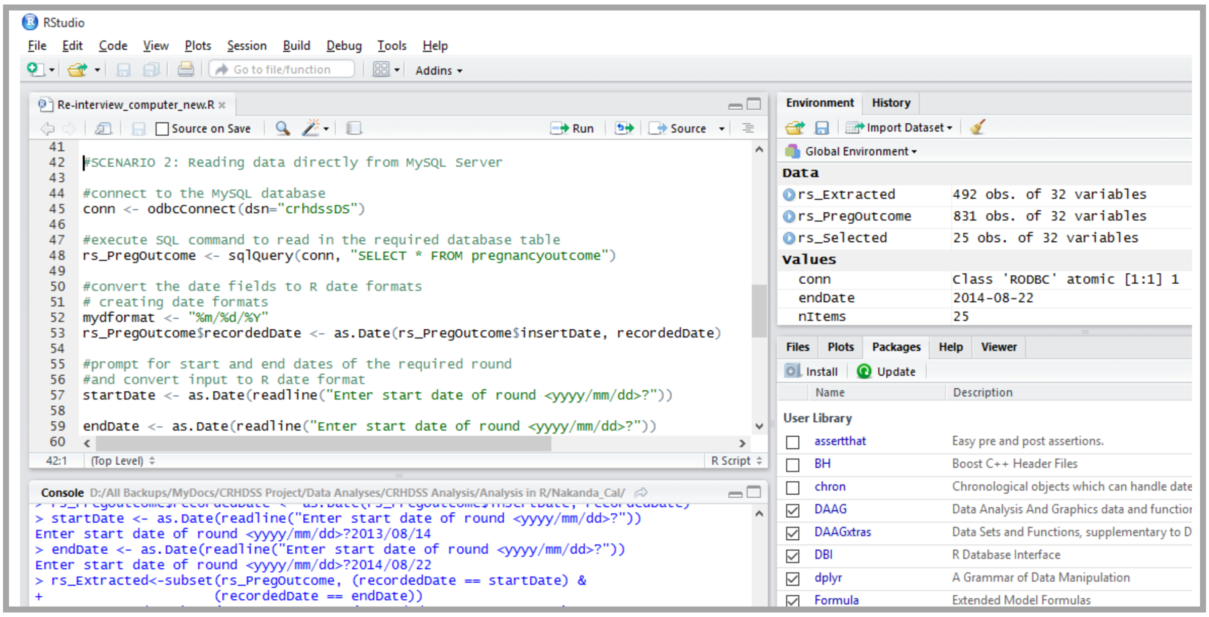
```
> startDate <- as.Date(readline("Enter start date of round <yyyy/mm/dd>?"))
Enter start date of round <yyyy/mm/dd>?2013/08/14
> endDate <- as.Date(readline("Enter start date of round <yyyy/mm/dd>?"))
Enter start date of round <yyyy/mm/dd>?2014/08/22
> rs_Extracted<-subset(rs_PregOutcome, (recordedDate == startDate) &
```

Data	
rs_Extracted	492 obs. of 32 variables
rs_PregOutcome	831 obs. of 32 variables
rs_Selected	25 obs. of 32 variables
Values	
conn	Class 'RODBC' atomic [1:1] 1
endDate	2014-08-22
nItems	25

	Name	Description
User Library		
□	assertthat	Easy pre and post assertions.
□	BH	Boost C++ Header Files
□	chron	Chronological objects which can handle date
☑	DAAG	Data Analysis And Graphics data and functior
☑	DAAGxtras	Data Sets and Functions, supplementary to D.
☑	DBI	R Database Interface
☑	dplyr	A Grammar of Data Manipulation

Figure 9: Section of R code for the scenario where data is in CSV format

The second scenario is a situation where the data is collected and transmitted straight onto a relational database management system (RDBMS), such as MySQL, as in the case of the OpenHDS software (*described earlier*), a section of the R software script will execute. In this case, the script speaks directly to the RDBMS, (*MySQL in this case*) extracting the data directly into R. It will then prompt the user for the start and end dates of the update round in question, to enable the extraction of a subset of the data from the result set. The rest of the process is as described in the first scenario above. Figure 10 shows a section of the R code for the second scenario.



```
#SCENARIO 2: Reading data directly from MySQL Server

#connect to the MySQL database
conn <- odbcConnect(dsn="crhdssDS")

#execute SQL command to read in the required database table
rs_PregOutcome <- sqlQuery(conn, "SELECT * FROM pregnancyoutcome")

#convert the date fields to R date formats
# creating date formats
mydformat <- "%m/%d/%Y"
rs_PregOutcome$recordedDate <- as.Date(rs_PregOutcome$insertDate, recordedDate)

#prompt for start and end dates of the required round
#and convert input to R date format
startDate <- as.Date(readline("Enter start date of round <yyyy/mm/dd>?"))

endDate <- as.Date(readline("Enter start date of round <yyyy/mm/dd>?"))
```

Console D:/All Backups/MyDocs/CRHDSS Project/Data Analyses/CRHDSS Analysis/Analysis in R/Nakanda_Cal/

```
> startDate <- as.Date(readline("Enter start date of round <yyyy/mm/dd>?"))
Enter start date of round <yyyy/mm/dd>?2013/08/14
> endDate <- as.Date(readline("Enter start date of round <yyyy/mm/dd>?"))
Enter start date of round <yyyy/mm/dd>?2014/08/22
> rs_Extracted<-subset(rs_PregOutcome, (recordedDate == startDate) &
+                      (recordedDate == endDate))
```

Data

rs_Extracted	492 obs. of 32 variables
rs_PregOutcome	831 obs. of 32 variables
rs_Selected	25 obs. of 32 variables

Values

conn	Class 'RODBC' atomic [1:1] 1
endDate	2014-08-22
nItems	25

User Library

	Name	Description
□	assertthat	Easy pre and post assertions.
□	BH	Boost C++ Header Files
□	chron	Chronological objects which can handle date
☑	DAAG	Data Analysis And Graphics data and functior
☑	DAAGxtras	Data Sets and Functions, supplementary to D
☑	DBI	R Database Interface
☑	dplyr	A Grammar of Data Manipulation
☑	Formula	Extended Model Formulas

Figure 10: Section of R code for the scenario where data is MySQL server

4. Discussion

The Solution Option 1 proffered in this study is meant for health and demographic surveillance system sites that do not have strong information technology teams; that is, sites where the data team has just basic computing skills, including proficiency in the Office suite. All that is required is an Internet connection to run Randomizer, the free software to help generate random numbers for lookup, and then select the records for repeat interview. However, this solution has a few limitations. Because of the requirement for Internet connectivity, there may be delays in generating the random matching records, in areas where Internet connectivity is poor or near-absent (*or not even available*). Another drawback to this option is the fact that, the user has to manually compute the 5% of total records for the event and then feed this into Randomizer. This can be prone to error.

Solution Option 2, on the other hand, is for sites with moderately proficient computing teams, with some knowledge of programming. The first scenario for this option applies to situations where the data has been collected into any form, and the required update round data extracted into Excel/CSV format. The R code imports this data into an R result set, performs computation of the 5% records, extracts the matching data, and exports (*into Excel/CSV*) the selected records without any manual involvement. The second scenario for this option, is applicable to situations where the data is collected and transmitted directly into a database server such as MySQL. This solution is faster, as the R code speaks directly to the database server, requiring only the start and end dates (*for the update round*) from the user. Unlike the first scenario, no manual extraction of update round data is required. We like to remark here that, Solution Option 2, as presented in this paper, does not require the Internet to work, and so overcomes the challenge in Solution Option 1 that may encounter delays due to poor Internet connectivity.

Notwithstanding the merits in electronic data validation checks in a paperless HDSS, there are challenges. For instance, the web-based Research Randomizer requires active Internet connection to function. The consequence is that, generation of random records can be delayed when there is no access to the Internet. In addition to this, the subset of the total records in the data to be randomly-generated is manually calculated before entering the value as input to the application's required number of random integers to generate. Another limitation to our information technology (IT)-based approach is the fact that, re-interviews are still done with paper forms, instead of mobile devices.

5. Conclusion

Automating data quality checks in health and demographic surveillance systems (HDSS) that are driven by information technology, comes with the advantage of overcoming the delays prevalent in paper-based HDSS processes. For instance, in the manual method, fields in paper forms are scrutinized manually for correctness. This is both time consuming and error-prone. Besides, in a manual system, paper forms are counted manually before selecting 5% of the event forms, a process very prone to high level of bias. For sites with large populations, the process of selecting forms for repeat interviews can take several hours; while the same process will take less than 20 minutes in a paperless system. This is not to claim that the automated solutions proffered in this study are a silver bullet.

Apart from Solution Option 1 requiring Internet connectivity, the requirement of computing proficiency in the Solution Option 2 can be a tall order for HDSS sites that are predominantly paper-based and trying to migrate to paperless. These sites did not require any proficient computing staff to run, before now, and so may not have the right mix of these staff. Nevertheless, these challenges are almost overcome by the increasing information technology (IT) penetration and increasing availability of IT workforce in the low- and middle-income countries (LMICs). The costs of providing Internet and training IT personnel is reducing considerably in these countries. One major limitation to the solutions provided in this study is the fact that, repeat interviews are still performed with paper forms. Future research may therefore attempt to design solutions that can facilitate repeat interviews using mobile devices.

References

Arikpo, I. M., Okoro, A., James, U., Aquaisua, E., Osonwa, K., Ushie, M., Meremikwu, M. (2013). Optimum hardware, software and personnel requirements for a paperless health and demographic surveillance system: a case study of Cross River HDSS, Nigeria. *Information and Knowledge Management,* Vol 3, No 3, pp 77-83.

Byass, P., Berhane, Y., Emmelin, A., Kebede, D., Anderson, T., Hogberg, U., & Wall, S. (2002). The role of demographic surveillance systems (DSS) in assessing the health of communities: an example from rural Ethiopia. *Public Health,* Vol 116, No 3, pp 145-150.

INDEPTH Network. (2006). *INDEPTH Resource Kit for Demographic Surveillance Systems.* Ghana: INDEPTH Network.

INDEPTH Network. (2008, June 03). *Field Quality Control*. Retrieved September 25, 2015, from INDEPTH Resource Kit for Demographic Surveillance Systems: http://www.indepth-network.org/Resource%20Kit/INDEPTH%20DSS%20Resource%20Kit/Fieldqualitycontrol.htm [Retrieved: 25 September 2015]

Kaneko, S., K'opiyo, J., Kiche, I., Wanyua, S., Goto, K., Tanaka, J., Shimada, M. (2012, May). Health and Demographic Surveillance System in the Western and Coastal Areas of Kenya: An Infrastructure for Epidemiologic Studies in Africa. *Journal of Epidemiology,* Vol 22, No 3, pp 276-285.

Laney, C., Kaasa, S., Morris, E., Berkowitz, S., Bernstein, D., & Loftus, E. (2008). The Red Herring technique: a methodological response to the problem of demand characteristics. *Psychological Research,* 72, 362-375. doi:10.1007/s00426-007-0122-6

Madeleine, F., YoonJoung, C., & Sandra, B. (2012). A systematic review of Demographic and Health Surveys: data availability and utilization for research. *Bulletin of the World Health Organization,* 90, 604-612.

Mirth Corporation. (2015). *Mirth Connect*. https://www.mirth.com/Products-and-Services/Mirth-Connect?utm_source=google&utm_medium=cpc&utm_term=Download&utm_content=Mirth-Connect&utm_campaign=ADW_15_Mirth_Connect&gclid=CjwKEAiA3_axBRD5qKDc__XdqQ0SJAC6lecA7Z0uqiRWWuIxXzCI5lN0w630T1pnwVr88S2Z8wGnOxoC [Retrieved: 7 November 2015]

Open Data Kit. (2015). *Open Data Kit*. https://opendatakit.org/use/aggregate/ [Retrieved: 7 November 2015]

Pasquale, A., Mitra, R., & Maire, N. (n.d.). *OpenHDS manual.*

Social Psychology Network. (2015, September 25). *Research Randomizer*. https://www.randomizer.org/ [Retrieved:]

The R Foundation. (2016). *R: The R Project for Statistical Computing*, 3.2.2. Retrieved March 28, 2016, from The R Project for Statistical Computing: https://www.r-project.org/ [Retrieved: 28 March 2016]

Ye, Y., Wamukoya, M., Ezeh, A., Emina, J., & Sankoh, O. (2012). Health and demographic surveillance systems: a step towards full civil registration and vital statistics system in sub-Sahara Africa? *BMC Public Health*, Vol 12, p 741.

Table of Contents

Vol 21, No 2, 2017

UWS UNIVERSITY OF THE WEST *of* SCOTLAND

Printed by Printing Services, University of the West of Scotland
University of the West of Scotland is a registered Scottish charity. Charity number SC002520

Computing and Information Systems is published normally three times per year, in February, May, and October, by the University of the West of Scotland.

From the next issue the editorial address is Dr Abel Usoro, School of Computing, University of Paisley PA1 2BE; tel (+44) 141 848 3959; fax (+44) 141 848 3542, e-mail: cis@uws.ac.uk or abel.usoro@uws.ac.uk.

Editorial Policy

Computing and Information Systems offers an opportunity for the development of novel approaches, and the reinterpretation and further development of traditional methodologies taking into account the rate of change in computing technology, and its usage and impact in organisations.

Computing and Information Systems welcomes articles and short communications in a range of disciplines:

- Organisational Information Systems
- Computational Intelligence
- E-Business
- Knowledge and Information Management
- Interactive and Strategic Systems
- Engineering
- E-Learning
- Cloud Computing
- Computing Science

The website for Computing and Information Systems is http://cis.uws.ac.uk