ISSN No:-2456-2165

Implementation of a Community Emergency Security Alert System

Victor O. Matthews
Department of Electrical and Information Engineering
Covenant University
Ota, Ogun State, Nigeria

Uzairue Stanley Idiake
Department of Electrical and Information Engineering
Covenant University
Ota, Ogun State, Nigeria

Etinosa Noma-Osaghae
Department of Electrical and Information Engineering
Covenant University
Ota, Ogun State, Nigeria

Morgan Kubiat Enefiok
Department of Electrical and Information Engineering
Covenant University
Ota, Ogun State, Nigeria

Praise Jude Ogukah
Department of Electrical and Information Engineering
Covenant University
Ota, Ogun State, Nigeria

Abstract:- Emergency alert and response is carried out in different ways around the world. Governments, corporate bodies and individuals take emergencies very seriously and continue to develop ingenious ways of responding to emergencies very swiftly. Most Urban areas have well-developed emergency response systems but this is not true of rural and sub-urban settlements. Security risk keeps increasing by the day due to rapid population growth. This is particularly true at the grassroots or community level. This paper proposes a very effective and economical way of alerting a community to all kinds of security emergencies. It incorporates the use of a mobile application that was codenamed "CEMAS" (Community Emergency Alarm System). The mobile application with a "Panic button" on it provides all inhabitants of the community with the means of triggering two SMS-activated central alarms. The first alarm is located at the community center and the second at the community police station. The central alarm system is activated by pressing the "Panic Button" whenever there is a security threat. The designed and implemented system worked satisfactorily well.

Keywords:- *Alarm, anti-intruder, motion sensing, images, cloud, server, application programme interface, security, home.*

I. INTRODUCTION

Security is a source of primary concern for individuals, organizations and governments [1]. Nations have adopted various measures and methods to ensure the safety and survival of her citizens [2]. Urban areas have well-developed emergency response systems but very few rural, sub-urban and community-like settlements have the luxury of even the simplest emergency response system [3]. The most alarming part of this glaring gap is the fact that in most developed and developing countries, rural, sub-urban and community-like populations are growing at an exponential rate [4]. This has made security risk for community dwellers higher than it used to be in the past [5].

The time it takes to respond to security threats usually determines the gravity of the eventual lot of those who are directly and indirectly affected [6]. However, the response of the police to just about any security threat situation is also very dependent on how fast information about security threats gets to the police [7]. The responsibility of reporting emergencies such as security threats lies with the inhabitants. Thus, an ineffective security alert system will increase response time and the severity of the aftermath of such emergencies [8]. This paper addresses the challenge reporting security threats at the grassroots or community level. The system functions as a panic alert system [9]. It uses a mobile application that enables every inhabitant of the community to send security alerts as a broadcast via a "Panic button" [10]. The broadcast will also trigger the alarm installed at the community centre and at the community police station. Members of the community can respond to security alerts or escalate it [11].

This paper proposes a collaborative security alert system that can facilitate the reporting of security emergencies to all dwellers in a community and to law enforcement agencies. It is a mobile-based security alert system for reporting security emergencies the instant they occur. This ensures that response time is reduced to a minimum. The CEMAS (Community Emergency Alert System) software application (containing the

"Panic button") was installed on the mobile devices of community members. The mobile application was developed using a cross-platform mobile framework (Ionic Framework). The Ionic framework is a software development kit for hybrid mobile application development based on the JavaScript language and uses several web technologies including CSS, HTML and SASS to develop mobile applications that can be deployed across Android, iOS and Windows platforms by leveraging on Cordova – a mobile application development framework that wraps web files into a native container accessible on several mobile platforms [12]. The mobile application implements geolocation to detect the victim's precise location which will be included in the panic broadcast sent to all members of the community. The proposed alarm system has a SMS module that can broadcast panic messages offline to all members of the community when a security emergency occurs [13]. The system can also collect relevant up-to-date contact information concerning all members of the community. Two SMS-triggered alarms deployed at the community centre and at the community police station serve as a form of verbose alert each time a panic alert is broadcast [14].

II. LITERATURE REVIEW

One of the most crucial aspects of emergency management is emergency response. Information and Communication Technology (ICT) have been used extensively and innovatively to manage emergencies[15]. The integration of ICT solutions into real-life emergency scenarios in the health sector has caused the birth of a now popular concept, mHealth - which has been defined as "healthcare to anyone, anytime and anywhere by removing temporal and locational constraints while increasing both the coverage and the quality of healthcare" [16]. mHealth is a concept that has been actualized via mobile applications that depend on user behavior, geographic location and online community characteristics to offer medical emergency support and significantly reduce medical emergency response time [17].

Amongst the vast array of ICT solutions available to combat security disparities in the world today, the short message service (SMS) also known as text-message – "a service component that uses standardized communication protocols to enable mobile devices to exchange short text messages" [18] - has been one of the commonly used. This is because it is widely available for virtually every type of mobile device and 95% of the world population currently live in areas with cellular network coverage [19]. Palmieri et al [20] proposed a hybrid cloud-based architecture for managing computing and storage resources needed to control activities during emergency situations. The system also uses a novel positioning approach which utilizes signal data from physical landmarks placed by first responders in an emergency attack location as well as data from motion sensors. Their system leveraged the practically unlimited computing and storage resources provided by cloud architectures. Li et al [15] proposed a community-based collaborative information system for emergency management. The system had a focus on effective emergency management. The system created a distributed community-based virtual database based on a P2P (Peer-to-Peer) architecture which links local resource database of suppliers that provide information to foster multi-criteria decision making, thereby, enabling effective and timely emergency response. The P2P architecture used to manage the distributed datasets of the target community will allow a dataset to easily join and leave the network as well as allow for autonomous maintenance of each individual organization's dataset. The system was implemented as a social networking site, providing end users with access to information, good situational awareness and also a possibility of sharing such information with emergency partners at all levels.

Zhao and Liu [21] developed a decision support tool for optimizing urban emergency rescue facility locations to improve humanitarian logistics movement. The support tool integrates a number of loosely-coupled components into a uniform .NET application. These components include: a desktop geospatial database for storing geospatial data which also gives access to the stored datasets via an Application Programming Interface (API); a decision optimization model and NSGA-II algorithm which are encapsulated as a software component according to the Component Objective Model (COM) standard; a series of open-source Geographical Information System (GIS) APIs and a statically analysis module that is developed through third party data analysis applications. The developed software functioned to aid the optimal selection of emergency rescue facility locations in large-scale urban areas in order to foster public safety. [17] developed a simulation modeler for comparing Emergency Medical Services (EMS) with smartphone-based samaritan response. Their software compares the potential smartphone-initiated member response to traditional EMS response using certain parameters inputted into the application for specific health conditions in a given geographical region. They conducted experiments to establish adoption levels for certain Emergency Medical Services (EMS) as against smartphone-based samaritan response using various factors. This helped the researchers to determine the effectiveness of samaritan-based emergency response communities. The authors also emphasized the efficacy of deploying mHealth applications for emergency response. [22] carried out a research to determine the various factors affecting end user acceptance of Emergency Operation Centre Information Systems (EOCISs). Based on the model they developed, they were able to determine that social impact has a positive influence on technology uptake. They also determined that factors such as age, sex, and user experience greatly affected the adoption of new technologies. The effect of these factors on adoption of new technologies varied depending on the profile and behavioral differences of each user.

Jain, et al. [23] proposed the Punya framework that shortens the development time of android applications but still supports the communication and sensor features required to collect data in crisis scenarios. Its improved sensor abilities

and data collection components enable organizations build applications within a short amount of time that can collect data and visualize results. [24] proposed the use of a discrete optimization model on social media for the dissemination of emergency messages. The optimization model was aimed at helping organizations achieve optimal dissemination of information to targeted users. The model was implemented on a small scale twitter network with a hundred nodes and it proved successful.

Du and Zhu [25] proposed a public safety emergency management early-warning system based on IoT. It was realized that the system was capable of omni-directional monitoring as well as adequate predictions based on the data that it collects. This helped communities respond to emergencies faster and more accurately. The exact location of the emergency can be pin-pointed with IoT devices acting as sensors. [26] carried out a research on the social acceptance of location-based mobile government based services for emergency management. In their research they were able to establish that people's attitude to the application was highly based on its perceived usefulness and that the only negative impact on the system came as a result of people's apprehension towards the collection of personal data by the application.

Gomez, et al. [27] proposed an urban security system based on quadrants. This system was designed and developed to improve the response time of the police force to criminal activities. The urban area was divided into quadrants and each member of a quadrant had the system's application installed on their mobile device. It was tested out and found that the application improved the police respond time by 60%.

III. METHOD AND MATERIAL

CEMAS is a system that is made up of two (2) primary architectures. These architectures are the software and hardware architecture. The hardware architecture of CEMAS encompasses the design and specifications of the SMS-triggered smart alarm that will be deployed in two central locations in the community. Also, the software architecture details the design, processes and modules of the hybrid mobile application that the community members (client) will interact with in order to perform specific tasks.

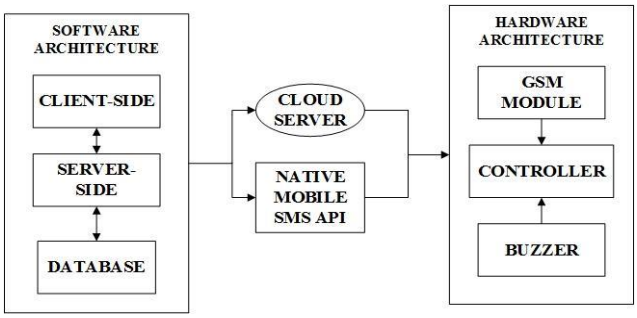

Fig 1:- Overall System Architecture

A. *Hardware Architecture*

The design specifications of the hardware architecture show the various components needed to move from the design phase to the development phase.

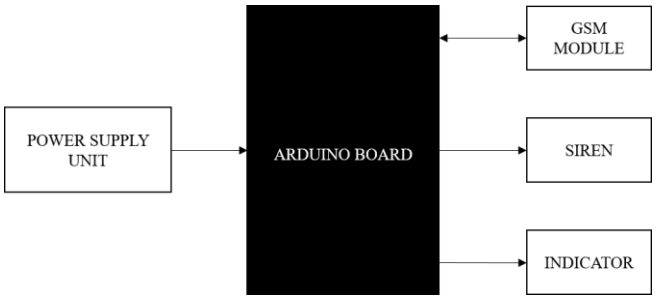

Fig 2:- The Hardware Architecture

B. *SMS Module*

This module handles the reception of the panic trigger SMS and serves this input into the microcontroller (Arduino UNO R3) which acts as the controller that activates the alarm module. The devices involved in this module are:

- Microcontroller (Arduino UNO R3)
- GSM module

C. *Microcontroller (Arduino Uno R3)*

The Arduino Uno can be programmed with the (Arduino Software ,IDE-Integrated development environment). The ATmega328 on the Arduino Uno comes pre-programmed with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. The program was written in C programming language.

D. *GSM Module*

The selected GSM module for this project is SIM 900 shield. This is a GSM/GPRS-compatible Quad-band cell phone, which works on a frequency of 850/900/1800/1900MHz. It can be used to access the internet as well as a GSM network. The module requires a supply of continuous energy of (between 3.4 and 4.5 V) [23] of which an AC adapter was bought to connect to a power outlet to energize the module. With the aid of the GSM module it becomes possible to make calls, receive calls, send text messages and receive text messages.

E. *Alarm Module*

This module is employed to alert the community of any distress call by triggering a siren/alarm on receipt of a signal from the microcontroller. The components involved in this module are.

- Relay
- Siren

F. *Software Architecture*

The software architecture of CEMAS was the Client-Server architecture. The client was a hybrid mobile application

ISSN No:-2456-2165

written in Ionic (a JavaScript hybrid mobile development framework) that was used to perform variety of tasks including initiating the panic broadcast. The client is the front-end while the server, a web server written in Node.JS (a JavaScript runtime for building web services and applications) is referred to as the back end.

G. Use Case Diagram

The use case diagram of CEMAS is as shown in figure 2. It clearly outlines all the actors in the CEMAS system. It also shows the activities that are performed by each of these individuals (actors) as well as defining relationships and dependencies between them.

H. Client Side _ Geolocation Component

This component consists of the geo-location component which utilizes the native mobile device's geo-location capabilities to detect the victim's current location in terms of latitude and longitude. However, in order to make the interface user-friendly and the current location data more readable by just about any user, the component employs a geocoding service. Geocoding is simply the process of converting a human-readable address of a given location into a set of coordinates (i.e. latitude and longitude). In the same vein, Reverse Geocoding is the process of converting a set of coordinates into a human-readable formatted address. Hence, the victim's current location, obtained from the geolocation module in the victim's mobile device, is reverse-geocoded into a readable address to provide a better description of the user's location to other members of the community.

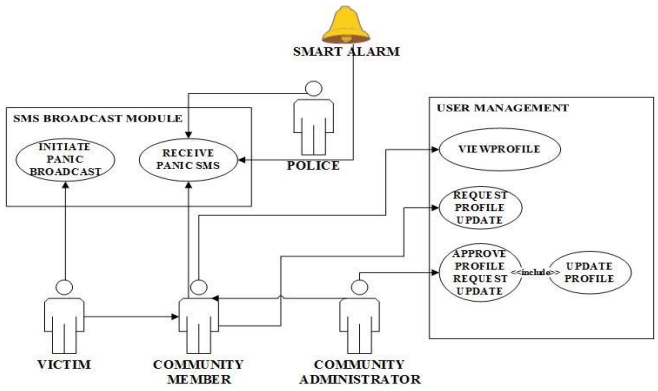

Fig 3:- The Use Case Diagram

I. The Panic Trigger Component

The panic trigger is implemented as a button on the mobile application that can be touched (pressed down) by the victim. In order to avoid accidental triggers of the panic button, the panic trigger does not respond when it is quickly tapped but sends the panic broadcast when it is pressed and held for at least 400ms.

J. The SMS Broadcast Component

This component compiles the panic message to be sent via SMS, the current location that has been set by the geolocation component and the phone numbers of everyone in the community fetched from the server using the community ID of the victim. The numbers are then queued. The component harnesses the native SMS application that resides on the victim's mobile phone to send out the panic information to each contact fetched from the database. This component finally returns a feedback to confirm the delivery of each of the panic messages.

K. The Server Side

This software architecture includes all the business logic as well as data structure and storage. It also includes description of the server's interaction with an external cloud storage and SMS server. Generally, the server-side utilizes a Non-Relational database - MongoDB. This database mainly uses collections to group data as opposed to the use of tables in Relational Databases such as MySQL Database. Also, data are stored as documents and can be of different types. Mongoose, a tool which provides an Object Data Model (ODM) or Object Relational Model (ORM) for interacting with the Mongo Database is also employed in order to interact with data using Models and Schema – a template in which documents (data) in a collection (table) will be structured as well as functions that will validate the type of data that is to be stored in the database. Mongoose provides a more flexible way of modeling data interaction between the application (client) and the database as it maps the data as JavaScript Objects (i.e. data can be served in JavaScript Object Notation (JSON) format).

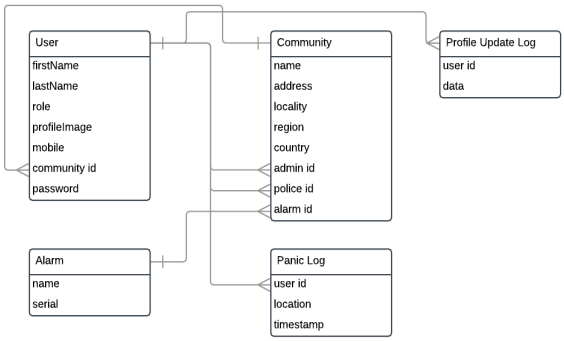

Fig 4:- Database Design_Model RelationshipThe Use Case Diagram

L. User Management Service

This component describes all functions and related methods that perform basic management operations on the Users Model. The basic actions carried out by this component include:

- Create – This action refers to the instantiation of a model which in turn creates a new document using the instantiated values, structured and validated by the schema of that object.
- Update – This is the act of modifying the value or data contained in the instance of a model or a database document.

- Delete – This act simply refers to the removal or destruction of an instance of a model or document in the database.
- Read – This action simply involves calling or invoking an existing instance of a model or document in the database in order to access data it contains or its properties.

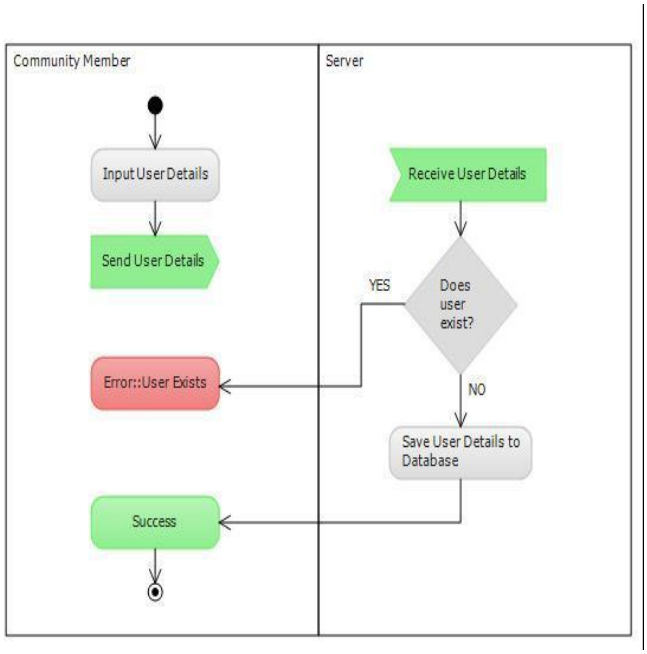

Fig 5:- Flowchart for Creating User Activity

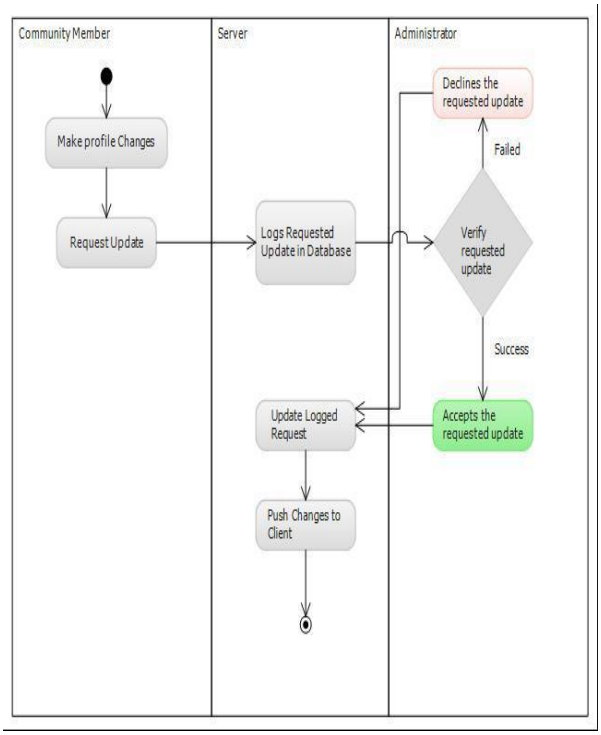

Fig 6:- Flowchart for Updating Users' Information

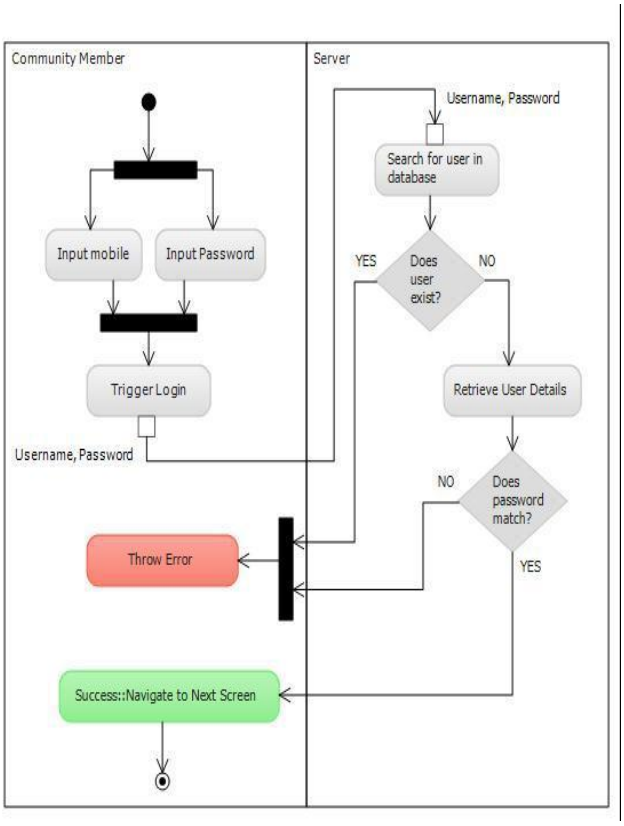

Fig 7:- Flowchart Illustrating User Login Activity

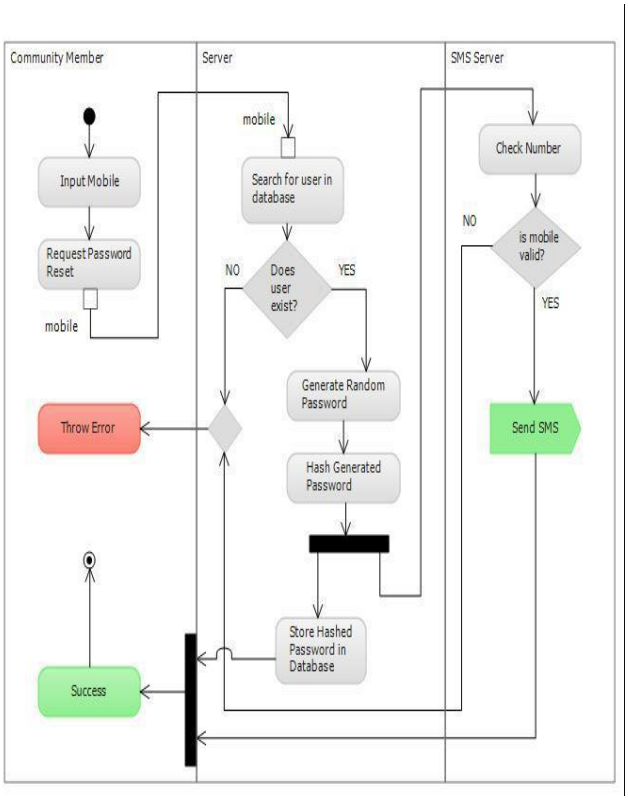

Fig 8:- Flowchart Illustrating Password Reset Activity

ISSN No:-2456-2165

IV. RESULTS AND DISCUSSION

A. *Retrieving List of Community Members*

Table 1 illustrates the process used to determine how well the designed system was able to retrieve the lists of all the members of the community stored in the database.

	FORMAT
Endpoint	/community/{cid}/{uid}
Method	GET
URI Parameters	Community id (cid), User id (uid)
Body Parameters	None
Response	An array of objects of all users in the specified community

Table 1. Format to Retrieve Lists of Community Members

Figure 10 and 11 shows a sample request and response for the query for the list of members in the community

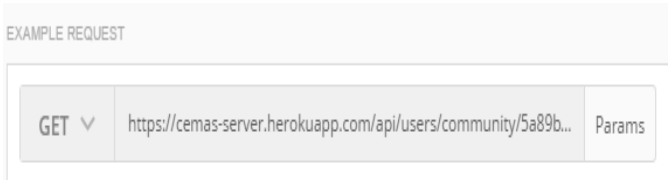

Fig 9:- A Test Request to Retrieve the List of Community members

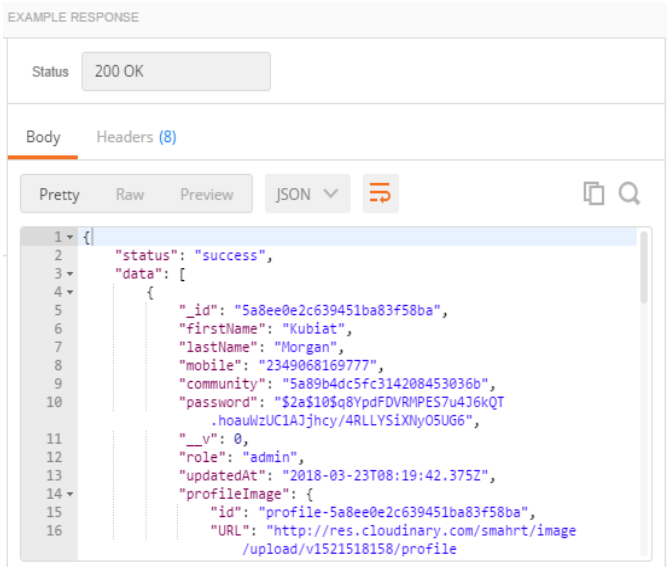

Fig 10:- The Response received for List of Community Members Test

B. *Password Update Service False*

Table 2 illustrates the process used to determine how well the designed system was able to update the password of registered members of the community.

Figure 12 and 13 shows a sample request and response for the query for the list of members in the community.

	FORMAT
Endpoint	/{uid}/password/update
Method	POST
URI Parameters	User id (uid)
Body Parameters	newPassword, oldPassword
Response	An array of objects of all users in the specified community

Table 2. Format for Updating Password

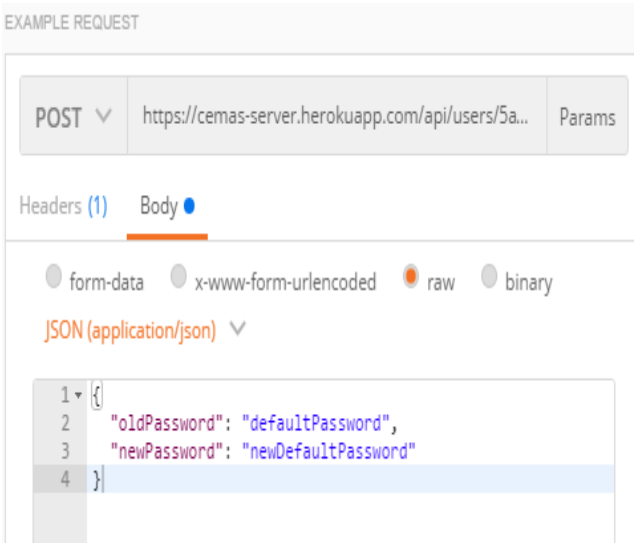

Fig 11:- A Test Request for Password Update Service

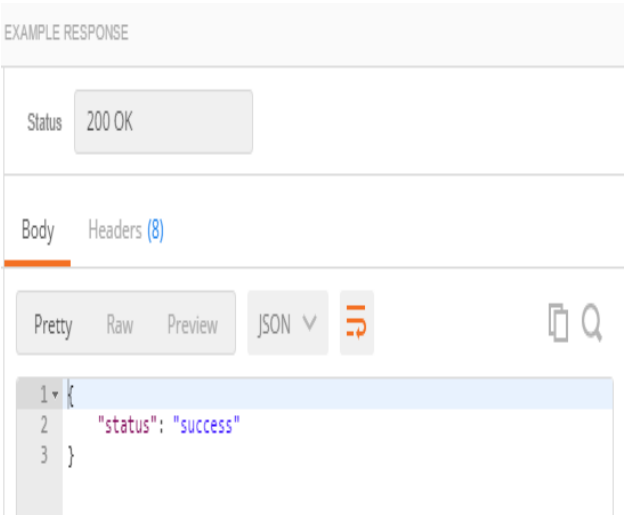

Fig 12:- Response from the Password Update Service

C. *Profile Update Service*

Table 3 illustrates the process used to determine how well the designed system was able to update the profile of registered members of the community.

	FORMAT
Endpoint	/{uid}/update/{action}
Method	POST
URI Parameters	User id (uid); Action (0-request update, 1-approve update)
Body Parameters	Community ID, Update Data (action=0), Log id (action =1)
Response	An array of objects of all users in the specified community

Table 3. Format for Updating Profiles

Figure 12 and 13 shows a sample request and response for the query for the list of members in the community.

Fig 13:- A Test Request Sent to Profile Update Request Service

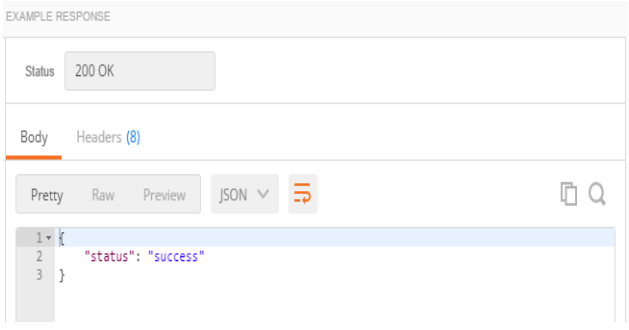

Fig 14:- Response Received from the Profile Update Approval Service Test

D. System Tests

The system was integrated by enrolling the mobile number assigned to the alarm hardware into the database and specifying authorized users of the system. This enabled the server to push a code via SMS to the alarm in order to trigger it in the event of a security emergency.

A case study was set up in a test community and the time taken for the panic SMS to be delivered as well as the SMS command to trigger the alarm was recorded and the results of the test were as follows:

Community Name: Covenant University
Average Alarm trigger time: 4.28s
Average SMS delivery time: 3.88s

This result is got from Table IV.

No. of Iterations	Alarm Trigger Time (s)	SMS Delivery Time (s)
1	6.78	4.99
2	2.33	6.13
3	4.81	2.11
4	2.47	3.81
5	3.82	6.99
6	6.22	1.98
7	3.80	1.17

Table 4. Showing SMS Delivery Times for Seven Panic Broadcasts.

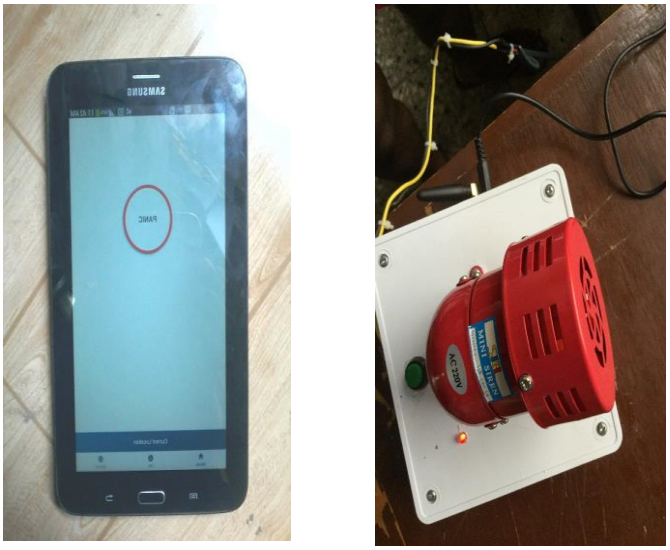

Fig 15:- An android Tablet Running the CEMAS Mobile Application and the Complete packaging of the SMS-triggered Alarm

V. CONCLUSION

A community security emergency alert system was developed and the processes and features were explained in this paper. The application which was built to serve this purpose was named CEMAS which is short for Community Emergency Alert System. The hardware developed was an SMS-triggered alarm system which activates on receipt of an SMS indicating a security threat to any of the community

members. The system provides an easy to use and efficient Mobile User Interface (UI) for reporting emergency threat situations. The proposed system improves emergency response time by reducing security emergency report time in rural, sub-urban and grassroots community dwellers. This can go a long way to mitigating bulk of the calamities experienced as a result of unreported or delayed reports of emergency threat situations prevalent in grassroots communities across the world.

VI. FURTHER WORK

The SMS-triggered alarm could be equipped with a display that would monitor the exact location of the victim. The SMS alarm hardware could be developed into portable sizes and installed in police patrol cars and not just central stations in the community. The application was deployed on the iOS and Android platforms which are easily the most popular in the world today, but there is a percentage, small as it may be, that adopt platforms like the Windows mobile, Blackberry OS etc. The application can be made to cover all platforms.

REFERENCES

[1]. K. Okokpujie, E. Noma-Osaghae, S. John, and P. C. Jumbo, "Automatic home appliance switching using speech recognition software and embedded system," in Computing Networking and Informatics (ICCNI), 2017 International Conference on, 2017, pp. 1-4.

[2]. C. Atuegwu, K. O. Okokpujie, and E. Noma-Osaghae, "A Bimodal Biometric Student Attendance System," 2017.

[3]. K. Okokpujie, N.-O. Etinosa, S. John, and E. Joy, "Comparative Analysis of Fingerprint Preprocessing Algorithms for Electronic Voting Processes," in International Conference on Information Theoretic Security, 2017, pp. 212-219.

[4]. K. O. Okokpujie, N.-O. Etinosa, O. J. Okesola, J. N. Samuel, and O. Robert, "Design and Implementation of a Student Attendance System Using Iris Biometric Recognition," in Computational Science and Computational Intelligence (CSCI), 2017, Las Vegas, USA, 2017.

[5]. N.-O. Etinosa, C. Okereke, O. Robert, O. J. Okesola, and K. O. Okokpujie, "Design and Implementation of an Iris Biometric Door Access Control System," in Computational Science and Computational Intelligence (CSCI), 2017, Las Vegas, USA, 2017.

[6]. K. Okokpujie, E. Noma-Osaghae, S. John, and R. Oputa, "Development of a facial recognition system with email identification message relay mechanism," in Computing Networking and Informatics (ICCNI), 2017 International Conference on, 2017, pp. 1-6.

[7]. C. Atuegwu, S. Daramola, K. O. Okokpujie, and E. Noma-Osaghae, "Development of an Improved Fingerprint Feature Extraction Algorithm for Personal Verification," International Journal of Applied Engineering Research, vol. 13, pp. 6608-6612, 2018.

[8]. K. O. Okokpujie, E. Noma-Osaghae, G. Kalu-Anyah, and I. P. Okokpujie, "A Face Recognition Attendance System with GSM Notification," 2017.

[9]. K. Okokpujie, E. Noma-Osaghae, S. John, and A. Ajulibe, "An Improved Iris Segmentation Technique Using Circular Hough Transform," in International Conference on Information Theoretic Security, 2017, pp. 203-211.

[10]. K. O. Okokpujie, A. Orimogunje, E. Noma-Osaghae, and O. Alashiri, "An Intelligent Online Diagnostic System With Epidemic Alert," An Intelligent Online Diagnostic System With Epidemic Alert, vol. 2, 2017.

[11]. K. O. Okokpujie, E. C. Chukwu, E. Noma-Osaghae, and I. P. Okokpujie, "Novel Active Queue Management Scheme for Routers in Wireless Networks," International Journal on Communications Antenna and Propagation (I. Re. CAP), vol. 8, pp. 53-61, 2018

[12]. D. Caldwell and R. E. Williams, "Seeking Security in an Insecure World," in Seeking Security in an Insecure World, ed, 2006, p. 241.

[13]. F. Vanderschueren. (2013, Febuary 2018). The Evolution and Challenges of Security within Cities. Available: https://unchronicle.un.org/article/evolution-and-challenges-security-within-cities

[14]. N. Adegoke, "THE NIGERIA POLICE AND THE CHALLENGES OF SECURITY IN NIGERIA," Review of Public Administration and Management, vol. 3, December 2014 2014.

[15]. J. Li, Q. Li, C. Liu, S. Ullah Khan, and N. Ghani, "Community-based collaborative information system for emergency management," Computers & Operations Research, vol. 42, pp. 116-124, 2// 2014.

[16]. U. Varshney. (2009, January, 2018). Pervasive Healthcare Computing.

[17]. M. Khalemsky and D. G. Schwartz, "Emergency Response Community Effectiveness: A simulation modeler for comparing Emergency Medical Services with smartphone-based Samaritan response," Decision Support Systems, vol. 102, pp. 57-68, 10// 2017.

[18]. Drj. (2001, January). Short Message Service (SMS) (3446 ed.). Available: https://en.wikipedia.org/wiki/SMS

[19]. S. J. Iribarren, W. Brown Iii, R. Giguere, P. Stone, R. Schnall, N. Staggers, et al., "Scoping review and evaluation of SMS/text messaging platforms for mHealth projects or clinical interventions," International Journal of Medical Informatics, vol. 101, pp. 28-40, 5// 2017.

[20]. F. Palmieri, M. Ficco, S. Pardi, and A. Castiglione, "A cloud-based architecture for emergency management and first responders localization in smart city environments," 2016.

[21]. M. Zhao and X. Liu, "Development of decision support tool for optimizing urban emergency rescue facility locations to improve humanitarian logistics

management," Safety Science, vol. 102, pp. 110-117, 2// 2018.

[22]. R. Prasanna and T. J. Huggins, "Factors affecting the acceptance of information systems supporting emergency operations centres," Computers in Human Behavior, vol. 57, pp. 168-181, 4// 2016.

[23]. A. Jain, J. Adebayo, E. de Leon, W. Li, L. Kagal, P. Meier, et al., "Mobile Application Development for Crisis Data," Procedia Engineering, vol. 107, pp. 255-262, // 2015.

[24]. X. Ma and J. Yates, "Multi-network multi-message social media message dissemination problem for emergency communication," Computers & Industrial Engineering, vol. 113, pp. 256-268, 11// 2017.

[25]. C. Du and S. Zhu, "Research on Urban Public Safety Emergency Management Early Warning System based on Technologies for the Internet of Things," Procedia Engineering, vol. 45, pp. 748-754, // 2012.

[26]. A. Aloudat, K. Michael, X. Chen, and M. M. Al-Debei, "Social acceptance of location-based mobile government services for emergency management," Telematics and Informatics, vol. 31, pp. 153-171, 2// 2014.

[27]. J. Gomez, H. Velssy, and L. Cobo, "Urban Security System Based on Quadrants," 2015.