**FRONT PAGE**

# CERTIFICATION

This is to certify that this project was carried out under my supervision by OLUWANIYI VICTORIA OLUWATOYIN with matriculation number 2016070501300 in the Department of Computer Studies, The Polytechnic, Ibadan.

**MRS. Y.O OGUNS DATE**

**(SUPERVISOR)**

**MR. B.O FADIORA DATE**

**(H.O.D)**

# DEDICATION

This project work is dedicated to God Almighty who has been my source of life and wisdom, without whom I am nothing.

# ACKNOWLEDGEMENT

My profound gratitude goes to Almighty God for His infinite mercy, blessings wisdom, knowledge and loving kindness that He bestowed upon me.

I greatly express my gratitude to my project supervisor, MR JOHN AWOYEMI and the Head of Department, Computer Science department whose advice and courage really made my project work a successful one.

A special thanks to my husband, my heart rob, MR. OLAWUYI SEGUN for his support through thick and thin, his care and advice. Thanks you for not giving up on me.

Special thanks goes to my parents, MR ABAYOMI AND MRS FOLASADE OLOWU I am really proud of them for their unrented effort, guidance and counselling.

My appreciation also goes to my mentor; Pastor JOSHUA ADEMOLA ADENEYE for fatherly advice and prayers all the time and MR ADEFUWA OLUWAPONMILE for is mentorship and financial support.

Finally, I do not forget the contributions of my course mates, MUKAILA MONSUR KOLAWOLE and colleagues of mine, just to mention but a few. I thank the rest of my friends and well-wishers who have in one way or the other contributed to my welfare. I pray that the Almighty God will reward each and every one of you abundantly.

# ABSTRACT

This study presents the development of a Location-Based System for Tracking Hospitals in the case of Medical Emergencies, focusing on the context of south-western Nigeria. In an era dominated by technology, the need for quick and efficient access to medical services during emergencies is paramount. The study identifies the challenges individuals face in locating medical centers, particularly during life-threatening situations. By introducing Location-Based Services (LBS), Geographic Information Systems (GIS), and various technological trends, the research formulates a solution. The system's architecture, functional and non-functional requirements, and the use of Unified Modeling Language (UML) diagrams are explored in detail. This system offers a three-tiered user structure, providing system administrators, users, and emergency seekers with defined roles and responsibilities. It promises to streamline access to medical assistance, reduce response times, and ultimately save lives. This innovation represents a critical intersection of technology and healthcare, offering a lifeline to those in need during medical emergencies.

Table of Contents

[CERTIFICATION i](#_Toc149170589)

[DEDICATION ii](#_Toc149170590)

[ACKNOWLEDGEMENT iii](#_Toc149170591)

[ABSTRACT iv](#_Toc149170592)

[CHAPTER ONE 1](#_Toc149170593)

[INTRODUCTION 1](#_Toc149170594)

[1.1 Background to the Study 1](#_Toc149170595)

[1.2 Statement of the Problem 3](#_Toc149170596)

[1.3 Aim and Objectives of the Study 4](#_Toc149170597)

[1.4 Methodology of the Study 4](#_Toc149170598)

[1.5 Justification of the Study 5](#_Toc149170599)

[1.6 Scope and Limitations of the Study 5](#_Toc149170600)

[1.7 Definition of Terms 5](#_Toc149170601)

[CHAPTER TWO 7](#_Toc149170602)

[LITERATURE REVIEW 7](#_Toc149170603)

[2.1 Medical Emergency Response 7](#_Toc149170604)

[2.2 Information Systems 8](#_Toc149170605)

[2.2.1 Types of information systems 9](#_Toc149170606)

[2.2.2 Health information systems 13](#_Toc149170607)

[2.3 System Development Life Cycle 14](#_Toc149170608)

[2.3.1 Information system development 15](#_Toc149170609)

[2.3.2 Modeling techniques – Unified modeling language (UML) 17](#_Toc149170610)

[2.4 Geographic Information Systems (GIS) 20](#_Toc149170611)

[2.4.1 Geographic object 20](#_Toc149170612)

[2.4.2 Data models 21](#_Toc149170613)

[2.5 Location-Based Services (LBS) Applications 25](#_Toc149170614)

[2.5.1 LBS for emergency services 27](#_Toc149170615)

[2.5.2 LBS for navigation 28](#_Toc149170616)

[2.5.3 LBS for information services 29](#_Toc149170617)

[2.5.4 LBS for tracking services 29](#_Toc149170618)

[2.5.5 LBS for billing services 30](#_Toc149170619)

[2.6 Related Works 30](#_Toc149170620)

[CHAPTER THREE 34](#_Toc149170621)

[METHODOLOGY OF THE STUDY 34](#_Toc149170622)

[3.1 Introduction 34](#_Toc149170623)

[3.2 Functional and Non-Functional Requirements of System 34](#_Toc149170624)

[3.2.1 Functional requirements analysis 34](#_Toc149170625)

[3.2.2 Non-functional requirement analysis 35](#_Toc149170626)

[3.2.3 Hardware requirements 36](#_Toc149170627)

[3.2.4 Software requirements 36](#_Toc149170628)

[3.3 System Analysis and Design 37](#_Toc149170629)

[3.3.1 Use case diagram 38](#_Toc149170630)

[3.3.2 Sequence diagram 40](#_Toc149170631)

[3.3.3 Class Diagram 43](#_Toc149170632)

[3.3.4 System architecture 44](#_Toc149170633)

[3.4 System Development Tools 46](#_Toc149170634)

[3.4.1 Cascading Style sheet 46](#_Toc149170635)

[3.4.2 PHP 46](#_Toc149170636)

[3.4.3 SQL 47](#_Toc149170637)

[3.4.4 Google map API 47](#_Toc149170638)

[4.2 Implementation Design 48](#_Toc149170639)

[4.3 Database Design 54](#_Toc149170640)

[CHAPTER FIVE 56](#_Toc149170641)

[SUMMARY AND CONCLUSION 56](#_Toc149170642)

[5.1 Summary 56](#_Toc149170643)

[5.2 Conclusion 57](#_Toc149170644)

[References 59](#_Toc149170645)

[Source Code 65](#_Toc149170646)

# List of figures

[Figure 1 Use case diagram showing user activities 39](#_Toc149170272)

[Figure 2 Sequence diagram of emergency seeker’s request 41](#_Toc149170273)

[Figure 3 Sequence diagram of emergency response officer 42](#_Toc149170274)

[Figure 4 Class diagram of the interaction between actors, locations and activities of the emergency response system 44](#_Toc149170275)

[Figure 5 System architecture of emergency response system 45](#_Toc149170276)

[Figure 6 User Login Interface 48](#_Toc149170277)

[Figure 7 User Registration Interface 49](#_Toc149170278)

[Figure 8 User dashboard 49](#_Toc149170279)

[Figure 9 Get direction 50](#_Toc149170280)

[Figure 10 Emergency Screen 50](#_Toc149170281)

[Figure 11 Admin Login 51](#_Toc149170282)

[Figure 12 Admin Dashboard 51](#_Toc149170283)

[Figure 13 Hospital List 52](#_Toc149170284)

[Figure 14 Add New Hospital 52](#_Toc149170285)

[Figure 15 User list 53](#_Toc149170286)

[Figure 16 Emergency List 53](#_Toc149170287)

[Figure 17 Detailed Emergency 54](#_Toc149170288)

[Figure 18 Admin table 54](#_Toc149170289)

[Figure 19 Complaint Table 54](#_Toc149170290)

[Figure 20 Hospital database table 55](#_Toc149170291)

# CHAPTER ONE

# INTRODUCTION

# 1.1 Background to the Study

An Emergency is an event that threaten the life style of an individual or people living within a specific location (Saeed, Bhatti, Ajmal, Waseem, Akbar & Mahmood, 2013).Situations that can be referred to as emergencies include: fire incidents, explosions, natural disasters, terrorism, road traffic accident, and seizures to mention a few. With current technology era where mostly everything runs on Smartphone’s and applications, the need of quick and efficient services is almost important in every aspect especially when it comes to medical services(Mod Sakriya & Samual, 2015). Patients are mostly having issues on searching for an ambulance, handling the locations and availability of the limited service in the time of emergency. The lack of such attention and information may lead to several casualties.

Emergency medical service (EMS) is a service which is responsible for leading the department in providing proper planned and organized emergency management resources which is capable of responding to public emergencies whenever it is need (El-Masri & McDaniel). Ambulance can be categorized as a limited resource in EMS and since the congested and rapid development of urbanization and concrete jungle in each and every part of the world; the route to search and rescue for human shelter is very complex. This reflects the performance of the ambulance driver to reach the emergency spot on time (Lee, Chuah, & Chieng, 2013).

Location-Based Services (LBS) are systems that employ accurate, real-time positioning in order to connect users to points of interests providing them with information about current conditions (Gholamhosseini Sadoughi & Safaei, 2019).Such conditions may include: traffic, weather conditions, or the provision of routing and tracking location of interest. Location based services have already proven to be useful as people need information related to their position. Such information is especially important when there is an emergency situation, under stress and in unfamiliar environments. Questions which arise in such situations include: Where can I find certain assets? How do I navigate my way there?

Location-based Services (LBS) emerged from the convergence of three major technological trends (Bridwell & Miller, 2016): geospatial technologies, including location-aware technologies, geographic information systems (GIS) and spatial databases; the Internet, and; information and communication technologies, in particular, personal computing devices and mobile communication. Location-Based Services Technology includes (Goodrich, 2021):

* 1. **Global Positioning Systems (GPS):** is an array of satellites that exist solely to help find things across the planet. This allows your phone to know exactly where you are and provide turn-by-turn navigation.
  2. **Wi-Fi Tracking**: uses IP for tracking the location of devices on the Internet. It is need for accurately sending information across Internet infrastructure.
  3. **Cellular Tracking**: works like GS but instead of connecting to satellites connects to cellular towers. It uses the concept of triangulation to locate mobile devices.
  4. **QR Codes**: can be used to scan and record the physical location of an entity thus tagging the location to the QR code.
  5. **Radio Frequency Identification (RFID) tracking**: accesses the location of a RFID tagged device connected to a static RFID scanner upon activation.

According to Idhoko and Ojaiko (2013), GIS is a computer system capable of capturing, storing, analyzing, and displaying geographically referenced information which works by relating information from different sources. The ontological changes and technological advancement have increased the awareness of GIS’s potential among the general public and also encouraged researchers to explore more powerful GIS techniques (Smiatek, 2005; Pearce et al., 2007). The recent development of web services, 3-dimensioanl (3D) visualization tools (e.g., Google Earth, World Wind) and Maps Application Programming Interfaces (APIs) have certainly contributed to the ever-increasing attention to the development and implementation of distributed GIS through the Internet (Butler, 2006).

The power of a GIS comes from the ability to relate different information in a spatial context and to reach a conclusion about this relationship. Since the emergence of the Internet in the 1990s, there has been a paradigmatic shift in all aspects of Geographic Information Systems (GIS). The conceptual model (and hence its technology) of GIS has undergone a trend of transformation – from an isolated architecture to an interoperable framework; from a standalone solution to a distributed approach; from individual proprietary data formats to open specification exchange of data; and from a desktop platform to an Internet environment (Chow, 2008).

# 1.2 Statement of the Problem

A number of people with all forms of medical emergency especially in isolated environments and unfamiliar locations are usually faced with the challenges of locating medical centers not too far from their location (Rahman and Zlatanova, 2006).Existing literature have focused attention on tracking emergency rescue, hospital facility monitoring, medication monitoring, in-patient/out-patient monitoring, and emergency ambulance service (Sakriya & Samual, 2015) using location-based services but have not considered tracking the location of medical centres. There is a need to implement a location-based system that will provide users facing medical emergency with the location of medical centres based on information about their current location, hence this study.

# 1.3 Aim and Objectives of the Study

The aim of this study is to develop a system that will facilitate the real-time identification and location of the nearest hospitals to a site of medical emergency based on information about the location of the site.

The specific research objectives are to

1. elicit knowledge on the user and system requirements of the proposed system;
2. specify the design of the proposed system based on (i);
3. implement the system; and
4. test the system.

# 1.4 Methodology of the Study

In order to meet up with the aforementioned objectives of this study, the following methods will be adopted.

1. The requirements of the system will be elicited by assessing the existing means of accessing the location of hospitals following which the coordinates of various hospitals will be collected for storing information about their location.
2. The system design will be specified using unified modeling languages (UML) such as: use-case diagrams for user actions, sequence diagrams for timing operations and class diagrams for data modeling based on the requirements identified in (a).
3. The front-end system will be implemented using hypertext markup language (HTML) and cascading styling sheets (CSS).
4. The database will be implemented using structured query language (SQL) and the digital map will be integrated using the Google Map API.
5. The system will be tested based on the user acceptance test (UAT) methodology.

# 1.5 Justification of the Study

The development of a system for tracking the location of nearby hospitals will assist stranded travelers involved in road accidents to easily locate emergency centres or hospitals in close proximity to their location at the push of a button. The system will also aid emergency-service providers to easily locate individuals in need of emergency services by navigating them through unfamiliar routes thereby reducing time of arrival at scene of need.

# 1.6 Scope and Limitations of the Study

This study is limited scope to the development of a system that can be used to track the location of nearby hospitals and emergency centres located in south-western Nigeria. The mobile part of the system is limited to the android operating system.

# 1.7 Definition of Terms

1. **CSS** – Cascading Styling Sheet
2. **EMS** – Emergency Medical Services
3. **GIS** – Geographical Information System
4. **GPS** – Geographical Positioning Satellite
5. **HIS** – Health Information Systems
6. **HTML** – Hypertext Markup Language
7. **LBS** – Location-Based Services
8. **SDLC** – System Development Life Cycle
9. **SQL –** Structured Query Language
10. **UAT** – User Acceptance Test
11. **UML** – Unified Modeling Language

# CHAPTER TWO

# LITERATURE REVIEW

# 2.1 Medical Emergency Response

Emergency can point towards all those events that threaten life styles of the people living in a state, especially the resources of that state prescribed for its sustainability and stability. All such situations might be carried out from fires, explosions, and traffic accident, terrorism, natural disasters all comprises of hazardous events (Rahman & Zlatanova, 2006).Tragic accidents happen every day all over the world causing injuries of different extend. The ambulance teams are the first to get to the place and provide medical help. The three basic tasks of the ambulances are: to get at the site of accident in the fastest manner; to provide the best treatment; and to transport the patient to a hospital. There are several specific characteristics within the work of medical personal on the field especially in large accidents (Torg *et al.*, 2005).

The types of injuries (e.g., skin damages in fire, or breathing problems in gas leakage) are very similar, which might need large amounts of the same medicines and specialists. Many injuries may require immediate and simultaneous high qualified treatment (e.g., surgery), an unusual situation which may face limited capacity (in terms of equipment and teams) in nearby hospitals (Montoya, 2003). The number of patients can rise significantly and will require mobilization of additional hospital units and equipment. Logistic problems, since many ambulances have to deliver several patients to more than one hospital. Stress and panic among the injured people, which usually results in bad estimates of the number injures and needed treatment (Togt, Beinat, Zlatanova & Scholten, 2005). A need for a very good real-time coordination and communication with police, fire brigade and the local authorities to speed up transportation of victims and evacuate non-injured to safe areas. A need for real-time communication and information to media, the public and especially relatives of injured people.

Clearly, a much better communication is required at several levels (Zlatanova, 2005).Firstly, the cooperation between medical institutions and all the other organizations involved in emergency response has to be improved. Secondly, a better organization within the hospitals is urgently needed concerning the deployment of specialists, availability of medicaments, transportation, rooms and equipment. The services have to demand-driven and patient-oriented. Thirdly, the communication between the hospitals and the ambulances on the field has to be strengthened and (when needed) enhanced with a real-time supervision .This support can be in two directions: guidance ambulances to the place of accident (especially important in large urbanized city areas) and supplying information about the condition of the injured people prior their transportations to the hospital.

# 2.2 Information Systems

According to Stair and Reynolds (2006), information systems can be defined as a set of interrelated components that can be used to collect, manipulate, and disseminate data and information for purpose of planning, control, coordination, analysis and decision-making by an organization (Correia, Chiodini, Dalfovo, Silva, & Teske, 2013). The set of components can be composed of people, hardware, software, communication networks and data storage resources. An information system is a group of interrelated components that work to carry out input, processing, storage, output and control actions in order to convert data into information that can be used to support forecasting, planning, control, coordination, decision making and operational activities in an organization (Al-Mamary, Shamsuddin, & Aziati, 2014). Today, every business organization needs an information system for keeping track of all business activities, right from business planning, till the product delivery via manufacturing and quality cycles (Nowduril & Al-Dossary, 2012).

## 2.2.1 Types of information systems

According to O’Brien & Marakas (2007), the applications of information systems that are implemented in today’s business world can be classified in several different ways. For example, several types of information systems can be classified as either operation which are used for supporting business operation such as day-to-day transactions, managerial decision-making process, for processing control systems and enterprise collaboration systems for providing office automation system (Patterson, 2005).

1. **Transaction processing systems (TPS)**

These are the basic business systems that serve the operational level of the organization. A transaction processing system is a computerized system that performs and records the daily routine transactions necessary to the conduct of the business (Laudon & Laudon, 2006). At the lowest level of the organizational hierarchy, there exist the transaction processing systems that support the day-today activities of the business (Belle, Eccles, & Nash, 2001).

1. **Process control systems**

Process control systems monitor and control industrial or physical processes. Examples: petroleum refining, power generation, and steel production systems. For example, a petroleum refinery uses electronic sensors linked to computers to monitor chemical processes continually and make instant (real-time) adjustments that control the refinery process (O’Brien & Marakas, 2007). A process control system comprises the whole range of: equipment, computer programs, operating procedures (Ciortea, 2004).

1. **Enterprise collaboration systems**

Enterprise Collaboration Systems (ECS) are one of the most widely used types of information systems that will help managers control the flow of information in organizations (Heidarkhani, Khomami, Jahanbazi, & Alipoor, 2013).Enterprise collaboration systems enhance team and workgroup communications and productivity (O’Brien & Marakas, 2007). Office automation systems are other types of information systems are not specific to any one level in the organization but provide important support for a broad range of users (Belle, Eccles, & Nash, 2001). Office information systems are designed to support office tasks with information technology. Voice mail, multimedia system, electronic mail, video conferencing, file transfer, and even group decisions can be achieved by office information systems (Shim, 2000).

1. **Management information systems**

Management Information Systems (MIS) are a kind of computer information systems that could collect and process information from different sources in institute decision making in level of management (Heidarkhani, Khomami, Jahanbazi, & Alipoor, 2013).Management information systems provide information in the form of pre-specified reports and displays to support business decision making (O’Brien & Marakas, 2007). The next level in the organizational hierarchy is occupied by low level managers and supervisors. This level contains computer systems that are intended to assist operational management in monitoring and controlling the transaction processing activities that occur at clerical level. Management information systems (MIS) use the data collected by the transaction processing systems to provide supervisors with the necessary control reports (Belle, Eccles, & Nash, 2001). According to Hasan, Shamsuddin and Aziati (2013), management information system is type of information systems that take internal data from the system and summarized it to meaningful and useful forms as management reports to use it to support management activities and decision making.

1. **Decision support systems**

A Decision Support System (DSS) is a computer-based system intended for use by a particular manager or usually a group of managers at any organizational level in making a decision in the process of solving a semi structured decision (Asemi, Safari & Zavareh, 2011). According to Heidarkhani, Khomami, Jahanbazi, and Alipoor (2013), decision Support Systems are Kind of organizational information computerize systems that help manager in decision making that needs modeling, formulation, calculating, comparing, selecting the best option or predict the scenarios. According to Khanore, Patil and Dand (2011), decision-support systems are specifically designed to help management make decisions in situations where there is uncertainty about the possible outcomes of those decisions. According to Shim (2000), a decision support system is computer-based information

1. **Executive information systems**

Executive Information Systems (EIS) have been developed, which provide rapid access toboth internal and external information, often presented in graphical format, but with the ability to present more detailed underlying data if it is required (Belle, Eccles, & Nash, 2001).Executive information systems provide critical information from a wide variety of internal and external sources (from MIS,DSS, and other sources tailored to the information needs of executives) in easy-to-use displays to executives and managers (O’Brien & Marakas, 2007). According to Patterson [2] An EIS provides senior managers with a system to assist in taking strategic and tactical decisions. According to Shim (2000), an executive information system is designed to generate information that is abstract enough to present the whole company operation in a simplified version to satisfy senior management.

1. **Expert systems**

Expert Systems (ES) are the category of artificial intelligence (AI) which has been used most successfully in building commercial applications (Belle, Eccles, & Nash, 2001).According to O’Brien & Marakas (2007), expert systems are knowledge-based systems that provide expert advice and act as expert consultants to users. According to Patterson (2005), an expert system is a computer program that tries to emulate human reasoning. According to Shim (2000), expert System is a set of computer programs that perform a task at the level of a human expert.

1. **Knowledge management systems**

Knowledge Management Systems (KMS) are knowledge-based information systems that support the creation, organization, and dissemination of business knowledge to employees and managers throughout a company (O’Brien & Marakas, 2007).Knowledge management is the deployment of a comprehensive system that enhances the growth of an organization's knowledge (Salisbury, 2003).

1. **Strategic information systems**

Strategic information systems apply information technology to a firm’s products, services, or business processes to help itgain a strategic advantage over its competitors (O’Brien & Marakas, 2007). According to Belle, Eccles and Nash (2001), strategic information systems are an important special type of organizational information system is used to secure or sustain competitive advantage in the market place.

## 2.2.2 Health information systems

Nowadays, the widespread use of Information and Communication Technologies (ICT) has permeated almost all aspects of life including the healthcare sector. Haux (2006), described systems that process data and provides information and knowledge in health care environments as health information systems. The aim of health information systems is to contribute to a high-quality, efficient patient care. Health information systems were introduced to fully utilize especially the Internet in providing better healthcare (Almunawar & Anshari, 2012). Health information systems refer to the interaction between people, process and technology to support operations, management in delivering essential information in order to improve the quality of healthcare services.

Some of terminologies related to health information systems are as follows. Health Informatics is the field that concerns itself with the cognitive, information processing, and communication tasks of medical practice, education, and research including the information science and technology that supports those tasks. Health informatics tools include computers as well as clinical guidelines, formal medical terminologies, and information and communication systems. In other words, it emphasis is on clinical and biomedical applications with added possibility of the integrating clinical components either among themselves or to more administrative-type health information systems (Conrick, 2006). In addition, Health information technology is the application of information processing involving both computer hardware and software that deals with the storage, retrieval, sharing, and use of health care information, data, and knowledge for communication and decision making(Goldschmidt, 2005).

Another important terminology in HIS is Electronic Medical Records (EMR), it resides at the centre of any health information systems. EMR is a medical record in a digital format, whereas electronic health record (EHR) refers to an individual patient’s medical record in a digital format. HER systems coordinate the storage and retrieval of individual records with the aid of computers, which are usually accessed on a computer, often through a computer network. One of the important trends is the move towards a universal electronic patient record (EPR). EPR is defined as electronically stored health information about one individual uniquely identified by an identifier. Essentially EPR technology entails capturing, storing, retrieving, transmitting, and manipulating patient-specific, healthcare related data singly and comprehensively, including clinical, administrative, and biographical data (Protti*et al.*, 2009).

# 2.3 System Development Life Cycle

Software development life cycle (SDLC) is a process followed for a software project within a software organization. In literature, most of the work defined SDLC processes and concluded that it is a method of software quality assurance and a way to ensure that software development teams stay on the same page. In Tatar and Tomur (2013), SDLC was defined as a conceptual model including a sequence of processes followed to develop information systems. It explained that the software industry follows the SDLC because it defines which task must be performed at each stage in the software development process and when carried out effectively, the SDLC produces high quality software that meets customer expectation and reaches completion within time and cost estimates.

As stated in Bhatnagar and Singh (2013), SDLC process is a sequence of activities for system designers and developers to follow for developing software efficiently and on prescribed time such as analysis, design, implementation, testing and maintenance. It further explained that SDLC is a concept used in project management that describes the stages involved in an information system development project starting from an initial feasibility study to maintenance of the completed application, and emphasized that the primary objectives of the SDLC are to ensure the delivery of high-quality systems using strong management controls to maximize productivity.

In Sanni and Kaur (2014), SDLC was described as the methodology by which the development of any software takes place. It further explained that before SDLC, process of developing the software was taken as informal activities with no formal rules and standards which may lead to various problems such as delay in development, cost overrun and low software quality. The introduction of the SDLC gives the precise standard and the steps for the development of the software.

There are many SDLC models and methodologies such as the waterfall model, rapid application development (RAD), joint application development (RAD), spiral model with each of them consisting of series of defined steps or phases (Tatar &Tomur, 2013). However, Bhatnagar and Singh (2013) highlighted the most well-known and widely used SDLC as the waterfall, V- shaped and evolution rapid. Gandhi et al. (2014) explained that regardless of the choice of model, different types of risks were associated with each of these SDLC phases. It proposed a model to help determine the impact of the risk on the project being developed, so that project failures due to these risks can be minimized.

## 2.3.1 Information system development

An Information system consists of input messages, message processing and output messages. It has processing rules which control the execution of the Information system. If the processing rules are formalized, we can have computer-based Information systems. But if the processing rules need a lot of personal knowledge, judgment and Intuition, the information systems a manual. A purposeful Information system shall help users to make good decisions and support their actions (Anders, 2012). Information systems, like other products, are going through a life cycle. Such as Information Systems Development, Information Systems in Use (Operation), Information Systems Maintenance Management and Information Systems Withdrawal. An information system development is the process of defining, designing, testing, and implementing a new software application or program (Farm Credit Association, 2007). The system developer uses different tools, techniques, procedure, method and philosophy to implement the information system development.

Modeling is a central part of all the activities that lead up to the deployment of good software. It is used to communicate the desired structure and behavior of a system. In addition, models build to visualize and control the system's architecture and to better understand the system we are building, often exposing opportunities for simplification, reuse and manage risks (Booch *et al.*, 1998). A Model provides the blueprints of a system and help the users visualize the final product. Different modeling approaches such as, structured and object-oriented, can be used in information system development. The traditional view of software development is referred to as structured system analysis and design approach. This view leads developers to focus on issues of control and the decomposition of larger algorithms into smaller ones. Such an approach tends to yield brittle systems.

As requirements change and the system grows, systems built with an algorithmic focus turn out to be very hard to maintain. The object-oriented paradigm is currently the most popular way of analyzing, designing and developing application systems, especially large ones. In this paradigm the elements of a given situation is viewed by decomposing them into objects and object relationships (Ramnath and Dathan, 2010). Systems developed with the OO approach are more flexible. These systems can be modified and enhanced easily, by changing some types of objects or by adding new types.

## 2.3.2 Modeling techniques – Unified modeling language (UML)

Notations enable to articulate complex ideas briefly and precisely. In projects involving many participants, often of different technical and cultural backgrounds, accuracy and clarity are critical as the cost of miscommunication increases rapidly. In the OO world UML is the industry-standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems (Laman, 2004). UML use different diagraming model to show the analysis and design of a system. There are nine artifacts defined in the UML modeling which mainly categorized under two different views of a system model. The static (or structural) view emphasizes the static structure of the system using objects, attributes, operations, and relationships. These static parts are represented by, use case, class, package, component, and deployment diagram (Padmanabhan, 2012).

1. **Use case diagram**

The use case diagram is concerned with the interaction between the system and actors (objects outside the system that interact directly with it). It presents a collection of use cases and their corresponding external actors. A use case is a generic description of an entire transaction involving several objects of the system. Use cases are represented as ellipses, and actors are depicted as icons connected with solid lines to the use cases they interact with. A use case diagram is helpful in visualizing the context of a system and the boundaries of the system’s behavior. Each use cases in the use case diagram can also be described using a narrative form (Elkoutbi *et al.*, 2012).

1. **Class diagram**

The class diagram represents the static structure of the system. It identifies all the classes for a proposed system and specifies for each class its attributes, operations, and relationships to other classes. Relationships include inheritance, association, and aggregation.

1. **Component diagram**

A component diagram provides a physical view of the system. Its purpose is to show the dependencies that the software has on the other software components in the system. It is built as part of architectural specification and developed by architects and programmers.

1. **Deployment diagram**

The deployment diagram shows how a system will be physically deployed in the hardware environment. Its purpose is to show where the different components of the system will physically run and how they will communicate with each other. It is used to identify performance bottlenecks, and is developed by architects, networking engineers, and system engineers.

1. **Package diagram**

Package diagram shows how the various classes are grouped into packages. Packages are UML constructs that enable you to organize model elements into groups. It makes your UML diagrams simpler and easier to understand.

Behavioral diagrams basically capture the dynamic aspect of the system by showing collaborations among objects and changes to the internal states of objects. Dynamic aspect can be further described as the changing or moving parts of a system. These dynamic parts are represented by sequence, collaboration, state chart, and activity diagram.

1. **Activity diagram**

Active diagrams are used to model the flow of an object as it moves from state to state at different points in the flow of control. It is essentially a flow chart that emphasizes the activity that takes place over time. Activity diagrams can be used to model higher-level business process at the business unit level, or to model low-level internal class actions. It is l*ess technical* in appearance, compared to sequence diagrams, and business-minded people tend to understand them more quickly.

1. **Sequence diagram**

A sequence diagram shows interaction among a set of objects in temporal order, which is good for understanding timing issues. It shows a detailed flow for a specific use case or even just part of a specific use case. A sequence diagram deals with the sequence of messages flowing from one object to another. It is mainly used to visualize the sequence of calls in a system to perform a specific functionality.

1. **Collaboration diagram**

Collaboration diagram is used to explore the dynamic nature of the software. Collaboration diagrams show the message flow between objects in an object-oriented application, and also imply the basic associations (relationships) between classes. The purpose of collaboration diagram is similar to sequence diagram. But the specific purpose of collaboration diagram is to visualize the organization of objects and their interaction.

1. **State chart diagram**

The state chart diagram models the different states that a class can be in and how that class transitions from state to state. Every class has a state (which is a situation during the life of an object, which satisfies some condition, performs some activity or waits from some event), but that every class shouldn't have a state chart diagram. The state chart diagrams are used to capture event-oriented dynamic behavior, model object. While the UML provides nine artifacts to model the system, it is not important to use all as each of them provides a different perspective of the same system.

# 2.4 Geographic Information Systems (GIS)

A geographic information system (GIS) is a location-based information system modeling the real world. It digitally captures, stores, manages, analyzes and presents location-based datasets as alpha-numerical or graphical output (Lange, 2006). Relating objects to a geographical position within a reference system creates a geographical object. Usually, we use geographical coordinates, i.e., latitude and longitude, to specify its position on the surface of the earth. A GIS is a computer-based system to aid in the collection, maintenance, storage, analysis, output, and distribution of spatial data and information (Bolstad, 2007).

## 2.4.1 Geographic object

A geographic object is the fundamental unit of a GIS. It represents a unique entity of the earth which is physically, geometrically or thematically limited (Reinhardt *et al.*, 2003). Norbert de Lange defines geographical objects as follow: Geographical objects are spatial elements which exhibit geometrical, topological and temporal properties in addition to their semantic information (Lange, 2006). As such, geographic objects are an abstraction of reality. The produced representation of the real world is a digital model with a defined precision. Geographical objects can be classified into points, lines, and areas features, and solid figures (Imhof, 1975).

For example, point features can define border stones or Points-of-Interest, line features can represent water pipelines or roads, area features display municipal areas or land-cover and solid figures represent 3D buildings or trees. This feature-based classification defines one possible organization of geographic data. Another approach consists in an object-oriented model, i.e., general objects can be derived into specialized objects. A child object (e.g., a motorway) would inherit its attributes from a base object (e.g., a road) (Lange, 2006). For managing, processing and visualizing these objects we must create appropriate structures, called data models.

## 2.4.2 Data models

A data model is the abstraction, representation and organization of real-world elements (Kappas, 2001). Therein, the geometry, topology, semantic and relationship of real objects has to be abstracted enough to generate a corresponding data model representation. This allows us to map reality to data structures for computational and visualization purposes in a GIS. On a higher level, we organize geographical objects using two fundamental principles: within a layer or within an object-oriented model. On a lower level, we differentiate between a raster-based and vector-based model.

1. **Thematic layer concept**

The thematic layer concept originates from cartography, where mapmakers created transparencies that could be overlaid on a light table. Hence, by combining different layers, they could create their desired information density in an analog map. This concept represents the default form of data organization within a GIS. It follows a top-down approach to create a thematic sorting of all geographic input information. Each layer represents a distinct data theme consisting of a collection of common geographic elements, e.g., a road network, a digital elevation model or urban areas (Fig. 2.1).Thematic layers have several key advantages. First, they represent an intuitive way to organize and view data in a GIS. Second, errors occurring in a layer only have a local impact. Finally, they are efficient resource-wise, because only requested layers are processed and visualized.

1. **Vector, raster and hybrid models**

Vector data models represent information as points, lines and polygons (Fig. 2.2). In a GIS, the OGC and ISO committees define these basic geometrical elements as Simple Features (ISO, 2004). This model discretizes the geometry of real-world elements. All geographic elements of the vector data model are based on point coordinates, e.g., latitude and longitude. The topological relationship is stored explicitly, e.g., which points create a line or an area (Lange, 2006).Using further attributes, we define the thematic relationship, e.g., whether a line is a road. Therefore, the vector data model is also called the geo-relational data model (Bartelme, 2005).

This model presents several advantages (Buckey, 1997). Geographical data can be represented with its originally captured resolution. In a cartographic representation, the graphical output is usually more aesthetically pleasing. Also, simple geometrical elements can usually be very efficiently encoded into vector data, e.g., a road network. Topology is easily stored and enables efficient topological operations, e.g., network analysis. However, continuous data, e.g., temperature or elevation data is not effectively stored in vector form. Furthermore, the complexity of data operations is proportional to the number of simple features present.

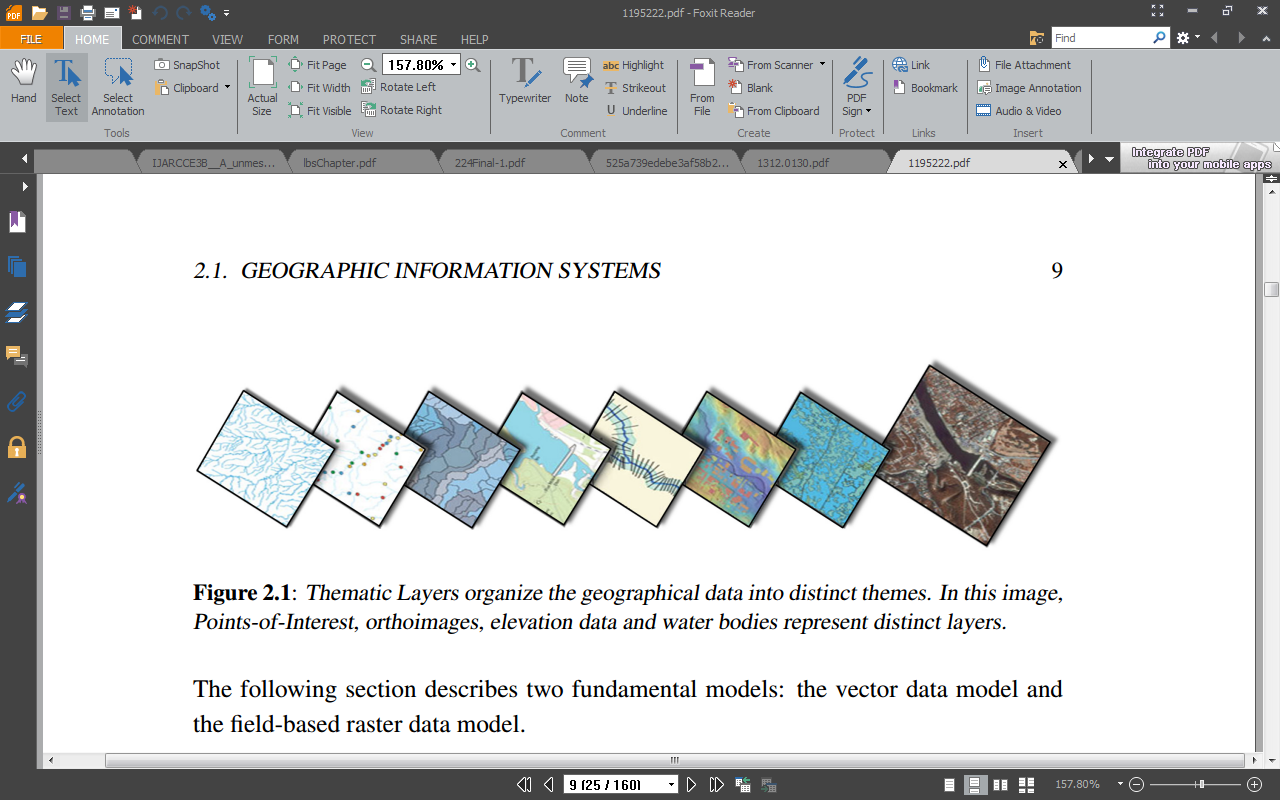


Figure 2.1: Thematic layers showing distinct images of geographical data

(Source: Vaaraniemi, 2014)

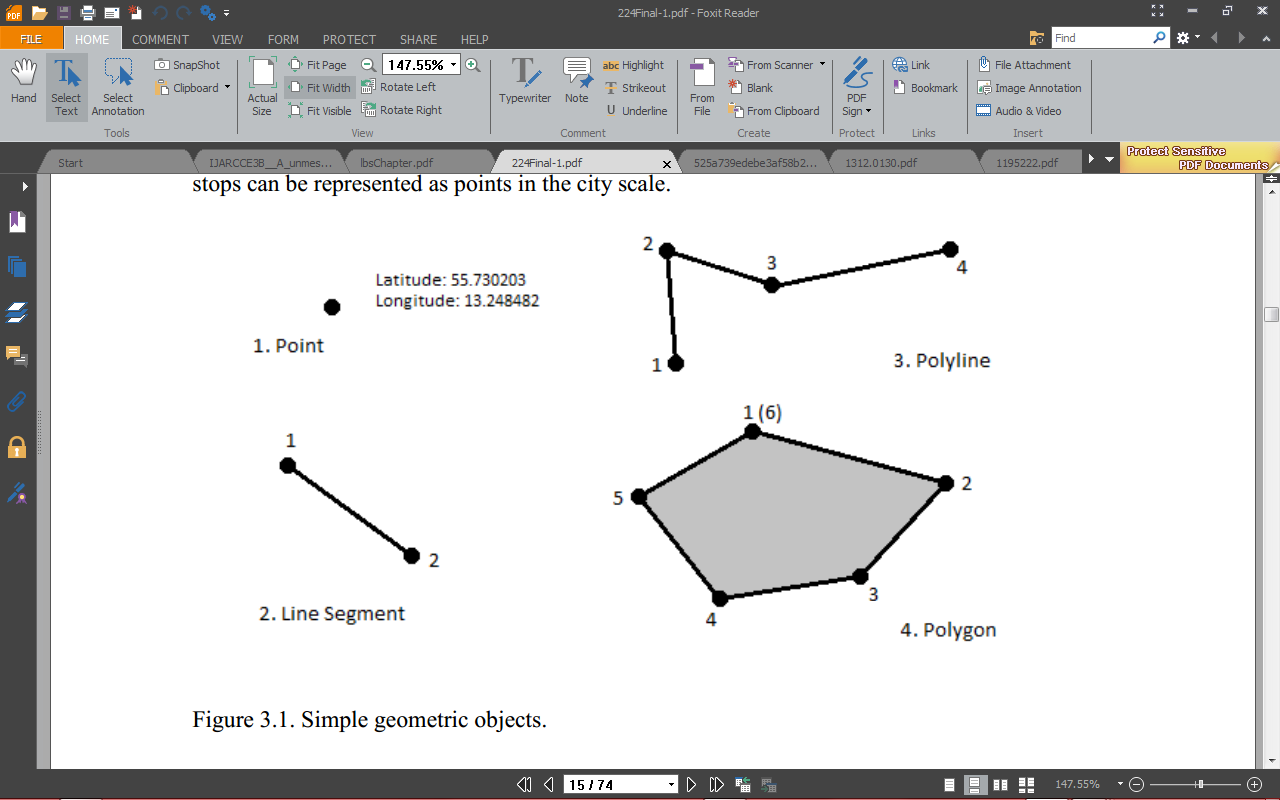


Figure 2.2: Vector data models

(Source: Bartelme, 2005)

1. **Field-based data model**

The field-based model partitions the theme of the geographic input surface into homogeneous areas (cells). The form and size of these cells can be defined freely. However, as a whole, they should cover the entire input surface (Bartelme, 2005). Therefore, each cell explicitly stores geo-referenced thematic information as shown in Figure 2.3. An example for the field-based concept is the DEM, wherein each cell represents the average height inside the covered input surface.

1. **Raster data model**

In GIS, the raster data model is used to represent continuous data over space. It is a specialization of the field-based model. The input surface is divided into equally sized areas, usually a quadratic cell, i.e., pixel. For example, every cell stores the ambient temperature or the average height. The size of the cells definesthe perceivable data resolution (Bartelme, 2005).The raster data model has several advantages (Buckey, 1997). In comparison to the vector model, the geographic coordinates are not explicitly stored. If the geographic location and extend of the entire grid is defined, the position of every pixel is implicit in the layout of the grid.

Moreover, the theme (e.g., the temperature) is given implicitly and not explicitly like in the vector model. Hence, data processing and analysis is usually quite simple to perform. It is perfectly suited for continuous data. However, the cell size determines the resolution for processing and visualizing the data. Hence, it is difficult to adequately represent linear geographical elements, i.e., simple features. Usually, storing this data at a high precision comes at the expense of a very high storage cost.

With the subsequent rapid expansion and development of the Internet and the World Wide Web (www), GIS extended onto websites were becoming ever more

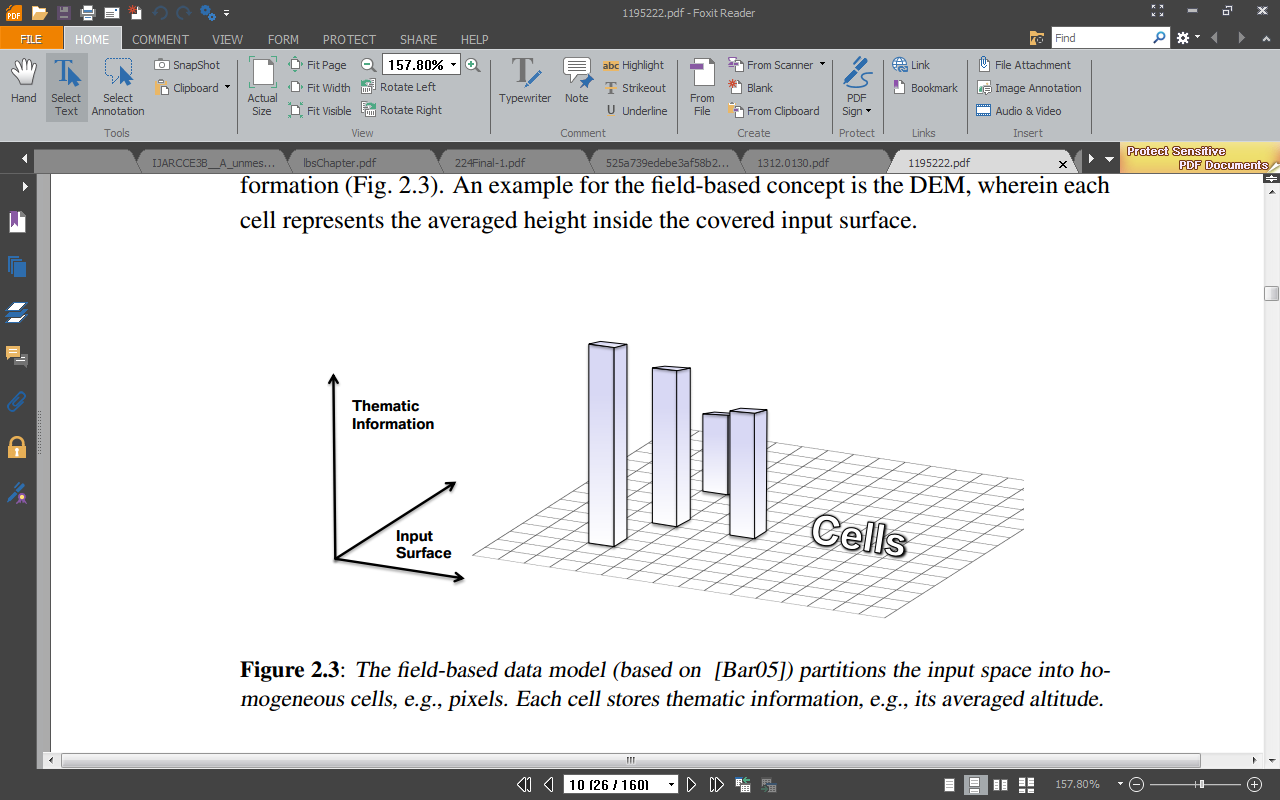


Figure 2.3: Field based data model showing partitions of the input space

popular and as a result numerous sites have added GIS capabilities on their websites (Zhuang, 1997). Furthermore, Mobile-GIS are emerging as a result of integrating GIS, global positioning systems (GPS), wireless communications and mobile computing technologies (Karimi, 2000). Providing this kind of functionality to the general public has sparked off the initiation and development of location-based services (LBS) (Francica, 2000).

1. **Location-Based Service (LBS)**

LBS are any service or application that extends spatial information processing or GIS capabilities to end users (based on their geographical location) via the Internet and/or wireless networks (ESRI, 2001). Such services combine scalable GIS technology, easy-to-use browsers, mobile and wireless devices, and wireless and Internet infrastructure with web servers to provide information and services whenever and wherever they are needed.

# 2.5 Location-Based Services (LBS) Applications

Nowadays with the rapid development of information and telecommunication technologies which are integrated in devices, identifying location of mobile devices has become important. Mobile devices are used in various areas like m-banking, m-government and also m-learning (Ghadirli & Rastgarpour, 2012).Several technologies with various accuracy and cost such as Geographical Information Systems(GIS), Global Positioning Systems (GPS) are also integrated in mobile devices which are used to determine the location of people, cars, etc. Some most recent location sensing technology based on ultra-wideband radio can even achieve accuracies on the order of centimeters in an indoor environment (Ververidis & Polyzos, 2002).

The term Location-based services (LBS) refer to set of applications which can identify the location of amobile device and offer value added service to the mobile user base on his/her location(Madadipouya, 2014). Value added services include emergency services, navigation services, entertainment services and such services and such services are increasingly being integrated in various combinations, and to various extents, to meet both user and regulatory requirements in a variety of operational environments (Hurson & Gao, 2009). Location-based applications can be generally classified as push or pull applications. In the push mode, services are pushed to the user automatically without his/her permission; an example is in an advertising application. By contrast in the pull mode, the user has this freedom to choose deliverable services from LBS providers based on his needs, which tourism application is the common example of pull mode LBS.LBS are an intersection of three technologies. It is based on new information and communication technology (NICTS) plus Geographic Information Systems (GIS) with special databases.

In order to use a location-based service five basic components are requested. The first element is a mobile device which a tool for the user to receive the needed information and interact with application. The next component is communication which transfers data between a user and a service provider. In other word, the communication network is responsible for interactions among them. The third one is positioning component and it is needed for the processing the user geographical location. The user location can be acquired by using GPS (Global Positioning System), mobile communication networks and the Internet. Service and application provider is the fourth part of the LBS. It offers a number of different services to the user and responsible for the processing of the requested service (Steiniger, Neun, Edwards, 2006). The last component is data and content provider. Service providers mostly do not store all of information which may needed by users. Hence, geographical location of the user and data related the user position is mainly gained from maintaining authority or business and industry partners such as traffic companies, mapping agencies.

Location-based services provide many advantages to the users and service providers, some of them are listed below:

1. It helps to offer information based on user request among lot of data available on the Internet;
2. Service providers with providing relevant information for users assist them to speed up their decisions and activities;
3. It decreases the volume of user input data for accessing a service. (LBSs acquire information about users' location automatically via smart mobile devices); and
4. With sharing location-tagged information, more and up to date data is available for all users.

## 2.5.1 LBS for emergency services

Emergencies and disasters have been inevitable parts of human life since the beginning of theworld. Wide ranges of potential known and formerly unknown hazards keep challenging society even nowadays. Emergency location based-services have been recommended, tested or utilized to overcome impacts of catastrophes. In this scenario LBS is used to determine the exact geographical location. In these conditions LBS can find an individual place after the cell phone user has made an emergency call or a distress short message service request for help who does not know about his/her current location or cannot reveal it due to an emergency circumstance (injury, criminal attack, and etc.).

As a solution LBS also can be utilized to deliver warning notifications and emergency alert information. For instance, emergency information to disaster areas could be broadcast via the new 4G standard “Multimedia Broadcast Multi-Cast Service (MBMS)” with rich multimedia content like voice instructions and evacuation maps (Aloudat, Michael, & Yan, 2007; Michael, Abass, Aloudat, & Al-Debei, 2011). Using LBS reduces the side effects and of the emergency and unpredictable disaster. In addition, broadcasting alerts in dangers areas can prevent human damage.

## 2.5.2 LBS for navigation

Navigation services have been the core of location-based services for many years (Rao & Minakakis, 2003). In this service, the location of mobile users or vehicles are determined to offer value added services regarding traffic jams, close points of interest (POI) such as gas stations or parking spaces (Grebmann, Klimek, & Turau, 2010).In addition, navigation system is able to compute the shortest path for the user’s destination based on his/her current location (Hand, Cardiff, Magee, & Doody, 2006). Users can also use the navigation service to find their direction and their current location (Steiner, Neun, & Edwardes, 2006).Information can be provided for the device via system update or the internet. In the second mode, data is loaded from the internet with one of the various positioning technologies (Grebmann, Klimek, & Turau, 2010): cellular network (GSM, UMTS) or satellite (GPS, A-GPS)

## 2.5.3 LBS for information services

Information service is, identifying the user’s location and offering information based on his/her location such as list of restaurants near the current place of the user or call a cap from the nearest taxi station, etc. Information services can be used in tourist industry as well. Visitors can utilize the service in their cell phones to easily get information like location of tourist attraction, transportation facilities, accommodation places, medical facilities. According to a report by Zickuhr (2012), showed that around 74 per cent of smart-phone users use their phones for getting real-time location-based information services based information also available in another form which is called location-sensitive information services. It is referring to distribution of proper information based on device location, user’s behavior and time (Steiner, Neun, & Edwardes, 2006).

## 2.5.4 LBS for tracking services

Tracking is finding the location of entities such as personal and staff security, goods in transit and vehicles. Initially, it is used to track employee locations. Today, tracking service is used in business area and home life as well. In business area delivery companies could use tracking service to know where their delivery vehicles are. With using tracking service companies’ can ensure about that all employees are where they should be during working hours. In addition, they can let customers know about the location of a truck that is due to deliver to their house (Brown & Sturza, 1995).In home life tracking service allows parents to track their children at anytime and anywhere without any age limitation.

Different devices such as GPS receivers, phone applications and car trackers could be used to tracking children with or without their awareness. They are always able to trace the exact location of their kids and monitor activities they are involved in (Applewhite, 2002). Some virtual boundaries such as around a school or home can be set up by parents that could alert them when the device arrives or leaves at the zone and moving speed (Burke, 2012). In addition, the application is applicable in different form for elderly people and patients as well.

## 2.5.5 LBS for billing services

The service offers calling plans to hand phone users based on their location which they make calls. Lowe cost calls are provided for the users who have unique geographic zones such as home, office or other desired locations. When an LBS user makes or receives a call, LBS checks whether the subscriber is in the preferential calling zones or not. Information is made to the billing system if the subscriber is in one of preferred areas and he/she is charged for the call with preferred rate.

Zones are managed by the carrier’s Customer Care with the LBS Web interface. If it is set properly, users can choose their preferred areas from their cell phones or via the internet. The LBS administrator can dedicate various radiuses for each rate area and exclude some zones in market scopes. In the exclusion zones subscribers are banned to create rate zones. For example, high-income domains or areas with low network coverage (Chan & Lars, 2003).

# 2.6 Related Works

Balogun, Kasali, Akinyemi, Akinyemi and Idowu (2021), worked on the development of an information system that was used for tracking terrorist attacks online. The system and user requirements of the proposed system were identified following the location of various security agencies across Nigeria were gotten by collecting their coordinates. The unified modeling language was used to specify the design of the system based on the system requirements identified. The system was implemented was a web-based information system using HTML, CSS, SQL and Google Map API. The system could effectively capture the location and information and reported crimes across various location in south-western Nigeria and display their spatial distribution on a digital map. The study was limited to the spatial distribution of reported crimes.

Gholamhosseini Sadoughi and Safaei (2019), performed an extensive review of various real-time location-based systems adopted by hospitals in Iran. The study gathered and analyzed information from textbooks and indexing sites related to hospital real-time location systems from IEEE, PubMed, Science Direct, Medica, Scopus, Web of Science, Elsevier and Google Scholar from 2006 to 2017. A total of 148 sources were selected from which 44 sources were reviewed. The results showed that most of the sources focused on tracking patients, elderly care, medical assets, medication; monitoring blood transfusion, healthcare environments and for the surveillance of children and elderly. Various technologies adopted include: radio frequency identification (RFID), wireless sensor networks (WSN), GIS, and global positioning systems (GPS).The study focused on the review of the various application of location-based systems and technologies adopted within the healthcare sector.

Mod Sakriya & Samual (2015), worked on the development of a geographical information system for ambulance emergency response. The study identified the challenges facing the existing system alongside the user and system requirements of the proposed emergency response system. The study specified the design of the system using UML. The study implemented the use of GPS for tracking the location of the emergency which can be identified by an emergency ambulance. The system was implemented as a mobile application for android OS and adopted the Google Map for location tracking. The results of the system implementation revealed that the system was able to provide the location of accidents to ambulance while plotting the shortest path to the location thus reducing arrival time. The study focused on a location-based system for tracking accident victims by emergency ambulance.

Martin-Gomez, Vergara-Falces, & Elvira-Zalduegui (2015), developed a location-based system for responding to fire emergencies at reional level. The system was developed with the purpose of storing and retrieving information relevant for tracking the various fire incidents across Nevarra territory. The system makes infromation stored avaibale to both the fire depertament, forest guard and the police department with respect to the type of vehicle used to respond to the emergencies and the type of road leading to the location of the emergencies. The sytem and user requirements were identified following which the design was specified based on the identifed requirements. The system was implemented as a web-based system which was integrated with a digiyal mapping system. The study was limited to the development of a system used for tracking and responding to fire emergencies.

Rahman and Zalatanova (2006), proposed architecture for the development of location-based emergency management system for hospitals in Malaysia. The study assessed the existing manual system which was used by hospitals for responding to emergencies and these as a basis for identifying the user requirements of the proposed system. The study specified a system architecture of the proposed system which composed of GPS-enabled mobile devices by the users which communicated with the hospital base system and an on-the-move spatial subsystem. The results of the study showed that the development of a computerized emergency management system would reduce the number of deaths caused by untimely response by emergency response teams. However, the study was unable to implement the proposed system design.The study focused on the development of an emergency management system that can help hospitals to easily respond to emergency from mobile users.

# CHAPTER THREE

# METHODOLOGY OF THE STUDY

# 3.1 Introduction

This chapter presents the process via which the aim of this study was achieved using a number of methods. It starts by presenting the user and system requirements of the proposed system for monitoring the location of medical centers and reported emergency locations across south –western Nigeria. The identification of the requirement was followed by the specification of the design using the Unified Modeling Language (UML) 2.0 diagrams such as the user-case for the user action specification and the class diagrams for the abstraction of the system classes and their relationships. Also, a description of the system development tools that were used for this study was also done.

# 3.2 Functional and Non-Functional Requirements of System

In order to develop a system that can be used for monitoring the location of medical centers and emergency locations, a set of requirements must be met by both users and systems which are generally classified as functional and non-functional requirements.

## 3.2.1 Functional requirements analysis

This involves the system functionality and behavioral analysis that how the system in view was able to perform the tasks required of it by the users. This involves the use of necessary unified modeling languages to analyze the system. Hence, the functional requirements for the development of the system by users were as follows:

1. The system will only allow authorized users access to the system using usernames and passwords provided by system administrator;
2. New users and existing users must be able to change their default passwords to their preferred password;
3. The system must allow new medical centers and new users to be created when required;
4. Existing users must be able to provide information of new emergencies upon reporting;
5. Existing emergency records can be viewed by users using tables and charts; and
6. System must provide a digital map showing the distribution of the location of emergencies captured by emergency response personnel.

## 3.2.2 Non-functional requirement analysis

Non-functional requirements define the overall qualities or attributes of the system. Non-functional requirements place restrictions on the system being developed, the development process, and specify external constraints that the system must meet which include:

1. **Accessibility**: accessibility analysis is a type of system analysis designed to determine whether individuals with disabilities will be able to use the system in question, which could be software, hardware, or some other type of system. Disabilities encompasses a wide range of physical problems, including learning disabilities as well as difficulties with sight, hearing and movement;
2. **Integrity control:** integrity in term of data and network security is the assurance that information can only be accessed by those authorized to do so.
3. **Security:** in information technology, security is the protection of information assets through the use of technology, processes, and training. This system was built such that: the access permissions for system data may only be changed by the system’s data administrator; all system data must be backed up every 24 hours and the backup copies stored in a secure location which is not in the same building as the system. All external communications between the system’s data server and clients must be encrypted
4. **Authentication:** is the process of determining whether someone or something is, in fact, who or what it is declared to be.
5. **Reliability**: is the ability of a system to perform its required functions under stated conditions for a specific period of time.
6. **Confidentiality:** is a set of rules or a promise that limits the access or places restrictions on certain types of information.
7. **Dependability**: the dependability of a computing system is the ability to deliver service that can justifiably be trusted by users.
8. **Usability**: usability is the ease with which a user can learn to operate, prepare inputs for, and interpret the outputs of the system or component. The usability requirements include: well-structured user manuals, informative error messages, help facilities; well-formed graphical user interfaces.

## 3.2.3 Hardware requirements

For effective and efficient performance of the project, certain hardware requirement must be met which are as follows: Android-powered mobile device with GPS and Internet functionality (minimum of 1 GB RAM, 8 GB internal memory and 1.5 GHz speed) alongside a personal computer with GPS and Internet functionality (Ethernet, minimum 2 GB RAM, 500 GB external storage and 2.5 GHz speed).

## 3.2.4 Software requirements

For flexible and effective use of the system via the internet, a network operating system must be running on the network server, a windows operating system for the client personal computer with a browser or alternatively an android OS installed on a mobile device.

# 3.3 System Analysis and Design

System analysis is the process of gathering and interpreting facts, diagnosing problems, and using the information to recommend improvements to the system. System design involves the analysis and configuration of the necessary hardware and software components to support a solution’s architecture. In order to solve the problem, it needed to be broken down into a set of interacting smaller problems whereby each of these smaller problems was decomposed into even smaller problems, until after enough iteration, each problem could be solved on its own.

Each decomposition provided a set of components thus providing a need to decide what those components are and how they fit together in the activity of system design. The design stage was required for transforming the detailed requirements of the definition stage into a complete, detailed specification of the system. Some of the most significant activities of this stage include:

1. Identification of all required data that was needed in managing the reported emergency information system and performing all the necessary activities that are required by the users of the system;
2. Representation and documentation of all related entities that exist in the emergency management system was performed;
3. Designing the components of the system using the unified modelling language (UML) diagram which showed the relationship that exists between entities in the emergency information system;the database structures, inputs, outputs, internal processing, manual procedures, system interfaces, technical environment, and overall system architecture;
4. Conducting walkthroughs of the design to ensure that it is programmable and technically complete; and
5. Beginning development of approaches to user support and system maintenance afterwards.

## 3.3.1 Use case diagram

The use case diagram was used to describe functions provided by the system that yielded a visible result for various actors that participated in the system use, such as the system administrator, system user who reports the emergency from their location and the emergency response officer. The identification of actors and use cases resulted in the definition of the boundary of the system that is, in differentiating the tasks accomplished by the system and the tasks accomplished by its environment. In the use case diagram, the actors are placed outside the boundary of the system, whereas the use cases are inside the boundary of the system. This projected the flow of activities and interaction of the emergency information system.

Following is a description of the identification of the system users alongside the various activities that can be performed by each. The use-case diagram shown in Figure 3.1 shows a description of the actors alongside their respective activities using the proposed system. The proposed system will have three (3) sets of users

* 1. **The System Administrator** – responsible for storing information about the location of hospital/emergency centres and for creating system users belonging to their respective medical centres.The primary responsibilities of the system administrator are to:

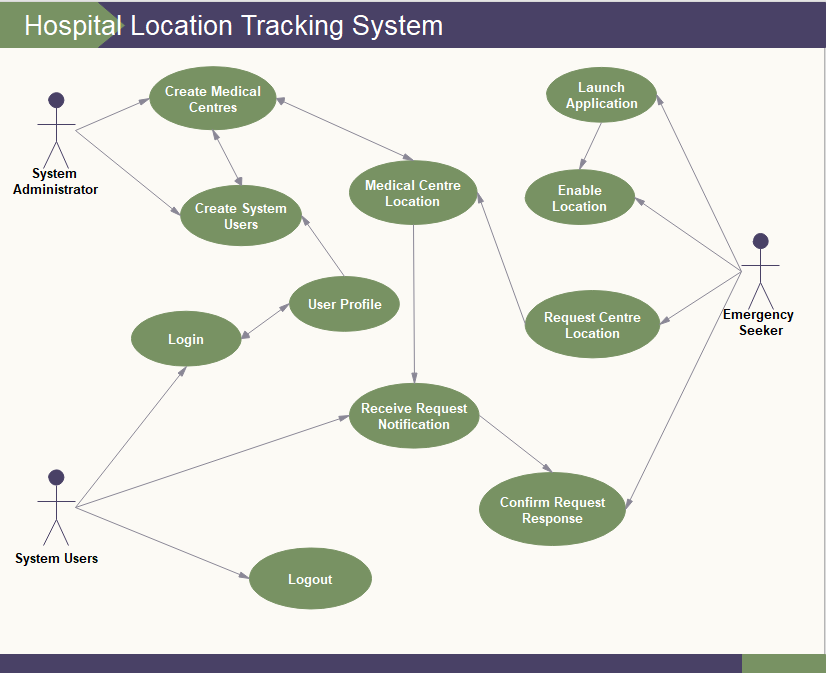


Figure 1 Use case diagram showing user activities

* + 1. Create the profiles for newly registered medical centers based on their respective locations;
    2. Create the profiles for emergency response officers within their respective medical centers;
    3. Manage the profiles of registered medical centers and their respective staff; and
    4. View information stored about medical centers created, users created and emergency events reported.
  1. **The System users** – responsible for responding to requests for emergency services made by the emergency seekers. Their primary responsibilities are to:
     1. View reported emergencies reported by emergency seekers;
     2. Respond to the reported emergency;
     3. Send an ambulance dispatch to respond to the emergency reported;
     4. Store information about the reported emergency.
  2. **Emergency Seekers** – required to search for nearby medical emergency centres using a push notification request. Their primary responsibilities are to:
     1. Search for nearby medical centers within their respective location;
     2. Send a notification of emergency report to nearby medical centers; and
     3. Communicate with emergency response team dispatch upon responding to reported emergency.

## 3.3.2 Sequence diagram

This diagram was used to describe the timings of operations performed by the users of the emergency response information system for locating nearby medical centers (hospitals). Figure 3.2 shows the sequence diagram for the activities that are to be performed by the emergency seeker while Figure 3.3 shows the sequence diagram for the activities performed by the emergency response personnel using the system.

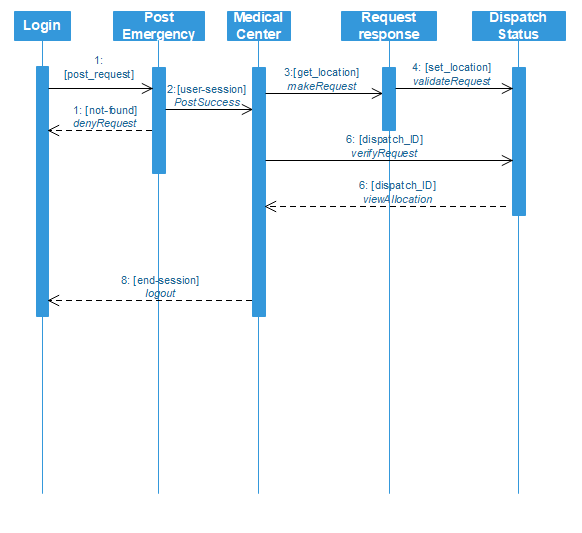
****

Figure 2 Sequence diagram of emergency seeker’s request

**Figure 3.2:** Sequence diagram of emergency seeker’s request

According to Figure 3.2, the emergency seeker is required to open the application on his mobile device or browser in order to access the system. Upon logging into the system, the user can send a notification requesting for emergency response from medical centers located around its location. The system will request the user to activate his location services following which the application will search the database for any medical center which is located around its location. The system will only look for medical centers which have been registered to the system. Upon successfully locating the medical center, the emergency response officer is notified of the location of the emergency seeker and a notification is sent to the emergency seeker that emergency response is on its way to their location.

According to Figure 3.3, the emergency response officer is required to log into the system using their username and password. This is required for restricting access to unauthorized users of the system who may be masquerading as emergency response officer. Following successful log into the system, the user can wait for notification of emergency to be sent by any emergency seeker. Once a notification for emergency has been sent by an emergency seeker, the emergency response officer forwards the request to the dispatch unit who then respond to the request made. Once the dispatch has responded to the reported emergency, the location of the emergency is sent to the dispatch and they can track the location of the emergency reported via the digital map. A response request status confirmation is sent to the emergency seeker in order to notify them of a dispatch on transit on transit to their location.

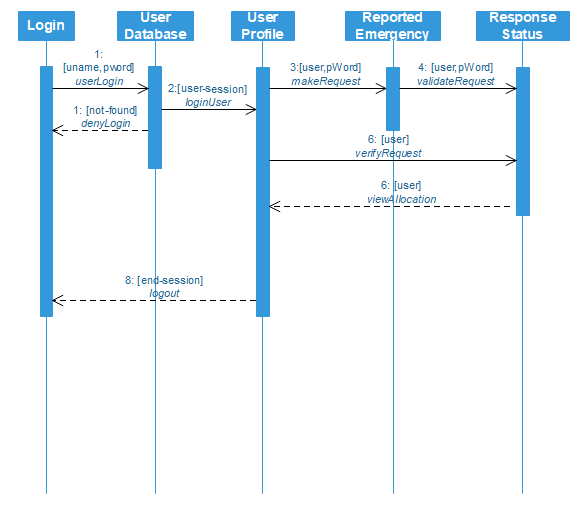
****

Figure 3 Sequence diagram of emergency response officer

## 3.3.3 Class Diagram

This diagram was used to identify the various types of classes, objects and their respective instances alongside their relationships within the context of the proposed system. The system has three main classes, namely: the actors, location and activity classes. Figure 3.4 shows a diagram of the class diagram of the emergency response information system.

According to the diagram, the parties have a number of categorizations, namely: the administrator, web user and the mobile user. The mobile and web users can be represented as the emergency seeker or the emergency response officers who is providing services to the emergency seeker. Also, the parties have a number of activities that are tied to each of them depending on the role of the party (or system user). In addition, the parties have their respective location which can be used to communicate their respective location to each other depending on the circumstance. For example, the location of the emergency seeker will allow the emergency response team to locate the emergency seeker upon responding to their response.

The location component can be used to describe the physical location defined by the address, electronic information defined by the email or phone number and by the spatial coordinates of the location. For the activity classes, some of the parties have the ability to view a set of activities taking place on the system such as the s=emergency response officer or the dispatch team responding to the emergency response request made. The administrator can create medical centers and emergency response officer and thus would also be able to view the list of registered officers and medical centers created on the system. Finally, some parties will be able to view maps, such as the emergency seeker; who would be able to view the location of the medical center responding to their emergency request alongside the emergency response dispatch who would be able to view the location of emergency seeker.

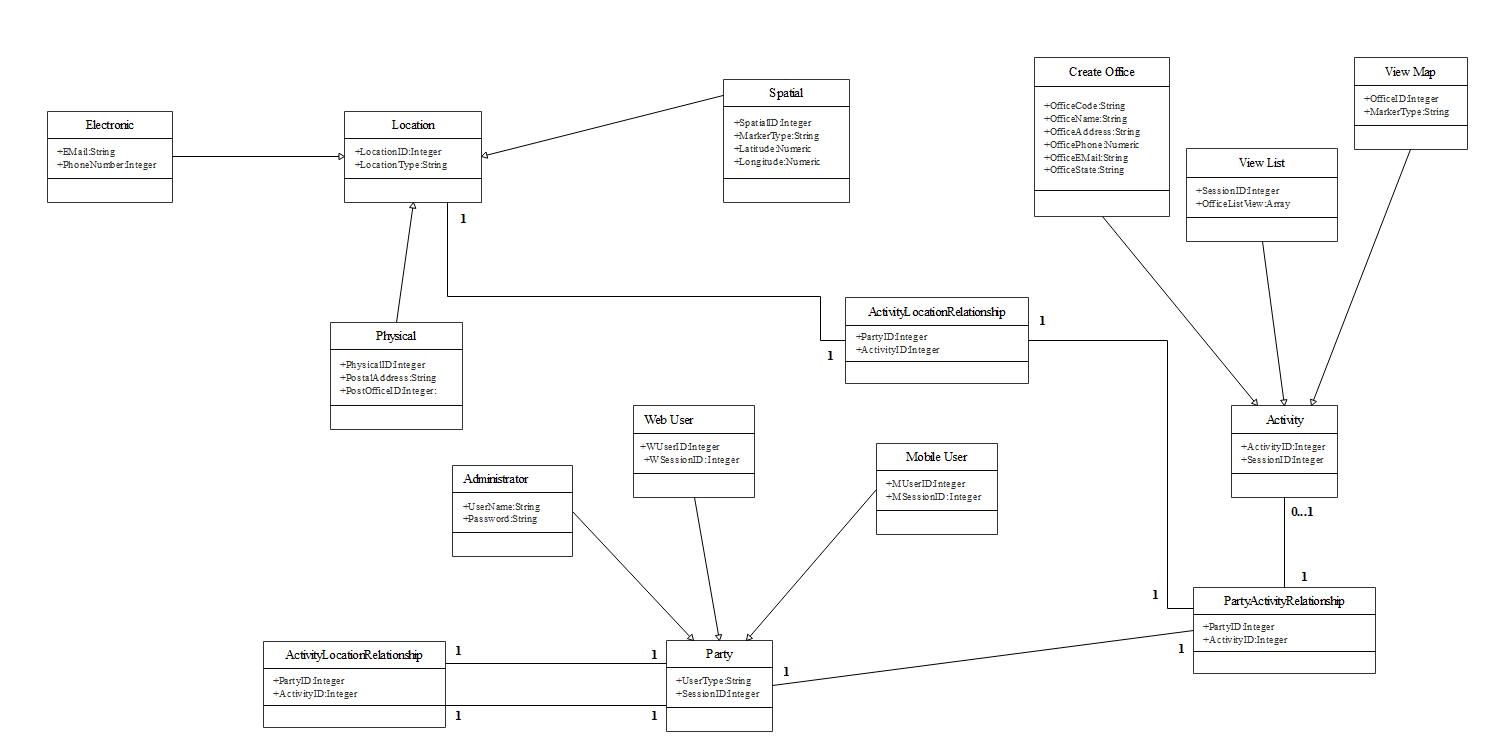


Figure 4 Class diagram of the interaction between actors, locations and activities of the emergency response system

## 3.3.4 System architecture

Figure 5 shows a conceptual diagram of the architecture of the spatial-based emergency response information system. The system architecture of the system is an n-tier architecture that consists of a client-side and a server side. The client side provides an interface for the primary users of the system to interact with using their system browser over the Internet. The client-side provides access to system users using various hardware consisting of portable devices (e.g., smartphones, tablets, laptops etc.) alongside personal computers via either LAN or wireless LAN networks. On the server side, a proxy server was required for hiding information about the IP addresses of system users so as to protect user information from system intruders. The information about the system is provided to users via the application server which is managed by the system administrator.

The system administrator uses the Internet to access the application server via which information about authorized users are provided to the system. However, data stored on the system were separated from the application server by storing them on the database server. Thus, every request for information provided to and retrieved from the system by users via the application server were managed strictly by the database server. Also, the Google map server which was accessed online was required for accessing the Google Map API needed for mapping the location of reported emergencies based on the coordinates of the location of the emergency seeker. The information about the coordinates were provided based on the longitude and latitude of the required location. In general, interaction between the severs located at the

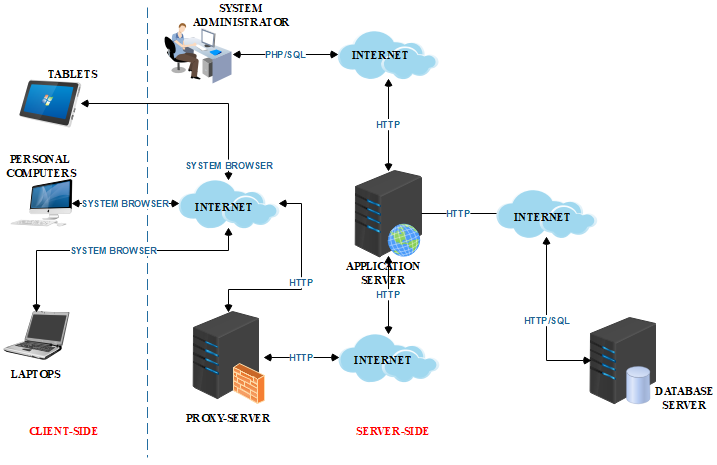
****

Figure 5 System architecture of emergency response system

server side were guaranteed using the hypertext Transfer Protocol (HTTP) for transferring requests made from one server to another during information processing.

# 3.4 System Development Tools

Implementation tools are tools needed to carry out the development of the emergency response information management system. The tools include: Cascading Style Sheet (CSS), Hypertext Pre-processor (PHP), Structured Query Language (SQL), and Google Map API.

## 3.4.1 Cascading Style sheet

Cascading Style Sheets (CSS) is the language used to tell computers how designs should look on the Web. With CSS, details can be specified (such as widths, heights, colors, margins, padding, borders, backgrounds, and type styles). CSS along with JavaScript is used by most websites to create visually engaging webpages, user interfaces for web applications and mobile applications. The CSS syntax is made up of three parts: a selector, a property and a value: selector {property:value}e.g.,p {color:blue} which indicates that the color of all paragraphs in that document where the CSS is used will be blue. CSS saved a lot of work in the implementation of the blood bank information management system as it controlled the layout of multiple web pages all at once where the external style sheets are stored in CSS files.

## 3.4.2 PHP

PHP is a scripting language designed specifically for use on the web. It has features to aid web developers in programming the tasks needed to develop dynamic web applications. PHP is a powerful tool for making dynamic and interactive web pages. PHP is an embedded scripting language when used in web pages. PHP is a widely used open-source general-purpose scripting language that is especially suited for the development of web-based spatial distributive fire outbreaks information management system. PHP is a module found in Apache HTTP server and its scripting engine can be built into the web server itself thereby leading to faster processing, more efficient memory allocation, and simplified maintenances. Many dynamic websites require a backend database. The database can contain information that the web pages display to the user, or the purpose of the database might be to store information provided by the user.

## 3.4.3 SQL

Structured Query Language (SQL) is a standard language for accessing and manipulating databases. Users communicate using Structured Query Language (SQL), which is a standard computer language understood by most database management systems. To make a request that MySQL can understand, an SQL statement is built and sent to the MySQL server. Some of the standard SQL commands are "Select", "Insert", "Update", "Delete", "Create", and "Drop". Examples of some common relational database management systems that use SQL are: Oracle, Sybase, Microsoft SQL Server, Access, Ingres, etc.

## 3.4.4 Google map API

A geographic information system is an organized collection of computer hardware and software, geographic and tabular data, and personnel and knowledge designed to capture, store, manipulate, update, analyze and display spatial data. The use of geographic information systems (GIS) as a major technology for map storage, production and dissemination has been fully recognized. Google Map was launched by Google in 2005 to allow web developers to integrate it into their web applications for free. Google Maps API allows the display of maps on the emergency response information system thereby allowing users to view the location of the nearest medical centers and to allow emergency response dispatch to locate emergency seeker.

**CHAPTER FOUR**

**IMPLEMENTATION AND DESIGN**

**4.1 Introduction**

This section explains the implementation design of the project. The system is a web application developed using HTML, CSS, Java Script and PHP. The database is created using MySQL.

# 4.2 Implementation Design

1. **User Login:** This interface is design for the user to login to the system. It allows user who have created to login with their username and password. And the **Register Now** allows user to redirect to the register page to create and account.

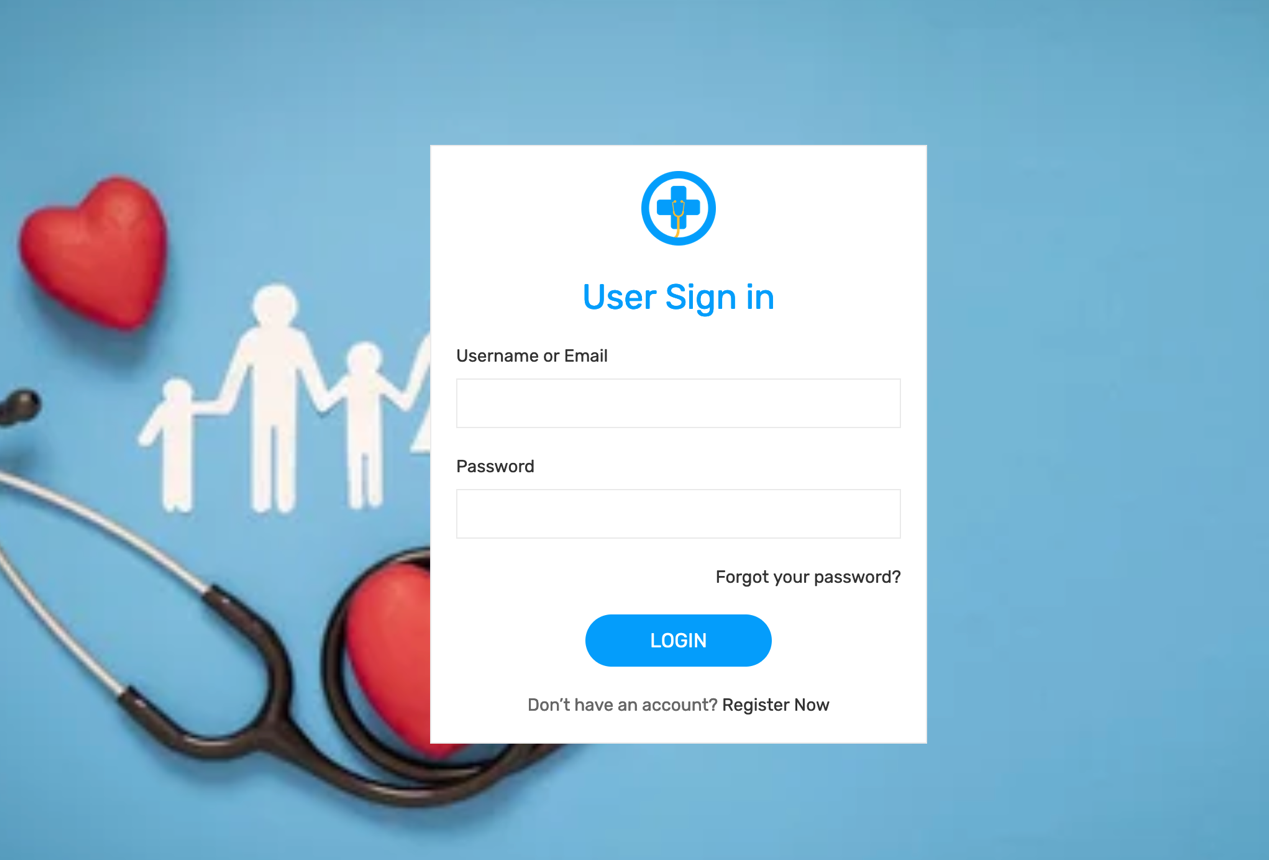


Figure 6 User Login Interface

1. **User Registration:** This interface allows new user to create an account so as to gain access to the system. It collects information such as the username, email address, password and phone number of the new user.

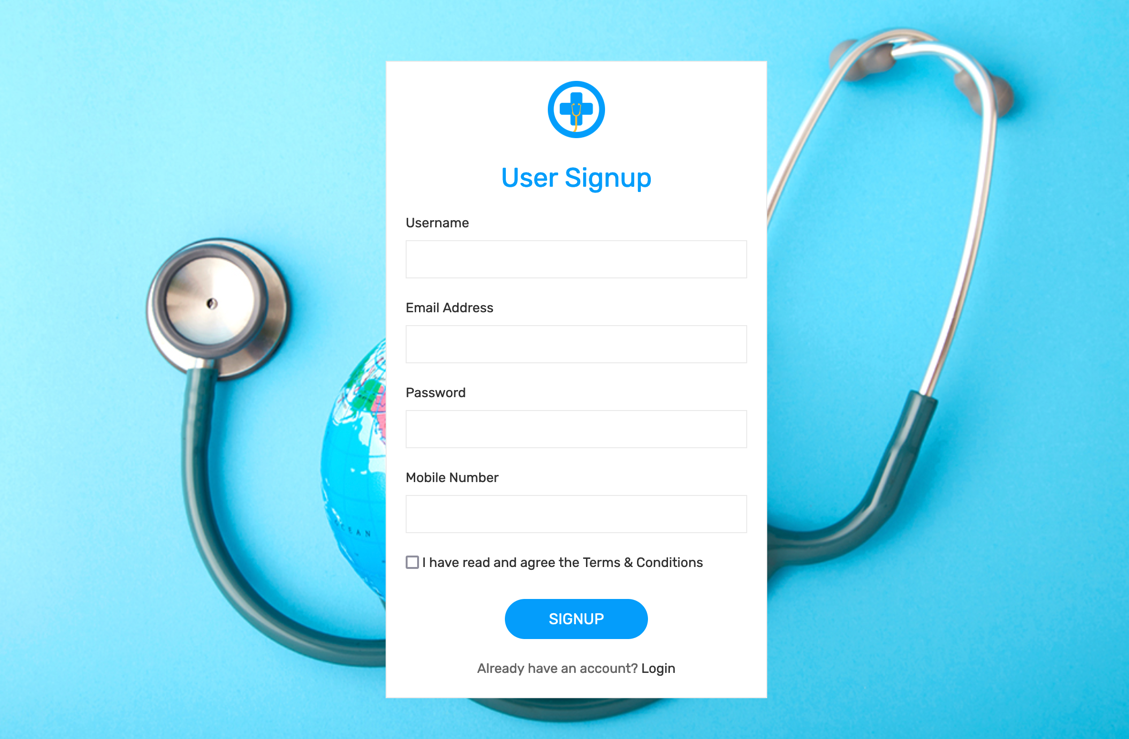
****

Figure 7 User Registration Interface

1. **User dashboard:** After a successful login of the user to the system. The user is redirected to the dashboard. The system automatically detects the user location and provide the nearest hospital to that location. With other information about the hospital facilities. The user can also click the view direction to see the direction from the user location to the hospital and the emergency complaints allows user to send a request to the admin.

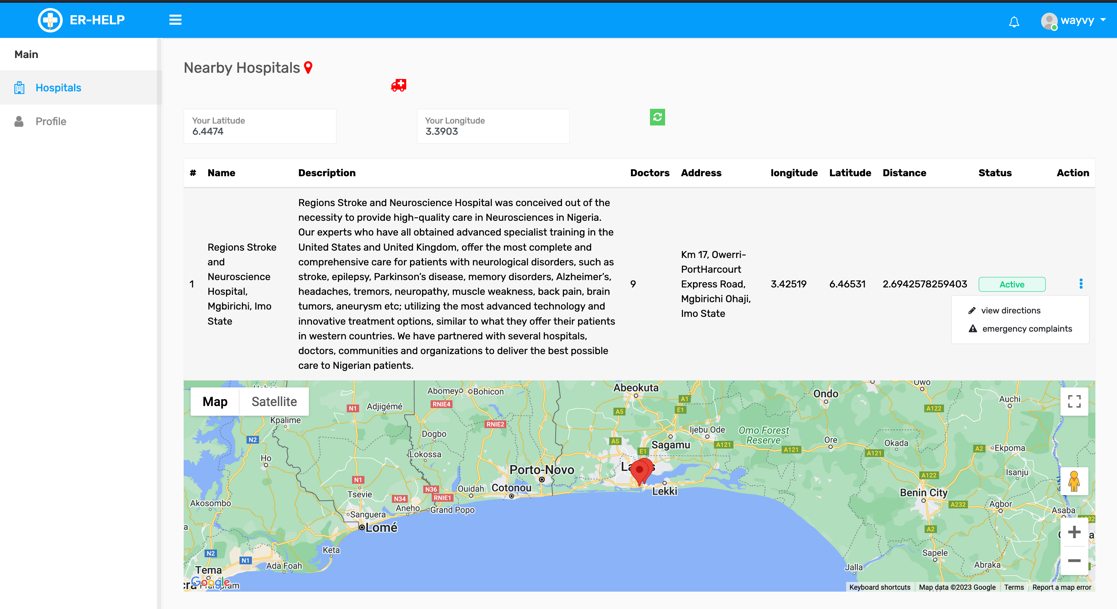


Figure 8 User dashboard

1. **User direction:** After clicking the view direction button, it allows the user to get the direction from his/her location to the nearest hospital identified.

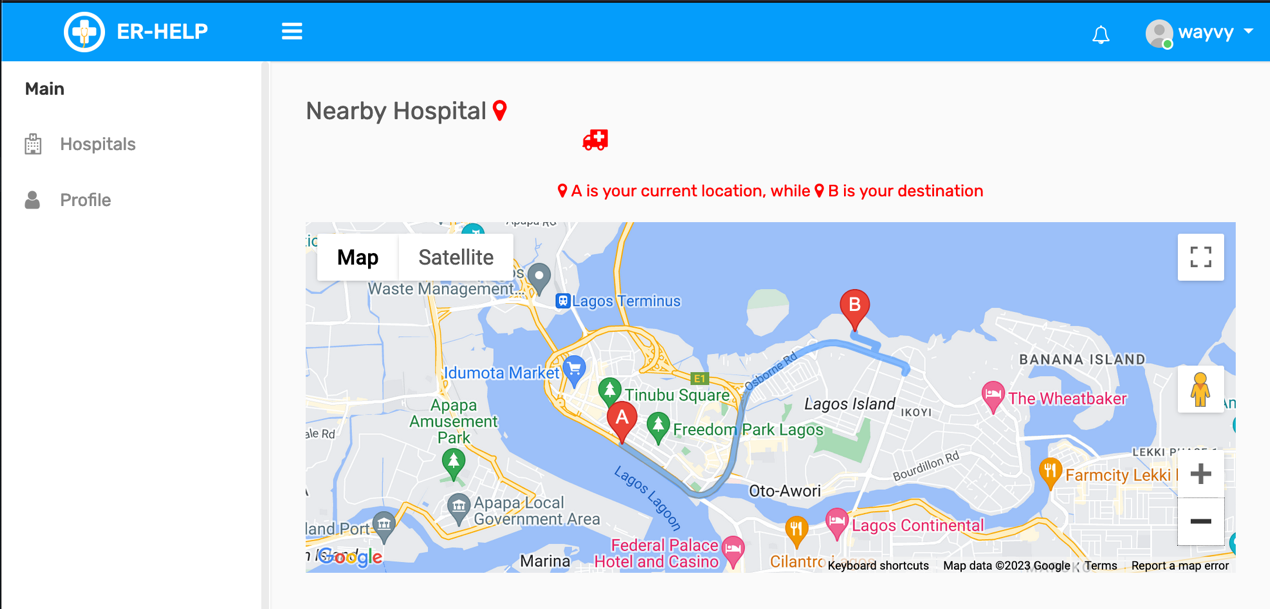
****

Figure 9 Get direction

1. **Send Request:** This form is created to allow the emergency seeker send a notification to the admin. The notification consists of the nearest hospital to the emergency seeker and his/her personal information. The admin can the put a call through to the nearest hospital to notify them an emergency will be arriving soon to put the staff of the hospital on standby.

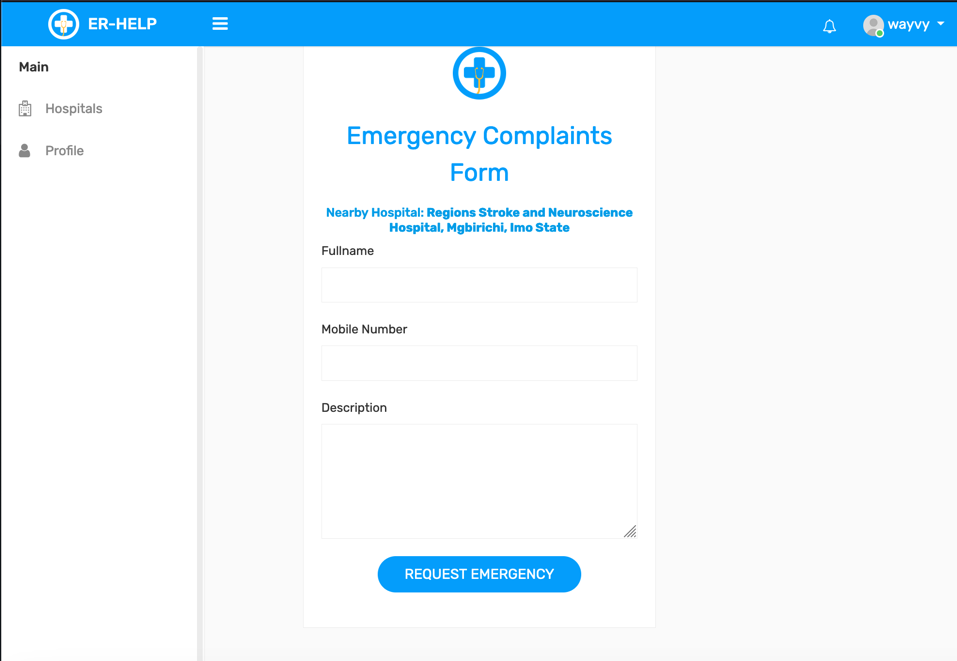
****

Figure 10 Emergency Screen

1. **Admin Login:** This interface is created for the admin to gain access to the system.

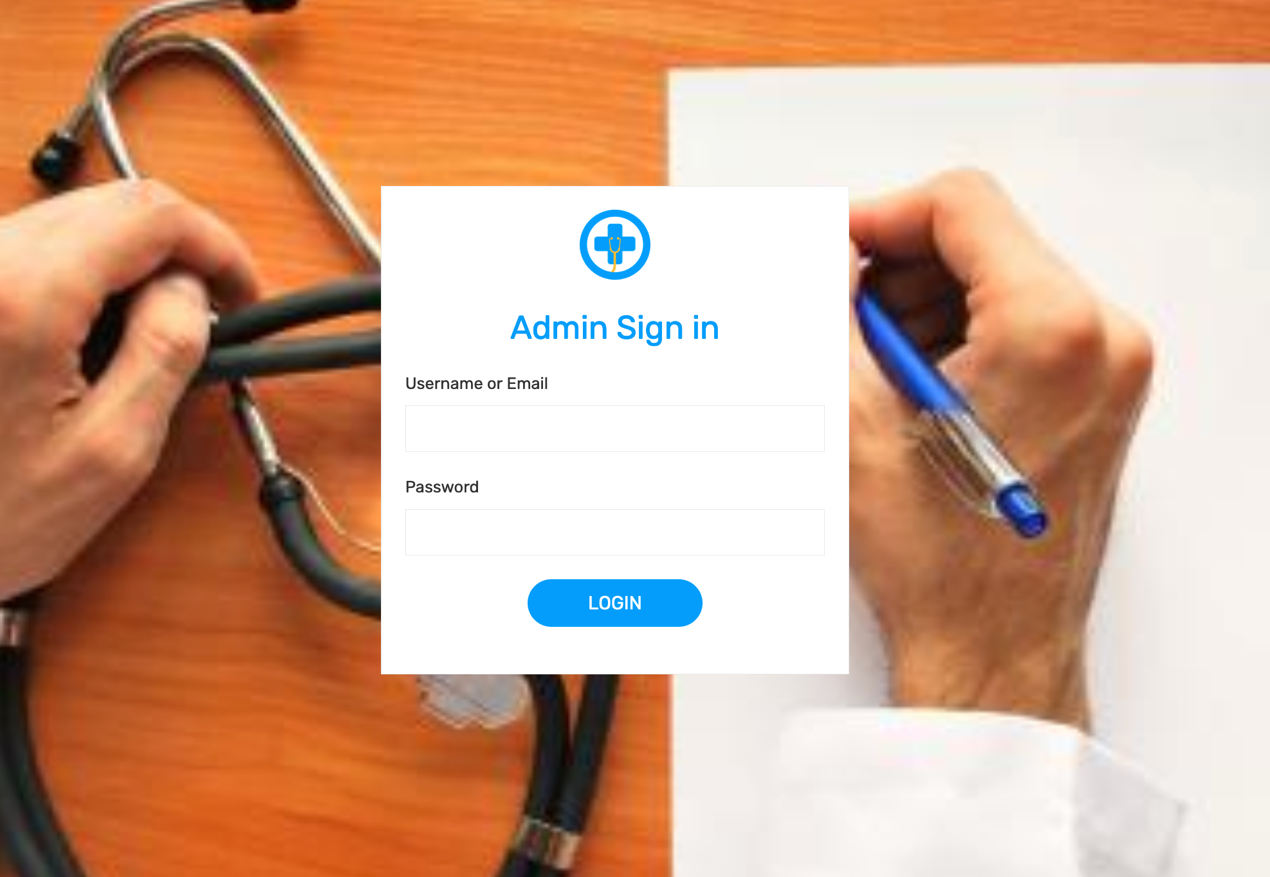
****

Figure 11 Admin Login

1. **Admin dashboard:** This dashboard shows the administrative operation of the system, such as adding hospital to the system, viewing all the users that have created an account and also list of emergencies sent to the system.

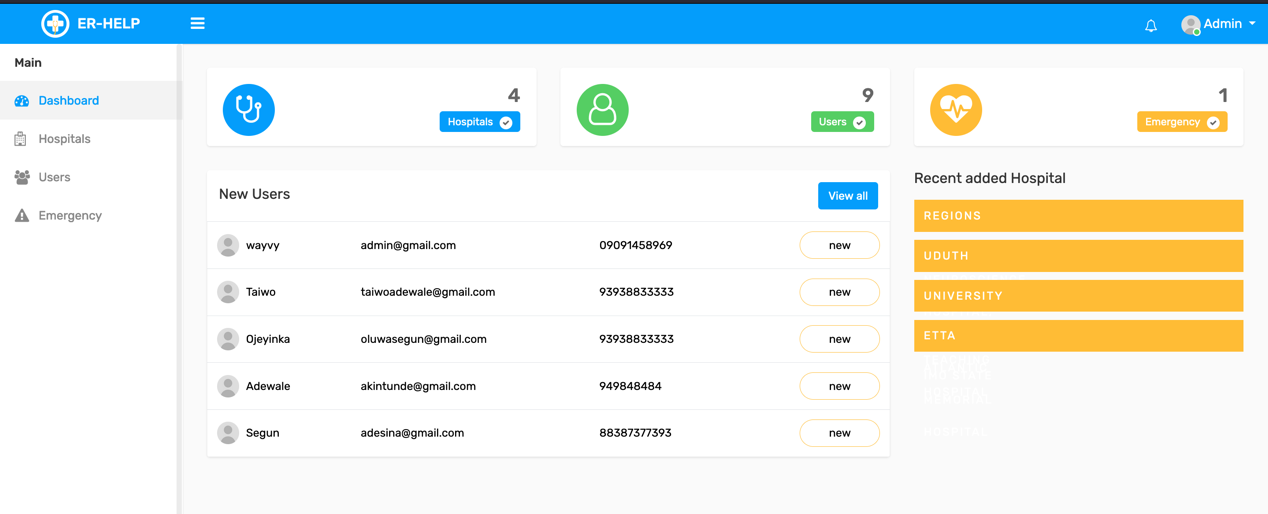
****

Figure 12 Admin Dashboard

1. **Hospital:** This section allows the admin to see the list of hospital that has been added to the system. The **Add New** button in the top right side of the screen opens up an interface to add new hospital information.

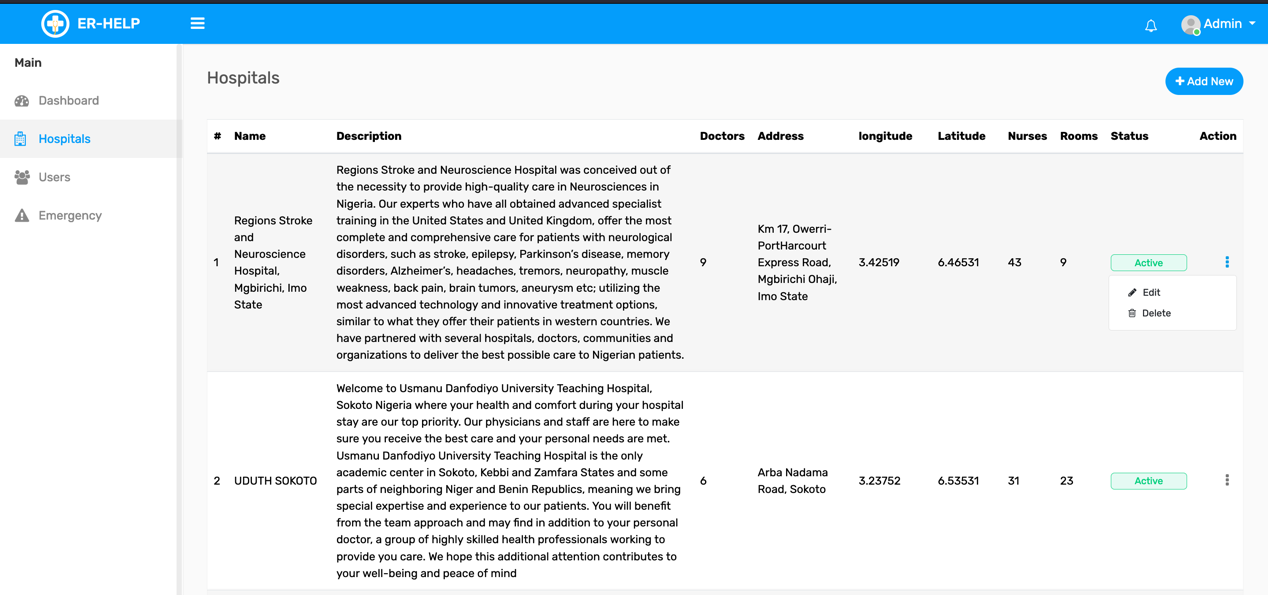
****

Figure 13 Hospital List

1. **Add New Hospital:** This interface collects information about the hospital and save it into the database.

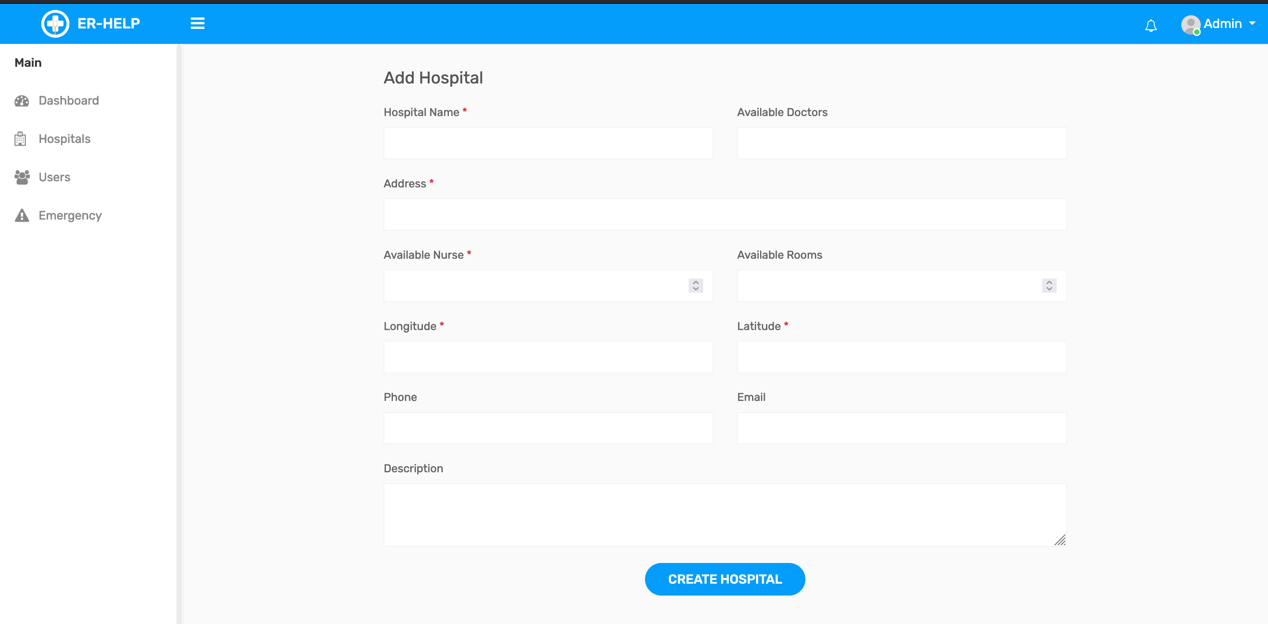
****

Figure 14 Add New Hospital

1. **User List:** This section shows the list of user that have created account on the system.

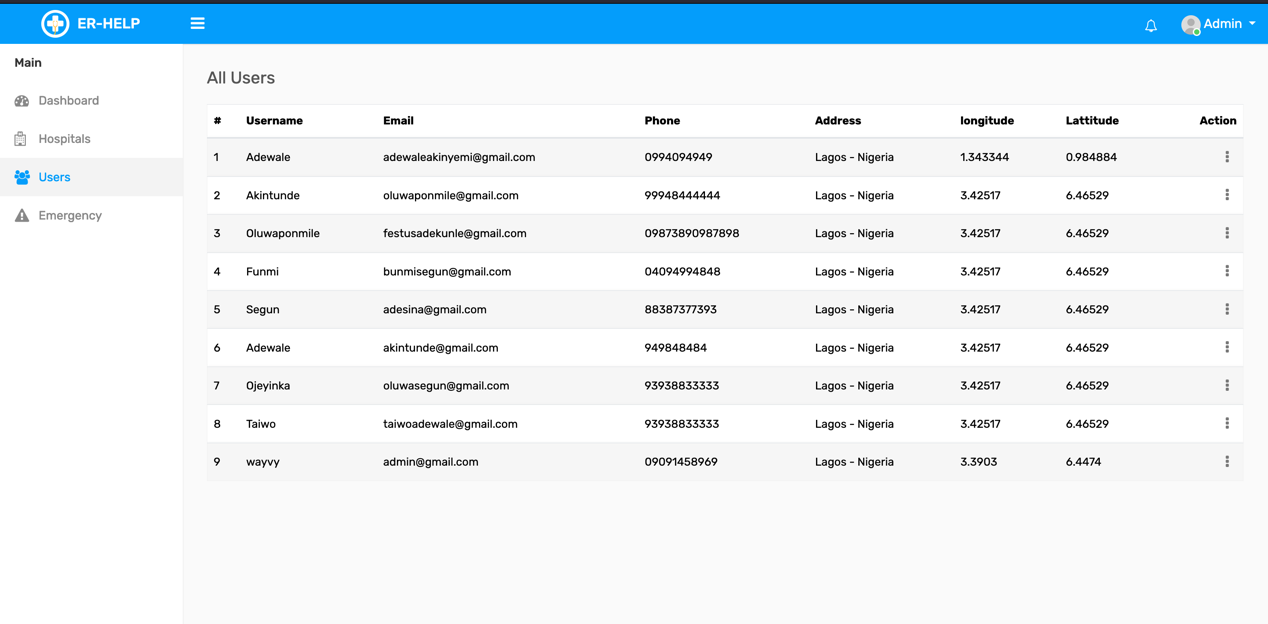
****

Figure 15 User list

1. **Emergency list:** This section shows the list of the emergency request being sent to the admin. The admin can click on any of it and view the details more.

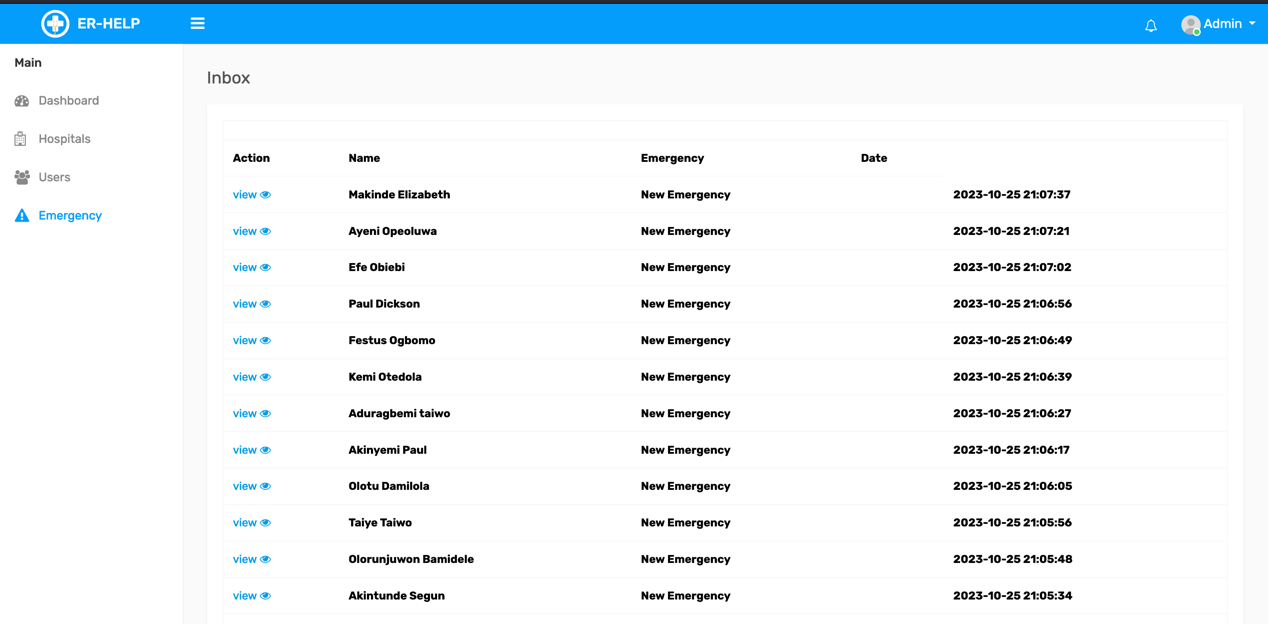
****

Figure 16 Emergency List

1. **Detailed Emergency:** This interface provide a detailed information about the emergency request sent to the admin for further processes.

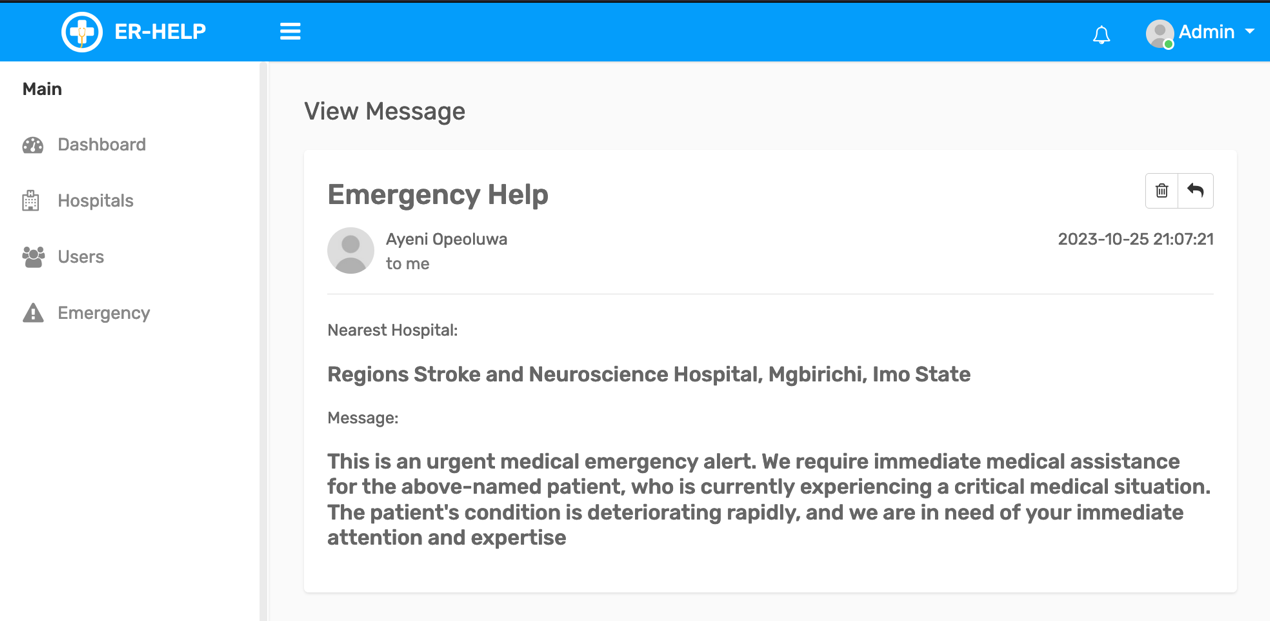
****

Figure 17 Detailed Emergency

# 4.3 Database Design

**Admin Database Table**

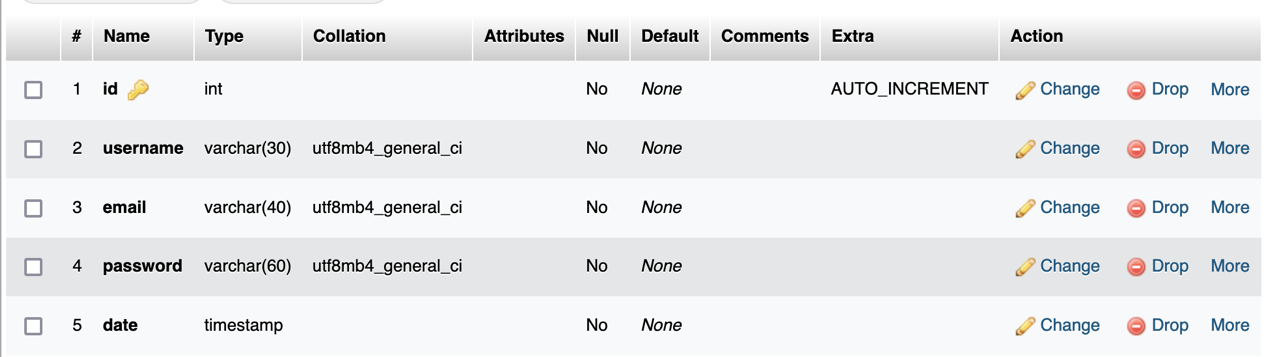


Figure 18 Admin table

**Complaints Database Table**

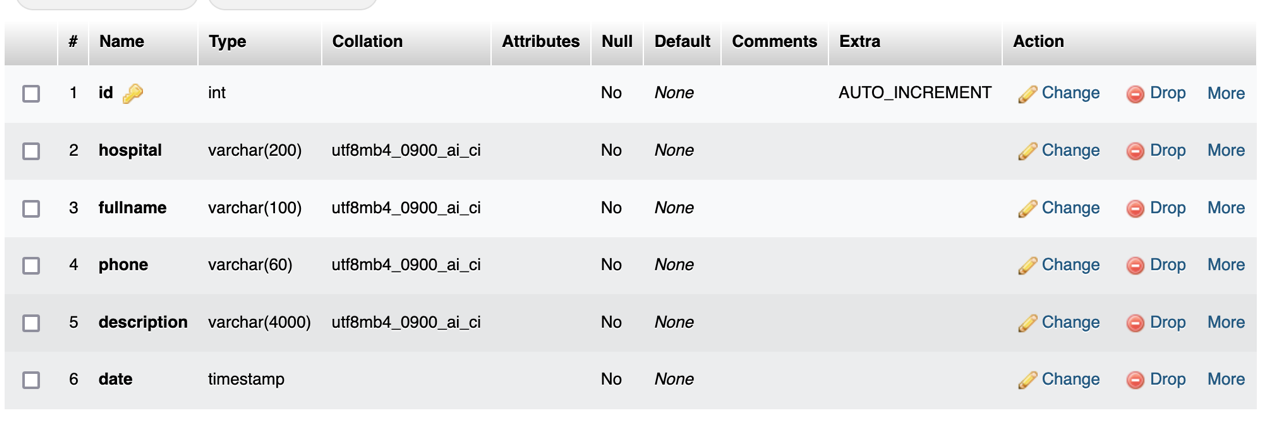
****

Figure 19 Complaint Table

**Hospital Database Table**

****

Figure 20 Hospital database table

# CHAPTER FIVE

# SUMMARY AND CONCLUSION

# 5.1 Summary

The development of a Location-Based System for Tracking Hospitals in the case of Medical Emergencies represents a significant step forward in addressing critical challenges in the accessibility and response to medical emergencies. This study has systematically outlined the methodologies, requirements, and tools employed in the development of this vital system, which holds the potential to save lives and improve emergency medical services. The study commenced by recognizing the pressing need for quick and efficient access to medical services, especially during life-threatening emergencies, in an era dominated by technology and smartphone applications. It highlighted the specific challenges that patients often face in locating medical assistance, particularly ambulance services, and underscored the complexities of navigating urban environments. The introduction of Location-Based Services (LBS), Geographic Information Systems (GIS), and various technological trends further reinforced the importance of technology in this context. The study identified a critical gap in existing literature, particularly in the context of south-western Nigeria, where the tracking of nearby medical centers had not been adequately addressed. This served as the catalyst for the development of a system designed to assist users in locating medical centers during medical emergencies. The specified objectives, both functional and non-functional, were outlined to guide the development process. The system's architecture, as described in the study, underlines the importance of client-server interaction, where the client-side interface allows users to interact with the system, and the server-side manages data processing and storage. The integration of tools such as CSS, PHP, SQL, and Google Map API was emphasized as crucial for the effective implementation of system functionalities. The study also delved into the use of various UML diagrams, including use case diagrams, sequence diagrams, and class diagrams, to illustrate the system's functionalities and relationships between entities and components. These diagrams played a crucial role in structuring and visualizing the development process. In practical terms, the proposed system offers a three-tiered user structure, including system administrators, system users, and emergency seekers, each with defined roles and responsibilities. This user structure allows for a streamlined interaction between emergency seekers and medical centers, ultimately leading to a faster response to medical emergencies. Overall, the development of this Location-Based System for Tracking Hospitals in the case of Medical Emergencies represents a significant contribution to public health and safety. By leveraging modern technology and GIS, this system is poised to improve the efficiency of emergency medical services, reduce response times, and, most importantly, save lives. It embodies the intersection of technology and healthcare in an increasingly interconnected world, where quick access to medical services during emergencies is paramount. As technology continues to advance and urbanization increases, the system stands as a beacon of hope, offering a lifeline to individuals in dire need of medical assistance. Through a systematic approach to development and a comprehensive understanding of user requirements, the system is positioned to make a lasting impact on the healthcare landscape, ensuring that medical help is never too far away in times of crisis.

# 5.2 Conclusion

In conclusion, the development of the Location-Based System for Tracking Hospitals in Medical Emergencies marks a pivotal advancement in enhancing emergency medical services. By harnessing technology and Geographic Information Systems, this system facilitates rapid access to healthcare resources during critical moments. It addresses the challenges of locating medical centers and expedites response times. The study's meticulous approach, including defining functional and non-functional requirements, utilizing UML diagrams, and selecting appropriate development tools, ensures a robust and efficient system. Ultimately, this innovation promises to save lives, bridging the gap between technology and healthcare, and ensuring that medical assistance is readily available when it matters most.

# References

Abdullah, K.& Al-Faris, K. (2013). Attitude to Blood Donation among Male Students at King Saud University. *Journal of Applied Hematology 12*(3): 1 – 13.

Akkas, K.M., Jahan, I., Islam, A. & Parvez, M.S. (2015). Blood Donation System. *American Journal of Engineering Research (AJER) 4*(6): 123 – 136.

Almetwally, M.M., Ahmed, E.Y. & Gamal, A. (2014). A Framework for a Smart Social Blood Donation System Based on Mobile Cloud Computing. Retrieved from <https://arxiv.org/ftp/arxiv/papers/1412/1412.7276.pdf> on March 12, 2018.

Aloudat, A., Michael, K., & Yan, J. (2007). *Location-Based Services in Emergency Management from Government to Citizens: Global Case Studies.* Faculty of Informatics.

Applewhite, A. (2002). What Knows Where You Are? *Pervasive Computing, 1*(4), 4-8.

Bhatnagar, M. and Singh, P.K. (2013). Research Methodology as SDLC Process in Image Processing. *International Journal of Computer Applications 77*(2):43 – 45.

Bloch, E.M., Vermeulen, M. & Murphy, E. (2012). Blood Transfusion Safety in Africa: A Literature Review of Infectious Disease and Organizational Challenges. *Journal of Transfusion Medical Review 2*: 1 – 4.

Bridwell, S.A. and Miller, H.J. (2016). Location-Based Services. n: Liu L., Özsu M. (Eds.) *Encyclopedia of Database Systems*. Springer, New York, NY. <https://doi.org/10.1007/978-1-4899-7993-3_494-2>

Brown, A., & Sturza, M. (1995). *Washington, DC Patent No. U.S. Patent No. 5,379,224.*

Bruno, B. (2010). Blindspots: The Many Ways We Cannot See. New York: Oxford University Press. ISBN 978-0-19-539426-9.

Burke, L. (2012). *Parents tracking Children with GPS*. Retrieved from Yahoo News Australia: <http://au.news.yahoo.com/thewest/a/-/newshome/13495616/parents-tracking-children-with-gps>

Chan, N., & Lars, H. (2003). *Introduction to Location-Based Services.* Lund: Lund University GIS Centre.

Chau, M., Cheng, E. & Chan, C.W. (2010). Data Analysis for Healthcare: A Case Study in BloodDonation Center Analysis. In *Proceedings of Sixteenth Americas Conference on Information Systems (AMICS):* 1 – 10.

El-Masri, S., & McDaniel, P. (n.d.). Android Market Reconstruction and Analysis. *IEEE Military Communications Conference*, (pp. 300-305). Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6956775

Frentu, F.D. and Briscoe, A.D. (2008). A Butterfly Eye’s View of Birds. *Journal of BioEssays 30*(11-12): 1 – 12. <https://doi.org/10.1002/bies.20828>

Ghadirli, H., & Rastgarpour, M. (2012). An Adaptive and Intelligent Tutr by Expert Systems for Mobile Devices. *International Journal of Managing Public Sector Information and Communications Technology, 3*(1), 1-9.

Gholamhosseini, L., Sadoughi, F., and Safaei, A. (2019). Hospital Real-Time Location System (A Practical Approach in Healthcare): A Narrative Review Article. *Iran Journal of Public Health, 48*, 593-602.

Gonzalez-Alwarez, J.A., Ortiz-Solis, G., Ramirez-Cordero, H., Carrillo-Santoscoy, J.P., Ocampo-Chavarria, A., Vejar-Aguirre, T., Jauregui-Ulloa, E. (2016). Evaluation of an electronic Information System for the Monitoring of Chronic Diseases. *Mexican Journal of Cardiology 27*(2): 77 – 86.

Goodrich, R. (2021). Location-Based Services use Real-Time Geodata from a Smartphone to provide Information, Entertainment or Security. Retrieved from <https://www.businessnewsdaily.com/5386-location-based-services.html> on June 4, 2021.

Grebmann, B., Klimek, H., & Turau, V. (2010). Towards Ubiquitous Indoor Location Based Services and Indoor Navigation. *7th Workshop on Positioning Navigation and Communication* (pp. 107-112). IEEE.

Hand, A., Cardiff, J., Magee, P., & Doody, J. (2006). An Architecture and Development Methodology for Location-Based Services. *Commerce Research and Applications, 5*(3), 201-208.

Hurson, A., & Gao, X. (2009). Location-Based Services. *Encyclopedia of Information Science and Technology*, 2456-2461.

Idhoko, K.E. and Ojaiko, J.C. (2014). Integration of Geographical Information Systems (GIS) and Spatial Data Mining Techniques in Fight against Boko Haram Terrorist in Nigeria. *International Journal of Science and research (IJSR)* 6(14): 1932 – 1934.

Ivan N. & Godfrey O. O. (2016). A Secure Application for Information Sharing in Organizations: A Case Study of Kabale District Local Government. International Journal of Computer (IJC). ISSN 2307-4523 (Print & Online). Volume 21, No 1, pp 64-77

Khan, A.R. & Qureshi, M.S. (2009). Web-Based Information System for Blood Donation. *International Journal of Digital Content technology and its Applications 3*(2): 137 – 142.

Land, M.F. and Fernald, R.D. (1992). The Evolution of Eyes. *Annual Review of Neuroscience 15*: 1 – 29.

Lee, J., Chuah, Y., & Chieng, K. (2013). Smart Elderly Home Monitoring System with an Android Phone. *International Journal of Smart Home, 7*(3), 17-32.

Macqueen, S., Bruce, E. &Gibson, F. (2012). *The Great Ormond Street Hospital Manual of Children's Nursing Practices*. John Wiley & Sons.

Madadipouya, K. (2014). An Examination and Report on Potential Methods of Strategic Location-Based Service Applications on Mobile Networks and Devices. *International Journal of Managing Public Sector Information and Communication Technologies (IJMPICT), 5*(3), 25-31. doi:10.5121/ijmpict.2014.5303

Martin-Gomez, C., Vergara-Falces, J., & Elvira-Zalduegui, A. (2015). Geographic Information System Software Application developed by a Regional Emergency Agency. *Case Studies in Fire Safety, 4*, 19-27.

Michael, K., Abass, R., Aloudat, A., & Al-Debei, M. (2011). The Value of Government mandated Location-Based Servoces in Emergencies in Australia. *Journal of Information Technology Research, 4*(4), 41-68.

Mod Sakriya, M., & Samual, J. (2015). Ambulance Emergency Response Application. *International Journal of Information System and Engineeering, 3*(2), 23-32. doi:10.24924/ijise/2015.11/v3.iss2/23.32

Osheroff J.A., Pifer E.A., Teich J.M., Sittig D.F. and Jenders, R.A. Improving outcomes with clinical decision support: an implementer’s guide. Chicago: Healthcare Information and Management Systems Society; 2005.

Pearce, J.(2006). Neighbourhoods andhealth: A GIS approach to measuringcommunityresource accessibility. *Journal of Epidemiology & Community Health 60*(5): 389–395.

Pelis, M. (2001). Taking Credit: The Canadian Army Medical Corps and the British Conversion to Blood Transfusion in WWI. *Journal of the History of Medicine and Allied Sciences 56*: 238– 277.

Rahman, A.A. and Zlatanova, S. (2006). Pre-Hospital Location-Based Services (LBS) for Emergency Management. In *Telegeomatics – Part II*, 49-58.

Rao, B., & Minakakis, L. (2003). Evolution of Mobile Location-Based Services. *Communications of the ACM, 46*(12), 61-65.

Rauth, P., Parab, P., Suthar, Y., Narwani, S. & Pandey, S. (2016). Blood Bank Management System. *International Journal of Advanced Computational Engineering and networking 4*(9): 27 – 31.

Saeed, A., Bhatti, M.S., Ajmal, M., Waseem, A., Akbar, A. and Mahmood, A. (2013). Android, GIS and Web Project, Emergency Management System (EMS) which overcomes Quick Emergency Response Challenges. In Rocha *et al.* (Eds.): *Advances in Information Systems and Technologies, AISC,* 269-278. Springer-Verlag, Berlin Heidelberg.

Sakriya, A.Z. and Samual, J. (2015) Ambulance Emergency Response Application. *International Journal of Information System and Engineering, 3*(2), 23-32.

Sanni, M. and Kaur, K. (2014). A Review of Open Source Software Development Life Cycle Models. *International Journal of Software Engineering and Its Applications 8*(3): 417 – 434.

Santosa, I., Romla, L. and Herawati, s. (2018). Expert System Diagnosis of Cataracts Eyes Using Fuzzy Logic. *Journal of Physics 953*: 1 – 6.

Shyam, S. &Santhanam, T. (2009). Classification of Blood Donors using Data Mining. In *Proceedings of the Semantic E-Business and Enterprise Computing*: 145 – 147.

Smiatek, G. (2005). SOAP Based web services in GIS/RDBMS Environment. *Journal of Environmental Modeling and Software 20*: 775 – 782.

Steiner, S., Neun, M., & Edwardes, A. (2006). *Foundations of Location-Based Services.*

Sumazly, S., Abdul, A. and Nurul, A.N. (2015). Development of a Blood Bank Management System. *Journal of Social and Behavioral Sciences 195*: 2008 2013.

Tatar, E. and Tomur, E. (2013). A Secure Development Framework for SaaS Applications in Cloud Computing. In *6th International Information Security & Cryptology Conference* held in Ankara Turkey: 338 – 342.

Ververidis, C., & Polyzos, G. (2002). Mobile Marketing using a Location-Based Service. *Proceedings of First International Conference on Mobile Business* (pp. 1-12). Prentice-Hall.

World health Organization, WHO (2011). Global database on Blood Safety. WHO Press

Zickuhr, K. (2012). *Three-Quarters if Smartphone Owners use Location-Based Services.* Pew Internet & American Life Project.

Zhou, Z., Rachir, S., Huiying, L., Xiangyu, C., Lixin, D., Damon, W.K.W., Chee, K.K & Tien, Y.W. (2014). A Survey on Computer Aided Diagnosis for Ocular Diseases. *BMC Medical Informatics and Decision Making 14*: 1 – 29.

# Source Code

<?php

//Require config config file

include\_once ('config.php');

$stmt = $mysqli->prepare("SELECT \* FROM settings ORDER BY id DESC LIMIT 1");

$stmt->execute();

$result = $stmt->get\_result();

$data = $result->fetch\_assoc();

$site\_name = $data['site\_name'];

//$site\_address = $data['site\_address'];

//$site\_phone = $data['site\_phone'];

//$site\_mail = $data['site\_mail'];

define('DS', DIRECTORY\_SEPARATOR);

/////site name

define('SITE\_NAME', $site\_name);

// define('SITE\_ADDR', $site\_address);

//define('SITE\_PHONE', $site\_phone);

/////App Root

define('APP\_ROOT', dirname(dirname(\_\_FILE\_\_)));

define('URL\_ROOT', '/emergency\_care');

define('URL\_SUBFOLDER','emergency\_care');

function secure($string){

$sec = htmlentities($string);

return $sec;

}

function user\_exist($username, $email, $table){

//Require Databse config file

require 'config.php';

//Sanitize function variables

$table = secure($table);

//Check if email exist

$stmt = $mysqli->prepare("SELECT \* FROM ".$table." WHERE username = ? || email = ?");

$stmt->bind\_param("ss", $username,$email);

if($stmt->execute()){

$result = $stmt->get\_result();

if ($result->num\_rows >0) {

return true;

}else{

return false;

}

}else{

die($mysqli->error);

}

}

function deleteF($del, $table){

require 'config.php';

$stmt = $mysqli->prepare("DELETE FROM ".$table." WHERE id='$del' LIMIT 1");

if($stmt->execute()){

return true;

}else{

return false;

}

$result = $stmt->get\_result();

return $result;

}

function getAdmin(){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("SELECT \* FROM admin");

$stmt->execute();

$result = $stmt->get\_result();

return $result;

}

function fetchrecord(){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("SELECT \* FROM ".$table."");

$stmt->execute();

$result = $stmt->get\_result();

return $result;

}

function char\_tbl($table,$created\_at\_month,$created\_at\_year){

require 'config.php';

$stmt = $mysqli->prepare("SELECT count(\*) AS total FROM ".$table." WHERE date\_format(date,'%m') = $created\_at\_month AND date\_format(date,'%Y') = $created\_at\_year");

$stmt->execute();

$res = $stmt->get\_result();

$re = $res->fetch\_assoc();

return $re["total"];

}

function char\_tbl\_($table,$created\_at\_year){

require 'config.php';

$stmt = $mysqli->prepare("SELECT count(\*) AS total FROM ".$table." WHERE date\_format(date,'%Y') = $created\_at\_year");

$stmt->execute();

$res = $stmt->get\_result();

$re = $res->fetch\_assoc();

return $re["total"];

}

function getAdminUsr($em){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("SELECT \* FROM admin where email = '$em' || username = '$em'");

$stmt->execute();

$result = $stmt->get\_result();

$data = $result->fetch\_assoc();

return $data['username'];

}

function getUsr($em){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("SELECT \* FROM users where email = '$em' || username = '$em'");

$stmt->execute();

$result = $stmt->get\_result();

$data = $result->fetch\_assoc();

return $data['username'];

}

function getUsrInfo($em){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("SELECT \* FROM users where email = '$em' || username = '$em'");

$stmt->execute();

$result = $stmt->get\_result();

return $result;

}

function getchkAdminUsr($username){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("SELECT \* FROM admin WHERE username = '$username'");

$stmt->execute();

$result = $stmt->get\_result();

$data = $result->fetch\_assoc();

$privilege = $data['access'];

if($privilege=='1'){

return true;

}else{

return false;

}

}

function getAdminUsrz($username,$email){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("SELECT \* FROM admin where username = '$username' || email = '$email'");

$stmt->execute();

$result = $stmt->get\_result();

return $result;

}

function getUsrz($username,$email){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("SELECT \* FROM users where username = '$username' || email = '$email'");

$stmt->execute();

$result = $stmt->get\_result();

return $result;

}

function getUsrz\_exist($username,$email){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("SELECT \* FROM users where username = '$username' || email = '$email'");

$stmt->execute();

$result = $stmt->get\_result();

if(mysqli\_num\_rows($result) > 0){

return true;

}else{

return false;

}

}

function admin\_login($username,$password){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("SELECT \* FROM admin WHERE username='$username' AND password = '$password'");

$stmt->execute();

$result = $stmt->get\_result();

return $result;

}

function add\_admin($username,$pass,$admin\_id){

//Require Databse config file

require 'config.php';

//add admin

$stmt = $mysqli->prepare("INSERT INTO admin(username, password,admin\_id) VALUES(?,?,?)");

$stmt->bind\_param("sss",$username,$pass,$admin\_id);

if($stmt->execute()){

return true;

}else{

return false;

}

$result = $stmt->get\_result();

return $result;

}

function add\_user($username,$email,$phone,$passw,$datejoined){

//Require Databse config file

require 'config.php';

//add user

$stmt = $mysqli->prepare("INSERT INTO users(username, email, phone, passw, datejoined) VALUES(?,?,?,?,?)");

$stmt->bind\_param("sssss",$username,$email,$phone,$passw,$datejoined);

if($stmt->execute()){

return true;

}else{

return false;

}

$result = $stmt->get\_result();

return $result;

}

function add\_category($category){

//Require Databse config file

require 'config.php';

//add admin

$stmt = $mysqli->prepare("INSERT INTO categories(category) VALUES(?)");

$stmt->bind\_param("s",$category);

if($stmt->execute()){

return true;

}else{

return false;

}

$result = $stmt->get\_result();

return $result;

}

function fetch\_admin\_username($table,$username){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("SELECT \* FROM ".$table." WHERE username='$username'");

$stmt->execute();

$result = $stmt->get\_result();

return $result;

}

///user functions

function fetch($table){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("SELECT \* FROM ".$table." ORDER BY id DESC");

$stmt->execute();

$result = $stmt->get\_result();

return $result;

}

function fetch\_set($table){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("SELECT \* FROM ".$table." ORDER BY id DESC LIMIT 5");

$stmt->execute();

$result = $stmt->get\_result();

return $result;

}

function fetch\_data($table){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("SELECT \* FROM ".$table."");

$stmt->execute();

$result = $stmt->get\_result();

return $result;

}

/////////////////////the functions to add new hospitals ///////////////////////////////////////

function add\_hospital($hospitalname, $description, $doctors, $address, $longitude, $latitude, $nurses, $rooms, $status){

//Require Databse config file

require 'config.php';

//add admin

$stmt = $mysqli->prepare("INSERT INTO hospitals(hospitalname, description, doctors, address, longitude, latitude, nurses, rooms, status) VALUES(?,?,?,?,?,?,?,?,?)");

$stmt->bind\_param("ssisssiis",$hospitalname, $description, $doctors, $address, $longitude, $latitude, $nurses, $rooms, $status);

if($stmt->execute()){

return true;

}else{

return false;

}

$result = $stmt->get\_result();

return $result;

}

function add\_new($username,$passw,$pass){

//Require Databse config file

require 'config.php';

//add admin

$stmt = $mysqli->prepare("INSERT INTO admin(username,password,pass) VALUES(?,?,?)");

$stmt->bind\_param("sss",$username,$passw,$pass);

if($stmt->execute()){

return true;

}else{

return false;

}

$result = $stmt->get\_result();

return $result;

}

function complaints($fullname,$phone,$des,$hosp){

//Require Databse config file

require 'config.php';

//add admin

$stmt = $mysqli->prepare("INSERT INTO complaints(fullname,phone,description,hospital) VALUES(?,?,?,?)");

$stmt->bind\_param("ssss",$fullname,$phone,$des,$hosp);

if($stmt->execute()){

return true;

}else{

return false;

}

$result = $stmt->get\_result();

return $result;

}

function update\_addr($longitude, $latitude, $address, $username){

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("UPDATE users SET longitude = ?, latitude = ?, address = ? WHERE username = ?");

$stmt->bind\_param("ssss",$longitude, $latitude, $address, $username);

if($stmt->execute()){

return true;

}else{

return false;

}

$result = $stmt->get\_result();

return $result;

}

function fetch\_loc($latt,$logg){

//Require Databse config file

require 'config.php';

$stmt = $mysqli->prepare("SELECT \* , (3956 \* 2 \* ASIN(SQRT( POWER(SIN(( $latt - latitude) \* pi()/180 / 2), 2) +COS( $latt \* pi()/180) \* COS(latitude \* pi()/180) \* POWER(SIN(( $logg - longitude) \* pi()/180 / 2), 2) ))) as distance

from hospitals having distance <= 10 order by distance");

//$stmt->bind\_param("ss", $latt,$logg);

$stmt->execute();

$result\_loc = $stmt->get\_result();

return $result\_loc;

}

function fetch\_prod($search){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("SELECT \* FROM products WHERE productname = '$search' || productid = '$search'");

$stmt->execute();

$result = $stmt->get\_result();

return $result;

}

function fetch\_cmp($eid){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("SELECT \* FROM complaints WHERE id = '$eid'");

$stmt->execute();

$result = $stmt->get\_result();

return $result;

}

function fetch\_hist\_all(){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("SELECT \* FROM pointsale");

$stmt->execute();

$result = $stmt->get\_result();

return $result;

}

function fetchinf($hid){

//Require Databse config file

require 'config.php';

//fetch all hospital

$stmt = $mysqli->prepare("SELECT \* FROM hospitals WHERE id = '$hid'");

$stmt->execute();

$result = $stmt->get\_result();

return $result;

}

///////////////////end dunno ///////////////////////////////////////////////

function fetch\_usr($table,$username){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("SELECT \* FROM ".$table." WHERE username='$username'");

$stmt->execute();

$result = $stmt->get\_result();

return $result;

}

function updateStat($pid,$table,$status){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("UPDATE ".$table." SET status = '$status' WHERE id = '$pid'");

if($stmt->execute()){

return true;

}else{

return false;

}

$result = $stmt->get\_result();

return $result;

}

function update\_hospital($hid, $hospitalname, $description, $doctors, $address, $longitude, $latitude, $nurses, $rooms){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("UPDATE hospitals SET hospitalname = ?, description = ?, doctors = ?, address = ?, longitude = ?, latitude = ?, nurses = ?, rooms = ? WHERE id = ?");

$stmt->bind\_param("ssisssiii",$hospitalname, $description, $doctors, $address, $longitude, $latitude, $nurses, $rooms, $hid);

if($stmt->execute()){

return true;

}else{

return false;

}

$result = $stmt->get\_result();

return $result;

}

/////site settings

function fetchsiteset(){

//Require Databse config file

require 'config.php';

//fetch all user

$stmt = $mysqli->prepare("SELECT \* FROM settings ORDER BY id DESC LIMIT 1");

$stmt->execute();

$result = $stmt->get\_result();

return $result;

}

function update\_site\_settings($token,$username){

//Require Databse config file

require 'config.php';

$stmt = $mysqli->prepare("UPDATE settings SET token = '$token' WHERE username = '$username'");

if($stmt->execute()){

return true;

}else{

return false;

}

$result = $stmt->get\_result();

return $result;

}

?>

<?php

include\_once('header.php');

?>

<div class="page-wrapper">

<div class="content">

<div class="row">

<div class="col-sm-5 col-5">

<h4 class="page-title">Hospitals</h4>

</div>

<div class="col-sm-7 col-7 text-right m-b-30">

<a href="add-hospital.php" class="btn btn-primary btn-rounded"><i class="fa fa-plus"></i> Add New</a>

</div>

</div>

<div class="row">

<div class="col-md-12">

<div class="table-responsive">

<table class="table table-striped custom-table mb-0 datatable">

<thead>

<tr>

<th>#</th>

<th>Name</th>

<th>Description</th>

<th>Doctors</th>

<th>Address</th>

<th>longitude</th>

<th>Latitude</th>

<th>Nurses</th>

<th>Rooms</th>

<th>Status</th>

<th class="text-right">Action</th>

</tr>

</thead>

<tbody>

<?php

$count = 0;

$table = 'hospitals';

$fetch\_usrs = fetch\_set($table);

if ($fetch\_usrs->num\_rows > 0) {

while ($fetch = $fetch\_usrs->fetch\_assoc()) { ?>

<tr>

<td><?php echo $count = $count + 1; ?></td>

<td><?php echo $fetch['hospitalname']; ?></td>

<td><?php echo $fetch['description']; ?></td>

<td><?php echo $fetch['doctors']; ?></td>

<td><?php echo $fetch['address']; ?></td>

<td><?php echo $fetch['longitude']; ?></td>

<td><?php echo $fetch['latitude']; ?></td>

<td><?php echo $fetch['nurses']; ?></td>

<td><?php echo $fetch['rooms']; ?></td>

<td><span class="custom-badge status-green">Active</span></td>

<td class="text-right">

<div class="dropdown dropdown-action">

<a href="#" class="action-icon dropdown-toggle" data-toggle="dropdown" aria-expanded="false"><i class="fa fa-ellipsis-v"></i></a>

<div class="dropdown-menu dropdown-menu-right">

<a class="dropdown-item" href="edit.php?hosp=<?php echo $fetch['id']; ?>"><i class="fa fa-pencil m-r-5"></i> Edit</a>

<a class="del dropdown-item" id="<?php echo $fetch['id']; ?>"><i class="fa fa-trash-o m-r-5"></i> Delete</a>

</div>

</div>

</td>

</tr>

<?php

}

}else{

echo '<tr><td>No records yet!</td></tr>';

}

?>

</tbody>

</table>

</div>

</div>

</div>

</div>

</div>

<script src="assets/js/jquery-3.2.1.min.js"></script>

<script type="text/javascript">

$(document).ready(function(){

$('.del').click(function(){

var del = $(this).attr("id");

if(confirm("Are you sure you want to delete?")){

$.ajax({

url:'ajaxcall.php',

method:'POST',

data:'del='+del,

success:function(data){

alert(data);

},

error: function(data){

}

});

location.reload();

}

});

})

</script>

<?php

include\_once('footer.php');

?>

<?php

include\_once('header.php');

if(isset($\_POST['emg'])){

$hosp = $\_GET['hosp'];

$fullname = secure($\_POST["fullname"]);

$phone = secure($\_POST["phone"]);

$des = secure($\_POST["description"]);

$chk\_u = complaints($fullname,$phone,$des,$hosp);

if($chk\_u==true){

$success = 'Emergency Complaints Submitted';

}else{

$error = 'Error, try again';

}

}

?>

<div class="main-wrapper account-wrapper">

<div class="account-page">

<div class="account-center">

<div class="account-box">

<form action="<?php $\_SERVER['PHP\_SELF'];?>" method="post" class="form-signin" enctype="multipart/form-data">

<div class="account-logo">

<a href="#"><img src="assets/img/logo-dark.png" alt=""></a>

</div>

<center><p style="font-size: 2em; color: #009efb;">Emergency Complaints Form</p></center>

<center><h5 style="color: #009ce7;">Nearby Hospital:<b> <?php echo $\_GET['hosp']; ?> </b></h5></center>

<?php

if(isset($error)){

echo '<center><h5 style="color: red;">'.$error.'</h5></center>';

}

if(isset($success)){

echo '<center><h5 style="color: green;">'.$success.'</h5></center>';

}

?>

<div class="form-group">

<label>Fullname</label>

<input type="text" class="form-control" name="fullname" required>

</div>

<div class="form-group">

<label>Mobile Number</label>

<input type="text" class="form-control" name="phone" required>

</div>

<div class="form-group">

<label>Description</label>

<textarea class="form-control" name="description" required></textarea>

</div>

<div class="form-group text-center">

<button class="btn btn-primary account-btn" type="submit"><input type="hidden" name="emg">Request Emergency</button>

</div>

</form>

</div>

</div>

</div>

</div>

<?php

include\_once('footer.php');

?>

<?php

include\_once('header.php');

?>

<div class="page-wrapper">

<div class="content">

<div class="row">

<div class="col-sm-5 col-5">

<h4 class="page-title">All Users</h4>

</div>

</div>

<div class="row">

<div class="col-md-12">

<div class="table-responsive">

<table class="table table-striped custom-table mb-0 datatable">

<thead>

<tr>

<th>#</th>

<th>Username</th>

<th>Email</th>

<th>Phone</th>

<th>Address</th>

<th>longitude</th>

<th>Lattitude</th>

<th class="text-right">Action</th>

</tr>

</thead>

<tbody>

<?php

$count = 0;

$table = 'users';

$fetch\_usrs = fetch\_data($table);

if ($fetch\_usrs->num\_rows > 0) {

while ($fetch = $fetch\_usrs->fetch\_assoc()) { ?>

<tr>

<td><?php echo $count = $count + 1; ?></td>

<td><?php echo $fetch['username']; ?></td>

<td><?php echo $fetch['email']; ?></td>

<td><?php echo $fetch['phone']; ?></td>

<td><?php echo $fetch['address']; ?></td>

<td><?php echo $fetch['longitude']; ?></td>

<td><?php echo $fetch['latitude']; ?></td>

<td class="text-right">

<div class="dropdown dropdown-action">

<a href="#" class="action-icon dropdown-toggle" data-toggle="dropdown" aria-expanded="false"><i class="fa fa-ellipsis-v"></i></a>

<div class="dropdown-menu dropdown-menu-right">

<a class="delu dropdown-item" id="<?php echo $fetch['id']; ?>"><i class="fa fa-trash-o m-r-5"></i> Delete</a>

</div>

</div>

</td>

</tr>

<?php

}

}else{

echo '<tr><td>No records yet!</td></tr>';

}

?>

</tbody>

</table>

</div>

</div>

</div>

</div>

</div>

<script src="assets/js/jquery-3.2.1.min.js"></script>

<script type="text/javascript">

$(document).ready(function(){

$('.delu').click(function(){

var delu = $(this).attr("id");

if(confirm("Are you sure you want to delete?")){

$.ajax({

url:'ajaxcall.php',

method:'POST',

data:'delu='+delu,

success:function(data){

alert(data);

},

error: function(data){

}

});

location.reload();

}

});

})

</script>

<?php

include\_once('footer.php');

?>