The International Journal of Multi-Disciplinary Research

ISSN: 3471-7102

FIGHTING COUNTERFEIT DRUGS USING INFORMATION COMMUNICATION TECHNOLOGY (FCDICT) - ZAMBIAN EXPERIENCE.

(Conference ID: CFP/167/2017)

Dr Patrick Kalyabanyama
Lecturer, School of Business Studies.
Mulungushi University
P.O.Box 80415 Kabwe

M. Sinyanda
Lecturer : School of Science, Engineering and Technology
Mulungushi University
P.O.Box 80415 Kabwe

Simukonda Luckson
Research Assistant:
School of Science, Engineering and Technology
Mulungushi University
P.O.Box 80415 Kabwe

1.0 ABSTRACT

Counterfeit drugs increase every year and Information Communication Technology is also increasing, therefore using Information technology concepts like computer visioning, content delivery and database concepts we are able to fight counterfeit drugs. Mobile technology is becoming more and more every day therefore people can use mobile technology to fight against counterfeit drugs.

The aim was to analyze, design; implement a prototype system to fight counterfeit drugs in Zambia using ICT tools.

The following objectives were realized and implemented

- *To mobilize and involve the stockholders to support the system*
- *Analyze and propose a design architecture for Counterfeit Drug Identifier System (CDIS)*
- *Implement a system that is going to use Ge-location and produce statistical reports*
- *Using web services the client and the server should be able to communicate to transfer information to and from each other.*

The system presented successfully managed to solve the problem of counterfeit drugs, through the use of ICT mainly through two components that were developed. The user of the system is able to enter a code and take picture of product codes on drugs using a mobile phone. Through these two components a user is able to send a code or a picture to the relevant server, once the message is received by the server, it is processed and a notification message is then sent back to the user to inform whether the code which was received is valid or not. This process is providing a mechanism of authenticating drugs with a remote server and able to notify all the relevant stakeholders in the fight against counterfeit drugs on geo-location and authenticity of drugs sent through the system. Therefore the objectives of this project were met and successfully fulfilled.

Key Words: *Information Communication Technology, content delivery, Counterfeit Drugs, Web services, Geo-location*

2.0 INTRODUCTION AND BACKGROUND

Over the past years more and more technologies have been developed, and these technologies have enabled people all around the world to produce and manufacture products at a very high rate. The worrying thing all around the world is that people will begin to manufacture products that are of sub stand, products that do not work, and products that are harmful to human beings. One may wonder if technology has brought more harm than good, in that in the 21st century counterfeit products have increased tremendously and among these counterfeit products are counterfeit drugs, and this has worried the world's health sector because people are making drugs that do not work or drugs that have incorrect measurements of medicinal ingredients in the drugs that are being made.

Recent statistics from (Southwick, 2013) have proved that over 1 million people die annually from counterfeit drugs, which is evidence that we are all not secure no country or state is exempted from this problem of counterfeit drugs. Surveys that have been taken in Africa by WHO have proved that between 20% and 90% of all anti-malarial failed quality testing because these anti-malarial drugs contained medicinal ingredients that are incorrect.

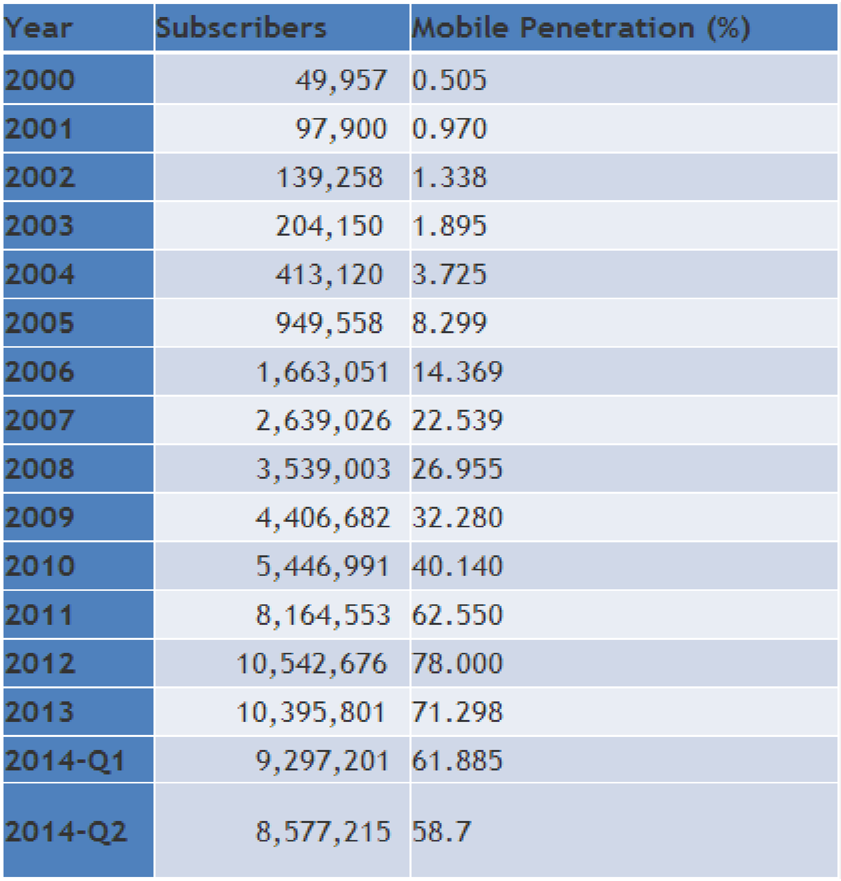
Unfortunately Zambia is also affected in that it is not spared from having counterfeit drugs out on the market and people do not get healed because the drugs that they buy are of substandard or contains incorrect ingredients, some people even die because of counterfeit drugs.

Therefore what shall we do then to solve this problem? Well the solution is in our hands in that a system can be developed that is able to tell a counterfeit drug from a genuine drug from the covers of the drug, a system that is able to tell that this drug is expired and is not safe for consumption. The Vision 2030 plan which seeks to empower the Zambian populace with information among other things will help this system come to a realization (MoH, 2013).

3.0 LITERATURE REVIEW

Mobile phone usage has increased tremendously over the past few years and there are 3.5 times more mobile phones than PC's (Nisarg Gandhewar, 2010). Today mobile phones are not just used for texting or calling but are used for other things like entertainment, gathering information and many other important things.

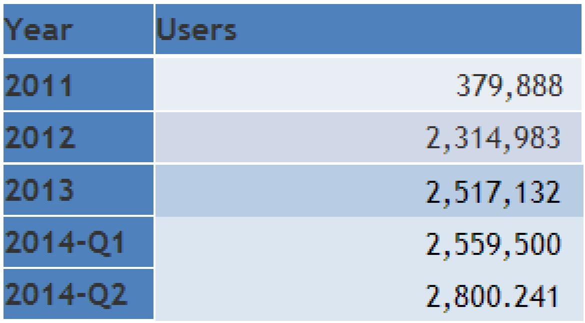
Zambia has had an increasing mobile usage increase from 2000 up to 2012 (ZICTA, 2014) until in 2013 and 2014 where the use of mobile phones dropped because people are now using tablets and other devices that are able to communicate like mobile phones. In 2000 there were only 49,957 mobile subscribers and by 2014 there are 8,577,215 which show a steady increase in the use of mobile Technology. The diagram below describes statistics provided by ZICTA:



Year	Subscribers	Mobile Penetration (%)
2000	49,957	0.505
2001	97,900	0.970
2002	139,258	1.338
2003	204,150	1.895
2004	413,120	3.725
2005	949,558	8.299
2006	1,663,051	14.369
2007	2,639,026	22.539
2008	3,539,003	26.955
2009	4,406,682	32.280
2010	5,446,991	40.140
2011	8,164,553	62.550
2012	10,542,676	78.000
2013	10,395,801	71.298
2014-Q1	9,297,201	61.885
2014-Q2	8,577,215	58.7

Figure 1 showing mobile penetration in Zambia from 2000-2014 (ZICTA, 2014)

But there has been an increase in the use of mobile internet as shown in the diagram below:



Year	Users
2011	379,888
2012	2,314,983
2013	2,517,132
2014-Q1	2,559,500
2014-Q2	2,800.241

Figure 2 showing the use of mobile internet (ZICTA, 2014)

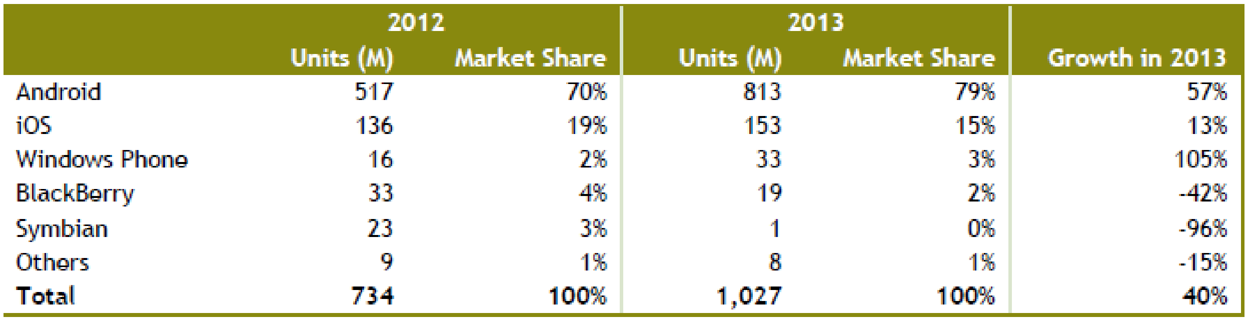
Using the statistics provided above we can safely say that mobile internet users keep increasing every day. These statistics also help achieve the Zambia e-health strategy, by providing solid evidence why mHealth should be implemented in Zambia. The problem statements stated in the e-health strategy is that there are high disease rates, and shortages of health practitioners (MoH, 2013), but with the help of mHealth these problems can be reduced.

Existing Mobile Platforms

The increasing demand of mobile phones has led to the development of different platforms, some of the big platforms available are Android, RIM's BlackBerry, Apple's iPhone, and Windows Mobile, and therefore in this part of the chapter we are going to elaborate more on each of the popular platforms available.

Android

Android is a software stack for mobile devices which includes an operating system, middleware and applications, which was developed by Google and it code license is under apache which means that it's code is free. According to statistics provide by (CSSInsight, 2014) android dominates in volume and market share terms as evidence shows in the diagram below:



	2012		2013		
	Units (M)	Market Share	Units (M)	Market Share	Growth in 2013
Android	517	70%	813	79%	57%
iOS	136	19%	153	15%	13%
Windows Phone	16	2%	33	3%	105%
BlackBerry	33	4%	19	2%	-42%
Symbian	23	3%	1	0%	-96%
Others	9	1%	8	1%	-15%
Total	**734**	**100%**	**1,027**	**100%**	**40%**

Figure 3 showing the market share of Mobile platforms as of 2012-2013 source (CSSInsight, 2014)

Android is belt on top of the Linux kernel because the Linux kernel has been improving tremendously over the past years (Marko Gargenta, 2014) and is suitable for the following reasons:

1. **Portability**
   Linux is a very relative system that is very easy to port to another hardware architecture, and because Android is based on the Linux kernel not much is to be considered for the underlying hardware features (Marko Gargenta, 2014).
2. **Security**
   Linux is a highly secure system, having been tested over the years through some harsh environments and having been passed these tests (Marko Gargenta, 2014) states that it has been concluded that it is a secure kernel, therefore Android relies on the security of Linux.
3. **Features**
   Linux comes with a lot of feature which are important to android, feature like power management, memory management and many other feature that android uses.

Structural overview

The Android software stack/ Architecture as shown in figure 2 can be subdivided into five layers: The kernel and low level tools, native libraries, the Android Runtime, the framework layer and on top of all the applications.

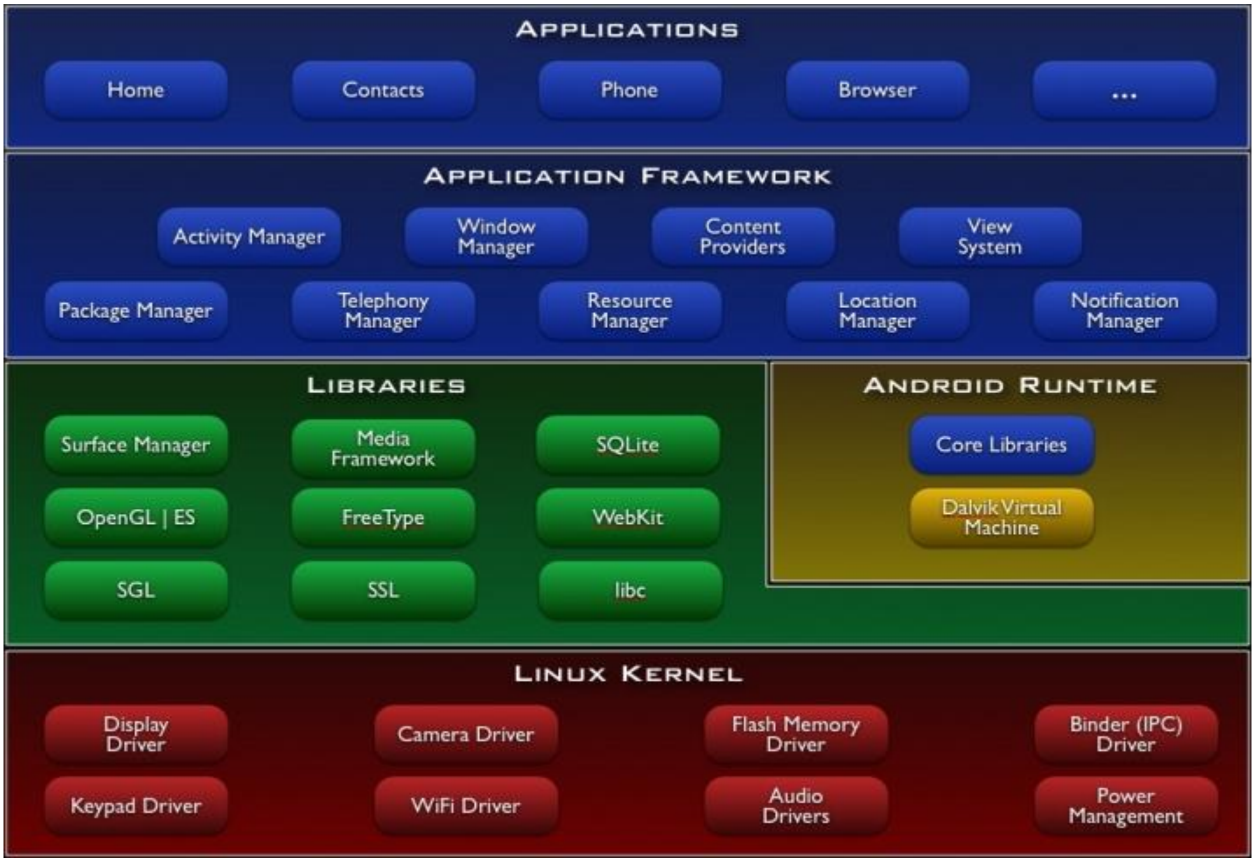


Figure 4 Android system architecture. Source (Romanovsky, 2013)

Linux kernel

Linux Kernel (Linux 2.6) is at the bottom layer of the software stack. The android operating system is implemented from this component. (Singh, 2014).

Native Libraries Layer

Just above the Linux Kernel layer is Android's native libraries. This layer enables the device to handle different types of data (Singh, 2014).

Android Runtime

The Android Runtime has Core Java libraries and Dalvik Virtual machine. It is located on the same level as the library layer. Dalvik Virtual Machine is a Virtual Machine used for running Android applications (Singh, 2014).

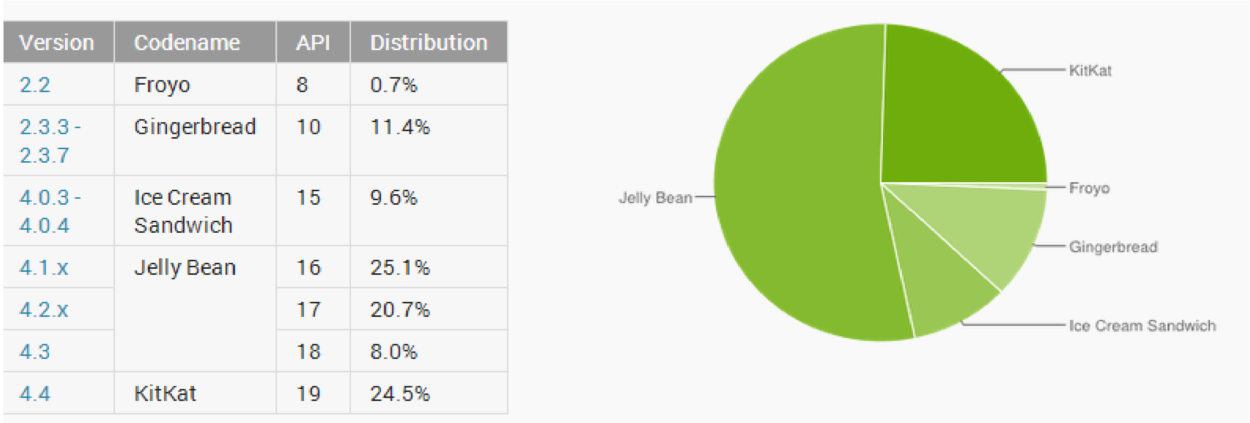
Application framework

The Application Fabric layer provides many higher-level services or important APIs to applications in the form of Java classes. Application developers are allowed to make use of these services in their applications (Singh, 2014).

Applications

The Applications Layer is the top layer in the Android architecture. A developer is able to write his or her own application and can replace it with the existing application (Singh, 2014).

Over the past years many edition of Android operating system have been released and the table below shows the versions and there dispersion:



Version	Codename	API	Distribution
2.2	Froyo	8	0.7%
2.3.3 - 2.3.7	Gingerbread	10	11.4%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	9.6%
4.1.x	Jelly Bean	16	25.1%
4.2.x		17	20.7%
4.3		18	8.0%
4.4	KitKat	19	24.5%

Figure 5 showing versions and public share of the android OS

IOS

IOS is the operating system that operates on apple products like iPad, iPhone, and iPod devices. The IOS operating system manages the hardware and supports the technologies required to manage old application. The system comes with different applications, applications such as Mail, and Safari, which provide default services to the user. IOS has four abstraction layers which are shown in following figure:

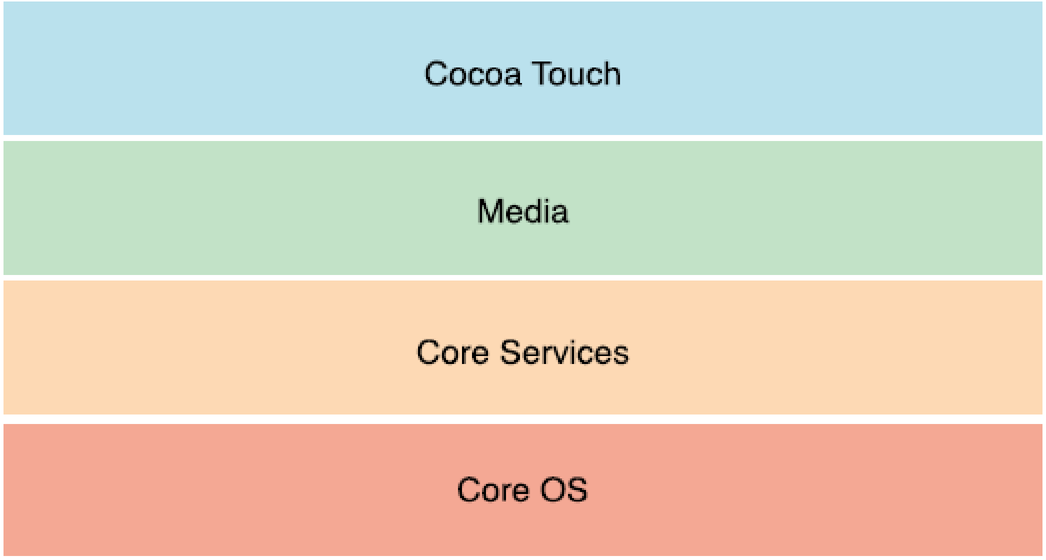


Figure 6 showing architecture of iOS (Bilal Ahmed yaseen, 2012)

- The Cocoa Touch layer contains key frameworks for building iOS apps. These frameworks define the appearance the application. The interfaces are made in C language and provide technologies such as Core Foundation, CF Network, SQLite, and access to POSIX threads and UNIX sockets among others (Bilal Ahmed yaseen, 2012).
- The Media layer contains the fundamental technologies used to support 2D and 3D, video and audio. This layer includes technologies that are built in C like OpenGL ES, Quartz, and Core Audio. It also contains Core Animation, which is an advanced Objective-C based animation engine. It uses a mixture of C-based and Objective-C based interfaces (Bilal Ahmed yaseen, 2012).
- The Core Services layer contains fundamental system services for apps. This layer also contains some technologies to support features such as location, iCloud, social media, and networking (Apple.Inc, 2012).
- The Core OS is considered the deepest layer of the iOS architecture. The Core OS layer contains the low-level features that most other technologies are built upon (Apple.Inc, 2012).

Windows Mobile

Microsoft has developed a mobile platform that it uses for phones, as seen from figure 1 windows mobile carries a 3% market share. Windows mobile has four layers of abstraction which will be documented about in a few lines to come.

Windows Mobile has plenty of features which a mobile device ever needs. When we look at the system as a whole, WM is one of the top operating systems for mobile devices. The OS is built with many cutting-edge features including (Le, 2009):

- **High-end features:**
  International language, GPS, speech recognition

- **Most-wanted features:**
  Media player Bluetooth, Wi-Fi, web browser
- **Basic features:**
  SMS messaging, vibration, Telephone API, screen resolution recognition

The Diagram is the windows mobile operating system architecture:

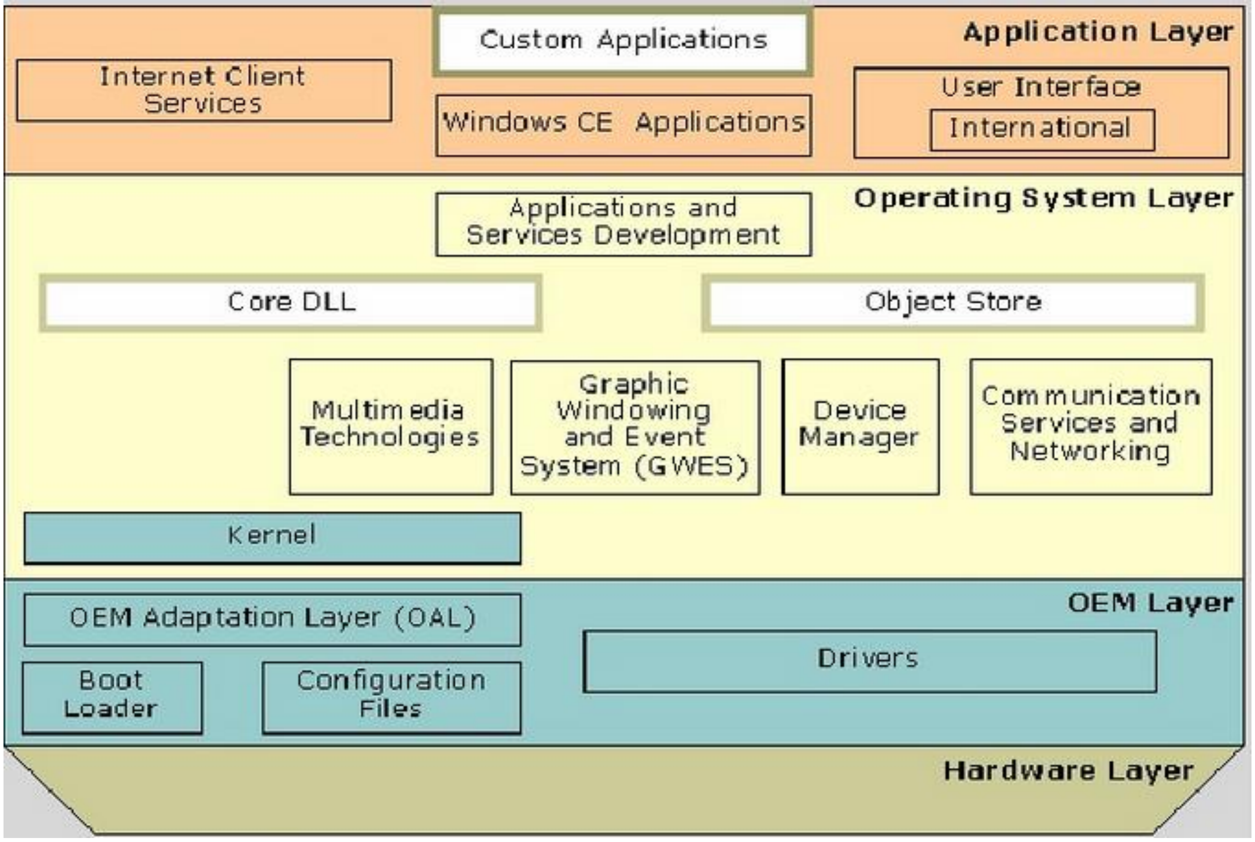


Figure 7 showing windows mobile architecture source (Le, 2009)

Application Layer:

This layer contains most custom applications. An application in this layer runs in user mode. Applications run in this layer are considered as un-trusted codes (Le, 2009). A crash in these applications in this layer will not cause a severe damage to the whole system. Usually, an application will be shut down by the system without bringing down other components.

Operating System Layer:

This layer contains all services that an operating system will need. The services include memory management, file system management, IO service, device manager, communication and networking (Le, 2009). All of services in this layer are operating in kernel mode.

OEM Layer:

This layer is reserved for OEM vendor components. OEMs are responsible for building very-low level components which communicate with hardware devices. These components include system bootstrap, memory map, interrupt handlers (Le, 2009).

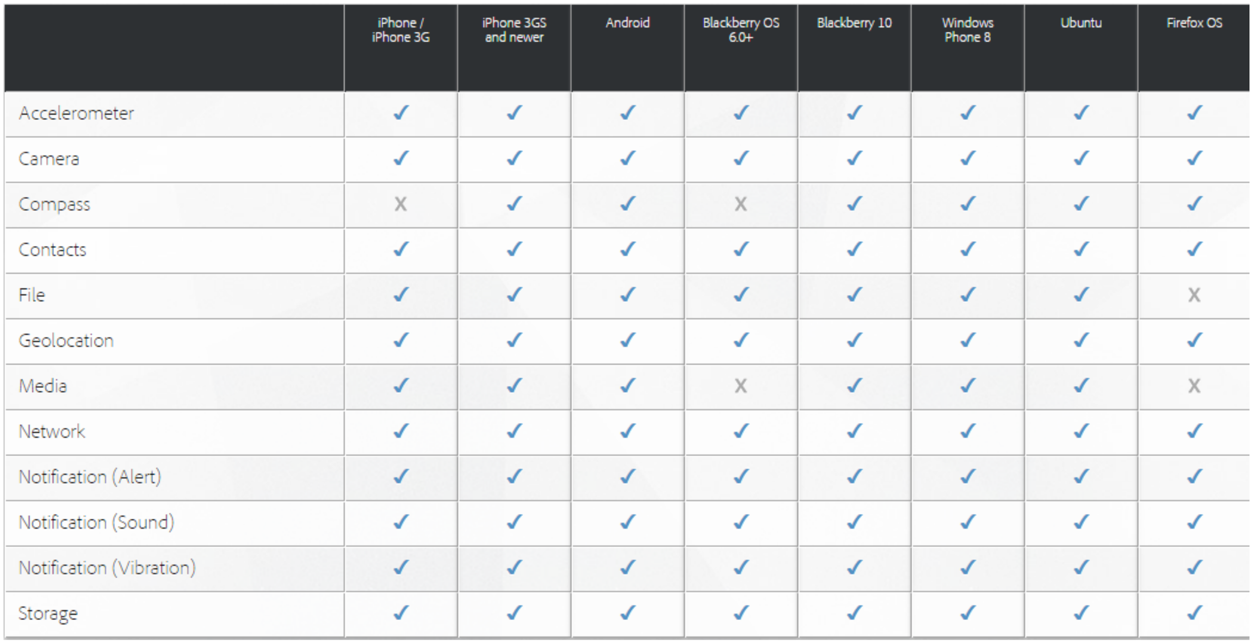
Hardware layer:

This layer represents all hardware components of a device (Le, 2009).

Phone Gap

Phone Gap is an open source framework that helps developers develop mobile applications with the use of JavaScript and html5, mobile platforms include iOS, Android, and BlackBerry (Rohit Ghatol, 2012). This framework lets you build HTML and JavaScript-based apps and still take advantage of mobile device capabilities like camera, local Storage, geolocation, storage and many more, for any mobile platform you target (Rohit Ghatol, 2012).

The diagram below shows Phone Gap compatibility with other platforms:



	iPhone / iPhone 3G	iPhone 3GS and newer	Android	Blackberry OS 6.0+	Blackberry 10	Windows Phone 8	Ubuntu	Firefox OS
Accelerometer	✓	✓	✓	✓	✓	✓	✓	✓
Camera	✓	✓	✓	✓	✓	✓	✓	✓
Compass	X	✓	✓	X	✓	✓	✓	✓
Contacts	✓	✓	✓	✓	✓	✓	✓	✓
File	✓	✓	✓	✓	✓	✓	✓	X
Geolocation	✓	✓	✓	✓	✓	✓	✓	✓
Media	✓	✓	✓	X	✓	✓	✓	X
Network	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Alert)	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Sound)	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Vibration)	✓	✓	✓	✓	✓	✓	✓	✓
Storage	✓	✓	✓	✓	✓	✓	✓	✓

Figure 8 showing Phone Gap platform compatibility (PhoneGap, 2014)

The Diagram below shows the architecture for Phone Gap:

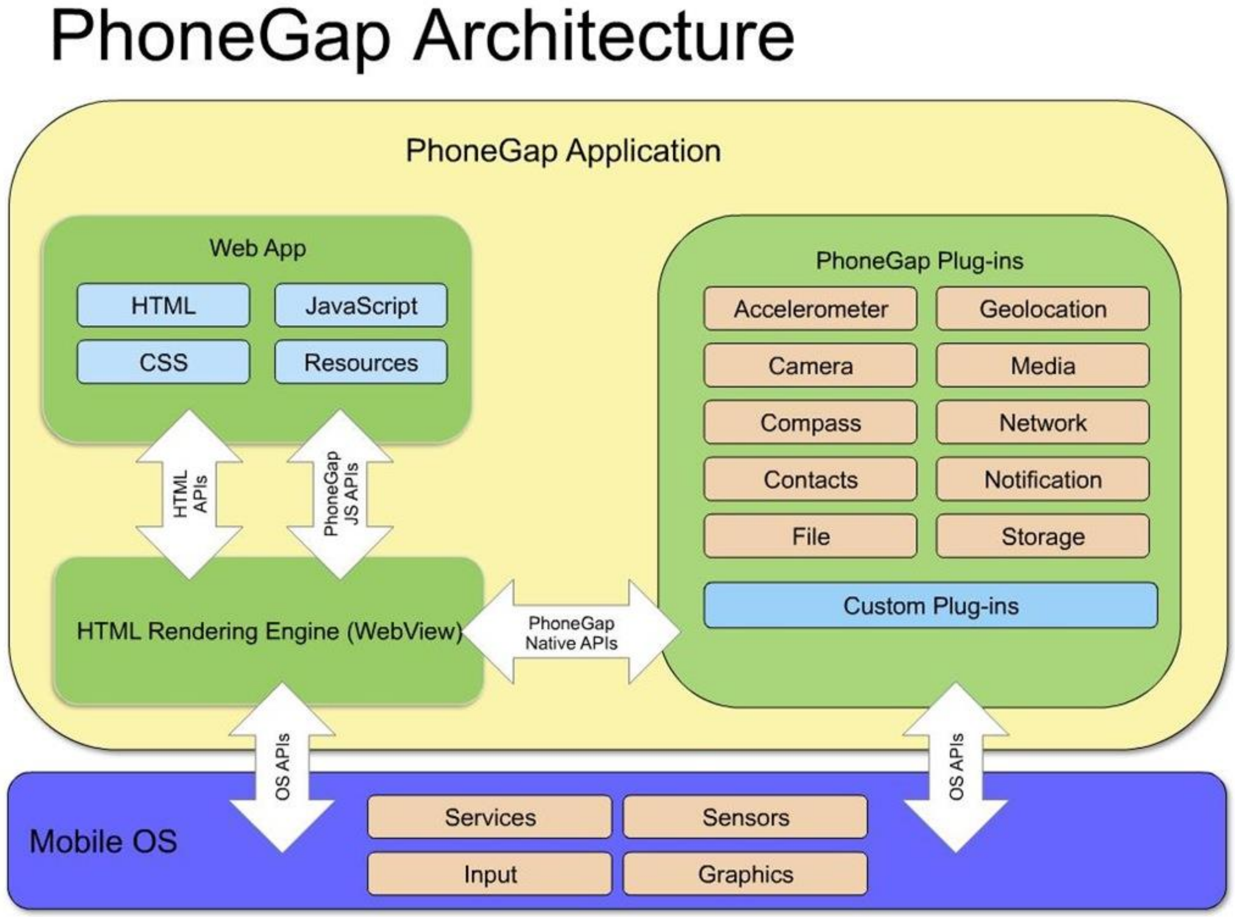


Figure 9 showing Phone Gap architecture (Shield, 2014)

1. **Web App** -these are web based languages can are used in Phone Gap in order to communicate with the Phone Gap API (PhoneGap, 2014).
2. **HTML Rendering Engine**- this is the main interface between the user interface, the mobile OS and the Phone Gap plugin-ins, it converts data from one form to another (Shield, 2014).
3. **Mobile OS**- this is the operating system and it’s main of any supported mobile platform (Shield, 2014).
4. **Phone Gap Plug-Ins**- these plugins contain call functions to the operating system, and when a functionality is needed an appropriate call to the system is made (Shield, 2014).

Platform Comparison and selection

The IOS and Windows platform can be used, but according to the statistics they are the least used by people compared to the android platform, therefore Android would be one suitable platform to use but it is still one platform what about the other percentage of platforms that exist, do we just forget about them? No, that is where we use Phone Gap. Phone Gap has proved to be the best platform independent platform to use to develop CIDS, but for now Phone Gap is not compatible with the OpenCV library which will be heavily used in the development of CDIS. Therefore according to the statistics that have been gathered and the system that is required to be developed, the better choice of the documented platforms is to use the Android.

Systems Review

In this part of the chapter is we are going to look at three existing mobile systems that are fighting counterfeit drugs, it will also look at countries that have implemented these systems, and finally it will review the platforms that these systems are running on and the mobile platform that CIDS will run on.

MPedigree

Bright Simons, a Ghanaian, developed a phone-based system called mPedigree to tackle the problem of counterfeit drugs (USAID, 2012). mPedigree solves the problem of counterfeit drugs in such way that mobile users can send a number that they scratch off a purchased drug, then within seconds a message will be received that tells them if the purchased drug is a genuine drug or a counterfeit drug. Using this method users are able to trace the origin and authenticity of the drug. mPedigree was launched in 2007 ever since then the mPedigree network has been growing and now this system is implemented in Ghana, Nigeria, East Africa and South Asia even India has started using this system (USAID, 2012).

The Diagram below shows a mobile user in Ghana verifying a drug:

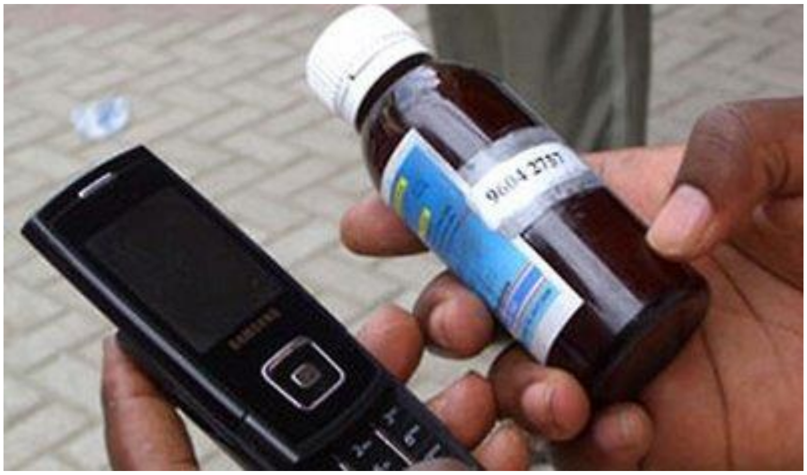
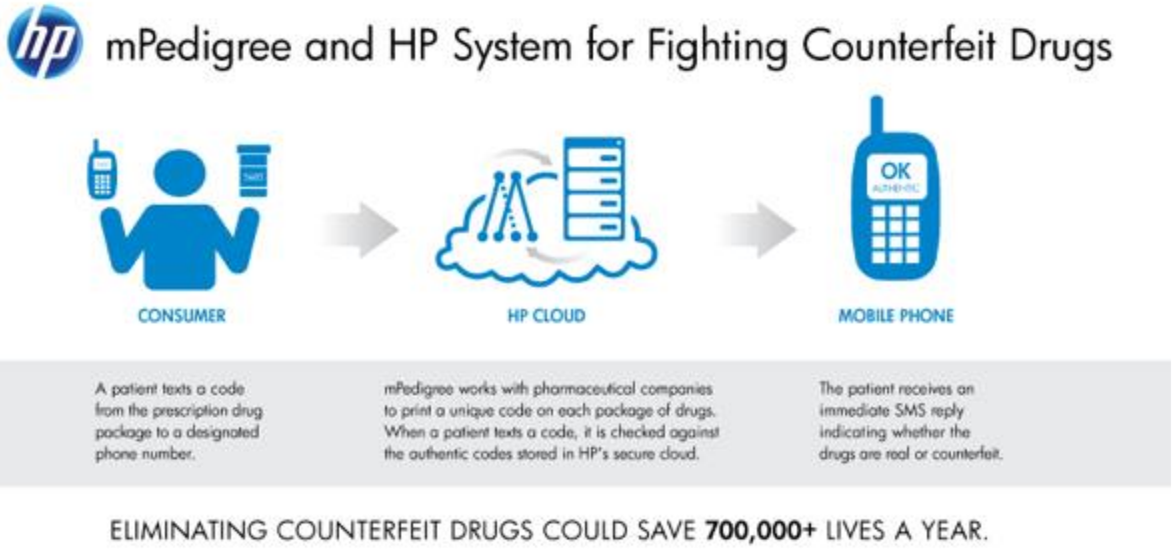


Figure 10 showing user authenticating a bought drug (Dolan, 2010)

mPedigree has established an understanding with authentic or know manufactures of drugs to be putting a scratch pad that when you scratch a number will be reviewed, then using that number the user is able to send a request. This request is routed to mPedigree servers and consumers receive a quick response to authenticate their purchase.

HP created a system for tracking pharmaceutical supplies in the supply chain, while leveraging the mobile penetration in region for the consumer interface (USAID, 2012). The diagram below shows the link between mPedigree system, the HP cloud-based system and the consumer:



A patient texts a code from the prescription drug package to a designated phone number.

mPedigree works with pharmaceutical companies to print a unique code on each package of drugs. When a patient texts a code, it is checked against the authentic codes stored in HP's secure cloud.

The patient receives an immediate SMS reply indicating whether the drugs are real or counterfeit.

ELIMINATING COUNTERFEIT DRUGS COULD SAVE **700,000+** LIVES A YEAR.

Figure 11: Framework of mPedigree *Image Source (USAID, 2012)*

HP runs the hosting infrastructure and the security systems for the service, out of its data centers in Frankfurt. Since there are many mobile phones in Nigeria and Ghana, and till date they are still increasing, the system has a higher chance of reaching most people who are at risk (Zax, 2010).

Benefits of mPedigree

- MPedigree boast confidence in the users to take their medicines with confidence and a right mind to say that the drug they are ingesting is genuine.
- Helps reduce the death rate of people that die from counterfeit drugs, therefore mPedigree is a lifesaving system.
- Helps reduce the profit made by counterfeiters in that they will not get as much income as they would get when people buy the counterfeit products.
- It is accessible on all mobile platforms, therefore accessible to anyone with a mobile phone.

Shortfalls of mPedigree

- The system mPedigree has not implemented a GPS tracking system that is very crucial in that counterfeiters or the location of a counterfeit drug can be located in real time (South, 2011).
- MPedigree does not provide its users with report forms on how they found and how the drugs have affected them.

Countries using mPedigree

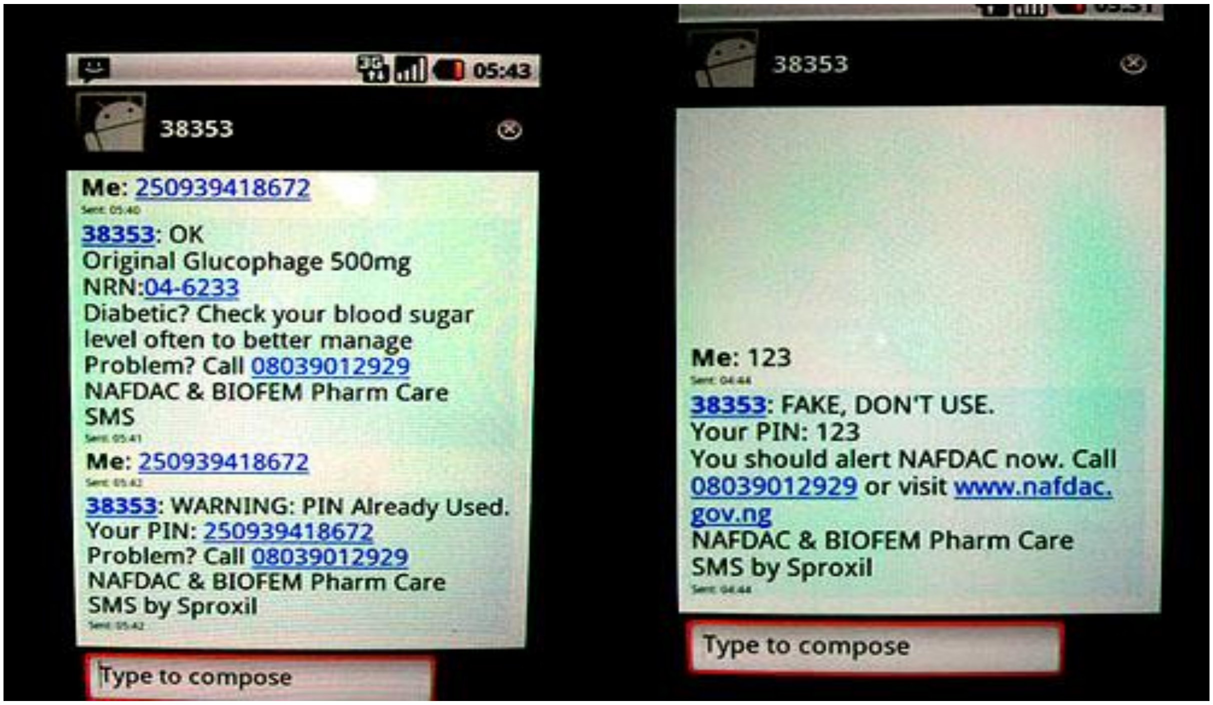
Because of mPedigree effectiveness, this system has extended to Nigeria, Rwanda, and Kenya and has gone as far as Asia in India (USAID, 2012).

1. 2.4.2 Sproxil

This system was founded by Ashifi Gogo in 2008 so that it can provide a simple and fast tool called the mobile authentication service (MAS) for drug consumers so that their medication is safe before buying the drug, this system help consumers determine if the drug is genuine or fake which allows them to participate in the fight against counterfeit drugs (Gogo, 2010).

Sproxil Mobile Product Authentication MPA solution uses scratch bars to tell whether a drug is an original one or a fake one. If it's indicated as fake, the consumer knows not to use it (thereby potentially saving his or her life) and is instructed how to report the counterfeit medicine to the proper authorities.

The Diagram Below show an SMS that authenticates and one that does not authenticate using Sproxil:



05:43

38353

Me: 250939418672
Sent 05:40

38353: OK
Original Glucophage 500mg
NRN:04-6233
Diabetic? Check your blood sugar level often to better manage
Problem? Call 08039012929
NAFDAC & BIOFEM Pharm Care SMS
Sent 05:41

Me: 250939418672
Sent 05:42

38353: WARNING: PIN Already Used.
Your PIN: 250939418672
Problem? Call 08039012929
NAFDAC & BIOFEM Pharm Care
SMS by Sproxil
Sent 05:42

38353

Me: 123
Sent 04:44

38353: FAKE, DON'T USE.
Your PIN: 123
You should alert NAFDAC now. Call 08039012929 or visit www.nafdac.gov.ng
NAFDAC & BIOFEM Pharm Care
SMS by Sproxil
Sent 04:44

Figure 12 showing two SMS received after sending the code on the product (Silberman, 2011).

The technology that Sproxil uses makes sure that it is as counterfeit-proof as possible, the company uses a scratch card method, similar to that used for replenishing mobile talk-time once the scratch card is scratched the number is never used (Gogo, 2010). Sproxil uses the following technologies so that it can function:

1. PHP- PHP is a popular scripting language for creating web pages. This is used by Sproxil for sever side scripting the main operation in the background.

2. JavaScript- JavaScript is a lightweight, object-oriented, cross-platform scripting language, and is used for client side scripting.
3. Apache- the Apache HTTP Server is a popular open source web server
4. JQuery CDN- The jQuery CDN hosts copies of the jQuery library, Sproxil uses this for content delivery.
5. Android –Sproxil has developed a smart android application and has managed to implement Google maps.

Benefits of Sproxil

- Sproxil also supports low literacy levels from people in that if the drug is a fake an X will appear on the screen and if it is a genuine a tick will appear therefore supporting low literacy levels. If response is sent back from the servers indicating that drug is fake the consumer is given a hotline number to call so that the fake product is reported (Sproxil, 2012).
- Sproxil now provides Geographical location services which means counterfeit products can be sported in real time.

Shortfalls of Sproxil

- The SMS based system is not free therefore, the mobile service providers charge users for texting that message.
- Sproxil does not provide its users with report forms on how they found and how the drugs have affected them.

Countries using Sproxil

Sproxil offers there service African and Asian countries some of the African countries are Chad, the Democratic Republic of the Congo, Rwanda ,Burkina Faso, Ghana, Kenya, Gabon, Malawi, Madagascar, Niger, Seychelles and works in conjunction with Airtel to provide a wider range of service (Gogo, 2010).

2. 2.4.3 EPedigree

This is a system that was developed by IBM to track goods throughout the supply chain, the ePedigree system, also known as an electronic certificate of authenticity, focuses on all aspects of the supply chain from the manufacturer to the pharmacy or hospital by tagging each bottle or package with a serial number (Sinha, 2009).

As a product moves from one place to place, each body or company that handles the product must carry forward all of the previous e-pedigree information. When it reaches its final destination, the retailer will have a complete document showing where that product has been and who has handled that product (Sinha, 2009). The first critical piece of the chain occurs right on the packaging line. The diagram below describes the movement of product information from one station to another until the retail seller:

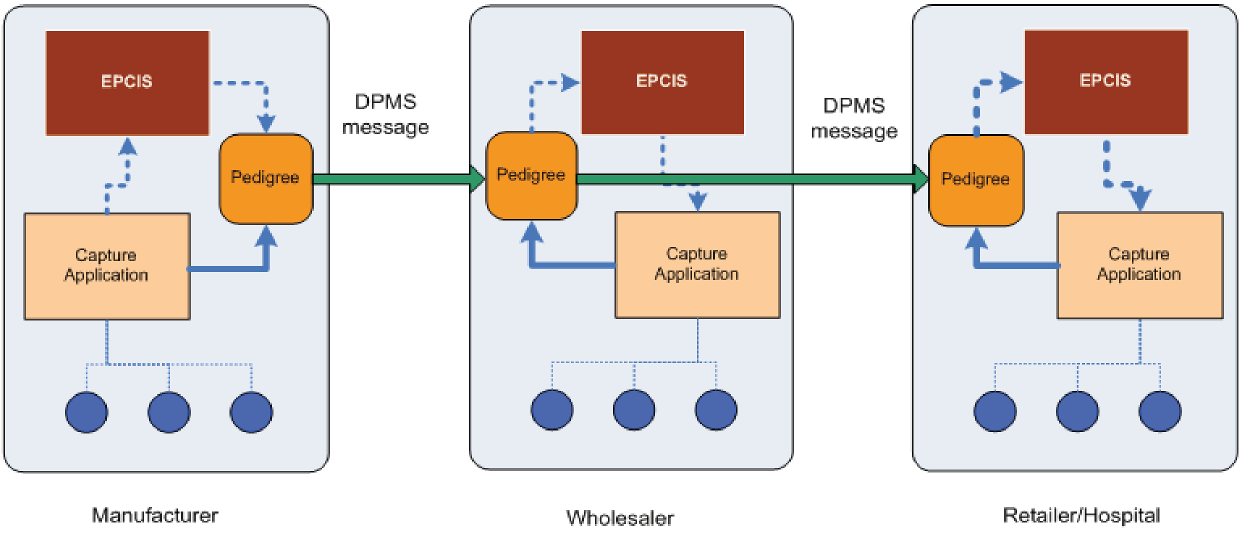


Figure 13 showing architecture of ePedigree (Sinha, 2009)

The diagram above shows DPMS messages being passed from one system to another from the manufacture to the wholesaler and finally the retailer or the hospital. Information is moved from one system to another using web services as show in the diagram below:

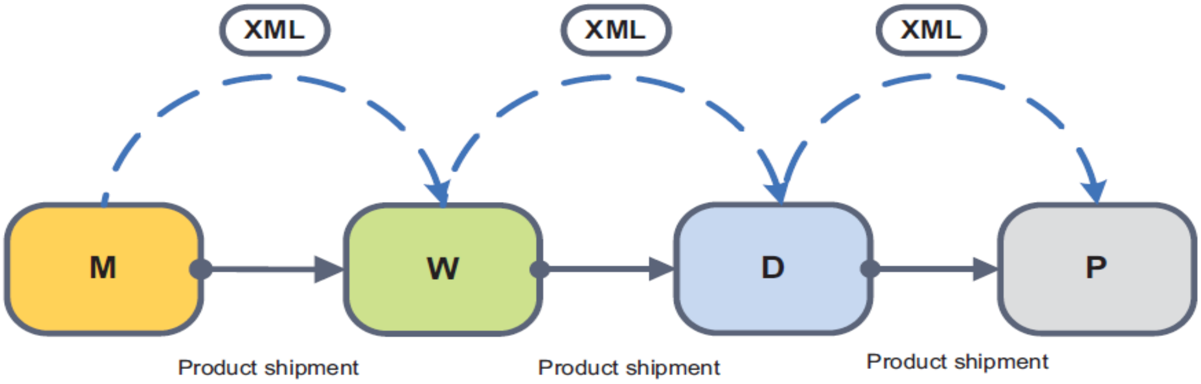


Figure 14 showing movement of information in xml format (Sinha, 2009)

The letters stand for manufacturers (M), distributors (D), wholesalers (W) and Pharmacy (P). Some of the standard protocols used for the transfer information are AS2, FTP, HTTP(s), fax and email. AS2 is the recommended standard and is the most commonly used standard for the transport.

Benefits of EPedigree

- It helps protect consumers from contaminated medicine and counterfeit drugs.
- It enable the authorities to track down where the product went wrong.

Shortfalls of EPedigree

- It is not available on the mobile platform therefore does not have a wide range of coverage.

- It does not offer location services, to tell where the counterfeit product is found.

Countries using EPedigree

This system is only available in America specifically California.

The Implemented System (CIDS)

CIDS has eliminated the shortfalls of the previous systems by Implement a location aware system for producing relevant statistical reports. It has promoted Pharmacovigilance to the users so that they can say what they think about a drug or how a drug is affecting them. Pharmacovigilance (PV) is defined as the science and activities relating to the detection, assessment, understanding and prevention of adverse effects or any other drug-related problem (WHO, 2014).

4.0 METHODOLOGY

This Section describes the essentials that were used in order to come up with CDIS, The researcher is first going to discuss the software development methodology that was used, it is also going to describe the technologies that were used and the implemented architecture of CDIS. It concludes by setting out the limitations during the research study.

Mobile-D Agile Methodology

Mobile-D is inspired by Extreme Programming, Crystal Methodologies and Rational Unified Process (RUP) (Abrahamsson, 2002). It is recommended to be used by a small, co-located team, working in a short development cycle. Mobile-D is structured in five phases (Explore, Initialize, Productionize, Stabilize and System Fix) sequentially arranged. Mobile –D methodology encourages iterations even though it is sequentially arranged.

1. **Explore**: The aim is to establish initial project as well as to prepare foundation for future development (Abrahamsson, 2002). This phase was done in the first part of this project because stakeholders were established and the scope of the project and the plan was defined.
2. **Initialize**: The purpose of this stage is to describe and prepare all components of application, as well as to predict possible critical issues of project (Abrahamsson, 2002). This phase was done when proposing this project in the first phases of this project.
3. **Productionize**: Includes implementation of required functionalities using iterative and incremental development cycle (Abrahamsson, 2002). The development of the system was done in the stipulated time.
4. **Stabilize**: The goal is to finalize the implementation, including integrating subsystems if needed (VTTElectronics, 2006). This phase involves fixing errors and making sure the system is functioning well, this phases is still in progress and will take time because there is always something wrong with the system.
5. **System test & fix**: review whether the application is developed according to requirements of clients and whether it contains all required functionalities (VTTElectronics, 2006). This phase has also been done under the testing section.

The diagram below shows a diagrammatic representation of the mobile-D development methodology:

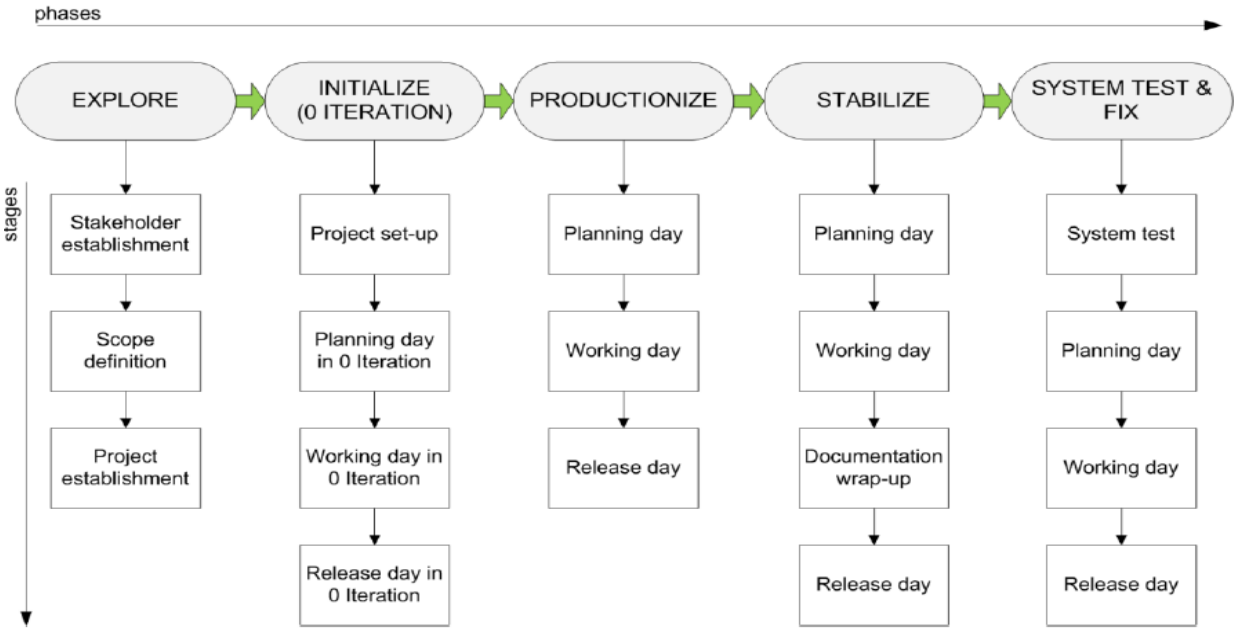


Figure 15 showing mobile-D life cycle (VTTElectronics, 2006)

After the implementation of Mobile-D there has been some advantages that have been observed, such as earlier discovery and repair of technical issues, increased progress visibility, low defect density in the final product, and a constant progress in development (Abrahamsson, 2002).

Technologies

The following technologies were used in order to make this system come to a realization:

External Libraries

OpenCV

OpenCV is an open source computer vision library written in C and C++ and runs under Linux, Windows and Mac OS X (Kaehler, 2010). There is active development on interfaces for Python, Java, Ruby, Mat lab, and other languages. OpenCV was designed for computational efficiency and with a strong focus on real-time applications. OpenCV is written in optimized C and can take advantage of multicore processors (Kaehler, 2010).

NUSOAP

This is an open source library that is being used in the system to enable communication between php and C# web services.

Development Language

- Android Object Oriented Programming using Java.
- C# for the backend
- PHP which will act as middleware
- C++ for image comparison

Database Management Tool

- MSSQL

Development Platform

- Eclipse ADT Bundle

Implemented Architecture

The diagram below is going to show the proposed architecture of CDIS:

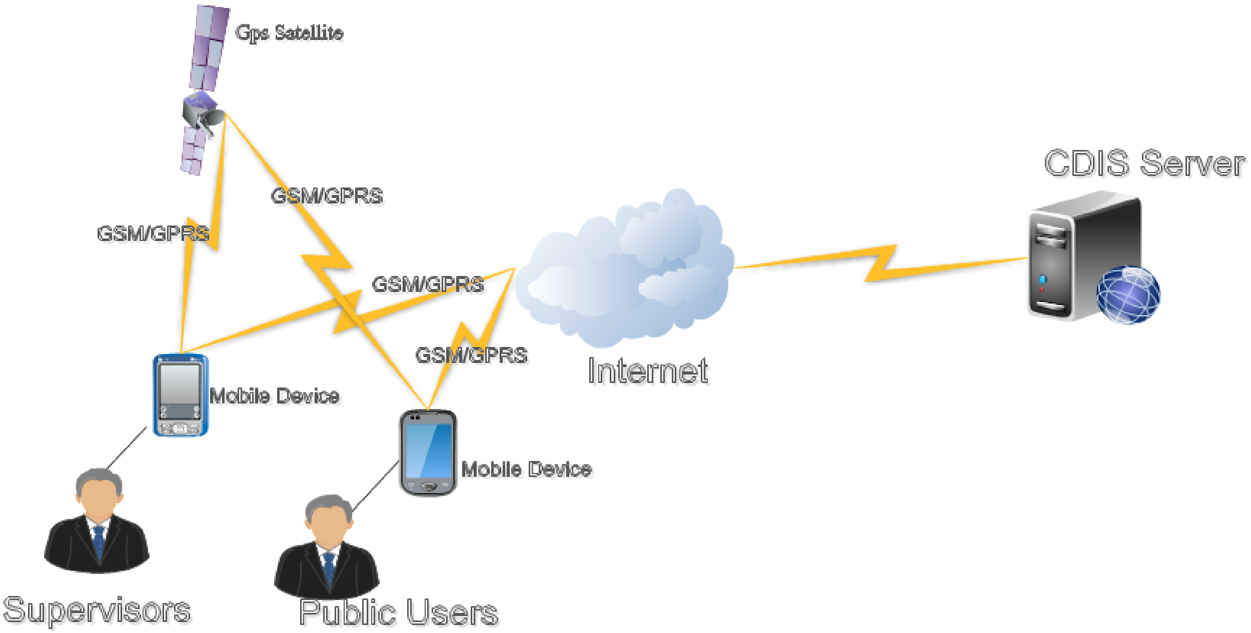


Figure 16 showing CDIS proposed architecture (Source: author 2014)

Any person with a smart mobile device is able to access and use this system considering if it is approved and put up. When a request to authenticate a drug either comes from the supervisor or the public users, first of all the smart phones queries for content from the CDIS server, were the main system will be located then content is sent back to the mobile devices and the users will be able to see the contents that he or she requested for from the CDIS system, after using the CDIS service with permission from the user, geographical location of the user is pin pointed using the GPRS services on the mobile phones, and that information is sent to CDIS server. The statistical server is available for gathering statistics about counterfeit drugs which might be useful to external system, and this will only be possible with the help of database systems that are available on the CDIS server. OpenCV will reside on the CDIS server and will help in the identification of counterfeit drugs.

Scheduling and Work Plan

The work plan and the Gantt chart are shown in the appendix under section scheduling and work plan (figure 43 - 45):

The CDIS system is using a software development methodology called mobile-D, and when constructing the system the following technologies were used, OpenCV for visioning on the camera using the phone, MSSQL database for storing data on the server and the development languages that will be used will be Java and C#, Finally it has given a schedule of the project using a schedule plan and a Gantt chart.

5.0 ANALYSIS AND DESIGN

Object-oriented analysis and design (OOAD) is a popular technical approach to analyzing, designing an application, scheme , or occupation by applying the object-oriented prototype and visual modeling throughout the developing life rhythm to foster better stakeholder communication and product quality (Sommerville, 2011).

Systems Analysis

Systems analysis is the dissection of a system into its component pieces for purposes of studying how those component pieces interact and work (Boehm, 2000).System analysis involves gathering of information, information such as finding out the business processes, gathering operational data.

This is mostly done by system analysts and there major objective is to develop a system or give birth to a system that satisfies the users need and has potential to grow as the organization grows, this process continues until a preferred and acceptable solution emerges.

Systems Design

After all the requirements have been gathered, the system has been studied and all the user requirements have been gathered, now the system can be designed. According to (Poulin, 1997) Systems design is highly dependent on the user requirements. The logical system design is a resultant of systems analysis, which is converted into physical system design.

Analysis and Design Tools

There are a number of tools that were used to model the system and these are as follows:

Use Cases

Use cases are a requirements discovery technique that were first introduced in the Objectory method (Jacobson et al., 1993). In this project the following use cases were discovered, Enter

Code which is describes the user authenticating a drug by entering a code, the other case identified is taking a picture were a user authenticates a drug using the camera, other cases like send location have been identified and many others as shown in the diagram below which describes the CDIS uses case diagram.

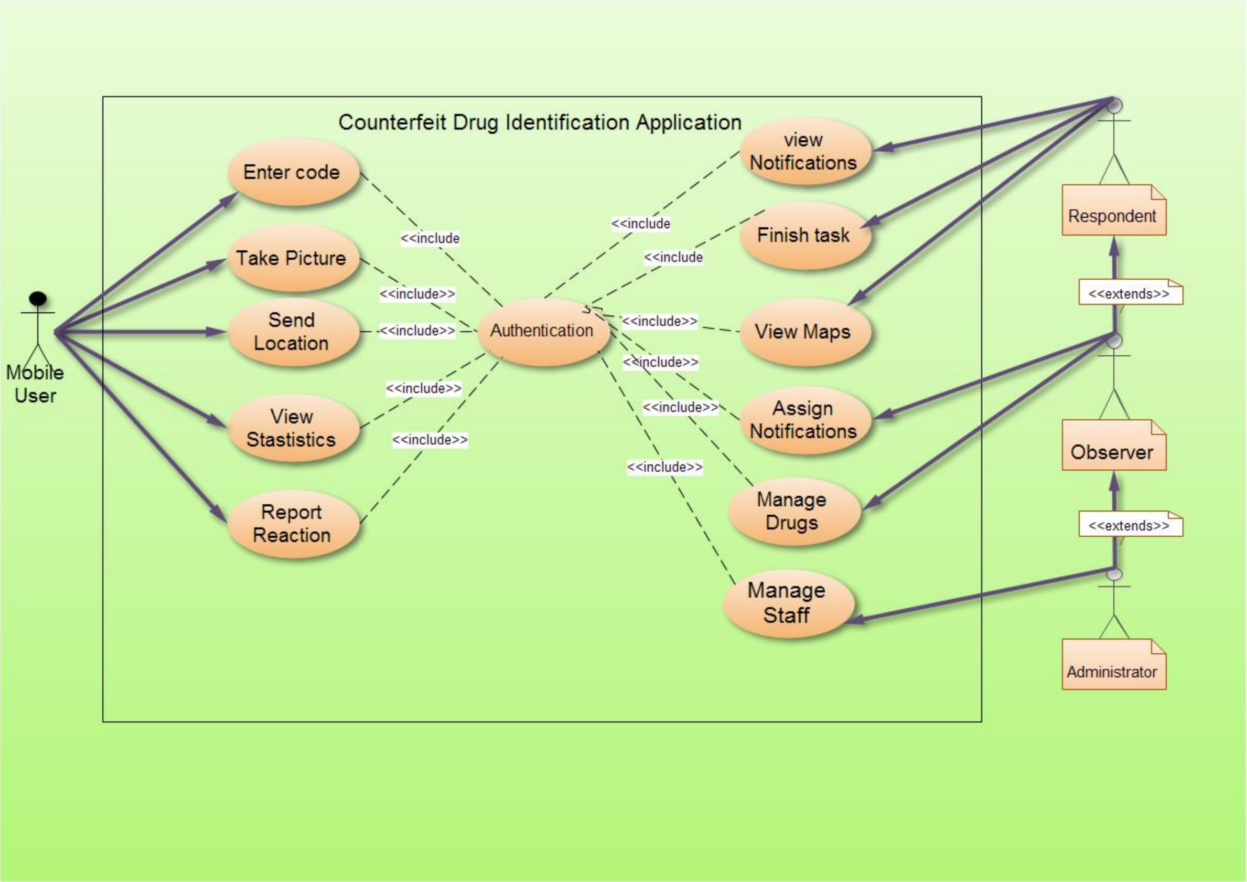


Figure 17: showing use case diagram for CDIS

The above diagram shows the systems use case and shows all the actors that are in the system.

3. 4.3.2 Scenarios

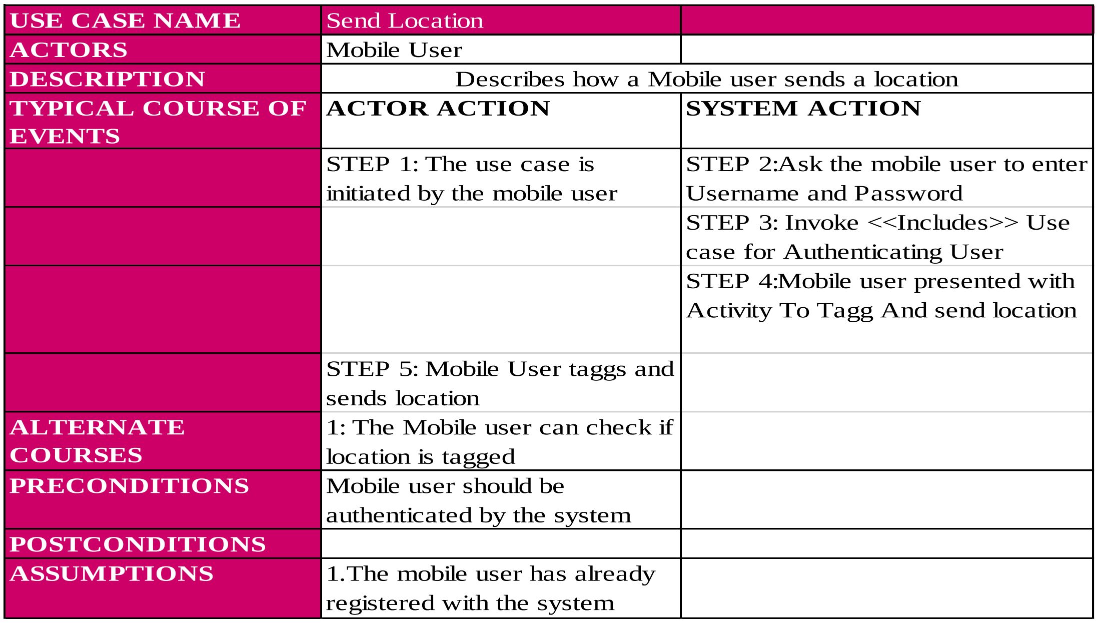
Scenarios are a deep representation of each action of the use case, these scenarios aid in the development of object sequences diagrams. Each uses case above was taken and a scenario was generated, below are the scenarios for the CDIS.

USE CASE NAME	Enter code	
ACTORS	Mobile User	
DESCRIPTION	Describes how a user enter a code to authenticate it	
TYPICAL COURSE OF EVENTS	**ACTOR ACTION**	**SYSTEM ACTION**
	STEP 1: The use case is initiated by the mobile user	STEP 2:Ask the mobile user to enter Username and Password
		STEP 3: Invoke <<Includes>> Use case for Authenticating User
		STEP 4:Mobile user presented with Activity To enter product And code
	STEP 5: Mobile User enters Name of product and code	Step 6:system verifies product name and code
		STEP 7: Mobile User provided with response
ALTERNATE COURSES	1. Report counterfeit product	
PRECONDITIONS	Mobile user should be authenticated	
POSTCONDITIONS	Results about the drug are shown	
ASSUMPTIONS	1.The mobile user has already registered with the system	

Figure 18: Enter code scenario

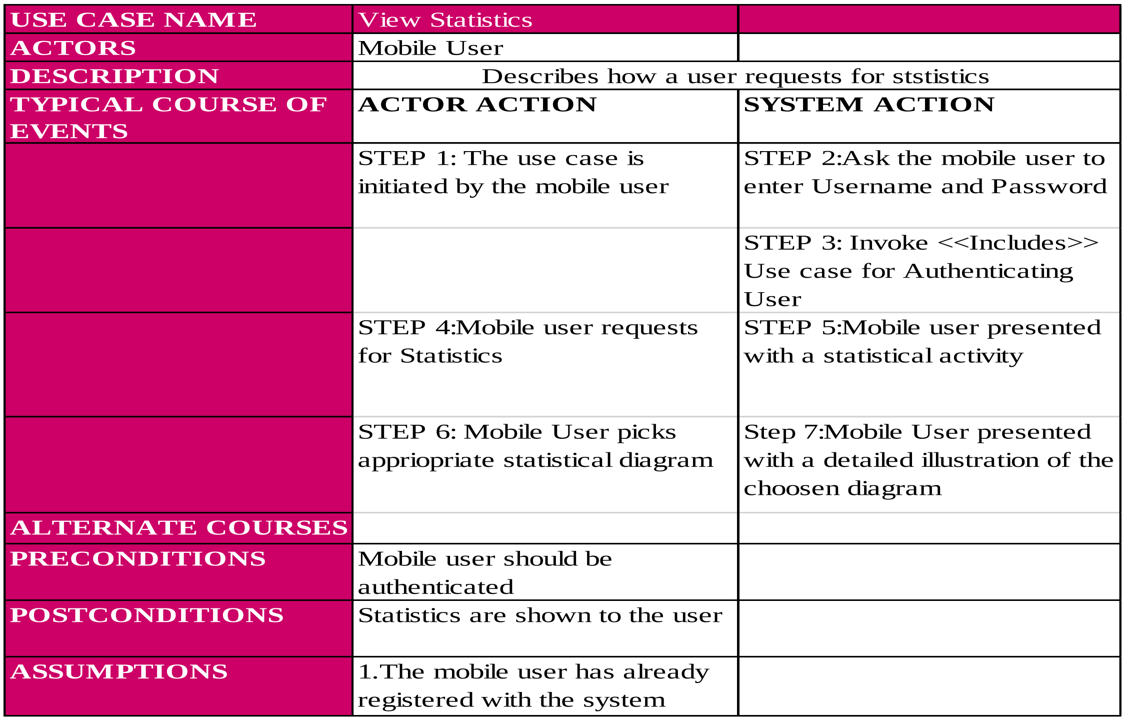
USE CASE NAME	Take Picture	
ACTORS	Mobile User	
DESCRIPTION	Describes how a user take a picture to authenticate it	
TYPICAL COURSE OF EVENTS	**ACTOR ACTION**	**SYSTEM ACTION**
	STEP 1: The use case is initiated by the mobile user	STEP 2:Ask the mobile user to enter Username and Password
		STEP 3: Invoke <<Includes>> Use case for Authenticating User
		STEP 4:Mobile user presented with Activity To enter product
	STEP 5: Mobile User enters Name of product	Step 6:Mobile User presented with Camera
	STEP 7: Mobile User Takes picture	
PRECONDITIONS	Mobile user should be authenticated	
POSTCONDITIONS	Results about the picture are shown	
ASSUMPTIONS	1.The mobile user has already registered with the system	

Figure 19 Use case for taking a picture



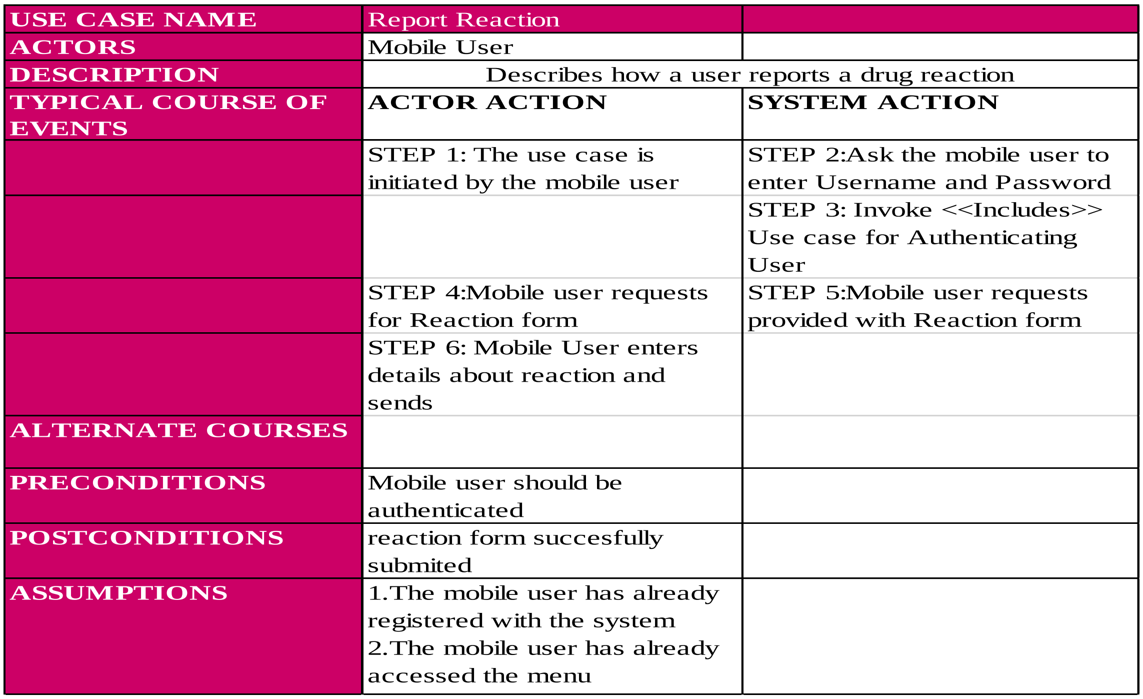
USE CASE NAME	Send Location	
ACTORS	Mobile User	
DESCRIPTION	Describes how a Mobile user sends a location	
TYPICAL COURSE OF EVENTS	ACTOR ACTION	SYSTEM ACTION
	STEP 1: The use case is initiated by the mobile user	STEP 2:Ask the mobile user to enter Username and Password
		STEP 3: Invoke <<Includes>> Use case for Authenticating User
		STEP 4:Mobile user presented with Activity To Tagg And send location
	STEP 5: Mobile User taggs and sends location	
ALTERNATE COURSES	1: The Mobile user can check if location is tagged	
PRECONDITIONS	Mobile user should be authenticated by the system	
POSTCONDITIONS		
ASSUMPTIONS	1.The mobile user has already registered with the system	

Figure 20: Use case for sending location



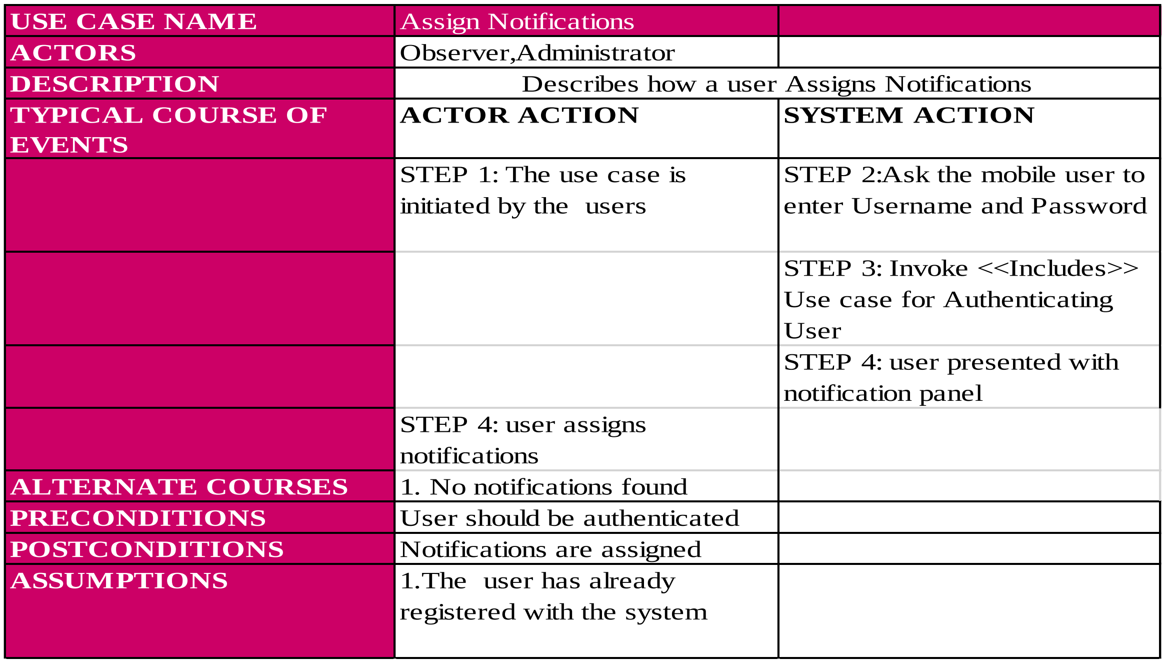
USE CASE NAME	View Statistics	
ACTORS	Mobile User	
DESCRIPTION	Describes how a user requests for ststistics	
TYPICAL COURSE OF EVENTS	ACTOR ACTION	SYSTEM ACTION
	STEP 1: The use case is initiated by the mobile user	STEP 2:Ask the mobile user to enter Username and Password
		STEP 3: Invoke <<Includes>> Use case for Authenticating User
	STEP 4:Mobile user requests for Statistics	STEP 5:Mobile user presented with a statistical activity
	STEP 6: Mobile User picks appriopriate statistical diagram	Step 7:Mobile User presented with a detailed illustration of the choosen diagram
ALTERNATE COURSES		
PRECONDITIONS	Mobile user should be authenticated	
POSTCONDITIONS	Statistics are shown to the user	
ASSUMPTIONS	1.The mobile user has already registered with the system	

Figure 21: Use case for viewing statistics



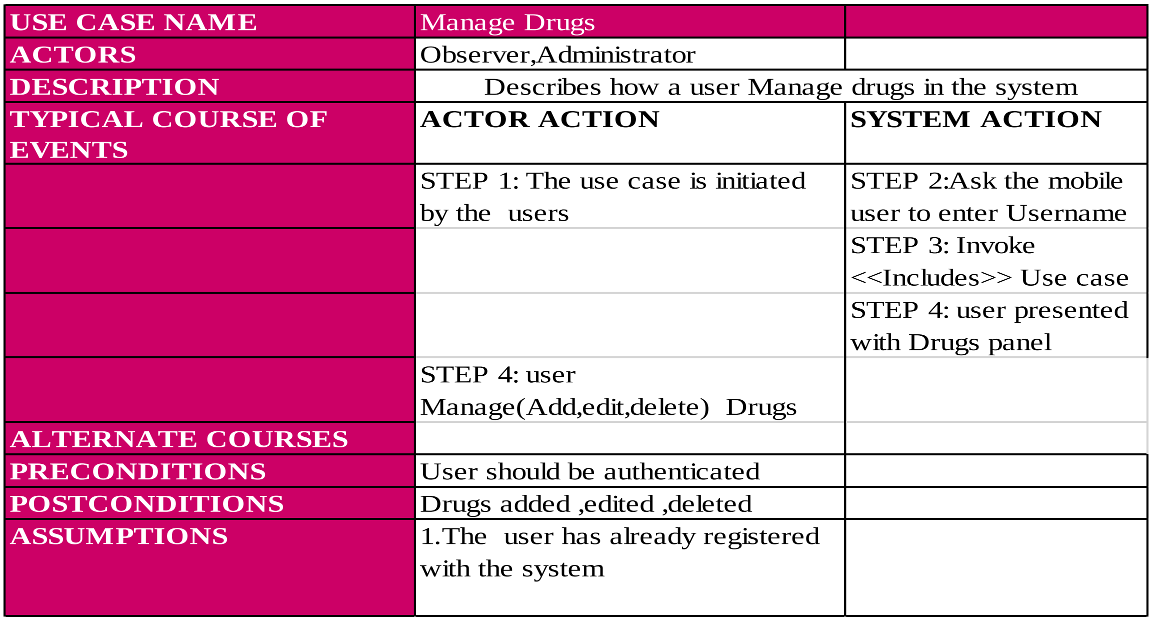
USE CASE NAME	Report Reaction	
ACTORS	Mobile User	
DESCRIPTION	Describes how a user reports a drug reaction	
TYPICAL COURSE OF EVENTS	ACTOR ACTION	SYSTEM ACTION
	STEP 1: The use case is initiated by the mobile user	STEP 2:Ask the mobile user to enter Username and Password
		STEP 3: Invoke <<Includes>> Use case for Authenticating User
	STEP 4:Mobile user requests for Reaction form	STEP 5:Mobile user requests provided with Reaction form
	STEP 6: Mobile User enters details about reaction and sends	
ALTERNATE COURSES		
PRECONDITIONS	Mobile user should be authenticated	
POSTCONDITIONS	reaction form succesfully submited	
ASSUMPTIONS	1.The mobile user has already registered with the system 2.The mobile user has already accessed the menu	

Figure 22: Use case for Reporting drug reactions



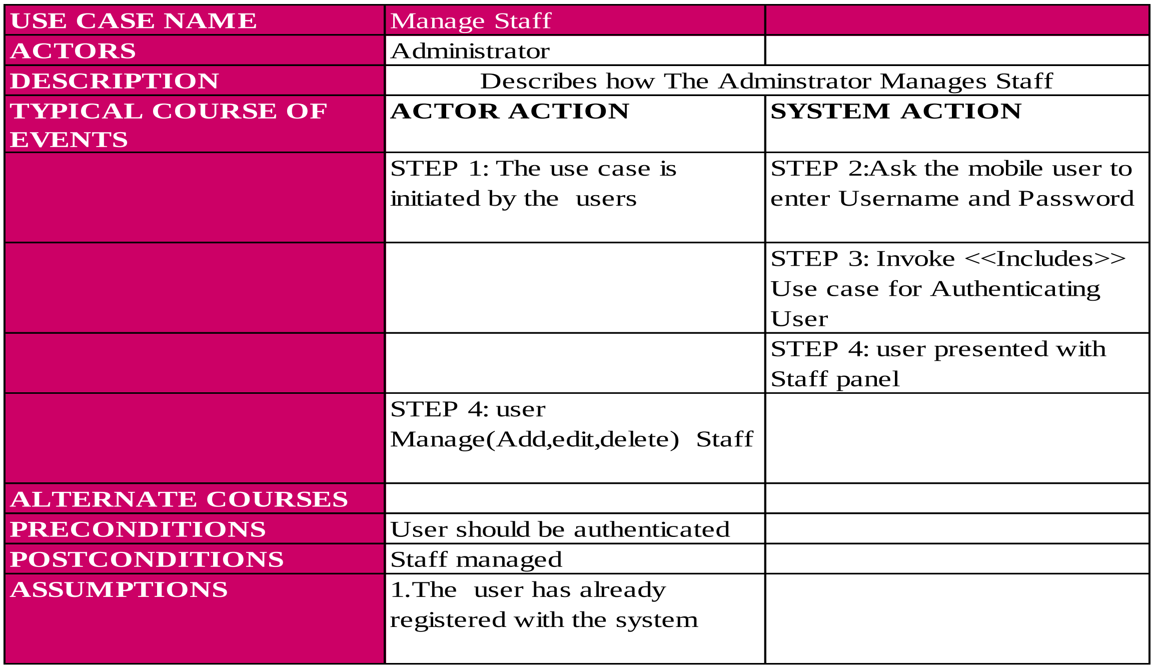
USE CASE NAME	Assign Notifications	
ACTORS	Observer,Administrator	
DESCRIPTION	Describes how a user Assigns Notifications	
TYPICAL COURSE OF EVENTS	ACTOR ACTION	SYSTEM ACTION
	STEP 1: The use case is initiated by the users	STEP 2:Ask the mobile user to enter Username and Password
		STEP 3: Invoke <<Includes>> Use case for Authenticating User
		STEP 4: user presented with notification panel
	STEP 4: user assigns notifications	
ALTERNATE COURSES	1. No notifications found	
PRECONDITIONS	User should be authenticated	
POSTCONDITIONS	Notifications are assigned	
ASSUMPTIONS	1.The user has already registered with the system	

Figure 23: Use case for assigning notifications



USE CASE NAME	Manage Drugs	
ACTORS	Observer,Administrator	
DESCRIPTION	Describes how a user Manage drugs in the system	
TYPICAL COURSE OF EVENTS	ACTOR ACTION	SYSTEM ACTION
	STEP 1: The use case is initiated by the users	STEP 2:Ask the mobile user to enter Username
		STEP 3: Invoke <<Includes>> Use case
		STEP 4: user presented with Drugs panel
	STEP 4: user Manage(Add,edit,delete) Drugs	
ALTERNATE COURSES		
PRECONDITIONS	User should be authenticated	
POSTCONDITIONS	Drugs added ,edited ,deleted	
ASSUMPTIONS	1.The user has already registered with the system	

Figure 24: Use case for managing Drugs



USE CASE NAME	Manage Staff	
ACTORS	Administrator	
DESCRIPTION	Describes how The Adminstrator Manages Staff	
TYPICAL COURSE OF EVENTS	ACTOR ACTION	SYSTEM ACTION
	STEP 1: The use case is initiated by the users	STEP 2:Ask the mobile user to enter Username and Password
		STEP 3: Invoke <<Includes>> Use case for Authenticating User
		STEP 4: user presented with Staff panel
	STEP 4: user Manage(Add,edit,delete) Staff	
ALTERNATE COURSES		
PRECONDITIONS	User should be authenticated	
POSTCONDITIONS	Staff managed	
ASSUMPTIONS	1.The user has already registered with the system	

Figure 25: Use case for Managing staff

USE CASE NAME	View Maps	
ACTORS	Repondent	
DESCRIPTION	Describes how a user view the notifications on maps	
TYPICAL COURSE OF EVENTS	**ACTOR ACTION**	**SYSTEM ACTION**
	STEP 1: The use case is initiated by the users	STEP 2:Ask the mobile user to enter Username and Password
		STEP 3: Invoke <<Includes>> Use case for Authenticating User
		STEP 4: System provides user with Map view with codinates
ALTERNATE COURSES	1. No cordinates found	
PRECONDITIONS	User should be authenticated	
POSTCONDITIONS	Task finished	
ASSUMPTIONS	1.The user has already registered with the system	

Figure 26: Use case for viewing maps

Object Sequences Diagrams

The Sequence Diagram models the collaboration of objects based on a time sequence. It shows how the objects interact with others in a particular scenario of a use case (Abrahamsson, 2002). Each of the Scenarios have to be represented into object sequence diagrams, each of the scenarios above was translated into an object sequence diagram. Below are the object sequence diagrams for CDIS.

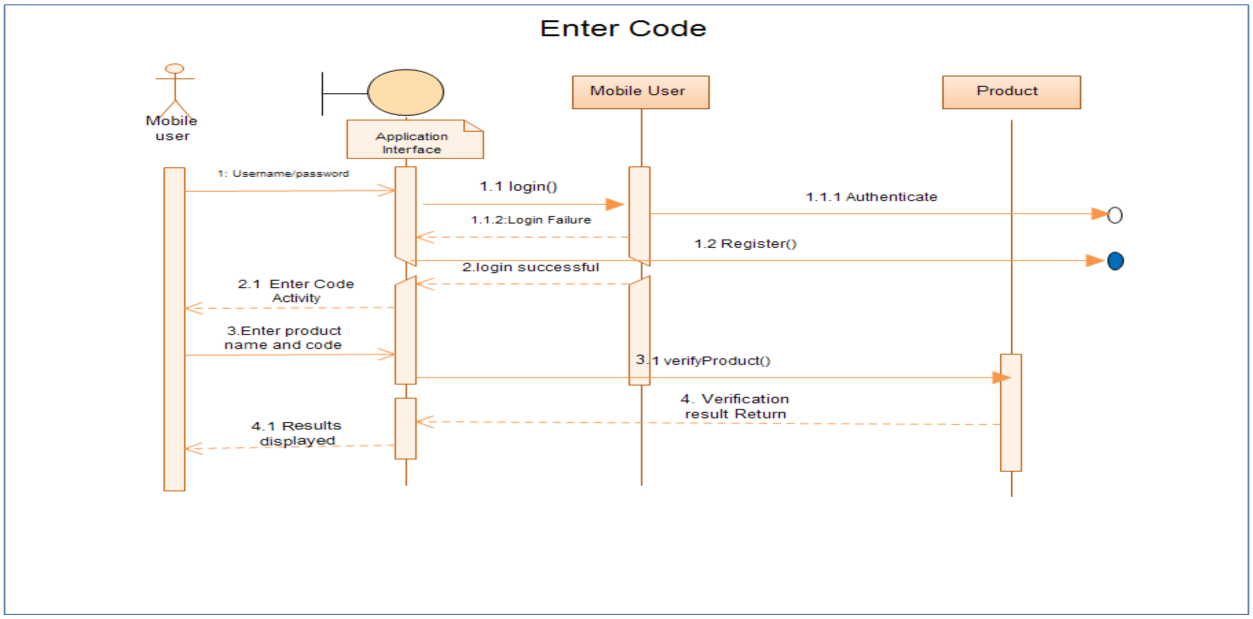


Figure 27: showing enter code Object Sequence

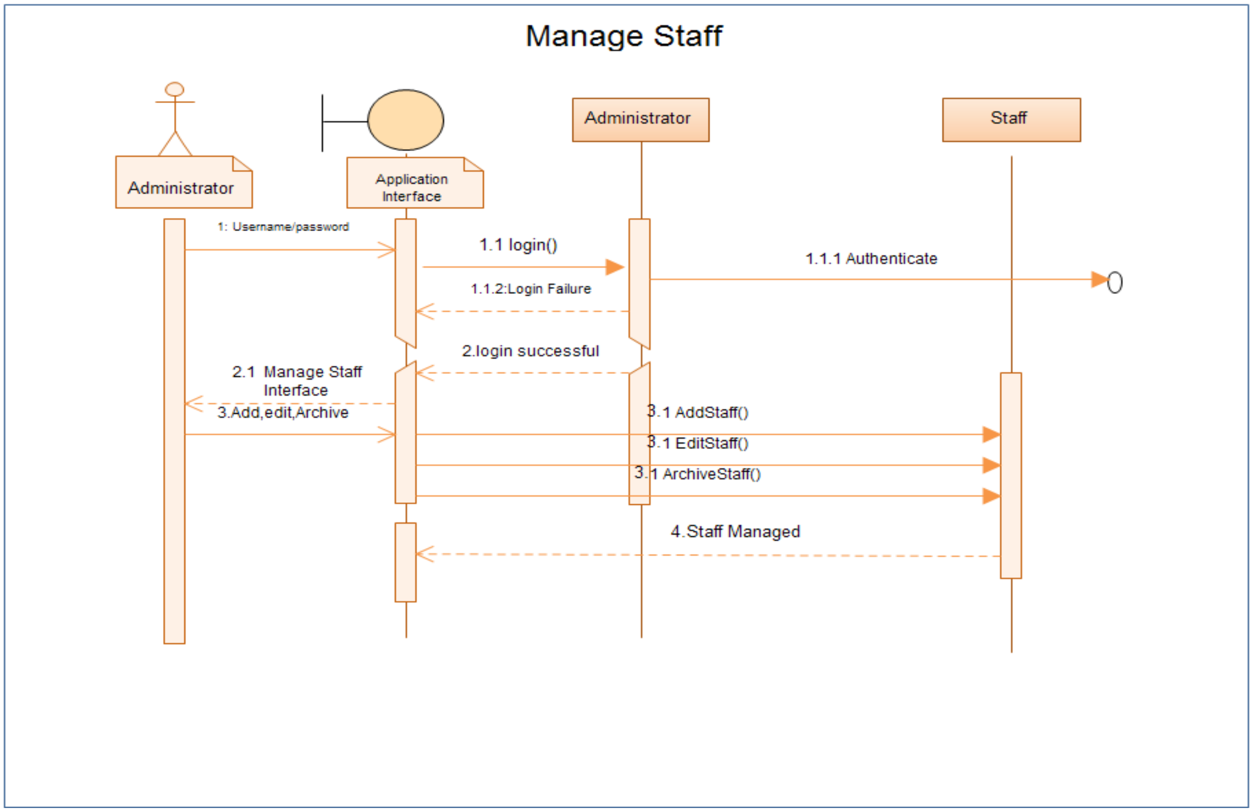


Figure 28: showing manage staff Object Sequence

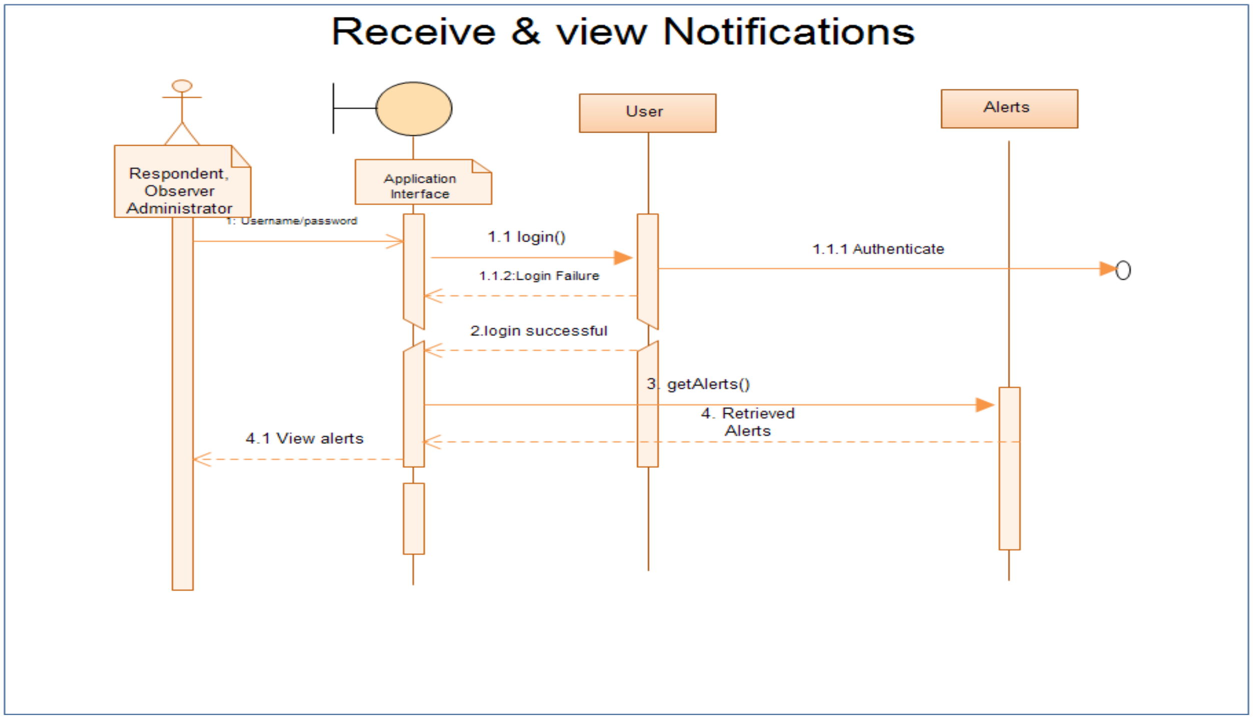


Figure 29: showing receive & view Object Sequence

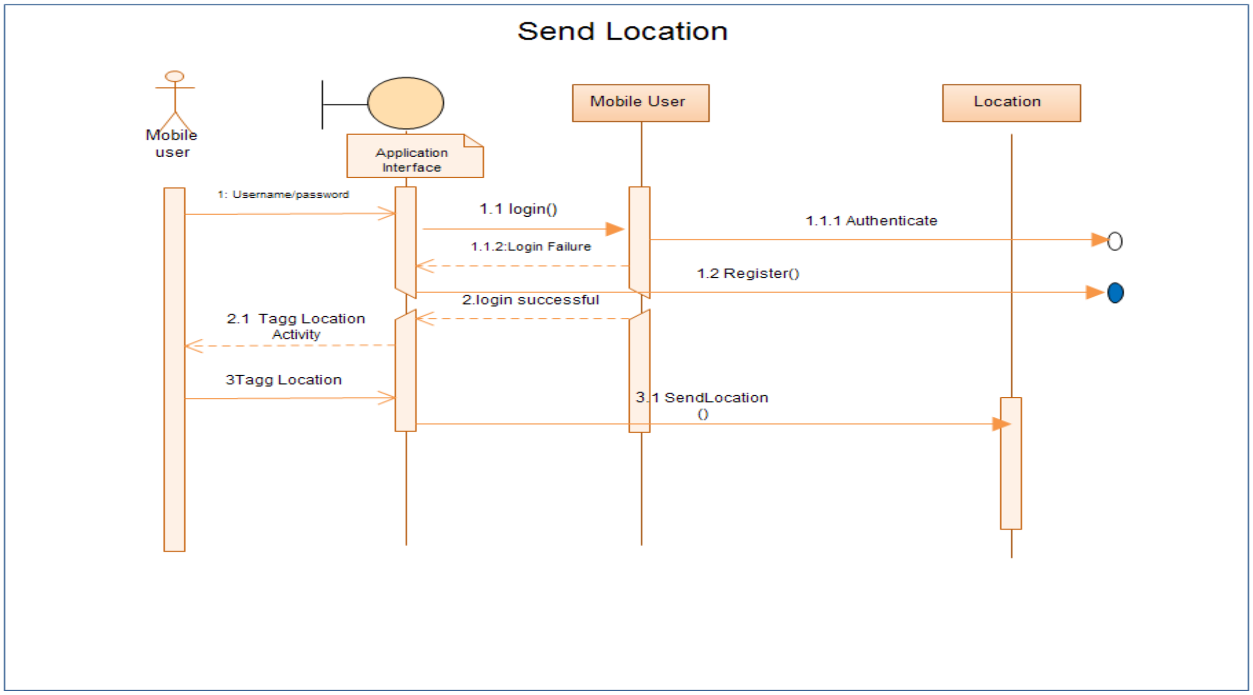


Figure 30: showing send location Object Sequence

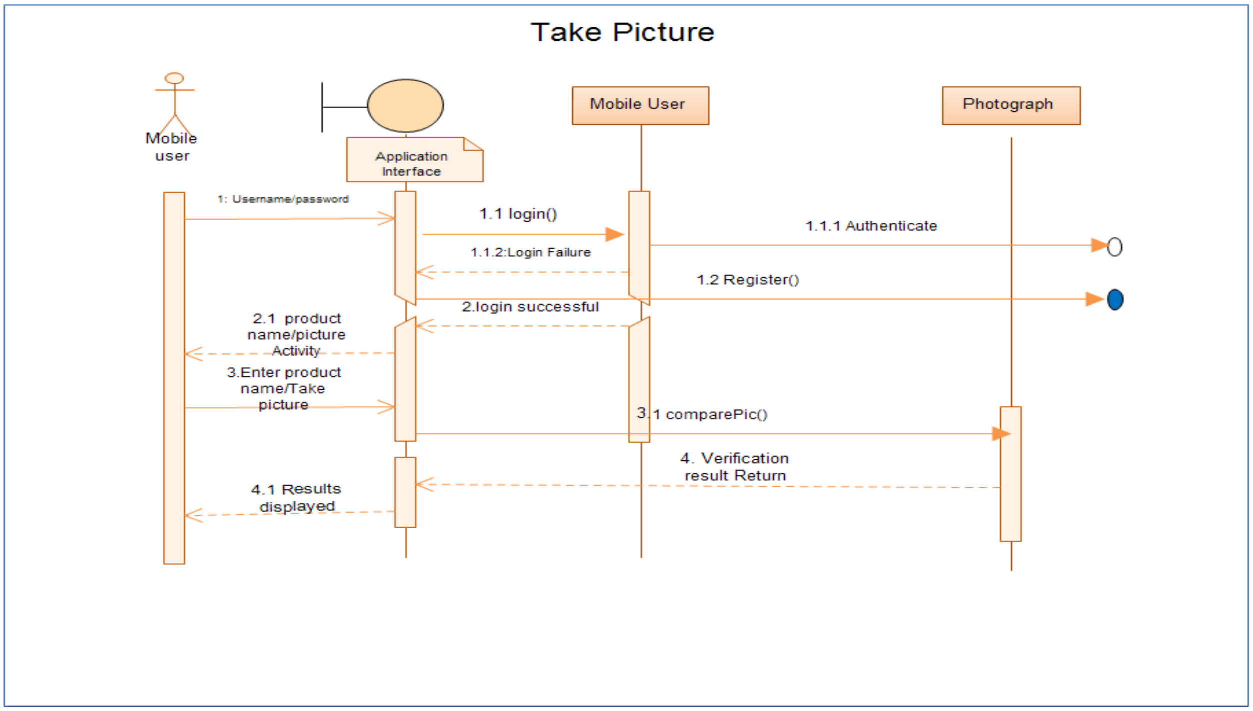


Figure 31: showing take picture Object Sequence

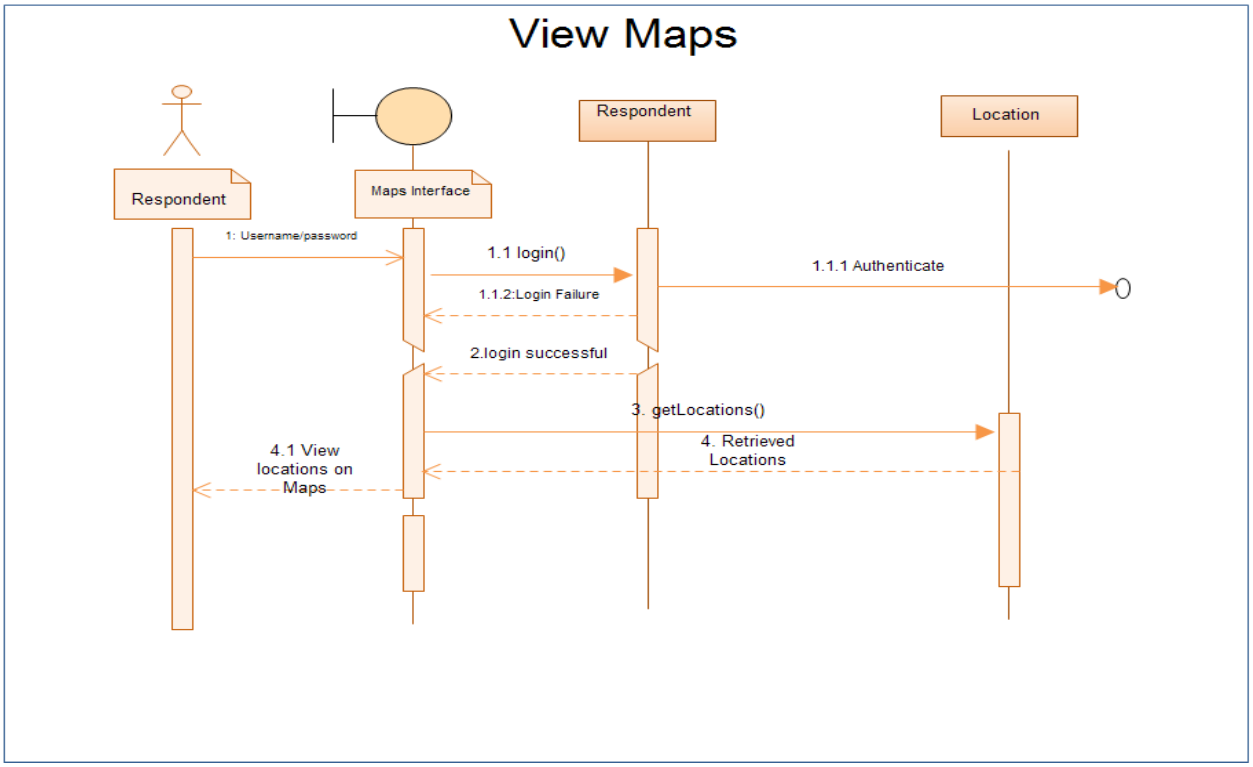


Figure 32: showing view maps Object Sequence

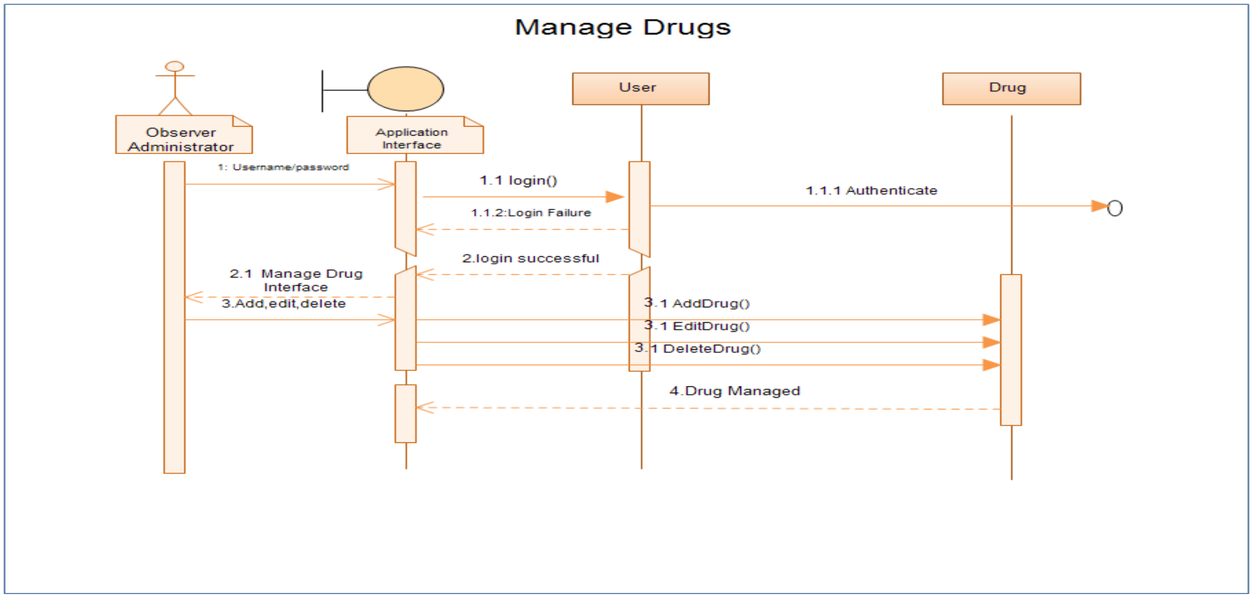


Figure 33: showing manage drugs Object Sequence

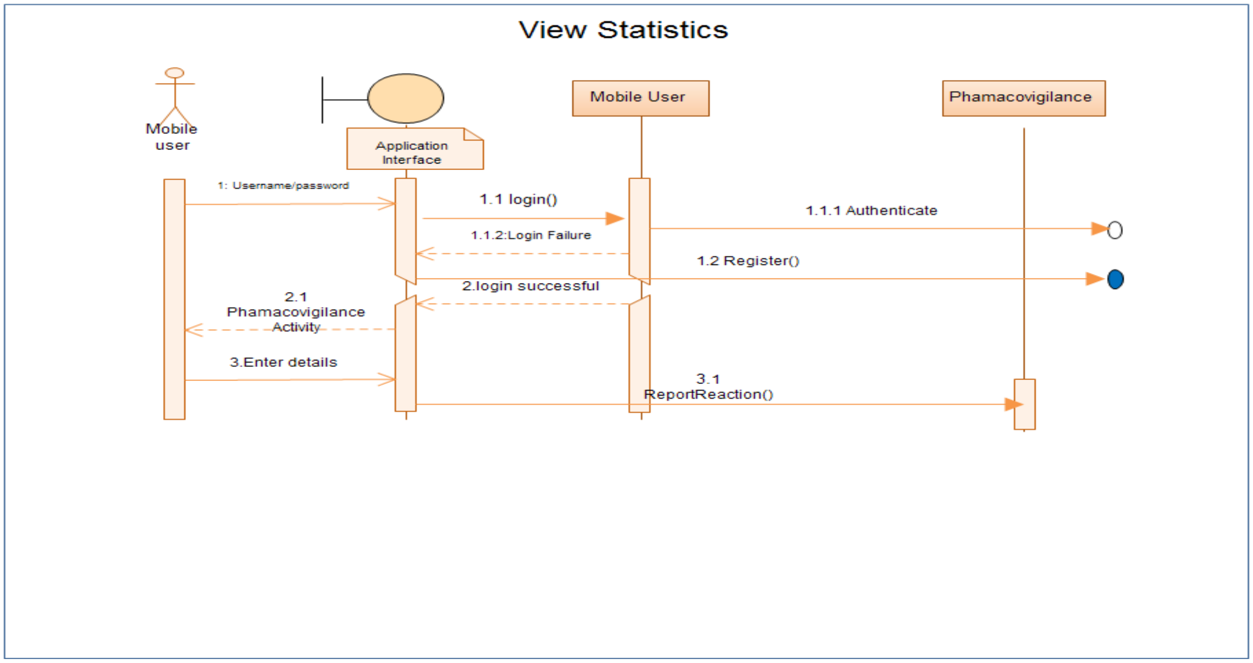


Figure 34: showing view statistics Object Sequence

Class Diagram

According to (Sommerville, 2011) a class diagram is UML diagram types that shows the object classes in a system and their relationships. From the object sequences diagrams we now develop a class diagram as represented below.

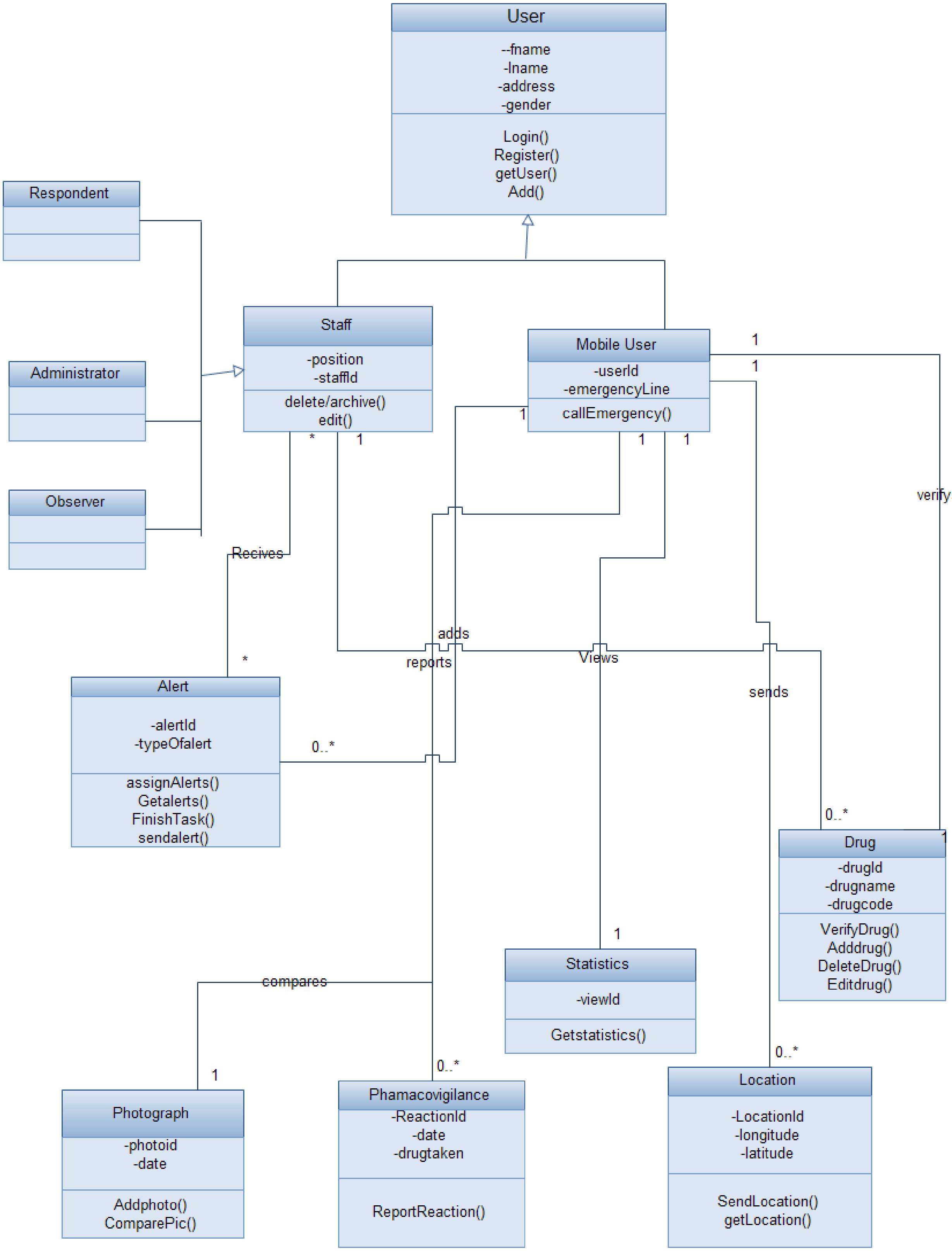

User
--fname
-lname
-address
-gender
Login()
Register()
getUser()
Add()
Respondent
Administrator
Observer
Staff
-position
-staffId
delete/archive()
edit()
Mobile User
-userId
-emergencyLine
callEmergency()
1
1
1
*
1
1
1
verifys
Recives
adds
reports
Views
sends
*
Alert
-alertId
-typeOfalert
assignAlerts()
Getalerts()
FinishTask()
sendalert()
0..*
0..*
1
Drug
-drugId
-drugname
-drugcode
VerifyDrug()
Adddrug()
DeleteDrug()
Editdrug()
1
Statistics
-viewId
Getstatistics()
compares
1
Photograph
-photoid
-date
Addphoto()
ComparePic()
0..*
Phamacovigilance
-ReactionId
-date
-drugtaken
ReportReaction()
0..*
Location
-LocationId
-longitude
-latitude
SendLocation()
getLocation()

Figure 35: showing class diagram

Entity Relationship Diagram

An Entity Relationship Diagram provides developers with an overall grasp of the data requirements, modeling and database structures of the information system before the implementation phase (Anon., 2013). The entity relationship diagram below represents the systems relations and objects and how each object interacts with other objects and the attributes that each object has.

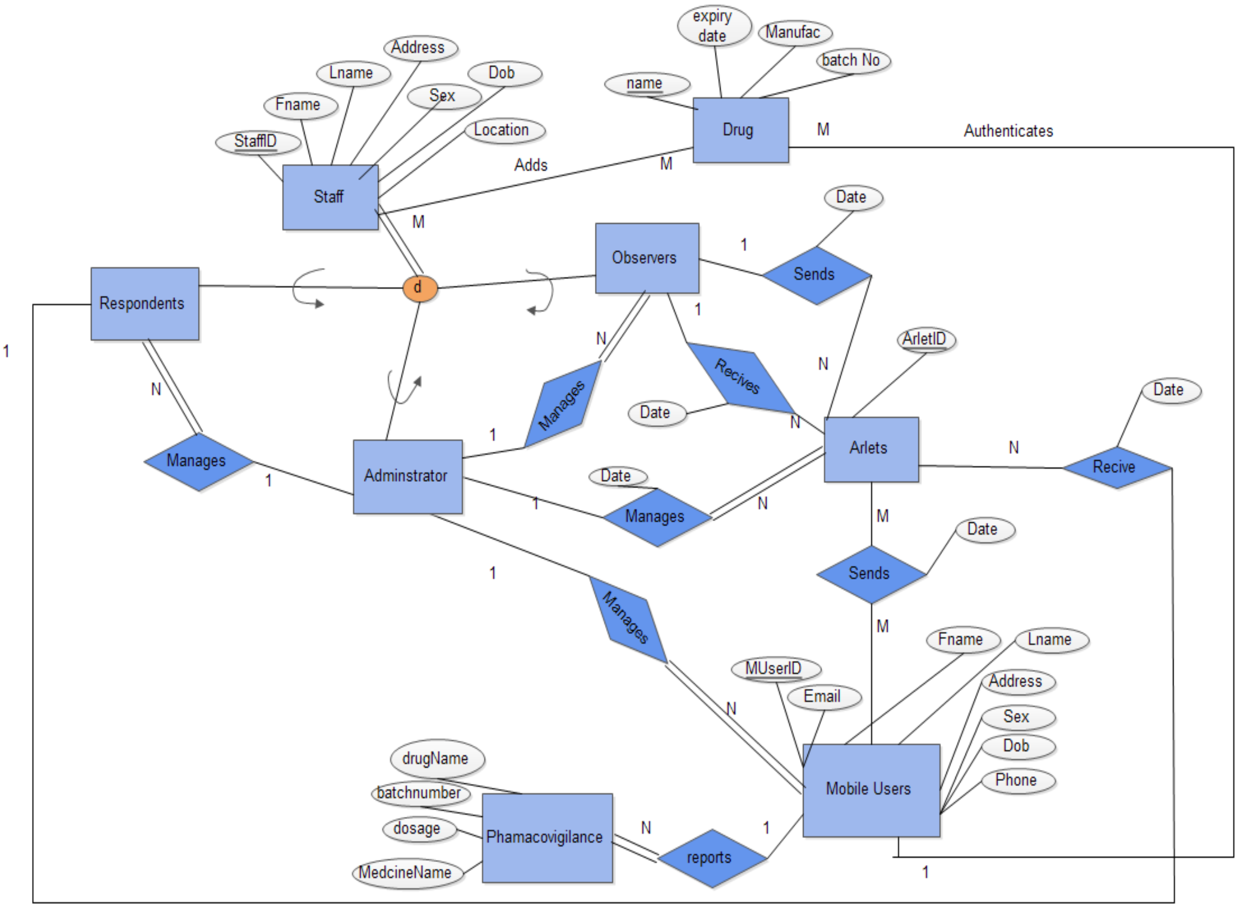


Figure 36: showing entity relationship diagram

Entity Relationship Matrix

Staff (**StaffID**, fname, lname, Adrress, Sex, Dob, position)

Mobile_users (**MuserID**, fname, lname, phone, Sex, Docnumber)

Phamaconvigilance (**MuserID**, **drugID***,batchnumber,dosage,medlenght,reasonforDrug,reaction)

Drugs(**DrugId**,Name,commoName,manufacture,)

Drug_image(**DrugId*,**imagepath)

Drugcode(**DrugID*,**drugcode,expirydate,batchcode)

Alerts(**AlertId**,MuserId*,date,latitude,longitude,status,description)

Staff_alerts(**staffID*,AlertID*,**date,status)

Data-Flow Diagram (Level 1)

Data-flow diagrams (DFDs) are system models that show a functional perspective where each transformation represents a single function or process. DFDs are used to show how data flows through a sequence of processing steps (Sommerville, 2011).

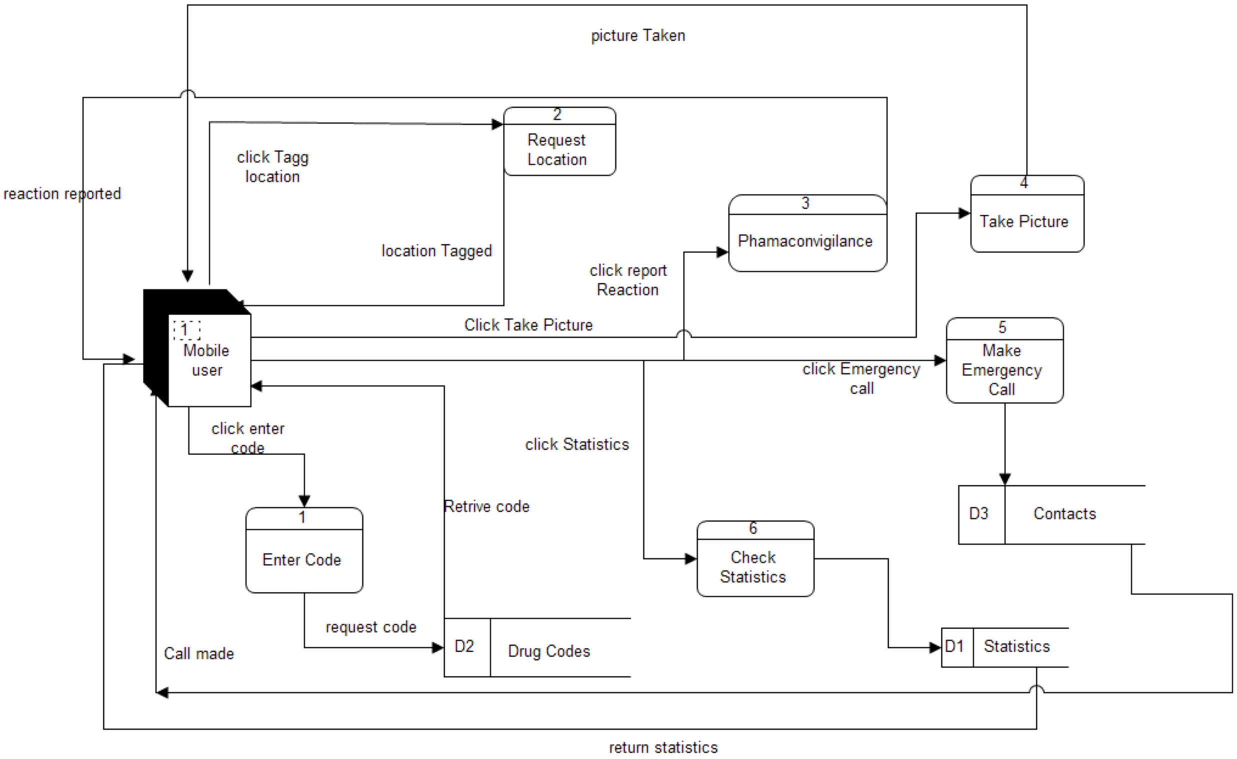


Figure 37: showing front end data flow diagram

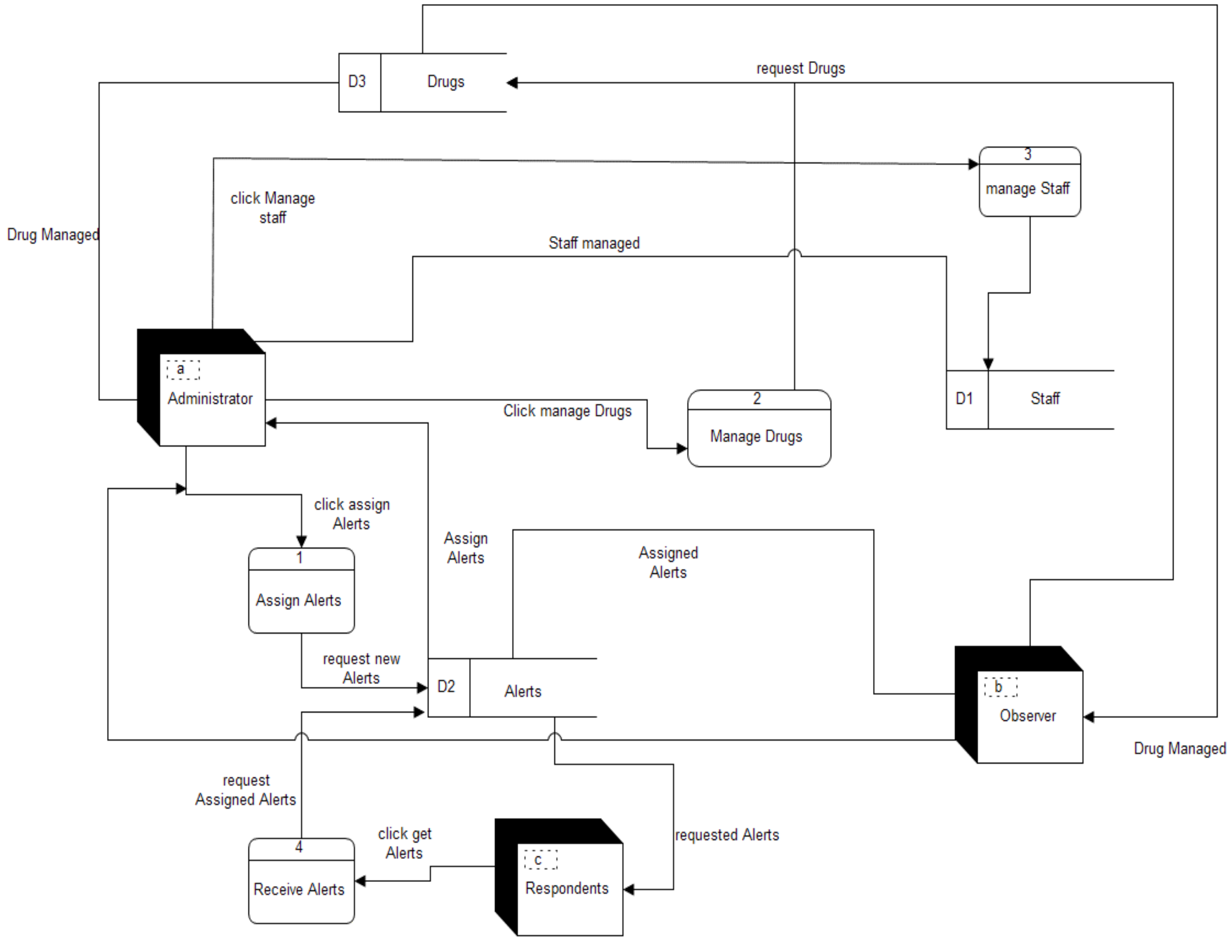


Figure 38: showing back end data flow diagram

Deployment Diagram

According to (Sommerville, 2011) UML deployment diagrams show how the systems components are distributed. The diagram below show the System deployment diagram:

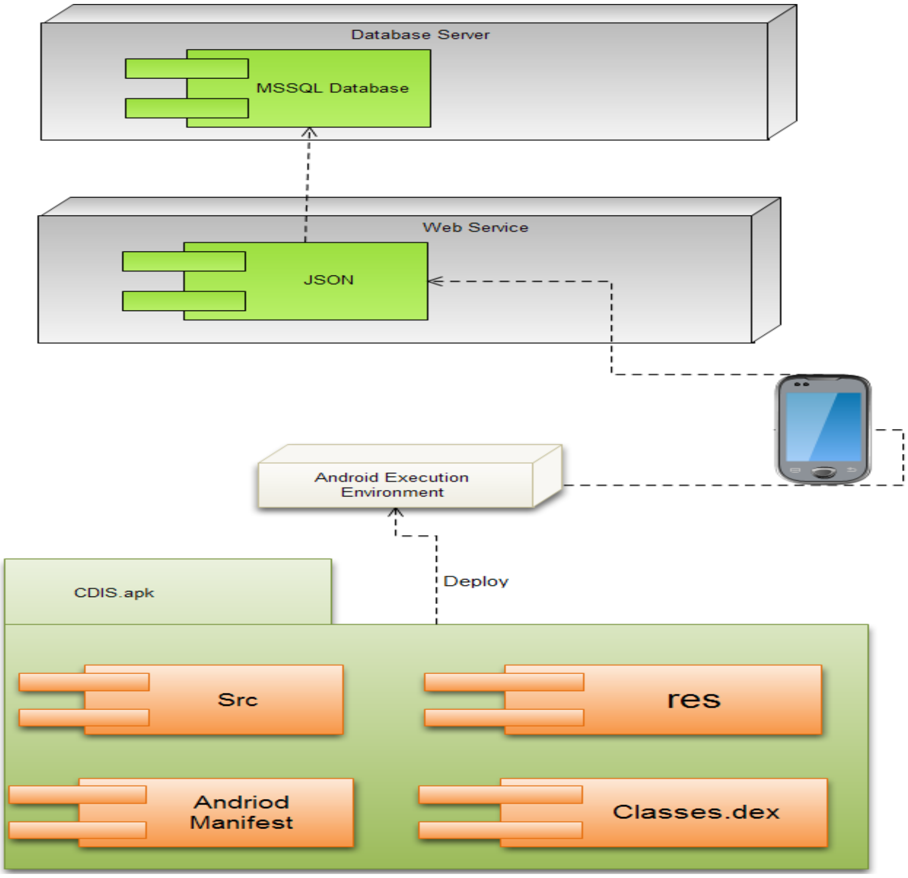


Figure 39: showing deployment diagram

Activity diagram

An Activity diagram is also known as a process model and it is used to show how a particular activity is done by a user, the activity diagram below shows the interaction between the user and the component called enter code, which is a component for verifying a drug.

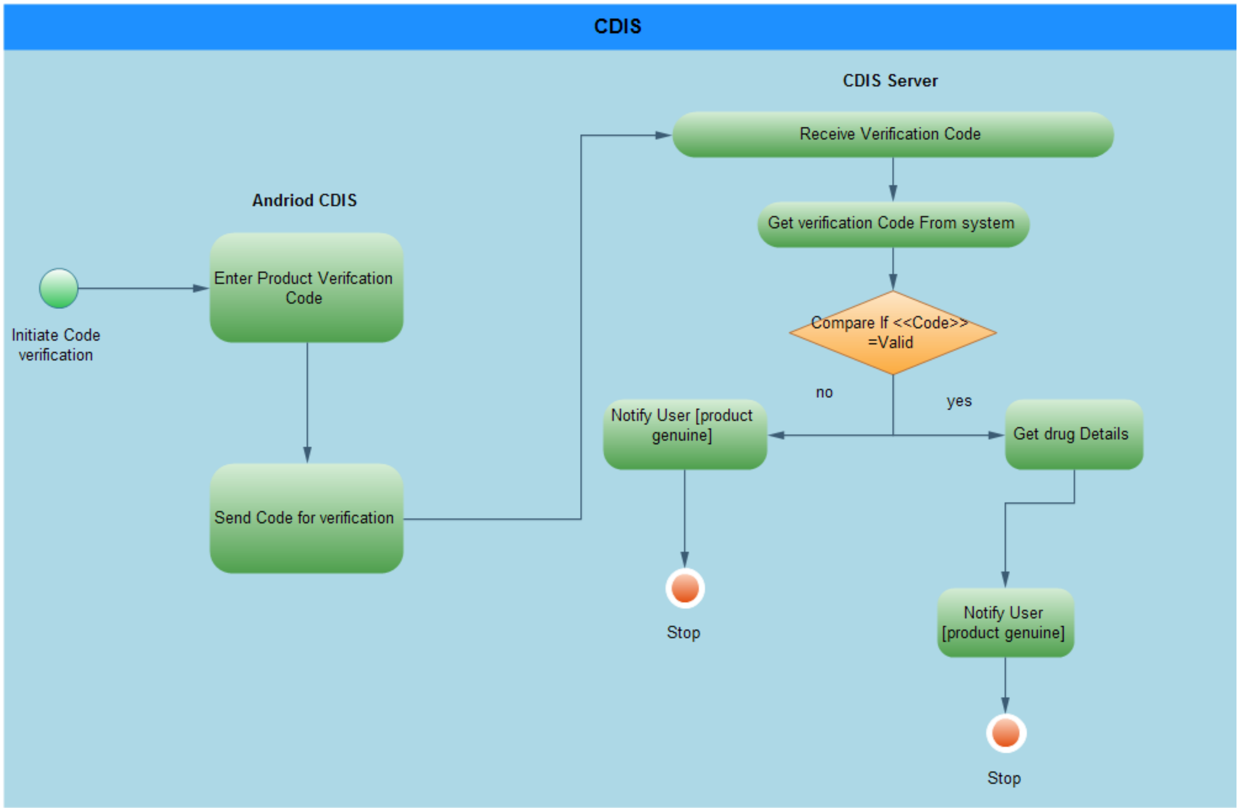

CDIS
CDIS Server
Receive Verification Code
Get verification Code From system
Compare If <<Code>> =Valid
no
yes
Notify User [product genuine]
Get drug Details
Stop
Notify User [product genuine]
Stop
Andriod CDIS
Enter Product Verifcation Code
Send Code for verification
Initiate Code verification

Figure 40: showing enter code activity diagram

In this section we have looked at different tools that were used to design the system. Uses cases were used which are requirement gathering technique, scenarios where generated from the use cases which later where useful in coming up with object sequences so that we may identify the objects present, which enabled the writer to develop a class diagram effectively. Entity relationship diagrams and entity matrixes were developed. The dataflow diagram was also drawn so that we can see the bounds of the system, a deployment diagram was developed which showed the software and the hardware part of the system and how they interact, and finally an activity diagram was done to show how users interact with the system.

6.0 DESIGNED PROTOTYPE

Testing

System testing is important before and after the system is put in place, because it helps to resolve any errors and helps to know the system better. In this chapter we are going to test the system so that we can find out if the system meets both functional and non-functional requirements. Testing can be in two forms namely white box and black box testing. White box testing or clear box testing is a way of testing a system's internal structures or how an application is working as opposed to the functionalities (Cesare Bartolini, 2011). Black box testing is a type of testing that looks at mostly the functionalities of the system rather the workings of the system (Cesare Bartolini, 2011).

Discussion of Faults and Failures

There are a number of problems in the CDIS and some of the major ones include:

- The image comparison algorithm does not fully authenticate the image, it will only authenticate an image provide the same conditions are given as the image in the system, therefore more work needs to be done.
- Tag location radius does not update as fast as it should be in the application.
- The application losses data some times and will log the user out then log them in.

Unit Testing

Unit testing is used for testing modules of the system so that it is confirmed that they meet objectives of the system that has been developed. The table below shows the test plan under user testing.

Input	Code/Method	Expected Output	Actual Output	Pass/Fail
Username, Password	Login()	Success	Success	Pass
correct Image	VerifyImage()	Original product	Image authenticated Original product	Pass
fake code	VerifyCode()	Product not authenticated	Product not authenticated	Pass
Check Tagged location	CheckLocation()	Area of interest within a radius of	Area of interest within a radius	Pass

		20m, and distance specifics.	of 20m.	
Check statistics	GetStatistics()	Statistics retrieved	Statistics retrieved	Pass
Pharmacovigilance	Reportdrug()	Successfully reported	Successfully reported	Pass
Correct Enter code	VerifyCode()	Product authenticated	Product authenticated	Pass
fake Image	VerifyImage()	Not Original product	Image Not Original product	Pass
Username, Password	Login()	Failure	Failure	Pass

Table 1: showing a test case that was followed

System Testing

Under system testing all the modules are combined and are tested as one, once these modules are combined they form a system. Under system testing we try to see how they interact with each other, in CDIS the different modules which are taking a picture, entering code, sending location and many others that have been defined above. The figures below show the main login screen and the menu which depict different components put together on one activity or page.

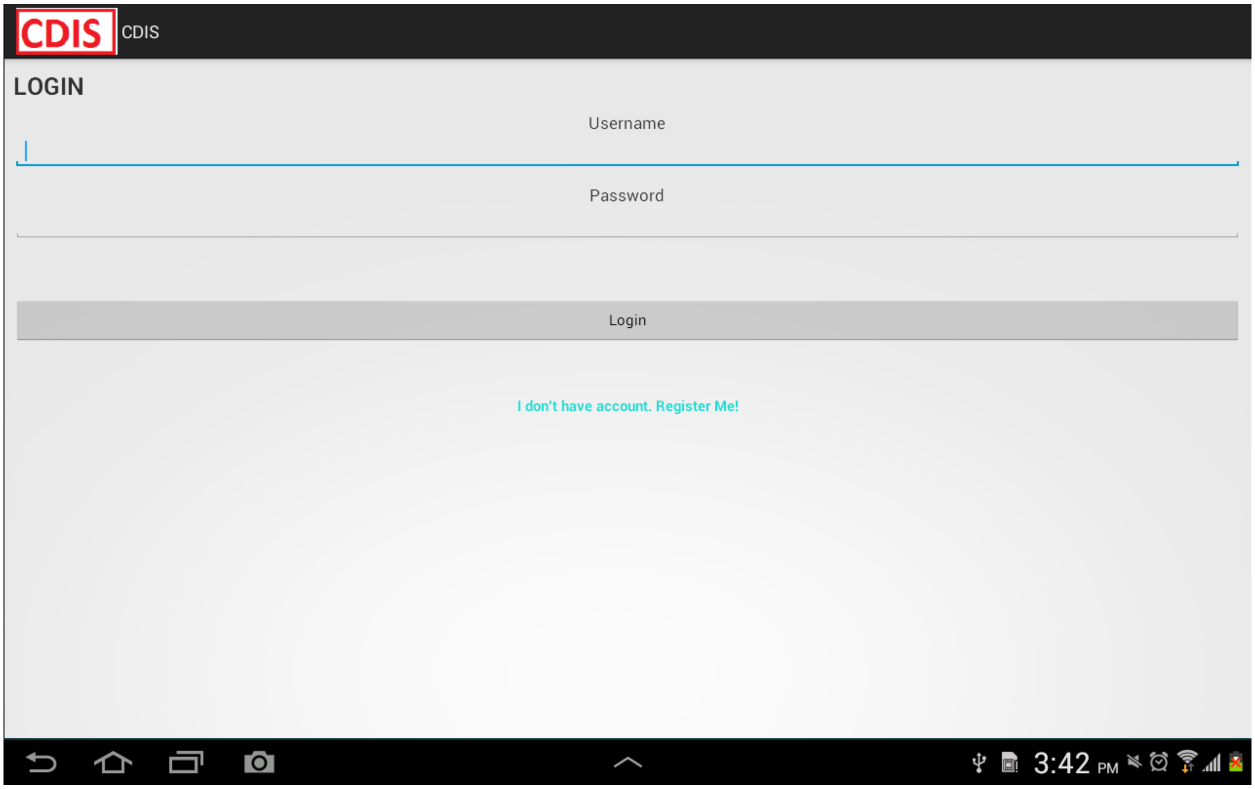


Figure 41: showing login screen

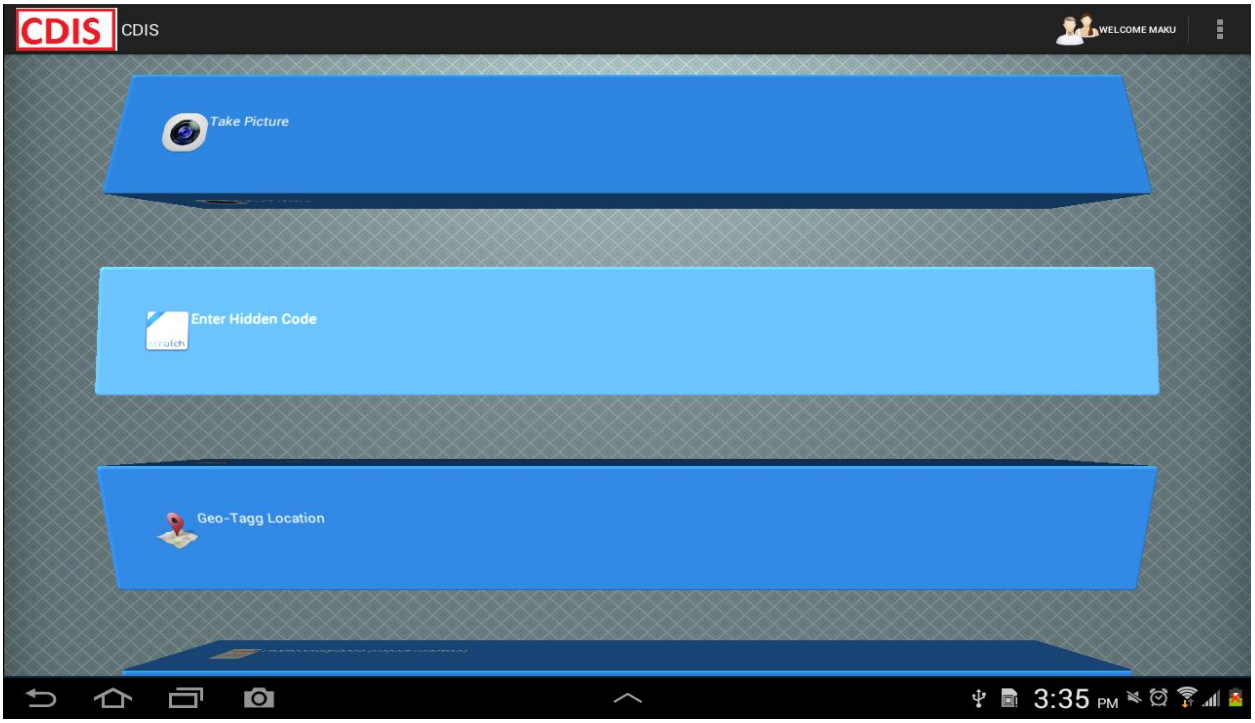


Figure 42: showing main menu for CDIS

User Acceptance Testing

Positive reviews about the application have been brought forth with the few user testing that have been done, the application was also tested at the trade fair which was held in Ndola in 2014 and many users said that the system was good.

Testing a system is very important because it allows the users and developers to correct and see how the application can perform better, it also helps the developers know exactly what the users want in the system being developed. This chapter has highlighted someone of the ways that were used to test the system, it has also highlighted the types of testing that were conducted.

7.0 CRITICAL EVALUATION

Introduction

One of the things that are important when developing a system is to measure how far the system is, therefore a qualitative analysis needs to be done. Qualitative analysis is analyzing the system to see whether the initial requirements have been fulfilled or the primary objective why the system was developed has been fulfilled. Therefore in this chapter we are going to critically evaluate what we have done and what was proposed.

Why it was undertaken

The critical evaluation was undertaken to ensure that all aspects of the system where fulfilled. The following aspects where to be made sure that they are achieved:

Objectives and user Requirements

1. The system should be able to generate statistical reports-**Achieved**
2. The system should be able to verify a product if it's a counterfeit or an original product-**Archived**
3. The system should be able to connect with other systems-**Archived**
4. The system should be able to pinpoint the location of a counterfeit product-**Archived**

Usability

1. Good interface-**Archived**
2. Easy to navigate-**Archived**

Analysis Tools

To critically analyze the system the developer used two tools to analyze the system:

1. Peer Review –The system was given to some people so that they can say what they think about the system, and there response was good, most of them said the system was good.
2. Stakeholder review- The system was also review by some of the stake holders at the trade fair in year 2015 and they gave positive responses about the system.

Fact Finding Techniques

For one to develop something there has to be facts as to why is was developed, and some of the ways these facts are gathered are through interview, recordings, observations, questioners and other tools which might not be mentioned.

In this research the developer used the following methods:

- Interviews
- Questioners

Design Process

During the designing of the system Edraw used to design Use cases, scenarios, object sequence diagrams, class diagrams, data flow diagram and many more. This process greatly helped in the implementation phase of the project.

In this section we have talked about the challenges, lessons learnt and what can be done to improve the system under development. Above all the project was a success because the achievements are more.

8.0 CONCLUSION

Mobile smart phones keep increasing in our country, more especially android smart phones and with the availability of such devices we are able to solve some of the problems are our country is experiencing. This project proposes a system where a mobile android smart phone can be used to verify the authenticity of a drug, and some of the services that the application offers are:

1. Image verification.
2. Code verification
3. Geographical tagging of location
4. Viewing of statistics: these show only ten most found drugs on the market

Therefore this is a very important application because it is applicable to the public and once it is adopted it will improve the living standards of this country.

During the development of this project the researchers came up with methods of how the application was going to be developed that is the methodology used, why we need this project, the tools that were going to be used, the designs of how the system was going to be like, and after the system was developed we began resting the application and finally we evaluated the project to see what we can learn from it and what the application is lacking so that we can plan what was needed in the future.

Therefore the following objectives were meet when developing this system by the researcher which were to mobilize and involve the stockholders to support the system, define, Analyze and propose a design architecture for CDIS, implement a CDIS system that is going to use Ge-location and produce statistical reports, using web services the client and the server should be able to communicate to transfer information to and from each other within CDIS, with all this in mind the researcher can safely say that the project is a success.

The research is relevant in that it can help the University in raising revenue especially if the research being conducted is linked to industry. The University would also benefit in terms of staff capacity building from WIPO Worldwide Academy or Ministry of Health if it takes an interest in Intellectual Property initiative.

Challenges Encountered

One of the main challenges that were faced in this project is the image comparison and verification component, this was a big problem because when someone take an image the algorithm in the background has to eliminate all surrounding obstacles and just pick the image, this was supposed to be done so that accuracy is achieved.

The other problem encountered was making the google maps working, this was so because we had to have our own API key on google.

Future Recommendations

1. The image component is still not complete and will have to be worked on in future, so that when the user is taking the image, it can be taken in any environment.
2. The user interface will have to be improved.
3. Improve the amount of time the tag radius checks a location.
4. Fix all data loses in the application.
5. Improve the amount of time the application takes to authenticate an image.
6. Make component for mobile user management

9.0 ACKNOWLEDGEMENT

We are grateful to Mulungushi University Management for giving us an opportunity to undertake this research. The financial support by Netherlands Initiative for capacity development in Higher Education (NICHE) without which this work would not have been possible. We also thank our research assistants: Lackson Simukonda students in the Centre for Information Communication Technology Education. He worked diligently and accurately. That quality of research assistantship was exemplary good. We thank him for the good work.

We also like to acknowledge Dr. Kunda:, School of Science, Engineering and Technology, Director and Prof. Yelokun: Project Director NICHE (MU) and former, Dean School of Agriculture and Nature Resources for the excellent support and knowledge delivery that they rendered to the team.

Finally, we thank the respondents who took time to complete and returned the questionnaires.

About the researchers

1. Dr Patrick Kalyabanyama Lecturer, Assistant Dean School of Business Studies, Mulungushi University
2. Mr M. Sinyinda Lecturer School of Science, Engineering and Technology, Mulungushi University

10.0 REFERENCES

[1] Abrahamsson, P. S. O. R. J. &. W. J., 2002. *Agile Software Development Metods: Review and Analysis.* [City Unknown]: VTT Electronics.

[2] Anon., 2013. Performing and analyzing non-formal inspections of entity relationship diagram (ERD). *Journal of Systems and Software,* August , Volume 86(Issue 8), p. Pp 2184–2195.

[3] Apple.Inc, 2012. *iOS Technology Overview.* [Online]
Available at:
https://developer.apple.com/library/ios/documentation/Miscellaneous/Conceptual/iPhoneOSTechOverview/Introduction/Introduction.html
[Accessed 26 October 2014].

[4] Apple.Inc, 2012. *iOS Technology Overview.* [Online]
Available at:
https://developer.apple.com/library/ios/documentation/Miscellaneous/Conceptual/iPhoneOSTechOverview/Introduction/Introduction.html [Accessed 26 October 2014].

[5] B.B. Agarwal, M. G. S. P. T., 2009. Software Engineering & Testing: An Introduction. 4 March, pp. Pg 211-215.

[6] Berg, H.-P., 2010. RISK MANAGEMENT: PROCEDURES, METHODS AND EXPERIENCES. June, Volume vol. 1, pp. pp79-95.

[7] Bilal Ahmed yaseen, M. A. T., 2012. *Comparison Between Android And iOS with respect To Their architecture,* [City Unknown]: punjab University.

[8] Boehm, B. e. a., 2000. *Software Cost Estimation with COCOMO II..* [City Unknow]: Prentice Hall PTR.

[9] Boehm, B., 1981. *Software Engineering Economics.* Los Angeles: Prentice Hall.

[10] Boehm, R. V. a. B. W., 2000. *A Systems Engineering Cost Model,* Los Angeles: [City Unknown].

[11] Cesare Bartolini, A. B., 2011. Bringing white-box testing to Service Oriented Architectures through a Service Oriented Approach. *Journal of Systems and Software,* vol 84(Issue 4), p. Pages 655–668.

[12] Chris Abts, B. C. S. D.-C., 2006. *COCOMO II Model Definition Manual,* California: University of Southern California.

[13] CSSInsight, 2014. *Global Smartphone Market Analysis and Outlook:.* [Online]
Available at: http://www.lenovo.com/transactions/pdf/CCS-Insight-Smartphone-Market-Analysis-Full-Report-07-2014.pdf [Accessed 23 October 2014]

[14] Dolan, B., 2010. *HP powers SMS-enabled pill authentication.* [Online] Available at: http://mobihealthnews.com/9717/hp-powers-sms-enabled-pill-authentication/

[15] Dr. Martin Glinz, A. M., 2003. *COCOMO (Constructive Cost Model),* Zurich: University of Zurich.

[16] Fairley, R. E. (., 2007. The influence of COCOMO on software engineering education and training. *Journal of Systems and Software,* August, Volume 80(Issue 8), pp. Pages 1201-1208.

[17] Gogo, A., 2010. *Mobile health,* Lagos: sproxil press.

[18] Kaehler, G. B. a. A., 2010. *Learning OpenCV.* Sebastopol: O'Reilly Media Inc..

[19] Le, T. M., 2009. *A STUDY ON WINDOWS MOBILE 6.5 OPERATING SYSTEM.* [Online] Available at: https://www.scribd.com/doc/22536065/A-Study-on-Windows-Mobile-6-5-Operation-System [Accessed 23 october 2014].

[20] Ligett, D., 2003. *Incremental COCOMO applied to components.* [Online] Available at: www.ifi.uzh.ch/req/courses/seminar_ws02/reports/
[Accessed 1 November 2014].

[21] Marko Gargenta, M. N., 2014. *Learning Android: Develop Mobile Apps Using Java and Eclipse.* s.l.:O'REILLY Media.

[22] Martin Fowler, J. H., 2001. The Agile Manifesto. *The LifeCycle Starts here,* August, Volume Vol. 1, pp. pp 1-7.

[23] MoH, M. O. H., 2013. *E-Health Strategy 2013-2016,* s.l.: s.n.

[24] Nisarg Gandhewar, R. S., 2010. Google Android: An Emerging Software Platform For Mobile Devices. *International Journal on Computer Science and Engineering (IJCSE),* p. p1.

[25] Ö.Özgür Tanrıövera, O. D., 2014. A process capability based assessment model for software workforce in emergent software organizations. *Computer Standards & Interfaces,* Volume Vol.37, pp. pp 29-40.

[26] PhoneGap, 2014. *Supported Features.* [Online] Available at: http://phonegap.com/about/feature/ [Accessed 21 October 2014].

[27] Poulin, J., 1997. *Measuring Software Reuse.* Reading Ma: Addison-Wesley.

[28] Robert Stern, J. C. A., 2011. REVIEW OF RISK MANAGEMENT METHODS. *Business Intelligence Journal,* vol.4(No.1), pp. pp 59-73.

[29] Rohit Ghatol, Y. P., 2012. *Beginning PhoneGap:Mobile Web Framework for JavaScript and HTML5.* [city Unknown]: Apress.

[30] Romanovsky, L., 2013. *Android Architecture For Beginners.* [Online] Available at: haifux.org/lectures/298/android.pdf [Accessed 3 November 2014].

[31] S.Balaji, 2012. WATEERFALLVs V-MODEL Vs AGILE: A COMPARATIVE STUDY ON SDLC. *International Journal of Information Technology and Business Management,* 30 June, Vol 2(Issue 1), pp. pp 26-30.

[32] Shield, J., 2014. *Cross-Platform Mobile Development: PhoneGap vs Xamarin.* [Online] Available at: http://www.mammoth.com.au/blog/cross-platform-mobile-development-phonegap-vs-xamarin [Accessed 21 October 2014].

[33] Silberman, R., 2011. *Meet Ashifi Gogo: Another Person for a Smarter Planet.* [Online] Available at: http://asmarterplanet.com/blog/2011/10/meet-ashifi-gogo.html [Accessed 17 October 2014].

[34] Singh, H., 2013. Mobile Data Collection Using an Android Device. *IJCST: Punjab Remote Sensing Centre,* Vol 4(Issues 1), pp. pp 200-202.

[35] Singh, R., 2014. An Overview of Android Operating System and Its Security Features. *Journal of Engineering Research and Applications,* February, Vol 4(Issue 2), pp. pp 517-521.

[36] Sinha, A., 2009. *Systems Design and Management (SDM),* Massachusetts: MIT.

[37] Sommerville, I., 2010. *Software Engineering.* 9th ed ed. Massachusetts: Pearson.

[38] Sommerville, I., 2011. *Software engineering..* [City Unknown]: Edwards Brothers.

[39] South, D., 2011. Southern Innovator. *Mobile Phones and Information Technology,* 11 May, Vol. 1(Issue 2), p. pp. 11.

[40] Southwick, N., 2013. *Counterfeit Drugs Kill 1 Mn People Annually: Interpol.* [Online] Available at: http://www.insightcrime.org/news-briefs/counterfeit-drugs-kill-1-million-annually-interpol [Accessed 28 September 2014].

[41] Sproxil, 2012. *Combating Counterfeit Drugs with Mobile Phones.* [Online] Available at: http://www.socialenterprisebuzz.com/2012/10/24/sproxil-uses-mobile-phones-to-combat-counterfeit-drugs/ [Accessed 11 October 2014].

[42] USAID, 2012. mHEALTH COMPENDIUM. 6 May.

[43] VTTElectronics, 2006. *Portal of Agile Software Development Methodologies..* [Online] Available at: http://virtual.vtt.fi/virtual/agile/mobiled.html [Accessed 23 October 2014].

[44] WHO, 2014. [Online]
Available at: http://www.who.int/medicines/areas/quality_safety/safety_efficacy/pharmvigi/en/
[Accessed 14 October 2014].

[45] Zax, D., 2010. *FIGHTING COUNTERFEIT DRUGS WITH MOBILE TECHNOLOGY.* [Online]
Available at: http://mhealthinfo.org/project/mobile-technology-fight-fake-drugs
[Accessed 21 October 2014].

[46] Zeynab Abbasi Khalifelu, F. S. G., 2012. Comparison and evaluation of data mining techniques with algorithmic models in software cost estimation. *Procedia Technology*, pp. Pages 65-71.

[47] ZICTA, 2014. *ICT Indicators.* [Online]
Available at: http://www.zicta.zm/index.php?option=com_content&view=article&id=58:ict-indicators&Itemid=56 [Accessed 15 October 2014].

[48] Zvonko Kremljak, C. K., 2014. Types of Risk in a System Engineering Environment and Software. *24th DAAAM International Symposium on Intelligent Manufacturing and Automation, 2013,* Volume Vol 69, pp. pp 177-183

11.0 Table of Figures