



ASHESI UNIVERSITY COLLEGE

**LOCATING THE NEAREST PHARMACY WITH THE DESIRED
MEDICINE**

APPLIED PROJECT

B.Sc. Computer Science

Brenda Akoth Mboya

2018

ASHESI UNIVERSITY COLLEGE

Locating the Nearest Pharmacy with the Desired Medicine

APPLIED PROJECT CAPSTONE

Applied project submitted to the Department of Computer Science, Ashesi
University College in partial fulfilment of the requirements for the award of
Bachelor of Science degree in Computer
Science

Brenda Akoth Mboya

April 2018

DECLARATION

I hereby declare that this applied project is the result of my own original work and that no part of it has been presented for another degree in this university or elsewhere.

Candidate's Signature:

.....

Candidate's Name:

.....

Date:

.....

I hereby declare that preparation and presentation of this applied project were supervised in accordance with the guidelines on supervision of applied project laid down by Ashesi University College.

Supervisor's Signature:

.....

Supervisor's Name:

.....

Date:

.....

ACKNOWLEDGEMENT

To all my friends who consistently motivated me during challenging moments as I did this project, I say thank you. More gratitude also to my family for their support.

I cannot forget my supervisor who was patient with me and never gave up on me. Thank you Mr. Francis Gatsi.

Special thank you to Dr. Ayorkoh Korsah, who gave me invaluable advice as I worked on this project.

Finally, yet importantly, my uttermost gratitude to God who made the successful completion of this project possible.

ABSTRACT

This project seeks to make locating medicine easier for patients. Through the proposed application, patients can search for medicine and the application would suggest the nearest pharmacy with the desired drug. In addition, patients can make medicine reservation and access directions to the pharmacy via google maps.

The proposed application also enables pharmacists to view medicine reservation from potential clients. In addition, they can update medicine availability on their drug database. They can also access the most searched drug to make better decisions on the drugs they need to have in their pharmacies. The result of this project is that patients do not need to worry about moving from one pharmacy to the other searching for medication. Just by a click of a button, they can locate the drugs they want and make a reservation for it.

Table of Contents

DECLARATION	iii
ACKNOWLEDGEMENT	iv
ABSTRACT	v
Chapter 1: Introduction	1
1.1 Background	1
1.2 Problem	2
1.3 Objective	2
1.4 Motivation and Potential Benefits	2
1.5 Related Work	4
Chapter 2: Software Requirements Analysis	6
2.1 Source of Requirements	6
2.2 User requirement.	7
2.3 System Requirements	7
2.3.1 Functional Requirements	8
2.3.2 Non-functional requirements.....	8
Chapter 3: Architecture and design.....	10
3.1 Introduction	10
3.2 Assumptions	10
3.3 System Overview	10
3.3.1 Use-case diagram	11
<i>1: Figure 3.1 use case diagram for the two users of the system</i>	11
3.3.2 Flow-chart diagram	12
<i>2 : Figure 3.2 flow chart showing how the application works</i>	12
3.4 System Architecture	13

3.4.1 Model View Architecture.....	13
3: <i>Figure 3.3 model-view-control diagram</i>	13
3.5 Database design	14
3.5.1 Entity-Relationship Diagram	14
4: <i>Figure 3.4 database Entity-Relation diagram</i>	14
3.5.2 Data Dictionary	15
1: <i>Table 3.1 Data Dictionary</i>	15
Chapter 4: Technologies and Implementation	17
4.1 Technologies Used	17
4.1.1 Materialize Cascading Style Sheet (CSS).....	17
4.1.2 Google maps API	17
4.1.3 MYSQL.....	18
4.1.4 Web Development Languages	18
4.1.5 AJAX	18
4.2 Haversine formula	19
4.3 Components of the Application	19
4.3.1 Patient dashboard for medicine search.....	20
6: <i>Figure 4.2 screenshot showing the patient dashboard</i>	21
4.3.2 nearby Pharmacies	21
7: <i>Figure 4.3 screenshot showing nearby pharmacies for a patient somewhere in Kwabenya estate</i>	21
4.3.3 Pharmacist’s dashboard.....	21
8: <i>Figure 4.4 screenshot showing pharmacist’s dashboard</i>	22
4.3.4 Closest pharmacy	22
4.3.5 Patient reservation.....	23

4.3.6 Most searched drug	23
figure 4.7 drug searches screenshot	23
Chapter 5: Testing and Results	24
5.1 Testing	24
5.2 Results	24
Chapter 6: Conclusion and Recommendation	25
6.1 Achievements	25
6.2 Challenges	25
6.3 Suggestions and future works	26
Reference	28

Chapter 1: Introduction

1.1 Background

According to a 2013 report by WHO, 80% of middle-income Africans rely on public health facilities (Pheage, 2017). This includes hospitals and pharmacies. It is critical that access to these facilities and their services are made easier especially with the advancement of technology. In a case study of the health sector in Benin with a focus on pharmaceutical delivery, Teege and Edoh (2010) highlight problems in the healthcare sector in developing countries. These include absence of health insurance, illiteracy, lack of proper delivery systems, inaccessibility of prescribed medicine and the “pharmacy tourism” problem. These researchers focused on solving the “pharmacy tourism” problem in Benin using information technology.

Teege and Edoh describe the “pharmacy tourism” problem as the scenario where patients are forced to move from one pharmacy to the other in search of prescribed medicine. The researchers attribute this issue to high prices of medicine and low demand for medicine due to substitution in the black market (Edoh & Teege, 2010). Imagine a scenario where a patient has been diagnosed of a disease and the doctor has given them the prescription. However, the hospital pharmacy does not have the desired medicine and the patient has to get it from another pharmacy. Most often, the patient has to move from one pharmacy to the other in search of the drug. This is the “pharmacy tourism” and it is often a very tedious and uncomfortable exercise especially for a sick patient.

This capstone explores the “pharmacy tourism” problem and seeks to solve it through a proposed web application. By a click of a button, patients can locate the nearest pharmacy with their desired medication.

1.2 Problem

Applications like Google maps can help users locate different facilities like hospitals or even pharmacies. However, when a patient needs a particular drug, the fact that a pharmacy is located on the map does not guarantee availability of the drug in these drug stores. Through this project, patients are assured of getting medication in a particular pharmacy.

In addition, this capstone seeks to reduce the time it takes patients to access medication. Therefore, instead of a sick patient moving from one pharmacy to another, they can use the proposed application to locate the nearest pharmacy with the drug they are looking for.

1.3 Objective

The proposed solution is a web application called “Dawa Pap”. This name comes from a Swahili phrase meaning instant medicine. The application has two main users: pharmacists and regular users (patients or anyone looking for medicine).The patient can locate all the pharmacies nearby and search for a drug. The application will suggest the nearest pharmacy with the searched medication. Once the search result is returned, the patient can reserve medication and get directions to the pharmacy.

The pharmacist can use the application to manage inventory as well as carry out analytics. In addition, they can view drug reservation made by patients and contact patients for further information.

1.4 Motivation and Potential Benefits

In a case study on m-health adoption and sustainability, Hwabamungu and Williams elaborate on m-health. They describe m-health as “ICT enabled healthcare service provision”. According to the researchers, the advantages of m-health include increased access to healthcare,

better diagnosis and disease tracking and access to medical education (Hwabamungu & Williams, 2010).

In an article on m-health, Martin et al. (2014) describe medical applications as the tools that have decentralized healthcare. This decentralization decreases financial resource required to access healthcare and increasing the adherence in the treatment and monitoring of patients. The authors further elaborate on how m-health is changing healthcare. First, m-health allows patient autonomy where patients oversee their health. Secondly, these applications are changing patient-doctor relationships since doctors can monitor their patients in real time. Thirdly, m-health is changing health management since patients can book and cancel appointments through these applications. Finally, these applications are improving patient monitoring since sensors attached to gadgets owned by patients can pick up health data like blood pressure and body temperature.

This project falls under m-health and the proposed solution is a medical application, thus the advantages discussed by these researchers apply in this context. The proposed solution, "DawaPap", would make access to medication faster. All the patient needs is access to the web application and search for the desired drug thus locate the pharmacies. In addition, the statistics on the most searched for medication will also be beneficial to the pharmaceutical industry. This information might inform drug prices and lead to more production of particular medication to increase accessibility of these drugs.

The above potential benefits of m-health applications motivated this project. In addition, my encounter with the "pharmacy tourism" on various occasions was also a motivation.

1.5 Related Work

This section analyses different literature and applications related to the project and emphasizes the importance of this capstone. A lot of research has been conducted in the health industry with the aim of making patients worry less about accessing medication. Neotech Kenya Limited, an Android developer company, released an application called “Livia Dawa”. This application allows Kenyans to purchase medicine and personal care items via their mobile devices. Users choose between having the medicine delivered to them and having it prepared and ready for collection at the nearest pharmacy. To purchase prescribed medicine, users take a picture of the prescription, select delivery or pick up point then finally make the payment. The only payment method used in the application is “M-Pesa” (a mobile money system popular in East Africa).

There also exist pharmacy locators like the “Pharmacy Finder” web application that displays a list of pharmacies near the users within a range of 10km. Imagine using the “Pharmacy Finder” to locate the nearest pharmacy but you do not find the medicine you need when you get to the pharmacy.

This is one loophole in this pharmacy locator. The “Pharmacy Finder” does not guarantee the users of the availability of the medicine in the pharmacy. This capstone addresses this by allowing patients to search for pharmacy based on the medicine they are looking for.

In Ghana, a mobile application called “Amatsii” was developed to help patients search for any drugs and pharmacies. The mobile application helps users locate pharmacies within a particular region that accepts a specific insurance. In addition, consumers can search for a particular drug and the application would suggest all the nearby pharmacies with the drug. The

proposed application in this project is different from the Amatsii application as it allows pharmacists to interact with patients through reservations. When a patient reserves medication, the pharmacist can view the patient details and get in touch with them. In addition, the graphical representation of the nearest pharmacies when a patient searches for a drug also makes this project unique.

One gap in the “Amatsii” solution is the fact that customers cannot locate the pharmacy and must rely on a delivery system. The goal of my capstone is to bridge this gap by integrating Google Maps with the final solution. Thus, a user of the application would locate the nearest pharmacy on the map. In addition, users would have the option of directly picking up the medicine from the pharmacy.

Cary Byrd founded an online pharmacy store (“ePharmacies.com”) based in San Antonio that helps patients and other pharmacies purchase prescribed medicine. The store has collaborated with various authorized pharmacies in America and Canada. Through this platform, patients can buy safe and quality medicine and have them delivered. When a buyer searches for medicine, they are directed to the highly rated online pharmacies selling safe medication. The buyer then has to make a choice from the pool of pharmacies based on their budget since different pharmacies might be selling the medication at different prices.

Chapter 2: Software Requirements Analysis

2.1 Source of Requirements

This section outlines both user and system requirement of the application. The user requirements were gathered from potential application users (pharmacists and potential patients). The main mode of gathering this data was one to one interviews. I also interviewed people in the pharmaceutical industry like pharmacists and those with a strong interest in this subject. I got most participants through recommendations from friends and lecturers who knew people in this field. In addition, my supervisor also played a vital role in the process of coming up with the user requirements. Existing m-health applications were also great resources for some of the user and system requirements.

Some of the questions asked during the interview included:

- Would you patronize an application that would help you locate the nearest pharmacy with your desired drug?
- Have you had to walk from one pharmacy to another in search of medication?
- As a pharmacist, how willing are you to patronize this application?
- Would you prefer online transaction or cash?
- Would you want delivery to be integrated with this application?

At the onset of the project, I had thought of a cashless transaction (use of visa cards/mobile money) so that a patient can order for the drug and have it delivered to them. However, after talking to some pharmacists, I realized that they were reluctant in accepting card payments. Most of them claimed that the process of getting the money from the banks that give out visa cards is tedious and in the long ran, they lose some cash. In addition, the pharmacists I spoke to were not willing to incur an extra cost of hiring a delivery person as I had imagined.

Therefore, delivery operations would have to be independent of the pharmacies. The original idea of the project was to locate the nearest pharmacy with the cheapest drug. However, during the user requirement stage, I found out that pharmacists are reluctant in giving out price information about the drugs they are selling. I attributed this to competition in the pharmaceutical industry. In addition, I found out that the difference in drug prices between most close-by pharmacies is often negligible. Thus, I had to narrow down my focus to the nearest pharmacy with the desired drug.

2.2 User requirement.

The user requirement section describes some of the things that potential users expect the application to do. The key user requirements include:

1. Searching for the pharmacies with the required medicine
2. Suggesting the closest pharmacy with the required medicine
3. The patient should be able to reserve medication at a particular pharmacy
4. The pharmacist should be able to update drug information
5. The pharmacist should view reserved medication and the most searched drug

2.3 System Requirements

The system requirements included in this document describe the functionalities the system should have. These requirements are divided into functional and non-functional requirements.

The non-functional requirements ensure the system performs efficiently while the functional requirements describe what the system should be able to do.

2.3.1 Functional Requirements

These include:

1. Access medicine availability data for all pharmacies registered in the application
2. Display correct results when a patient searches for a particular drug
3. Only have up-to-date information on the database
4. Connect to the google maps server to access the user location and get the nearest pharmacy
5. Keep track of the searched drugs to monitor pharmaceutical trends.
6. Allow users to register to use the application
7. Allow only registered users to login into the system

2.3.2 Non-functional requirements

Non-functional requirements ensure that the application is optimal, reliable and efficient. Some of these requirements are:

1. Security

To ensure database and systems security, authorization is required to access information from the system. Thus, only authorized users can successfully login into the system. Moreover, the information various users can view is different depending on the user type. The administrator of the system has full access to the database, the pharmacist can only view data of their pharmacy and the potential patient only searches for the nearest pharmacy. Patients cannot access the database and manipulate it.

2. Reliability

To ensure reliability, database back- ups have to be created for the system for example by the use of cloud technology. This technology has not been patronized in this project due to

a limited budget. Therefore, it is a recommendation to researchers interested in this subject. However, the application is to be deployed on the Ashesi server to ensure reliability.

Chapter 3: Architecture and design

3.1 Introduction

This section discusses the design of the project. It gives a detailed report on how the proposed system is designed and how it works with elaborate diagrams. These diagrams include use case diagrams, flow charts and database entity-relation diagrams. This section helps software developers understand in detail how the system works and how the various components of the system are connected.

3.2 Assumptions

In designing the “Dawa Pap” application, the below assumptions were made:

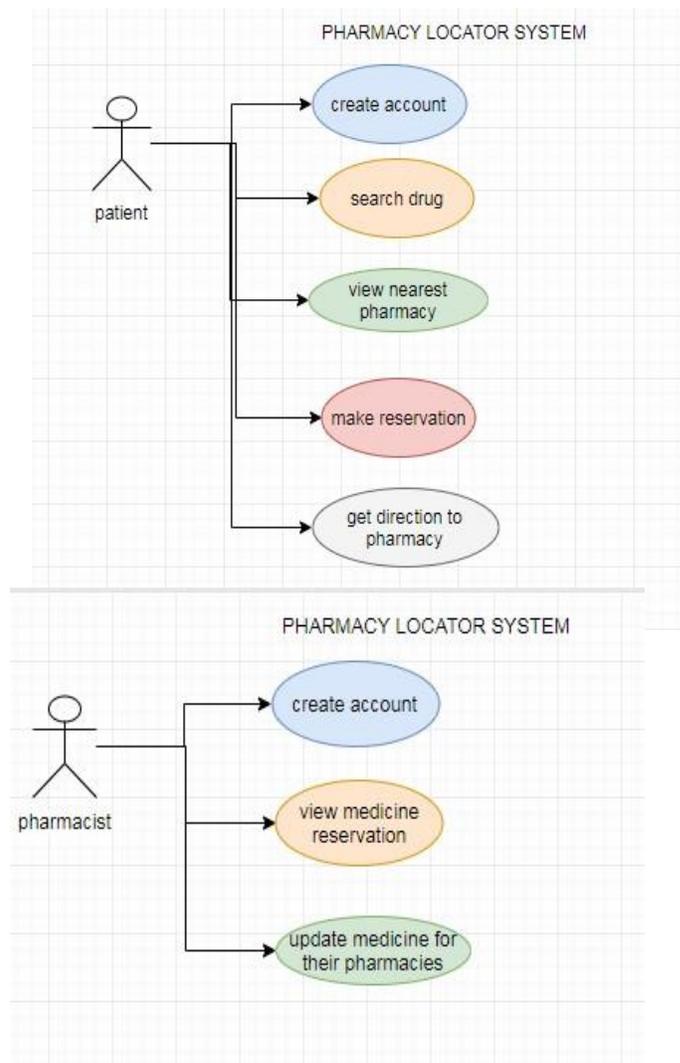
- The administrator of the application has access to the drug database of all registered pharmacies.
- The pharmacies genuinely update drug availability in their database in real time such that a patient does not reserve medicine that is not available in a particular pharmacy
- Only certified pharmacies are registered to use the application
- The users have to show up at the pharmacies with the prescription

3.3 System Overview

The application is a web application that allows patients to search for drugs and the application would suggest the closest pharmacy desired drug. The application has two main users (potential patients and the pharmacist each with their own dashboard. Patients can search for drugs and get directions to the suggested pharmacy. However, pharmacists can update the data on the drugs available in their various pharmacies as well as view medicine reservation.

3.3.1 Use-case diagram

The below diagram shows how the two users can interact with the application. After registering into the system, a patient can login and search for a drug, view nearby pharmacies, view the nearest pharmacy with the desired drug and reserve medicine. On the other hand, once the pharmacist has registered to use the system, he/she can log in and view reservations made by patients to get medication from their pharmacy. In addition, they can view the most searched drug.

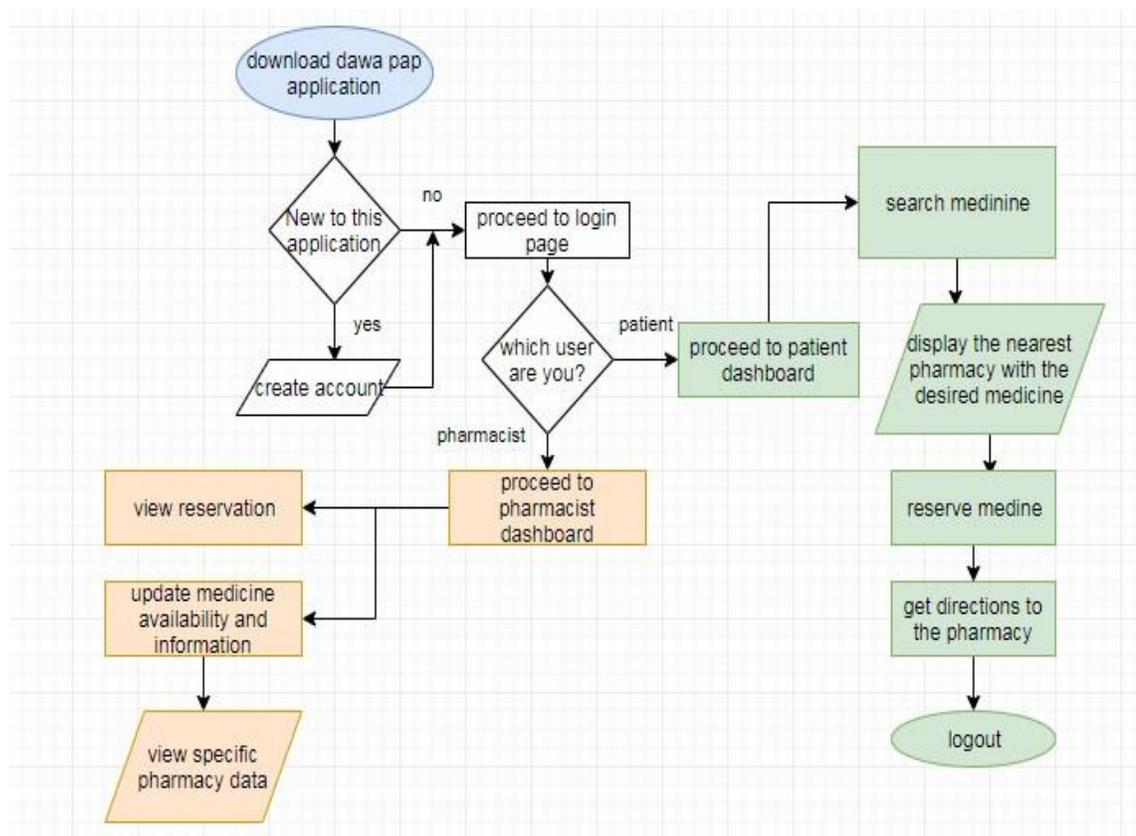


1: Figure 3.1 use case diagram for the two users of the system

3.3.2 Flow-chart diagram

The flow chart further elaborates how the application works. Internet access is needed to use the web application. On successful login, different users are directed to specific pages. Pharmacist are redirected to the pharmacist dashboard and the patients to the patient's dashboard.

Each user is then able to access the functionalities elaborated in the use case diagram above.



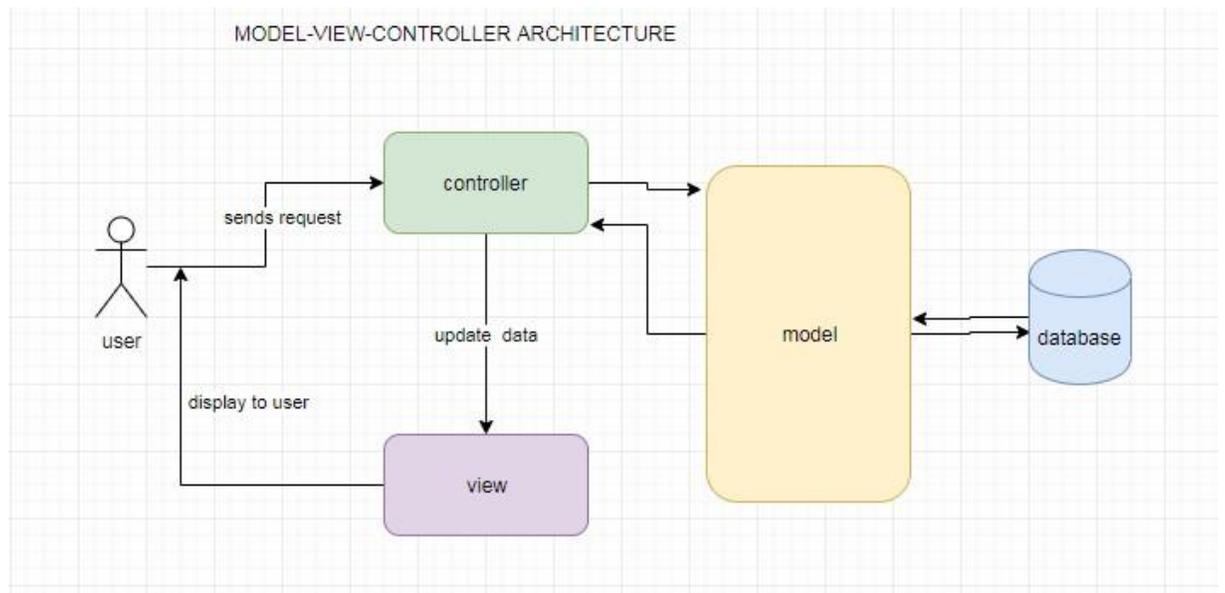
2 : Figure 3.2 flow chart showing how the application works

3.4 System Architecture

3.4.1 Model View Architecture

The model-view-controller architecture is used in designing the system. This architecture divides the system into three: view, model and controller. The login pages, registration pages, the pharmacy, and the patient dashboards represent the system views. The model represents the implementation of the database used. The controllers are the various file that link the model and the various views.

Below is a representation of the MVC architecture:

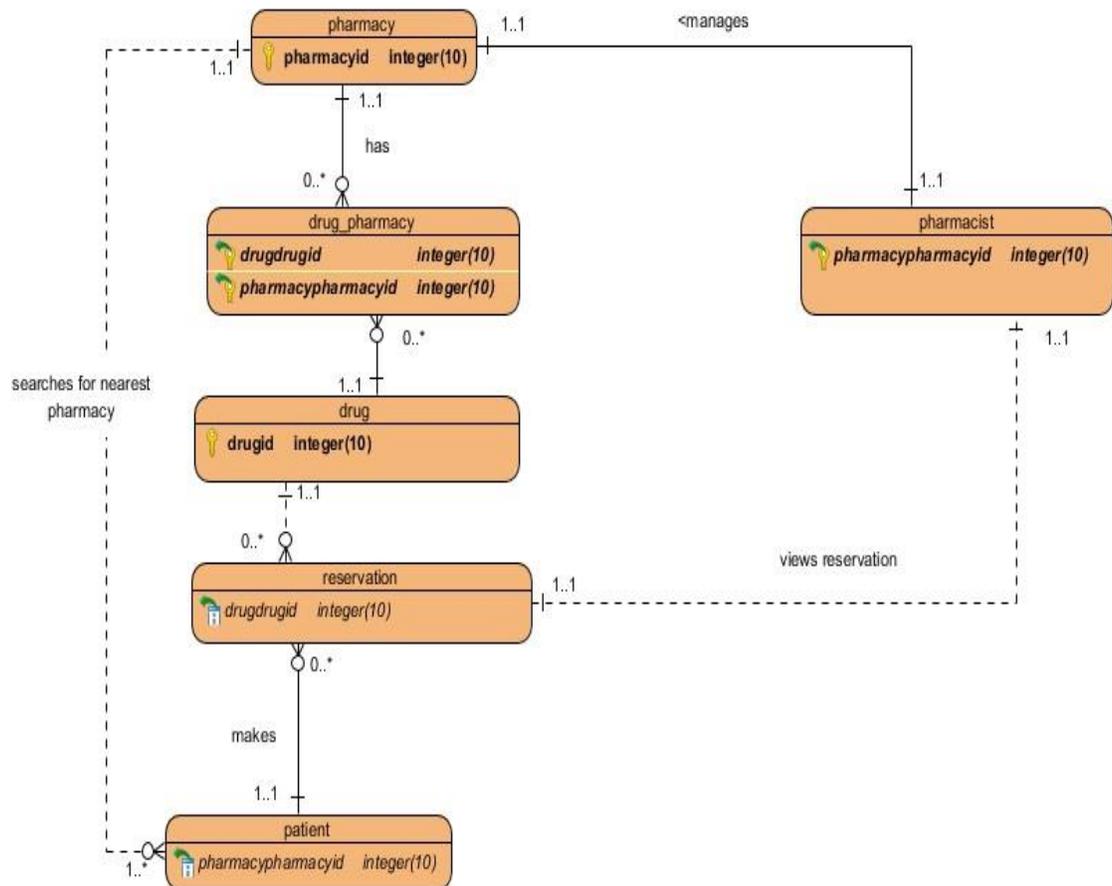


3: Figure 3.3 model-view-control diagram

3.5 Database design

3.5.1 Entity-Relationship Diagram

The below Entity-relationship diagram was used in implementing the database. The database contains six entities as illustrated below in the ER diagram.



4: Figure 3.4 database Entity-Relation diagram

3.5.2 Data Dictionary

A data dictionary that describes all the table fields present in the database. It shows the various tables, field name, data type, whether the different attributes accept null values and description of each attribute. The below table represents the data dictionary of this project:

1: Table 3.1 Data Dictionary

Table Name	Attribute	Data Type	does it accept null values?	Description of attribute
patients	Patientid	int	no	Id of patient
	Fname	varchar(80)	no	First name of patient
	Lname	varchar(80)	no	Last name of patient
	Phonenumber	char(10)	no	Patient mobile number
	Email password	varchar(100) varchar(255)	no no	Patient email number Password for login
Pharmacy	pharmacyid	int	no	Id of pharmacy
	pharmacyName	varchar(150)	no	Name of pharmacy
	pharmacyNumber	int	no	Contact number
	pharmacyEmail	varchar(100)	no	Email of pharmacy
	city address	varchar(100)	no	Which city it located
	latitude	varchar(100)	no	Address of the pharmacy
	longitude	varchar(250) float(15,10) float(15,10)	no no no	Latitude value Of location Longitude value of location
Pharmacist	pharmacistid	int	No	id of pharmacist
	fname lname	varchar(80)	No No	first name of pharmacist

	pharmacyNumber password AssociatedPharmacy	varchar(80) char(10) varchar(255) varchar(150)	No No no	last name of pharmacist contact of affiliated_pharmacy password for login associated pharmacy for verification
Drugs	drugid, drugName type weight SearchTime	int varchar(200) varchar(100) varchar(80) int	No No No YES	Id of drug Name of drug e.g amoxil Form of drug e.g syrup or tablet Weight of drug How many times the drug has been searched in app.
PharmacyDrugs	pharmacyid drugid drugName drugAvailability	int int varchar(200) varchar(20)	No No No no	Id of pharmacy Id of drug Name of drug Is the drug available or not?
reservation	reservationid drugid drugName prescription pharmacyName patient_phone	varchar(200) Long blob varchar(150) char(150)	No No No No No no	Reservation id The id of drug being reserved Name of drug being reserved. Picture of prescription Name of pharmacy Contact of patient

Chapter 4: Technologies and Implementation

“DawaPap” is a web application meaning that can be loaded from any web browser including Chrome and Mozilla. Some of the technologies used in creating this application include:

- Materialize CSS
- Google maps API
- MYSQL
- Web development languages
- Ajax

4.1 Technologies Used

4.1.1 Materialize Cascading Style Sheet (CSS)

This is a responsive framework created by google. This framework helps developers build great user interfaces. The user interfaces that use material design. Combine the classic principles of successful design along with innovation and technology. This framework was chosen because of its’ ease of use and the fact that a lot of documentation is available on material design. Thus, help is readily available when one rans into an issue while using this framework.

4.1.2 Google maps API

An application-programming interface (API) provides a set of data structures, object classes, protocols and tools that developers can use in building software. The google maps API makes it easier for developers to access google maps functionality. However, before one can use this API, an API key is required. The key gives the application permission from google to user their service. This key can be accessed by visiting console.developer.google.com site.

The google places API, which falls under the google maps API, is used in building the system. This API allows users to query for place information in a variety of categories such as

geographic location. Places are searched based on proximity or by specifying the particular place. The API returns a list of places within a particular distance plus a summary information of each place. The places API used to view the various pharmacies nearby a patient on a map.

4.1.3 MYSQL

This is the database used in implementing the web application. This database is a relational database meaning data is stored in form of tables. XAMPP, an open-source database management system, is used to manage this database. This database uses a simple structured query language that is easy to understand and use. In addition, so much documentation is available on the database thus solution to errors are readily available.

4.1.4 Web Development Languages

The main web development application used in this project include HTML5, CSS, JavaScript and PHP. The CSS catered for the user interface while JavaScript and PHP helped with interacting with the database and to ensure validity of information entered into the database. HTML5 is the standard web development language used.

4.1.5 AJAX

Asynchronous JavaScript and XML (ajax) is a technology used to create asynchronous web application. This means that the page is not reloaded every time one needs to access the database.

For example, when a patient is searching for a specific drug, once they start typing the drug name, the application suggests the possible drug being searched without reloading the page.

4.2 Haversine formula

In order to get the nearest pharmacy based on the patient's location, this mathematical formula is used. This formula calculates the distance between two points from the latitude and longitudes of the places. Through this formula, the application can get the distance from the location of the patient to every other nearby pharmacy. Below is a screenshot showing the Haversine formula in use in the application:

```
function calculateDistance($lat1,$long1,$lat2,$long2){
    $earthRadius=6371;
    //convert the latitudes and longitudes to radians
    $lat1Rad=deg2rad($lat1);
    $long1Rad=deg2rad($long1);
    $lat2Rad=deg2rad($lat2);
    $long2Rad=deg2rad($long2);

    $latDelta= $lat2Rad-$lat1Rad;
    $longDelta=$long2Rad-$long1Rad;

    $distance=2*asin(sqrt(pow(sin($latDelta/2),2)+cos($lat1Rad)*cos($lat2Rad)*pow(sin($longDelta/2),2)));
    return $distance*$earthRadius;
}
```

5: Figure 4.1 screenshot showing the Haversine formula in use

4.3 Components of the Application

This section outlines the various components of the application. The main user interfaces include:

1. The patient dashboard for searching for medication
2. The page that displays the nearby pharmacies on the map
3. The closest pharmacy from a patient's location
4. The pharmacist's dashboard

The below is the home page that users have to access in order to login or sign up.

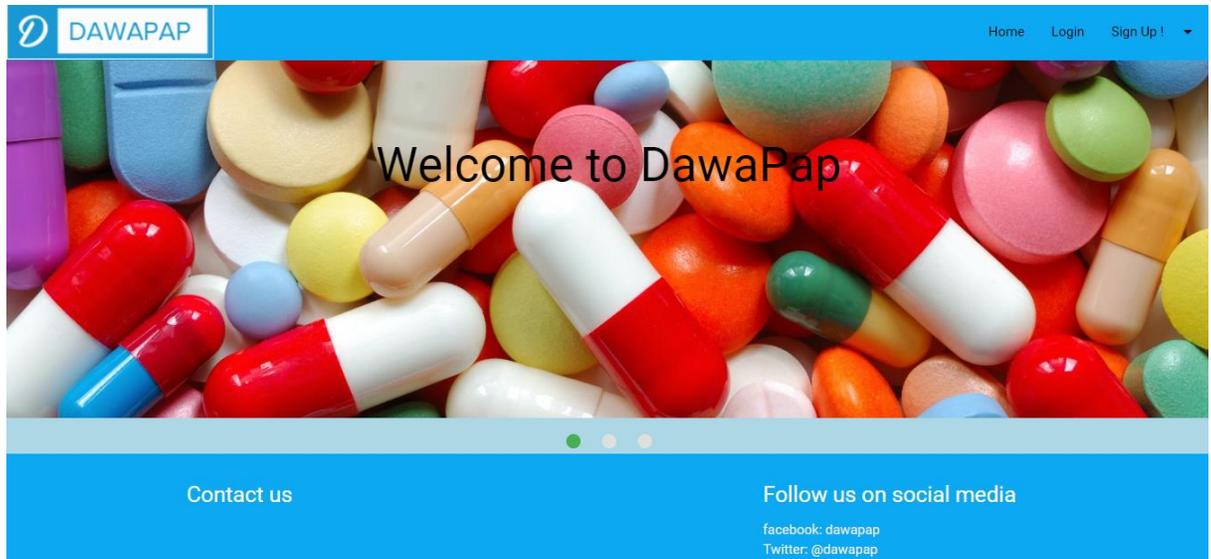
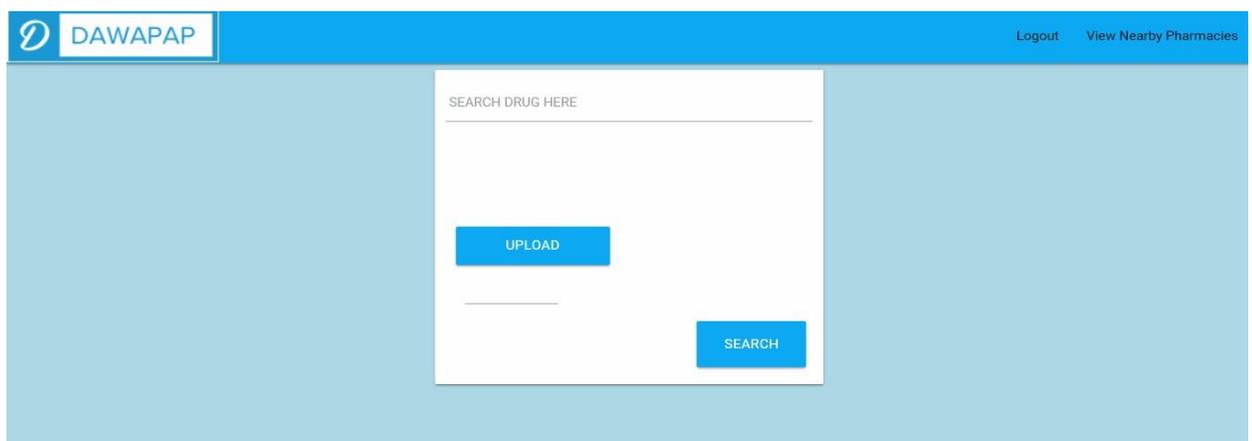


figure 4.3.1 showing the home page

4.3.1 Patient dashboard for medicine search

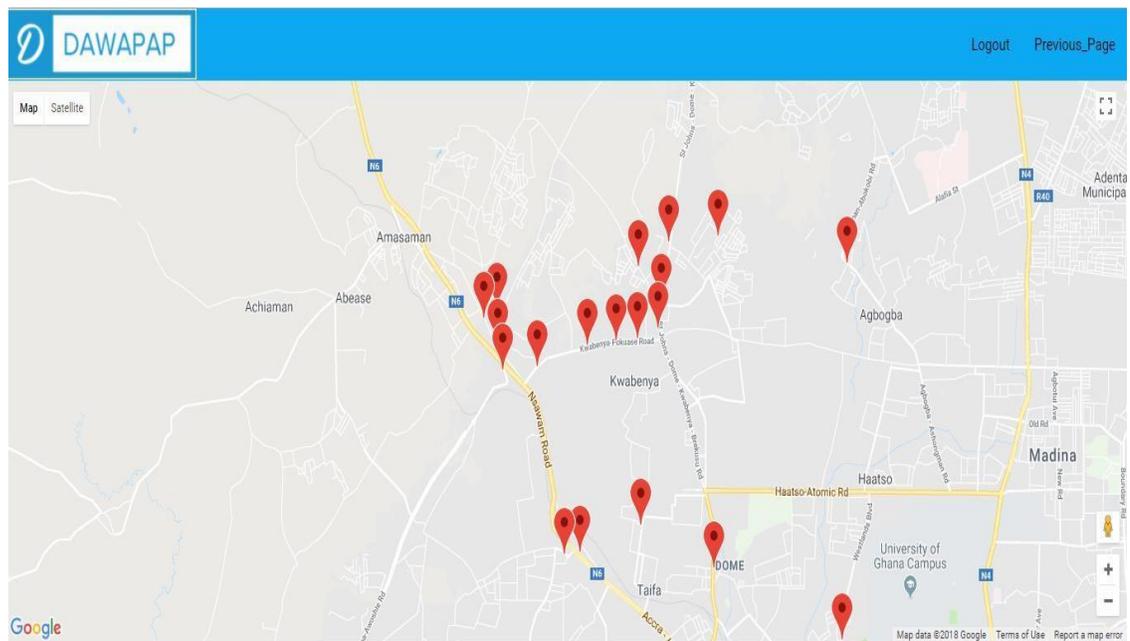
On successful login, the patient is redirected to the dashboard showed in figure 4.2. For a successful search, the patient has to upload a prescription, which will be useful if he/she want to reserve medication. In addition, the patient must type the name of the drug. To make this easier for user, real time data retrieval is used so that patients do not have to memorize spellings since when the begin typing a drug name is suggested. Once all this is done, the search button can be clicked.



6: Figure 4.2 screenshot showing the patient dashboard

4.3.2 nearby Pharmacies

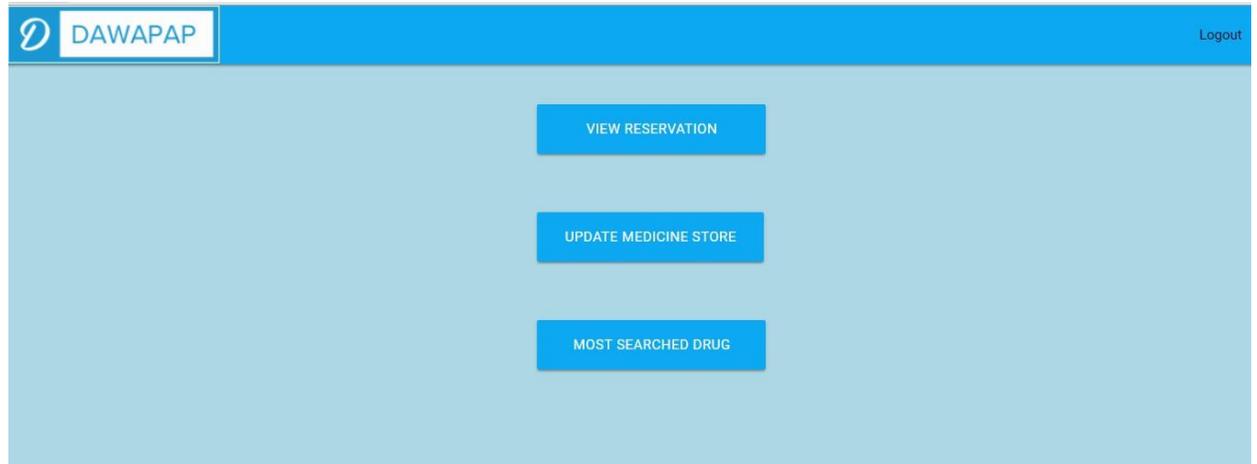
From the patient dashboard, the patient can also view the nearby pharmacies based on their location. The below screenshot shows the results of the nearby pharmacies of a patient in somewhere is Kwabenya estate in Ghana.



7: Figure 4.3 screenshot showing nearby pharmacies for a patient somewhere in Kwabenya estate

4.3.3 Pharmacist's dashboard

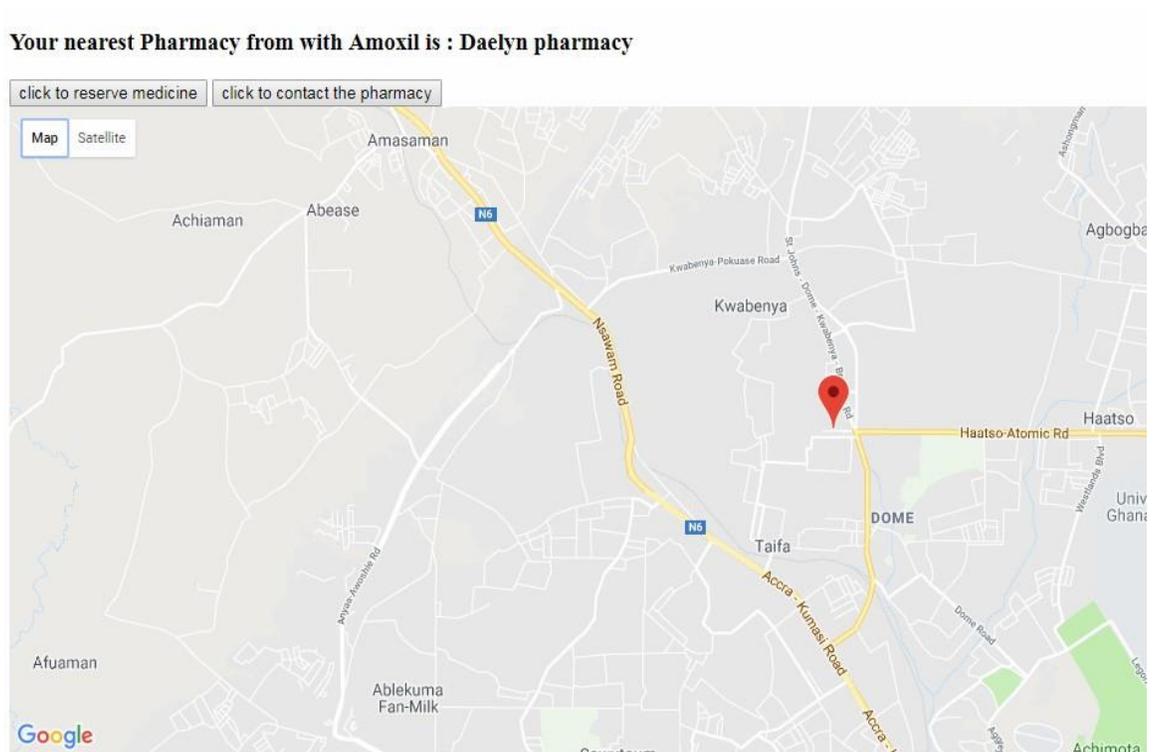
On successful login, a pharmacist views the below dashboard in figure 4.4. This dashboard allows the pharmacist to view reservations made by patients, update medicine availability for their pharmacy and to view the most searched drug.



8: Figure 4.4 screenshot showing pharmacist's dashboard

4.3.4 Closest pharmacy

A patient at Benjul Guest house need to find Amoxil. After logging into the application, the below figure displays the results of the search. The patient is instructed to head to **Daelyn Pharmacy** to get the drug. In addition, they can click the reserve button to reserve the drug or contact the pharmacy.



9: Figure 4.5 screenshot showing the nearest pharmacy from Benjul Guest House in Kwabenya

4.3.5 Patient reservation

When the pharmacist clicks the “view reservation” button, if the pharmacy has reservations, the pharmacist would see all the reservation plus patient contact details. The below is a screenshot of some reservations made

Reservation plus patient contact		
piriton	ladyTech.png	0504230217
quinine	ladyTech.png	0504230218
amoxil	ladyTech.png	0504230218
amoxil	ladyTech.png	0504230212
piriton	ladyTech.png	0504230218
piriton	ladyTech.png	1234567890

Figure 4.6 patient reservations screenshot

4.3.6 Most searched drug

One interesting features of this application is that the pharmacists can view the number of searches of each drug. This would inform their decisions when they are stocking the pharmacy. The below is a screenshot of the number of searched in the database. Amoxil is the most searched drug.

Drugs and number of searches	
Amoxil	25
Asprin	1
AmoxilSyrup	0
Albuterol	0
quinine	2
Amlodiphine	0
panadol	24
brufen	0
piriton	13
codeine	0
naproxen	0
doxycycline	0

figure 4.7 drug searches screenshot

Chapter 5: Testing and Results

5.1 Testing

This Section outlines how the final application was tested. It should be noted that the application was tested every time a functionality was implemented. This was done to ensure that the final end product had a few errors. The developer (myself) mainly did this unit test. However, upon completing all the code work, I had to let potential users navigate through the application. The main users of my application were Ashesi students. I did not train them on how to use the application. I just introduced the application to them and they had to explore. The lack of training was to help me monitor and observe how the user friendliness of the application.

The users, who used the application on volunteer basis, were able to sign up and login successfully as potential patients. In addition, they were able to search for their desired drug and upload a template prescription, which I had. Unfortunately, they had to use my laptop since the application was locally hosted on my machine.

5.2 Results

Most users were able to successfully register and login. I attribute this to the fact that they are students in an environment where technology is broadly used. It would have been ideal to test this application with people with less experience in technology. In addition, most users were impressed by the functionalities of the application and gave insightful feedback which are included in the next chapter.

Chapter 6: Conclusion and Recommendation

This chapter elaborates on the achievement of the project and some of the challenges faced. In addition, it also outlines some suggestion that can be implemented in future versions of the application. This project aimed at coming up with a solution to the “pharmacy tourism” problem. At the beginning of the project, I had proposed implementing a web application that would help patients search for drugs and the application would suggest the nearest pharmacy with the cheapest drug. However, realizing that pharmacists were unwilling to avail drug prices, the application had to be modified to only suggest the nearest pharmacy.

6.1 Achievements

The below are the achievement made through this project:

- Successfully built an application that allows patients to locate the nearby pharmacy with the desired drug. In addition, a pharmacist can use the application to monitor the number of searches per drugs and view patient reservation for drugs from their pharmacy.
- I explore google maps and gained skills in using it a developer
- I was able to successfully go through the software development process

6.2 Challenges

Integrating google maps API with the application was challenging. The places API returns a list of nearby places, yet the application only requires one nearby location. Thus, I had to calculate distances to the various pharmacies using the Haversine formula to determine the closest pharmacy. In addition, this project was not able to integrate google directions into the application.

6.3 Suggestions and future works

The MYSQL database used in developing the application is not easily scalable especially when the number of users increase. I would recommend MongoDB for future upgrades of the project. MongoDB is a non-relation database, which stores data in the form of collections and JSON documents. These collections are similar to table while the documents are similar to attributes in relational database.

Prescription does not only contain drug names. Sometimes doctors prescribe specific components of drug that a patient requires. When the prescription is presented to the pharmacist, he/she knows what drug to give the patient. The current implementation only searches for a pharmacy based on drug names. This means that users cannot search for specific drug components. It will be super helpful to implement this.

It will also be useful if the system can have a live chat feature integrated in the application such that patients can ask for real time advice from pharmacists.

As much as change is inevitable, pharmacists might be reluctant in using this application. Instead of creating a new pharmacy database when a pharmacist registers, the application should integrate with existing pharmacy databases. This way, pharmacists do not have to learn how to use the database from scratch. In addition, automating tasks like creating invoices would appeal to most pharmacies as this mean less work for them. Furthermore, the pharmacist does not have to delete a drug manually every time someone buys a drug. The system should automate this task.

Unfortunately, the delivery option was not integrated in the application. However, it would be more convenient is patients can have their medication delivered to them especially when they are not able to move due to fatigue. It will be great if this feature is implemented in

later version of the application. In addition, to ensure reliability of the application, automatic updates on the available drugs in the various databases should be updates consistently such that the system does not suggest a pharmacy that has ran out of stock of a particular drug.

In a 2017 report on the mobile economy of West Africa by the Global Systems for Mobile Communication, the penetration rate of mobile phones in the region is 50%. This is greater the 47% mobile penetration rate across the sub-Saharan Africa. It would be more impactful if a mobile version of the application is created since more people can use the application. It is thus recommended that a mobile version of the application is implemented.

Reference

- Edoh, O. T., & Teege G. (2010). *EPharmacyNet: an approach to improve the pharmaceutical care delivery in developing countries-study case-BENIN*. Proceedings of the 1st ACM International Health Informatics Symposium, 859-863. Retrieved from <https://dl.acm.org/citation.cfm?id=1883125>
- Epharmacies (2018). Retrieved from <https://epharmacies.com/>
- Hwabamungu, B., & Williams, Q. (2010). *M-Health adoption and sustainability prognosis from a care givers' and patients' perspective*. SAICAST '10 Proceedings of the 2010 Annual Researcher Conference of the South African Institute of Computer Science and Information Technologists. Retrieved from <https://dl.acm.org/citation.cfm?id=1899517>
- Livia Application(2018) .Retrieved from <https://liviaapp.org/>
- Martin, D., Vicente, O., Vincente, S., Ballesteros, J., & Maynar, A. (2014). *I will prescribe an app*. SummerSim '14 Proceedings of the 2014 Summer Simulation Multiconference. Article No. 58. Retrieved from <https://dl.acm.org/citation.cfm?id=2685617.2685675>