ASHESI UNIVERSITY

A MOBILE APPLICATION TO LOCATE SHOPS/ITEMS AND HANDYMEN

APPLIED PROJECT

B. Sc Management Information Systems

Nana Abena Sarpomaa Osei

2019

ASHESI UNIVERSITY

A MOBILE APPLICATION TO LOCATE SHOPS/ITEMS AND HANDYMEN

APPLIED PROJECT
Applied project submitted to the Department of Computer Science, Ashesi University in partial fulfilment of the requirements for the award of Bachelor of Science degree in Management Information Systems.

Nana Abena Sarpomaa Osei

April 2019

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 the preparation and presentation of this Applied Project were supervised in accordance with the guidelines on supervision of Applied Project laid down by Ashesi University.

Supervisor's Signature:

………………………………………………………………………………………………

……………………………..

Supervisor's Name:

………………………………………………………………………………………………

…………………………………

Date:

………………………………………………………………………………………………

……………………….

ACKNOWLEDGEMENT:

My first and foremost thanks go to the Almighty God for the idea and equipping me to pursue this project.

Secondly, I would like to thank my supervisor, Dr. Stephane Nwolley for all the help, support and supervision given to me throughout this project.

Finally, I would like to thank my friends, family, the CSIS department of Ashesi University and every other individual who helped me one way or the other to make this project a success.

Thank you all, and God bless you!

ABSTRACT:

Shopping is one essential activity in every individual’s life. Without shopping, individuals will not have access to things they need to live comfortably. Handymen also make our lives easier by offering services we cannot provide for ourselves either due to lack of skills, time constraints or unavailability of necessary tools.

Shopping and finding handymen are relatively easier in developed countries because of the ease of accessing shopping centers, and handymen. However, in developing countries, bad roads and traffic congestion, security and safety threats make these activities much more difficult.

In this project, a mobile application to locate shops, items, and handymen is developed to increase the accessibility rate of shops and items, while decreasing security and safety threats associated with hiring handymen.

Table of Contents

Chapter 1: Introduction:

1.1. Background:

With the increasing population, needs and wants of people, the need to shop and purchase items and services will also increase. With this, people will need to find a cheaper, convenient and safer method of obtaining what they need. The emergence of the internet and technology has made communicating with others, miles and continents away easier and simple over time. Consumers prefer to purchase goods and services from the source to ensure they receive the best quality and original product and do so within the minimum time possible.

The internet and technology have created a platform for this to be easily and readily available regardless of where a consumer is relative to the source of the product. Online shopping, as of 2001, was ranked as the third most popular activity on the internet [4]. An estimated population of 1.66 billion all over the world made online purchases, with a global income of 2.3 trillion USD [7]. The data shows that as the world advances, online shopping will be more common, even in Africa, that generated 1.8 percent of the global income in 2017 [7].

According to Li & Zhang [4], online shopping is the process of purchasing products or services via the Internet. In Ghana, online shopping is not as popular as it is in other parts of the world, mainly because of the rate of Cybercrime in the country – Ghana ranked second in cyber fraud [3], which can explain the fear of many Ghanaians when it comes to online shopping.

In Ghana, there is a difficulty when it comes to shopping and purchasing goods using the traditional method. This difficulty is because a consumer would have to move from shop to shop to locate the good, at an affordable price before deciding to buy or not, which is stressful and may pose an inconvenience. Highly congested roads also play a role in the difficulty that consumers face when locating products to purchase. The time needed to commute from one place to the other has almost doubled because of road congestion in Accra due to the increase in vehicles [1]. Others have resorted to online shopping, although it is not popular in the country yet.

According to the National Communication Authority (NCA), as of September 2017, there was a total mobile penetration of 37,445,048, which was a 0.09% increase from the previous month, August 2017 [8]. The data shows that more people use phones now, so having a mobile application that aids in their shopping needs will be a more convenient method.

For requesting services, such as plumbing, electrical services, and other basic household services, the increase in crimes in the country makes it difficult to trust “strangers” to come to one’s home to offer these services. As a result, people always rely on handymen of their friends and families to perform these functions for them. This is regardless of what or how much it will cost them because there is some level of trust.

1.2. Related Works:

Although the Ghanaian community does not commonly accept online shopping, there are some online shopping platforms, both websites and mobile applications that have emerged to solve this issue.

Utradia is a mobile application that allows users to buy goods. It lists various shops, along with their catalogues so users can contact them to make purchases. A user is required to create an account to use the application, and then access to the catalogues is granted, after downloading.

There is also Jumia, which has both a website and a mobile app for ease of accessibility. They offer a wide range of items from clothes, to electricals, to accessories, etc. They also offer discounts from time to time especially during occasions and other holidays. They offer customers delivery at a cost, to all ten regions in the country which also increase their availability [9].

Tonaton is another online platform that allows users to post items they are selling so others can request to purchase them. It connects sellers to buyers within the country. Buyers have to meet at an agreed upon location for sale to take place after connecting on the platform [6]. Both buyers and sellers are not 100% assured of the authenticity of the product and the identity of each other until they meet to make the exchange.

1.3. Objective:

The proposed solution that this project address is the development of a mobile app that gives customers a cheaper, stress-free and convenient way of finding items to purchase. The application will allow users to search and locate items, at the lowest prices, from their regular shops, to purchase before stepping out of their homes. It will reduce the time and stress involved in finding a product a customer needs and minimize cost because customers know exactly where to go to get what they need. Although there is the tendency of finding fake items, users are assured of original products only, that they can purchase securely and safely. It does not take away the joy of walking through a shop to pick the item off the shelf as preferred by others as it only shows the user where they can get what they need.

For the services, contacts and profiles of professionals will be put up on the application that users can go through to select their preferred choice to provide a service to them. Customers will have a platform that will give them easy access to trusted and skilled professionals where their security is assured.

1.4. Overview of Remaining Chapters:

This document has six chapters. Chapter 1 gives the reader an introduction to the project, the background, existing applications and websites that seem to solve this problem and the objective of the project.

Chapter 2 addresses the specifications of the proposed solution and gives the user an idea of the features and how it will solve the problem. Chapter 3 discusses into details, the design and architecture the application will possess.

The implementation, techniques, and tools used will be presented in chapter 4. The results and evaluations, the limitations of the system will be discussed in chapter 5. The last chapter, chapter 6 will present some reflections on learning throughout the project period, as well as some further work that can be undertaken to improve the system.

Chapter 2: Requirement Analysis:

2.1.Chapter Overview:

This chapter describes the scope of the project, the functionalities of the application and the target users of the application.

2.2.Project Overview:

This project aims to develop a mobile application that allows users to find items they desire to purchase easily and at the least price possible. Data on the inventory of registered shops will be collected and stored in a database, which will be retrieved when users search for a particular item.

Secondly, the app is to provide users with a platform that gives them easy access to basic home service professionals such as gardeners. A database of these professionals' information and ratings will be created. Users will be presented with options, based on ratings when they request for any service.

2.3.Requirement Gathering:

In gathering the requirements of the app, questionnaires, interviews, and observations were used. Service providers, customers, existing shopping apps were the resources used to collect information. This was to answer the following questions and others:

- Do people prefer to shop traditionally or online?
- Do people prefer their items delivered or handpicked?
- Are people satisfied with the condition in which they receive their ordered goods?
- How easy is it to find handymen their community?
- How easy is it to find items at the prices they are willing to pay for them?

- Will they prefer an easier method in finding shops and items to enable them to shop traditionally?

2.4.Requirements Analysis:

This section highlights the features of the app after analyzing the data gathered from the requirement gathering stage.

2.4.1. Core Features for User App:

2.4.1.1.Search for a shop:

- The user clicks on the find items' button and is directed to the view with registered shops.
- The user clicks on the search button.
- The search page opens, and input is made.
- The search query is run, and the shops with the item entered will be made available along with their prices, contacts, and working hours.
- The user selects a shop, and a map with directions to the shop opens.

2.4.1.2.Search for an artisan:

- The user clicks on the find artisan button and is directed to the view with registered artisans.
- The user clicks on the search button.
- The search page opens, and input is made.
- The search query is run, and the registered artisans who offer the service inputted is made available along with their ratings, working hours, location and contact.
- The user selects a preferred artisan, and a notification is sent to him/her.

2.4.1.3.Browse items or artisans:

- The user clicks on either the find items button or the find artisan button.

- Based on the selection, a shop view is made available, or an artisan view is made available.
- The shop view lists items available and the shops that have it, with their prices or price range, location, and contacts.
- The artisan view lists artisans registered onto the app, with their specialty, rating, and contact.
- A user can select either an item or artisan for further details.

2.4.1.4.Rate an artisan or leave feedback:

- A user clicks on his profile.
- The user selects the activities tab to view pending and completed activities.
- Upon completion of an activity, the user can mark it as completed and is linked to a rating and feedback page.
- The user then rates the quality of the job and leaves feedback or comment.

2.5.Core Features for Service Providers App:

2.5.1. Upload Items or Edit Existing Inventory:

- A shop owner logs in and can select the upload button or edit existing inventory button.
- On selecting the upload button, the user is directed to an upload page where an excel file with the shop inventory can be uploaded.
- On selecting the edit existing inventory button, the user is directed to a page where a table view of the existing inventory is shown.
- The user can click on the edit button and make changes to the inventory.

2.5.2. View and accept job requests:

- An artisan selects the request button.
- The artisan sees the list of job requests received.

- The artisan can view the details of the job request.
- The artisan accepts the request or rejects.

2.5.3. View completed and pending activities:

- An artisan selects the activities button.
- The artisan either selects the completed or pending activities tab.
- On selecting the completed activities tab, the artisan can view completed jobs, and ratings/feedback received on each activity.
- On selecting the pending activities tab, the artisan can view pending jobs, and cancel the request or contact the user.

2.6. Non-functional Requirements:

Security: Because it is an app that will connect people, i.e., shop owner to the buyer and artisan to the client, there is a need for a security check. Background checks will be run on artisans and shop owners before they are registered onto the app. Any applicant with a suspicious background or with non-verifiable details will not be registered onto the app to ensure the safety of users of the app.

With regards to shop owners, we must ensure that items uploaded are indeed available at their shops and in good condition, not close to expiry. To do this, there will be a representative to check on the inventory uploaded before it is made available to users.

Reliability: Users should be able to trust the information provided on the app.

Performance: The app should work as efficiently as possible at all times. Simple queries are used to communicate with the database. Hence, it takes a short period to return results to users.

Availability: The app should be available to serve users at all times, with no downtimes. It should be able to serve a vast number of users simultaneously without crashing or delays.

Chapter 3: Architecture and Design:

3.1.Chapter Introduction:

This chapter describes the design and architecture used in the implementation of the mobile application described in the previous chapters.

The architecture used in the design and implementation of the application is the client – server architecture or the two – tier architecture. The two – tier architecture "is a type of multi-tier computing architecture in which an entire application is distributed as two distinct layers or tiers. It divides the application logic, data, and processing between client and server devices". (Techopedia.com, n.d.).

The server device(s) contains all the necessary resources needed in the managing and running of the application, while the client device(s) are used to make and send requests to the server devices to make use of the resources contained on them. For this application, the server resources include XAMPP. The XAMPP application will be used in running the web server, with its inbuilt Apache web server and database server run on MySQL. The database will store all data to be used by the application.

For this application, the client will be a mobile device, on which the application will be installed and run.

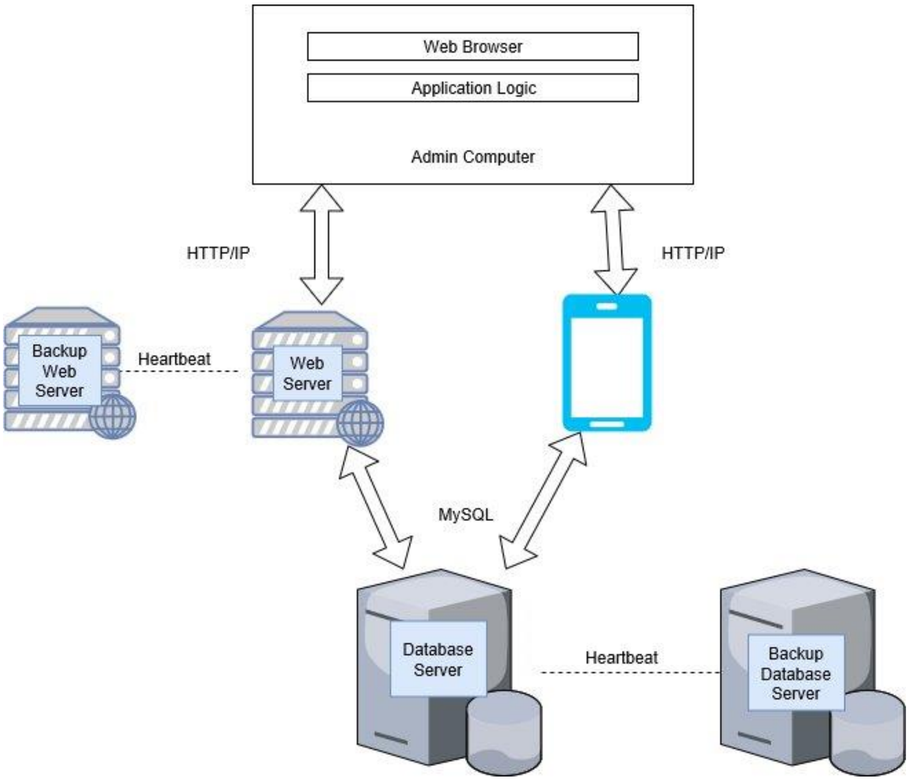


Figure 1

This figure shows the make-up of the connections between the units that form the mobile platform, both the hardware and software components. It also shows the interfaces and protocols used in communicating.

There is a backup web server and a backup database server, which makes the system a highly available system. They communicate using the heartbeat method, where the backup servers ping the main servers periodically to check if they are available. If there is a negative response from the ping, the backup servers immediately take over to run the system.

3.2.Presentation Layer:

The presentation layer handles what appears to the user and what they interact with on the application. It presents it in the simplest way for easy interaction between the user and the application. It considers the software running on the phone – the app will only run on certain software versions of the phone.

3.3.Data Management and Application Processing Layer:

This layer deals with the data to be processed and the functions the application will serve to the user. It also handles the data the user will be able to view, interact with, and the responses to the requests and queries they make to the database.

The queries are made, and server-side scripts are used to run those queries, to register a user, or record a requested job, or to request a map, etc.

3.4.Database Layer:

The last layer of the architecture is the database layer, which is also the key layer of the application because the system is run on the database. It is written with Structured Query Language (SQL)

The data stored in the database includes the details of the service providers (Artisans and Shop Owners), the services provided, job requests made, etc.

3.5.User Interface Requirements:

For this application, the interface is built on a principle of simplicity, beauty, and efficiency. There are two applications: one for a regular user, i.e., a user who wants to locate a shop or find an artisan and another for a user who registers as a service provider. Screenshots of each app can be seen below.

3.6.User Classes and Characteristics:

This app is to help users easily locate items they want to purchase, from the closest shops and at the least high price. It is also to help users reach skilled professionals who pose no safety risk to them and their families.

3.6.1 Use Case Scenarios:

3.6.1.1 A user who wants to locate a shop:

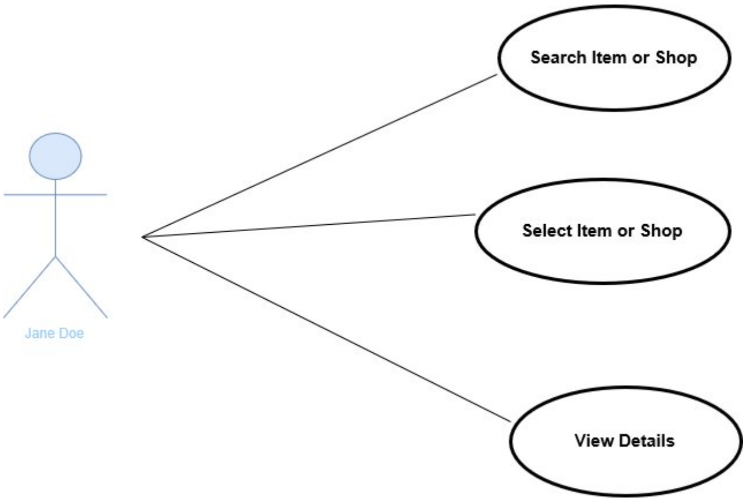


Figure 2

A mother wants to purchase some groceries for her home. The last time she went shopping, she was unable to get everything she needed, so she had to go elsewhere to complete her shopping, which caused a delay and messed up her plans. She doesn’t like online shopping because she loves to pick the things off the shelf herself and she fears they may not meet her specifications. She opens the app and skips the registration process as she is only a first-time user and wants to try it out before deciding to be a member. She proceeds to input her grocery list for the week and then selects a filter based on location, which means, she wants the shops closest to her. The app immediately returns her a table with the list of shops that has all her shopping items, based on the filter she selected. She proceeds to get ready to go out and do her shopping, from one shop, within 15 minutes’ drive from her home.

3.6.1.2 A user who wants to request an Artisan to provide a service:

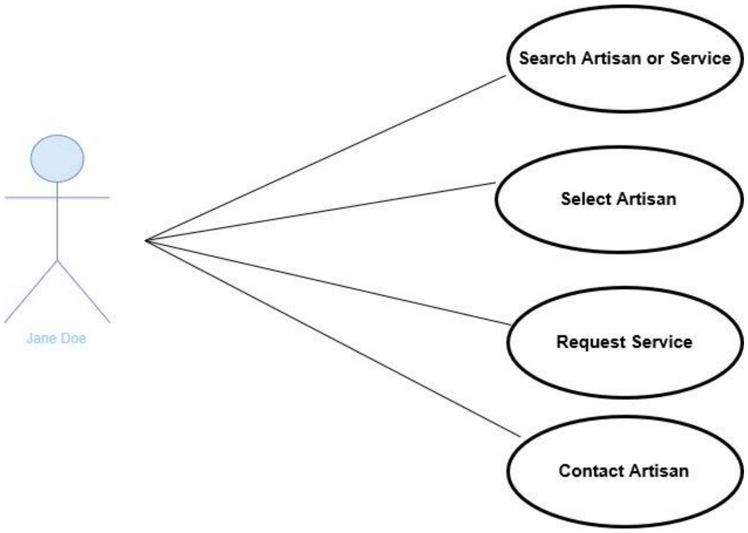


Figure 3

A father who just began working two jobs is unable to perform his gardening responsibilities due to time constraints and the fact that he is always tired when he returns home. He notices his garden is growing out of shape and needs to be trimmed. He also remembers that because of the drainage system close to their home, rodents and other dangerous animals like snakes make the garden their home when it is overgrown. He is trying to find a good gardener who can keep his garden for him, for about three months, while he adjusts to his new schedule. He opens the app and creates the required account that he needs to enable his request for a professional service. This is to ensure the security of both the professional and the user. The father then proceeds to request for the services a gardener and logs in his location and when he needs the service provided. The app returns a list of available gardeners based on his search and shows their price ranges and ratings. He picks his preferred gardener and waits for him to come by and provide the service.

3.6.1.3 A Shop Owner/Artisan who wants to sign up and be visible to buyers:

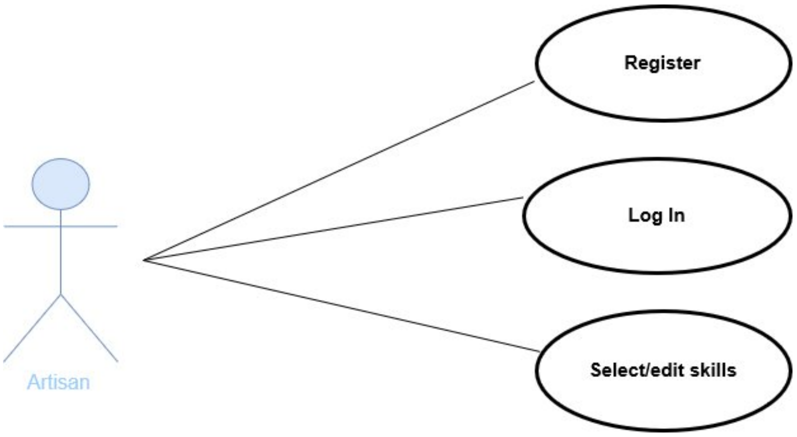


Figure 4

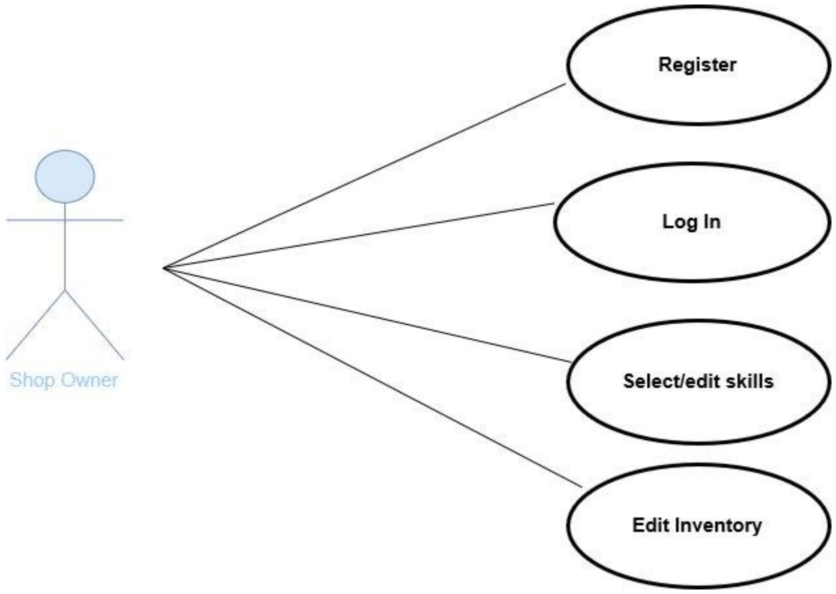


Figure 5

A service provider has noticed that each time he meets with his colleagues, they speak well of their business and how others are referring them, so they are making lots of money. Unfortunately for him, he just moved to a new neighborhood, provided few services; hence, he is not known by many and is yet to receive some recommendations and referrals. This has made business slow for him, and he is struggling to make ends meet.

He downloads the app, sends an email to the administrator and puts in an application to register. He is called for an interview; a background check is run, a few of his old customers are contacted for more information. He passes all the test, he is registered, and he is ready to be reached for business.

In a week, he receives about ten requests from his neighborhood and surrounding ones. Now, he can also have positive comments on work with his friends, and he can provide for himself.

3.7 Data Design:

3.7.1 Data Description:

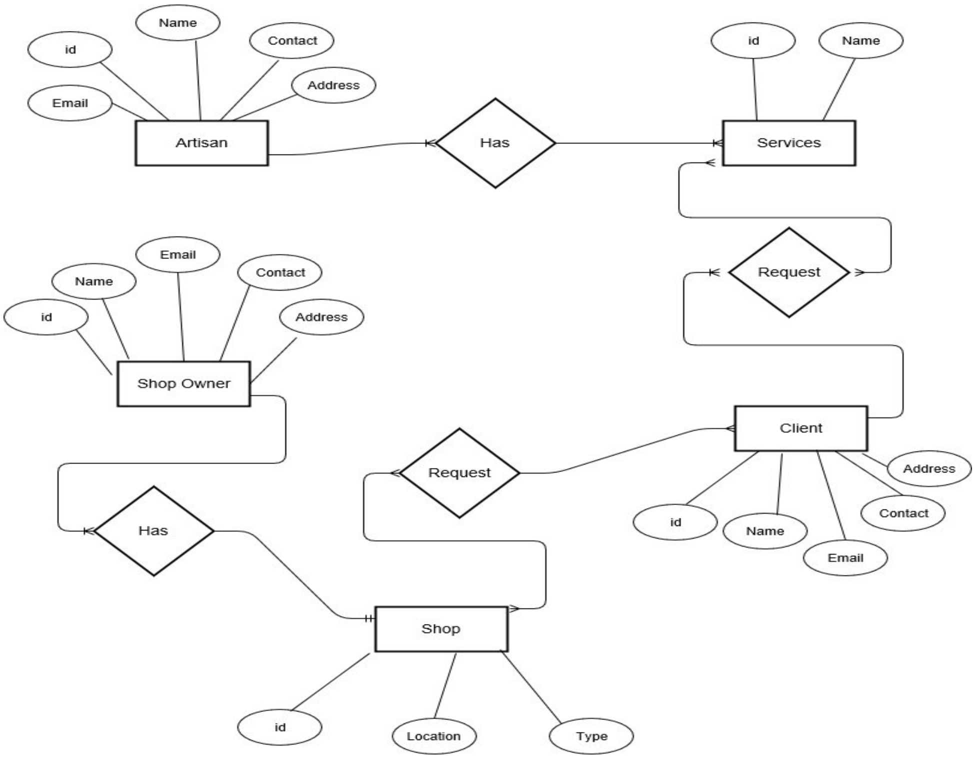


Figure 6

3.7.2 Data Dictionary:

Document Name	Field Name	Data Type	Allow Nulls	Field Description

Service Provider	id	Integer	No	This field stores a registered service provider with a unique number.
	FirstName	Text	No	It stores the first name of a registered service provider.
	LastName	Text	No	It stores the last name of a registered service provider.
	Email	Text	No	It stores the email address of a registered service provider.
	City	Text	No	It stores the city where a registered service provider resides.
	Telephone_no	Text	No	It stores the phone number of a registered service provider.
	Service	Text	No	It stores the service provided by a registered artisan.
	User_type	Text	No	It stores which type of service provider a registered service provider is; i.e., an artisan or shop owner.
	address	Text	No	It stores the address of a registered service provider.
	shopName	Text	No	It stores the name of a registered shop.
UserType	Id	Integer	No	This field stores a unique identifier for the user types.
	Usertype	Text	No	It stores the type of user types there are; i.e., an artisan or service provider.
Clients	id	Integer	No	This field stores a registered user with a unique number.
	clientName	Text	No	It stores the name of a registered user.
	dateOfBirth	Date	No	It stores the date of birth of a registered user.
	Address	Text	No	It stores the address of a registered user.

	City	Text	No	It stores the city in which a registered user can be found.
	Gender	Text	No	It stores the gender of a registered user.
	Email	Text	No	It stores the email address of a registered user.
	phoneNumber	Text	No	It stores the phone number of a registered user.
	dateJoined	Datetime	No	It stores an autogenerated timestamp of when a user register.
ItemRequests	Id	Integer	No	This field stores a requested item with a unique identifier.
	requestDate	Datetime	No	It stores an autogenerated timestamp of when a user requests an item.
	Status	Text	No	It stores the status of a requested item.
	Item	Text	No	It stores the name of a requested item.
	Quantity	Text	No	It stores the number of items requested.
	Client	Text	No	It stores the name of a client who has made a request.
	Shop	Text	No	It stores the name of the shop from which a request has been made.
	Rating	Integer	No	It stores the rating of a shop.
ServiceRequests	Id	Integer	No	This field stores a requested service with a unique identifier.
	requestDate	Datetime	No	It stores an autogenerated timestamp of when a user requests a service.
	Status	Text	No	It stores the status of a requested service.
	Service	Text	No	It stores the service type that has been requested.
	Client	Text	No	It stores the name of the client requesting the service.
	Artisan	Text	No	It stores the name of the artisan to provide the service requested.

	Rating	Integer	No	It stores the rating given by the client for the service provided.
Services	id	Integer	No	It stores the records of services with a unique identifier.
	serviceName	Text	No	It stores the name of the services provided.
	serviceDescription	Text	No	It stores the description of services provided.
shopItems	Id	Integer	No	It stores the items in a registered shop with a unique identifier.
	Item	Text	No	It stores the item name of items in a registered shop.
	Description	Text	No	It stores a description of items of a registered shop.
	Price	Decimal	No	It stores the price of items of a registered shop.
	numberAvailable	Integer	No	It stores the quantity of an item of a registered shop.
	shop	Text	No	It stores the name of the shop.

Chapter 4: Implementation:

4.1. Chapter Introduction:

The purpose of this chapter is to outline the necessary implementation needs of the application clearly. The scope of the requirements will span from hardware, software and skill requirements.

4.2. Application Overview:

The application to be implemented will be a system that enables the general public to access and hire the services of handymen of various skills. Handymen are then rated based on the quality of their service provision by the public which then informs all new users on the service expertise of each handyman.

Also, the application allows the general public to easily search basic items and the shops from which they can obtain these items as well as the item's price.

The system comes with the following four interfaces;

1. A mobile application for users
2. A mobile application for artisans
3. A web portal for shop owners to enable them to upload shop items
4. A web portal for an administrator to manage processes and operations within the system.

4.3. Hardware and Software Requirements:

To be able to use this application, either as a customer searching for a shop/item, or a service provider (shop owner or handymen), a user will need to have an Android or Apple smartphone with internet access. The device will need to be running an Android OS or iOS with internet connectivity. The system is a hybrid application that will be built using HTML, CSS, and JavaScript for the frontend.

The administrator portal will be built using HTML, CSS, JavaScript, and SQL and will require a hardware device that runs windows or Mac operating system with internet connectivity.

The server side will be built using HTML, CSS, JavaScript, and PHP and to run, the computer hardware device will need to run a Windows or Mac operating system. It will also need to run an SQL database to be able to store the data of the shop owners and their shops, and the data of the handymen and their services.

4.4. **Mobile Application**:

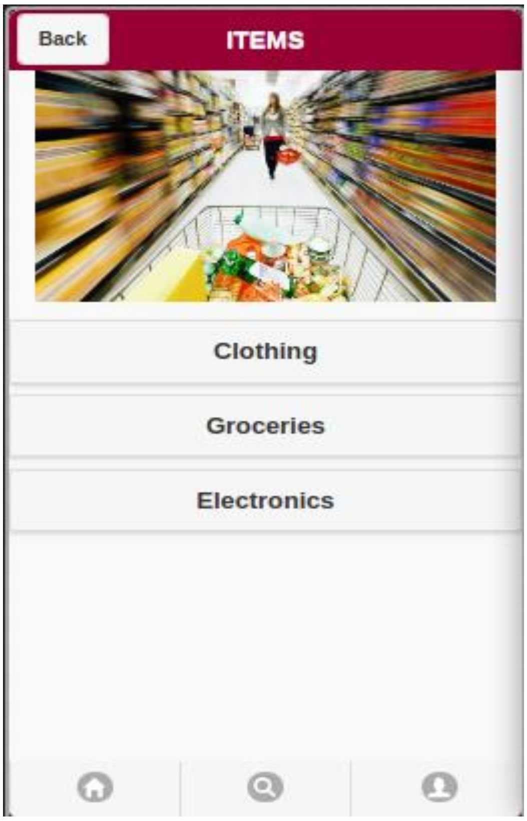
4.4.1 User Mobile Application:

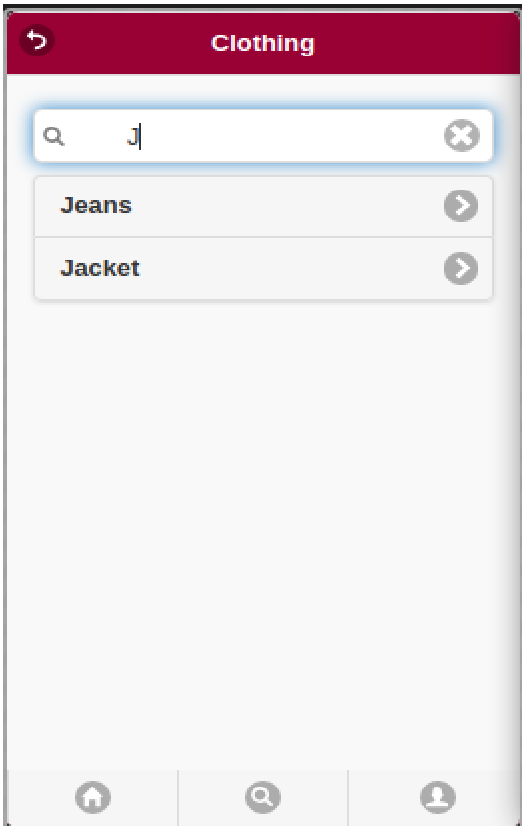
The mobile user application has four main functionalities; first, it allows a user to search for items they want to purchase, and which shops have them available. Secondly, it allows a user to search for handymen and the services they provide. It allows a user to request for the services of a handyman. Lastly, a user is allowed to rate a handyman after a service has been provided.

A user who downloads the application can have access to both the shops registered and the handymen. However, to request the services of a handyman, a user will have to register and create an account. This is to ensure security for both the handymen and customers as the administrator will have data on which customer requests for what services and which handyman accepts the job. A user is not required to register to access the shops because the application does not include delivery services; hence, there is little to no security concern in this instance.

A user can search for an item they would like to purchase. The application will return a list of all shops that have that item available for sale, together with the price of the item, the location of the shop and contact detail of the shop.

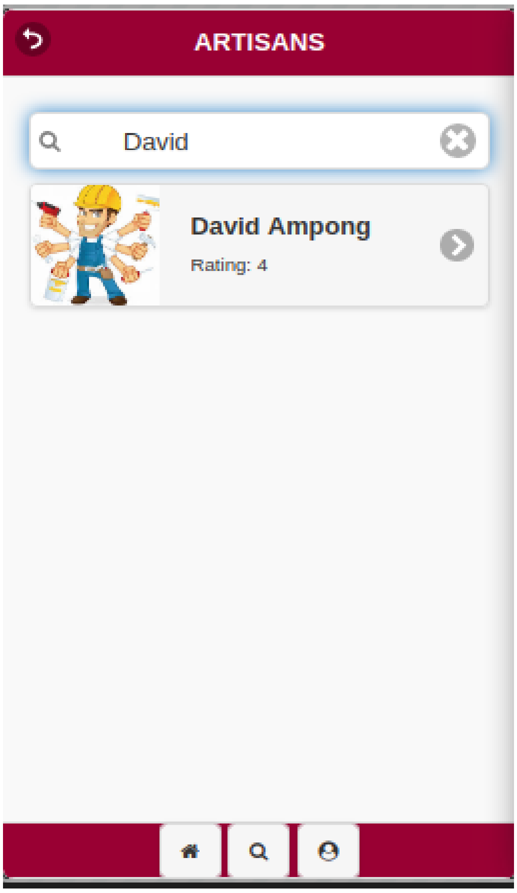
Search Item





A user can search for a service they need, and the application will return a list of all registered handymen who provide that service. The details of the handyman will be available if only the user is a registered user. For unregistered users, the system will prompt the to register before they can view the contact and location details of the handyman and then proceed to request for a job.

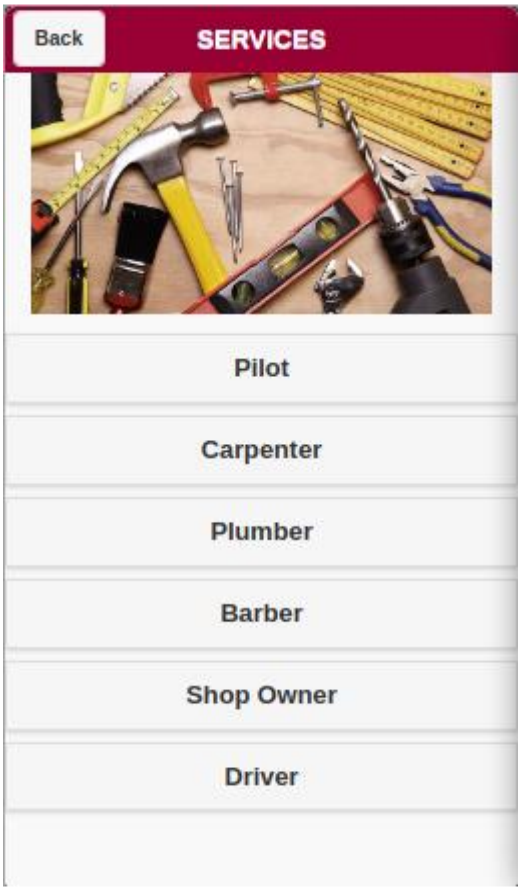
Search Service



David

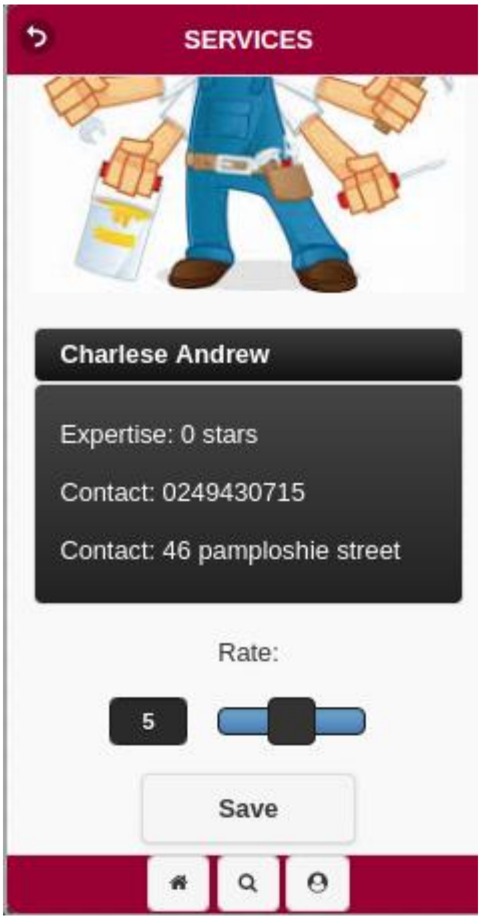
David Ampong

Rating: 4



After a service has been provided, a user can rate the handyman and leave feedback for the handyman.

User Rating



Charlese Andrew

Expertise: 0 stars

Contact: 0249430715

Contact: 46 pamploshie street

Rate:

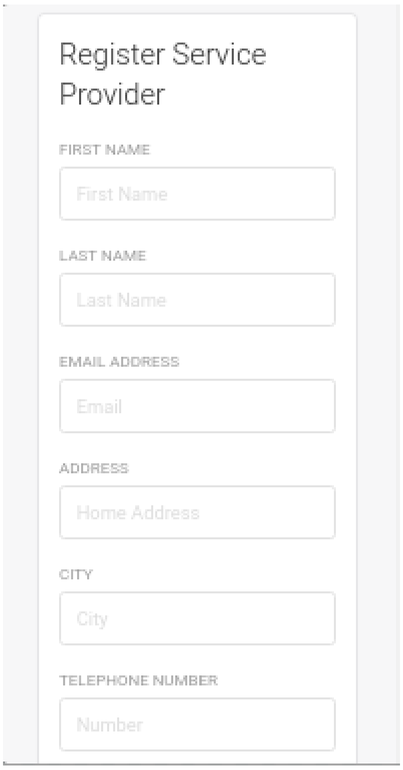
5

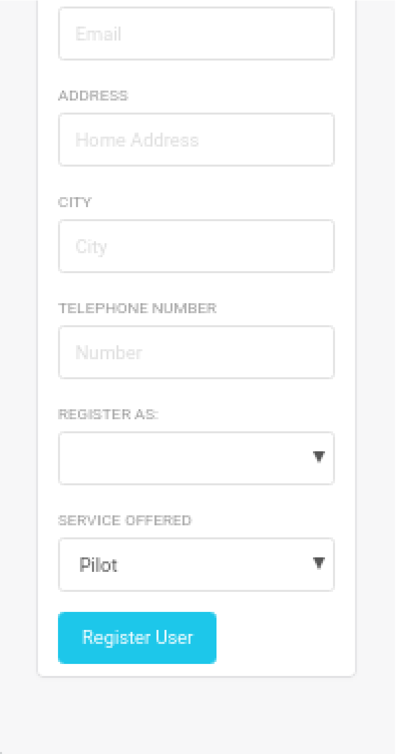
4.4.2 Service Mobile Application:

The mobile service application has three main functionalities; first, it allows a service provider (shop owner or handyman) to register onto the system. Secondly, it allows a shop owner to add or modify their inventory and allows a handyman to add or modify the services they offer. Lastly, it allows a handyman to accept or request a job offer.

A service provider who downloads the application will have to register and create an account to use the application. A service provider can register as a shop owner or a handyman.

Service Provider Registering

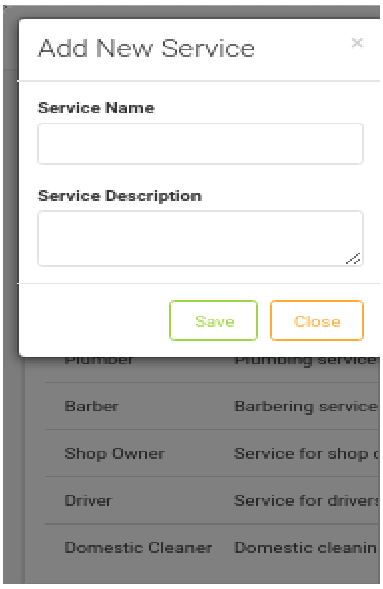




SERVICE OFFERED: Pilot

As a handyman, a service provider can add the services they provide or edit them.

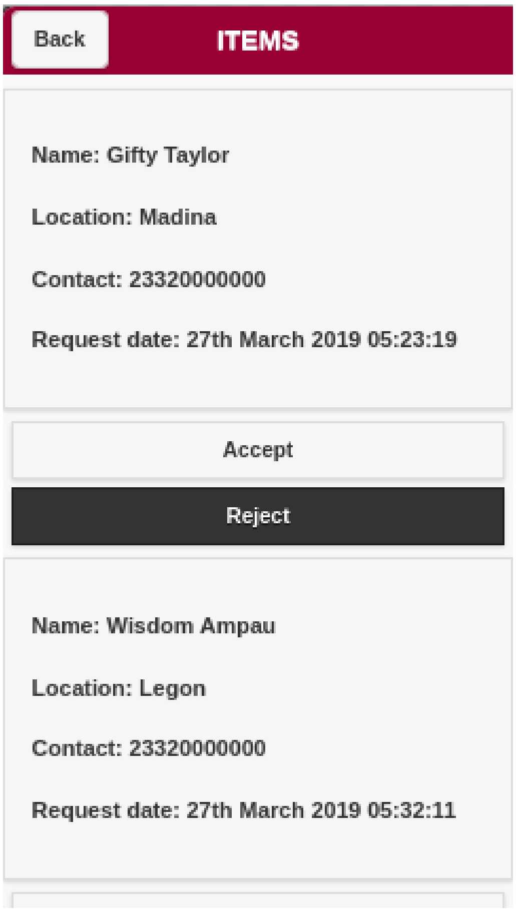
Handyman Adding Services



Plumber	Plumbing service
Barber	Barbering service
Shop Owner	Service for shop o
Driver	Service for drivers
Domestic Cleaner	Domestic cleanin

A handyman can also add or reject a request for a job received.

Handyman Accepting/Rejecting Request



Name: Gifty Taylor

Location: Madina

Contact: 23320000000

Request date: 27th March 2019 05:23:19

Name: Wisdom Ampau

Location: Legon

Contact: 23320000000

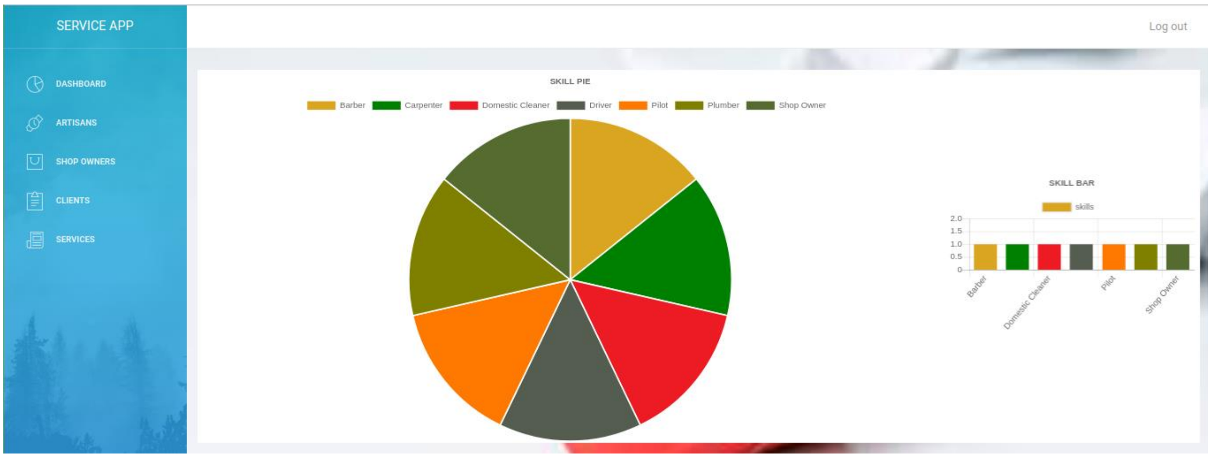
Request date: 27th March 2019 05:32:11

4.5. Administrator Portal:

The administrator portal is a web platform that allows the administrator to manage what the user sees on the frontend of the application. The portal also allows a shop owner to add their inventory, edit their inventory and edit their details.

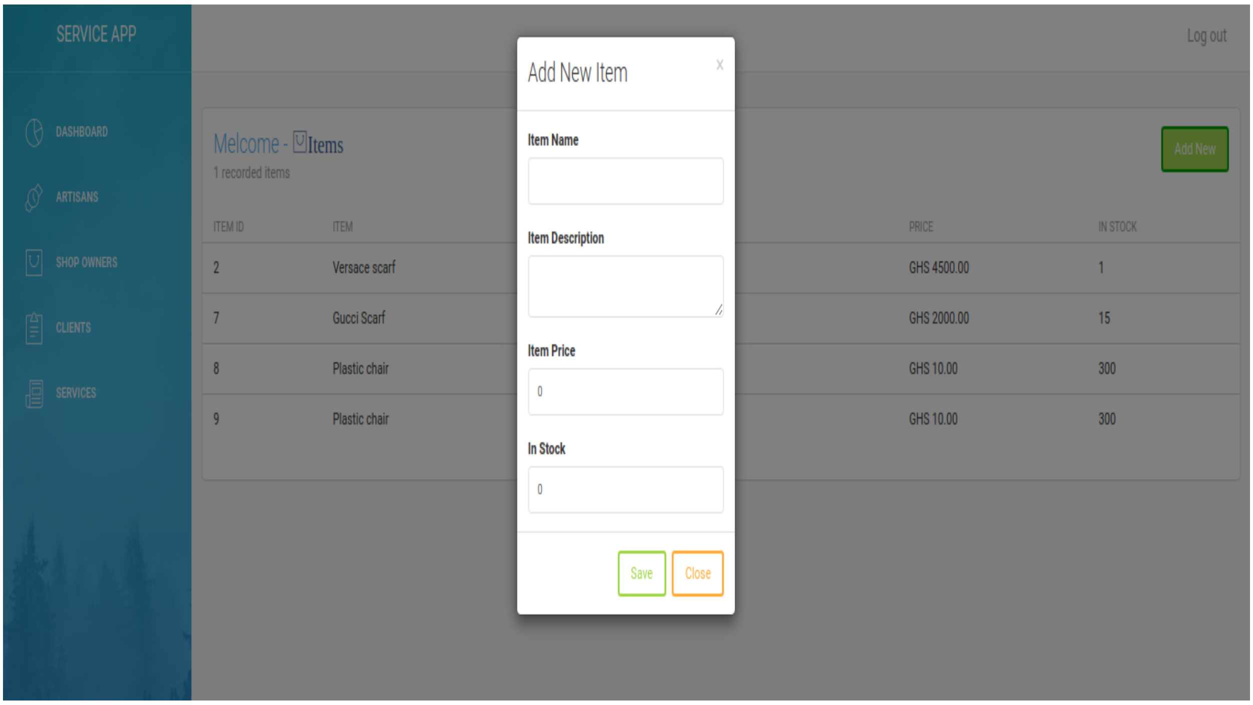
The administrator can view the percentage of registered handymen who fall into each service category.

Dashboard



The administrator can add service providers, a handyman or shop owner to the system. The reason for this is to ensure the security and reliability of service providers. Before a service provider is added, a background check will be run on the individual or shop to ensure the reliability and accuracy of the information provided. The administrator can also edit the information of the service providers and what the users see.

Admin Adding/Modifying Inventory



Melcome - Items

1 recorded items

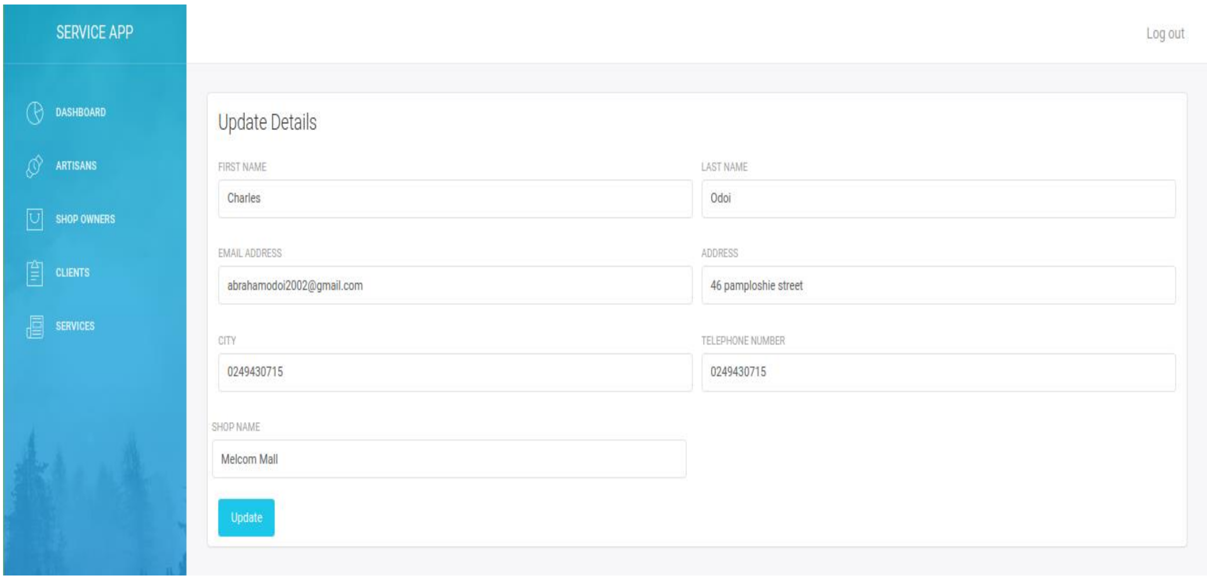
ITEM ID	ITEM	PRICE	IN STOCK
2	Versace scarf	GHS 4500.00	1
7	Gucci Scarf	GHS 2000.00	15
8	Plastic chair	GHS 10.00	300
9	Plastic chair	GHS 10.00	300

Item Price: 0

In Stock: 0

The administrator portal also allows shop owners to edit their data and their inventory.

Shop Owner Modifying Details



Update Details

FIRST NAME: Charles

LAST NAME: Odoi

EMAIL ADDRESS: abrahamodoi2002@gmail.com

ADDRESS: 46 pamploshie street

CITY: 0249430715

TELEPHONE NUMBER: 0249430715

SHOP NAME: Melcom Mall

CHAPTER 5: Testing and Results:

5.1. Chapter Overview:

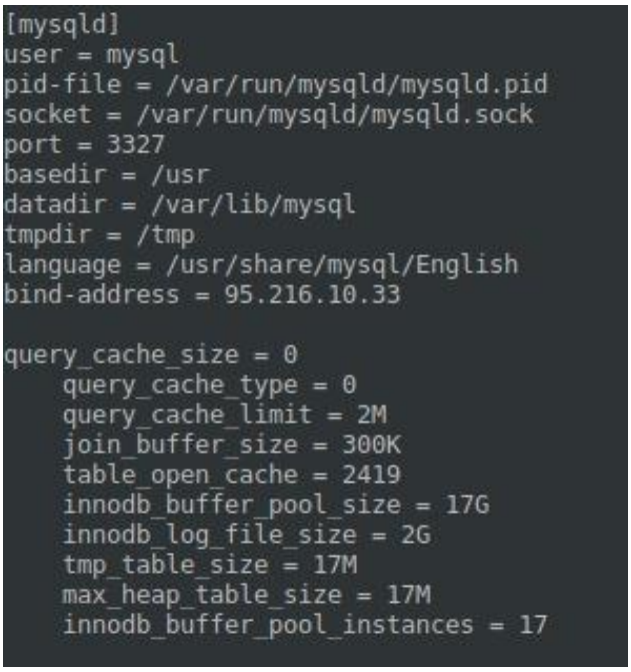
This chapter discusses the various tests that were carried out to check if the functional requirements were met and at its maximum capacity. It also addresses how errors were corrected. In this paper, two tests were carried out – component testing and user testing.

5.2. Component testing:

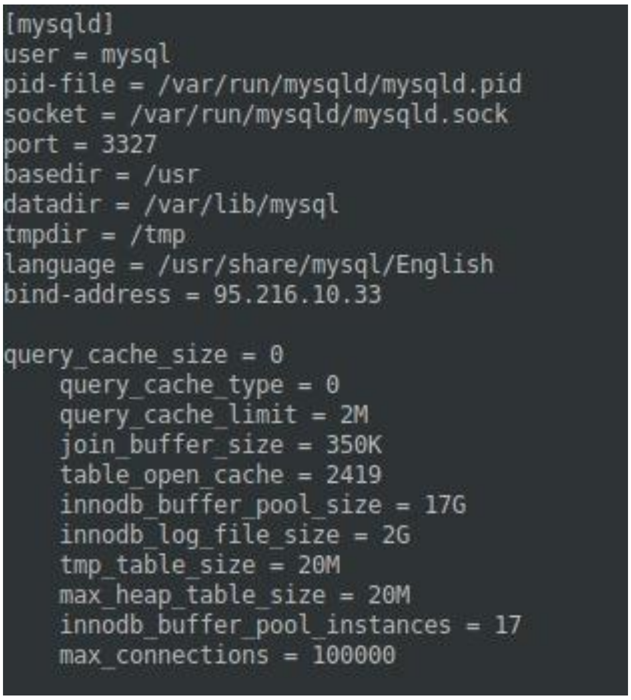
The different components of the system were subjected through various test sequences to ensure efficiency and reliability of each component is guaranteed. These tests included a database test run on the underlying database server’s capability to accept large data volumes from multiple sources seamlessly.

To do this, 20 different shop owners were registered and given access to concurrently upload their shop items onto the system, with each owner inserting 30 records. During the process of the upload it was observed that the database error logs recorded an error message; **“Too many connections”** and as a result, some inserts into the database failed due to this configuration limitation. A total of only 450 data points were inserted into the database during the test.

To solve this issue, the MySQL database configuration file with a default name as **“my.cnf”** was updated to include the line **“max_connections” = 100000** which allows up to a maximum of 100000 concurrent connections to access the database.



```
[mysqld]
user = mysql
pid-file = /var/run/mysqld/mysqld.pid
socket = /var/run/mysqld/mysqld.sock
port = 3327
basedir = /usr
datadir = /var/lib/mysql
tmpdir = /tmp
language = /usr/share/mysql/English
bind-address = 95.216.10.33

query_cache_size = 0
    query_cache_type = 0
    query_cache_limit = 2M
    join_buffer_size = 300K
    table_open_cache = 2419
    innodb_buffer_pool_size = 17G
    innodb_log_file_size = 2G
    tmp_table_size = 17M
    max_heap_table_size = 17M
    innodb_buffer_pool_instances = 17
```



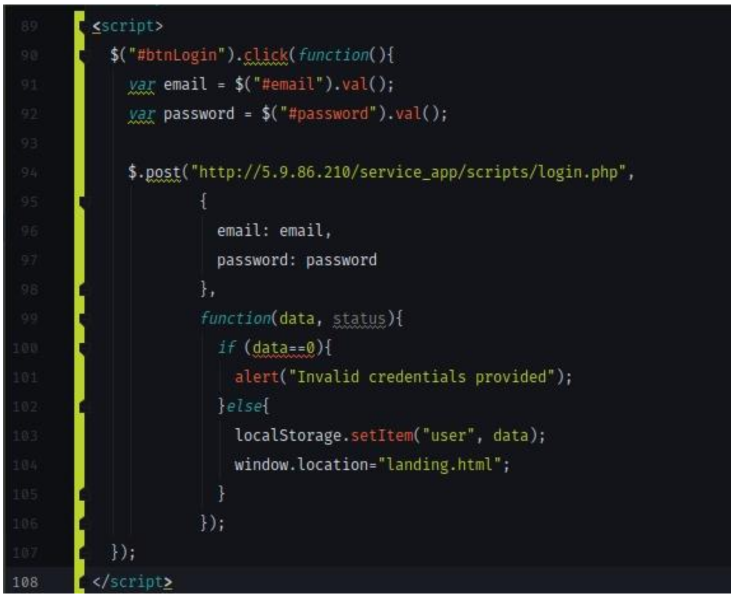
```
[mysqld]
user = mysql
pid-file = /var/run/mysqld/mysqld.pid
socket = /var/run/mysqld/mysqld.sock
port = 3327
basedir = /usr
datadir = /var/lib/mysql
tmpdir = /tmp
language = /usr/share/mysql/English
bind-address = 95.216.10.33

query_cache_size = 0
    query_cache_type = 0
    query_cache_limit = 2M
    join_buffer_size = 350K
    table_open_cache = 2419
    innodb_buffer_pool_size = 17G
    innodb_log_file_size = 2G
    tmp_table_size = 20M
    max_heap_table_size = 20M
    innodb_buffer_pool_instances = 17
    max_connections = 100000
```

5.3. User testing:

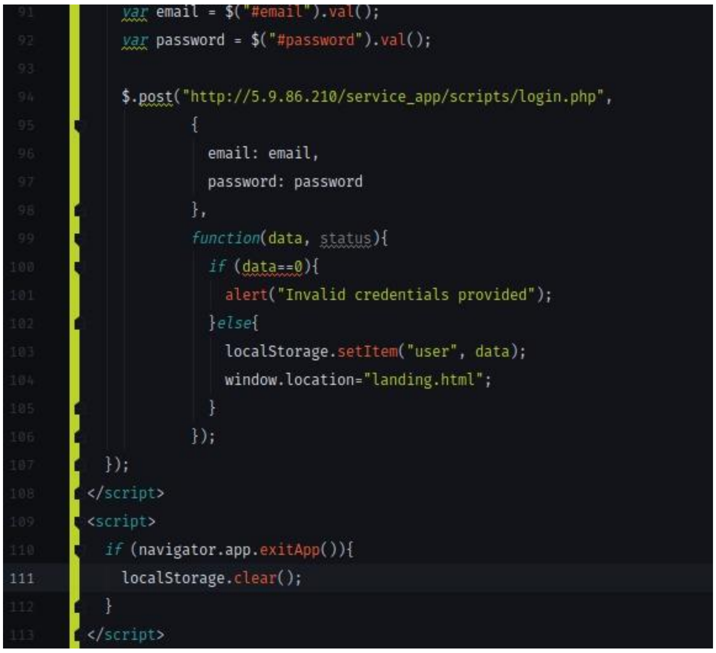
Following the component testing, a user test was then run, during which the application was installed on a Samsung NOTE 8 phone and used over 24 hours. During this period, the user accessed and used each function of the application multiple times.

During this period, it was observed that once the user logged onto the application, his session did not get terminated even when the application was closed, allowing him to skip the login page each time he re-accessed the application. After analyzing further, it was discovered that this was because the application locally stores a user’s session on the phone and only gets erased automatically when the device is restarted.

To check this security breach, a condition was introduced to ensure a user’s session is cleared from the device’s local storage each time the user attempts to quit the application. Doing this will protect the user’s usage details from others when they get access to your device.



```
<script>
  $("#btnLogin").click(function(){
    var email = $("#email").val();
    var password = $("#password").val();

    $.post("http://5.9.86.210/service_app/scripts/login.php",
            {
              email: email,
              password: password
            },
            function(data, status){
              if (data==0){
                alert("Invalid credentials provided");
              }else{
                localStorage.setItem("user", data);
                window.location="landing.html";
              }
            });
  });
</script>
```



```
    var email = $("#email").val();
    var password = $("#password").val();

    $.post("http://5.9.86.210/service_app/scripts/login.php",
            {
              email: email,
              password: password
            },
            function(data, status){
              if (data==0){
                alert("Invalid credentials provided");
              }else{
                localStorage.setItem("user", data);
                window.location="landing.html";
              }
            });
  });
</script>
<script>
  if (navigator.app.exitApp()){
    localStorage.clear();
  }
</script>
```

5.4. Regression Testing:

Considering the script updates and new code inclusions done, it was necessary to carry out a regression test to ensure existing functionalities were not affected by the updates. I did this by rerunning already run test cases against the updated system. The procedure I employed in doing this was to;

1. Identify the existing test scenarios to be run

2.Create the test approach templates

3.Categorize the templates according to the functions they apply to

4.Carry out the tests

5.Track the test results

Considering the changes done to the logout function as well as the quitting of the application; to check the security issues identified during my user testing, I rerun the user testing activities mainly focusing on the login and logout functions. After running these tests, it was confirmed that users were able to login successfully without any glitches. Other functions such as the logout feature also run successfully and satisfied the expected outcome.

As expected upon logging out of the system, all the users' details were deleted from the local storage and successfully cleared. The user was now required to provide his login details again to access the application.

5.5. Unit testing:

Both the web and mobile applications were taken through a series of unit tests during their developments. The tests were carried out incrementally as the development process progressed. The results of these tests were matched against a set of expected outcomes which then informed me on which corrections and tweaks were required to the source code. Examples of these tests are listed below;

Unit testing for services class

```
//Connecting  to DB
include '../connection.php';


if ($_POST['key']=='fetch'){


$responseArray=array();
$response = [];
$output  = array();

//Query details from the userTypes table
$queryDetails=mysqli_query($connect,"SELECT *FROM services");
  if($queryDetails){
    while($details=mysqli_fetch_array($queryDetails)){
       $serviceName=$details['serviceName'];
       $id=$details['id'];

         $response['x'] = $id;
         $response['y'] =  $serviceName;
         array_push($output,$serviceName);

      }
      echo print_r(json_encode($output));
  }else{
    echo "Query failed".mysqli_error($connect);
  }
}else{

$id=$_POST['id'];
$firstName=$_POST['firstName'];
$lastName=$_POST['lastName'];
$email=$_POST['email'];
$address=$_POST['address'];
$city=$_POST['city'];
$phoneNumber=$_POST['phoneNumber'];
$profession=$_POST['profession'];
$artisanService=$_POST['artisanService'];
$shopName=$_POST['shopName'];


$insertQuery=mysqli_query($connect,"UPDATE Service_providers SET
FirstName='$firstName',LastName='$lastName',Email='$email',
address='$address',City='$phoneNumber',Telephone_no='$phoneNumber',Service='$artisanService' WHERE id='$id'");
if($insertQuery){
  echo "Details Updated Successfully";
}else{
  echo "Update failed".mysqli_error($connect);
}

}
```

Summary of service class unit test

FUNCTION	EXPECTED RESULT	ACTUAL RESULT
Add new service	Service populates applications service list	Service populates applications service list
Edit service details	New service details update old details on application	New service details update old details on application
Delete service details	Service details should be removed from the application	Service details should be removed from the application

Unit testing for shop items class

```
include '../connection.php';


$shop=$_POST['shop'];
$itemName=$_POST['itemName'];
$itemDescription=$_POST['itemDescription'];
$itemPrice=$_POST['itemPrice'];
$itemStock  =$_POST['itemStock'];


$insertQuery=mysqli_query($connect,"INSERT INTO shopItems (item,description,price,numberAvailable,shop)
VALUES ('$itemName','$itemDescription','$itemPrice','$itemStock','$shop')");
  if($insertQuery){
    echo "Item Successfully created";
  }else{
    echo "insert failed".mysqli_error($connect);
  }
```

```
//Fetch details

//Connecting  to DB
include '../connection.php';

$id=$_POST['id'];

$holdingVariable="
    <table class='table table-hover'>
        <thead>
            <th>Item ID</th>
            <th>Item</th>
            <th>Item Description</th>
            <th>Price</th>
            <th>In Stock</th>
            <!--<th colspan='2'>Action</th>-->
        </thead>
        <tbody>
";
```

```
//Query details from the userTypes table
$queryDetails=mysqli_query($connect,"SELECT *FROM shopItems where shop='$id'");
  if($queryDetails){
    while($details=mysqli_fetch_array($queryDetails)){

            $itemID=$details['id'];
            $item=$details['item'];
            $description=$details['description'];
            $price=$details['price'];
```

Summary of shop items unit test

FUNCTION	EXPECTED RESULT	ACTUAL RESULT
Add new item	Item populates shops list of available items	Item populates shops list of available items
Edit item details	Item details should be updated on the application view	Item details should be updated on the application view
View item details	Item details should be populated in a tabular view on the app	Item details should be populated in a tabular view on the app

CHAPTER 6: Conclusion and Recommendations:

6.1. Chapter Introduction:

This chapter summarizes the complete project, discussing the functional requirements of the application and how they were met, limitations encountered and the future work that can be done to improve the system.

6.2. Summary:

This project set out to develop a mobile application that aids users in finding shops that have items they want to purchase and finding handymen who offer services they need. This requirement and other core functionalities of the application were achieved. However, there is room for improvement to make the system better.

6.3. Limitations:

One major limitation is the absence of the option to do a bulk upload when shop owners or the administrator is uploading a shop's inventory. This makes the process of adding an inventory slow and cumbersome; making it inefficient.

Another limitation is the need for internet connectivity to be able to access the updated data on the application. This means that in the absence of internet access, users and service providers will be unable to find shops and handymen.

6.4. Future Work:

Although all the stated requirements were met, there can be some additions to the application that can improve the system. Some suggested ways include:

- Bulk Upload: Introducing this function will make it easy and more efficient for shop owners and administrators to add a shop's inventory. It will increase the upload time also as multiple items can be added at a go instead of having to do it one after the other.

- Delivery option: Introducing the option to request for delivery will give users a wider range of options to choose from. They not only get to know the shops where they are purchasing from, but they get the option to get it delivered to them when they are unavailable.

6.5. Conclusion:

In conclusion, this application has taken the first step in providing an easier and secure means of conducting transactions involving shopping and hiring handymen. It has a simple and interactive interface that makes it easy to use and aids in understanding the application software.

The inclusion of the functions mentioned in section 6.4 will take it closer to being market ready.

References

[1] Abane, A. 1993. Tackling traffic congestion in Accra, Ghana: A road user's perspective. *Journal of Advanced Transportation*. 27, 2 (1993), 193-206.

[6] About us - Tonaton.com: 2018. *https://tonaton.com/en/help/about#help-content*. Accessed: 2019- 04- 23.

[4]"Consumer Online Shopping Attitudes and Behavior: An Assessment Of Research" by Na Li and Ping Zhang: 2002. *https://aisel.aisnet.org/amcis2002/74*. Accessed: 2019- 04- 23.

[3] Cybercrime in Ghana a Study of Offenders, Victims and the Law: 2015. *http://ugspace.ug.edu.gh/handle/123456789/8908?show=full*. Accessed: 2019- 04- 19.

[2] Internet and Online Shopping in an Emerging Economy: The Ghanaian Perspective: 2011. *http://pentvarsjournal.com/articles/january-2012/online-shopping.pdf*. Accessed: 2019- 04- 19.

[8] Telecom Voice Subscription » National Communications Authority: *https://nca.org.gh/industry-data-2/market-share-statistics-2/voice-2/*. Accessed: 2019- 04- 23.

[7] Topic: E-commerce worldwide: *https://www.statista.com/topics/871/online-shopping/*. Accessed: 2019- 04- 23.

[5] Understanding Cyber-Crime in Ghana: A View from Below: 2011. *http://cybercrimejournal.com/warner2011ijcc.pdf*. Accessed: 2019- 04- 23.

[9] What Is Jumia | Get To Know More About Us | Jumia Ghana: *https://www.jumia.com.gh/about_us/*. Accessed: 2019- 04- 23.