



**EGERTON UNIVERSITY**

**FACULTY OF SCIENCE**

**DEPARTMENT OF COMPUTER  
SCIENCE**

**PROJECT DOCUMENTATION FOR  
AN ONLINE AGRIBUSINESS SYSTEM**

**DEVELOPER: TOBIAS MUTUKU MUNYIVA**

**REG NO: S13/03125/18**

**SUPERVISOR: MR. INNOCENT NYALALA**

**CO-ORDINATOR: DR. BOSIRE OMARIBA**

**PROJECT PERIOD: OCTOBER 2022 -APRIL 2023**

## TABLE OF CONTENT

<b>ABSTRACT</b>	<b>vi</b>
<b>DOCUMENT I</b>	<b>1</b>
<b>PROJECT PROPOSAL</b>	<b>1</b>
<b>CHAPTER 1</b>	<b>1</b>
1.1 INTRODUCTION	1
1.2 CURRENT SYSTEM	1
1.3 PROBLEM STATEMENT	2
1.4 OBJECTIVES	3
1.4.1 GENERAL OBJECTIVES	3
1.4.2 SPECIAL OBJECTIVES	3
1.5 JUSTIFICATION	3
1.6 PROJECT SCOPE	4
<b>CHAPTER 2</b>	<b>5</b>
2.1 PROPOSED SYSTEM	5
2.2 FEASIBILITY STUDY	6
2.3 TECHNICAL FEASIBILITY	6
2.3.1 OPERATIONAL FEASIBILITY	6
2.3.2 ECONOMIC FEASIBILITY	7
2.3.3 SCHEDULE FEASIBILITY	7
2.4 SYSTEM DESIGN AND ANALYSIS	8
2.4.1 CONCEPTUAL FRAMEWORK	8
2.4.2 CONTEXT DIAGRAM	9
2.5 FUNCTIONAL REQUIREMENTS	10
2.6 NON-FUNCTIONAL REQUIREMENTS	10
2.7 PERFORMANCE REQUIREMENTS	11
2.8 SECURITY REQUIREMENTS	11
2.9 SOFTWARE REQUIREMENTS ATTRIBUTES	11
2.10 HARDWARE REQUIREMENTS	12
2.11 SOFTWARE TESTING	12

2.12 SOFTWARE DEPLOYMENT	12
<b>CHAPTER 3</b>	<b>13</b>
<b>CHAPTER 4</b>	<b>14</b>
<b>CHAPTER 5</b>	<b>15</b>
5.1 SCHEDULING	15
5.2 BUDGETTING	16
<b>CHAPTER 6</b>	<b>17</b>
6.1 CONCLUSION	17
6.2 FUTURE WORKS	17
<b>REFERENCES</b>	<b>18</b>
<b>DOCUMENT II</b>	<b>19</b>
<b>SOFTWARE REQUIREMENT SPECIFICATION(SRS)</b>	<b>19</b>
2.1 INTRODUCTION	19
2.2 OVERALL DESCRIPTION	19
2.2.1 PRODUCT PERSPECTIVE	19
2.2.2 PRODUCT FEATURES	22
2.2.3 USER PROBLEM STATEMENT	23
2.2.4 USER OBJECTIVES	23
2.2.5 ASSUMPTIONS	23
2.2.6 DEPENDENCIES	24
2.3 SYSTEM FEATURES	24
2.3.1 USER REGISTRATION AND AUTHENTICATAION	24
2.4 EXTERNAL INTERFACE REQUIREMENTS	25
2.5 FUNCTIONAL REQUIREMENTS	25
2.5.1 INPUT SPECIFICATIONS	25
2.5.2 PROCESSING SPECIFICATIONS	25
2.5.3 OUTPUT SPECIFICATIONS	26
2.5.4 SOFTWARE SPECIFICATIONS	26
2.5.5 HARDWARE SPECIFICATIONS	26
2.5.6 USER REQUIREMENTS	26

2.5.7 SYTEM REQUIREMENTS	26
2.6 NON-FUNCTIONAL REQUIREMENTS	28
2.6.1 PRODUCT REQUIREMENTS	28
2.6.2 ORGANIZATON REQUIREMENTS	28
2.7 PRELIMINARY OBJECT-ORIENTED DOMAIN ANALYSIS	28
<b>DOCUMENT III</b>	<b>30</b>
<b>SOFTWARE DESIGN DOCUMENTATION(SDD)</b>	<b>30</b>
3.1 INTRODUCTION	30
3.1.1 REVIEW SCOPE	30
3.1.2 EXECUTIVE SUMMARY	30
3.1.3 GLOSSARY	30
3.1.4 DEVELOPMENT METHODOLOGY	31
3.2 SYSTEM ARCHITECTURE	32
3.2.1 TOOLS USED	32
3.3 FILE AND DATABASE DESIGN	33
3.3.1 FILE DESIGN: -	33
3.3.2 DATABASE DESIGN: -	33
3.4 HUMAN MACHINE INTERFACE	34
3.4.1 USE CASE DIAGRAM	34
3.4.1.0 USE CASE TEMPLATES	34
3.5 DETAILED DESIGN	40
3.5.1 Data Flow Diagram (DFD)	40
3.5.2 SEQUENCE DIAGRAM	41
3.5.3 STATE CHART DIAGRAM	42
3.5.4 ACTIVITY DIAGRAM	43
3.5.4 ENTITY RELATIONSHIP DIAGRAMS	43
3.6 SYSTEM INTEGRITY CONTROLS	44
3.6.1 INTRODUCTION	44
3.6.2 ACCES CONTROLS	44
3.6.3 AUDIT TRAIL	44

3.6.4 DATA INTEGRITY	44
3.6.5 BACKUP AND RECOVERY	45
3.6.6 SECURITY MONITORING	45
<b>DOCUMENT IV</b>	<b>46</b>
<b>TEST PLAN</b>	<b>46</b>
4.1 INTRODUCTION	46
4.1.0 GOALS AND OBJECTIVES	46
4.1.1 SCOPE	46
4.2 TEST PLAN	47
4.2.1 TESTING STRATEGY	47
4.2.2 INTERFACES TO BE TESTED	47
4.2.3 BUG TRIAGE TESTING	47
4.3 TEST PROCEDURE	48
4.3.1 CRITERIA	48
4.3.2 UNIT TESTING	49
4.3.3 INTERGRATION TESTING	51
4.3.4 SYSTEM TESTING	52
4.4 TESTING RESOURCE PLANNING	52
4.4.1 SYSTEM RESOURCES REQUIRED	52
4.4.2 HUMAN RESOURCES REQUIRED	53
4.5 TEST ENVIRONMENT	53
4.5.1 HARDWARE REQUIREMENTS	53
4.5.2 SOFTWARE REQUIREMENTS	54
4.5.3 NETWORK REQUIREMENTS	54
4.6 TEST ESCHEDULE AND ESTIMATION	55
4.6.1 TESTING PHASES	55
4.6.2 TESTING SCHEDULE	55
<b>DOCUMENT IV</b>	<b>56</b>
<b>USER MANUAL</b>	<b>56</b>
5.1 SYSTEM OVERVIEW	56

5.2 SYSTEM REQUIREMENTS	56
5.2.1 HARDWARE REQUIREMENTS	56
5.2.2 SOFTWARE REQUIREMENTS	57
5.3 SYSTEM FEATURES	57
5.3.1 HOME PAGE AND NAVIGATION	57
5.3.2 NAVIGATION BAR	58
5.3.3 PRODUCTS CATALOGUE	58
5.3.4 LOG IN FORM	59
5.3.5 REGISTRATION FORM	59
5.3.6 CUSTOMER DASHBOARD	60
5.3.7 ADD TO CART	60
5.3.8 VIEW CART ITEMS	61
5.3.9 CHECKOUT	61
5.3.1.0 SHIPPING DETAILS AND PAYMENT OPTIONS	62
5.3.1.1 PAY	62
5.3.1.2 ADMIN DASHBOARD	63
5.3.1.3 ORDER MANAGEMENT	63
5.3.1.4 CUSTOMER MANAGEMENT	64
5.4 SYSTEM SUPPORT	64
5.4.1 SYSTEM INSTALLATION	64
5.4.2 TROUBLESHOOTING COMMON ISSUES	65

# ABSTRACT

An online agribusiness system is web system that enables farmers (sellers) to look for a ready market to sell their farm produce and also connect the customers (buyers) with the farmers directly and acquire fresh and quality products. The customers can register themselves in the website by clicking the register button, look through the products available in the system, identify them, check the price tag if it's suitable for him/her, add it to the cart and order anytime and at the comfort of his home or work place without going directly to the physical market place[1] This method is time consuming and tiring, since the customers have to physically find the products interested in manually in the market place and sometimes, they might not find the desired product and end up wasting a lot of energy and time since because it's not the season of such product.

Today's online agribusiness systems are designed to do a whole lot more than just reduce physical movement but also relinquish the middle man(brokers) in the physical market place who exploit customers and surcharge the products than normal. It also ensures that the products reach the buyers when fresh and do not overstay in the market since most of the farm produce are perishable and might go bad when they stay on the market place without customers. This makes the company more popular and attracts more customers increasing its sales activity.

*“UptownFresh”* which is the sales company buys farm produce from the farmers directly. Store the produce on their warehouses and do the shipping from there once a customer is authenticated and validated. The company monitors farmers throughout the planting season and estimated the harvesting time so that by that time they have ready market to sell because they have already advertised to the potential customers who sometimes might be the end users or resellers. This enables the company not to take too much time looking for customers while the products are in the store. Advertisement can be done on the website, flyers, roadshows, mainstream broadcasting or even one on one.

The main objective of this system is to design, develop and implement an efficient, user friendly, secure and interactive web-based agribusiness system.

# **DOCUMENT I**

## **PROJECT PROPOSAL**

### **CHAPTER 1**

#### **1.1 INTRODUCTION**

Computers have become part and parcel of our daily lives. A day can never pass by without humans interacting with at least one electronic gadget. Advancement of technology has increased over the years and it is very hard in this day and age for an organization to survive without the use of technology[2] Most of the companies, organizations and businesses nowadays have embraced the internet and have taken most of their functionalities online.

The earlier method of buying and selling which was done physically and manually has had so many issues. Some of the issues are middle men, tiresome means of transporting bulky products from physical market to the desired destination since customers are recording their financial statements manually and they may incur losses without their knowledge. Generating statistics and reports has also been cumbersome and prone to many errors[3] The system will generate financial report and purchase invoices for the customers once they make an order and also the company. This will enable company manage its finances and determine if they are making profits or losses in the business.

#### **1.2 CURRENT SYSTEM**

The current system is a manual system where everything is done with a minimum use of a computer. For every farm product you want to buy or sell, you have to appear physically at the market place on a specific day of the week. Most customers and farmers make long journeys to travel to market places and sometimes find the market filled up and sometimes queues outside so as to access the seller with some products which are in high demand and low supply. These farmers and buyers make



long queues in the market, burning in the scorching sun or feeling cold outside for products which they are not sure they will get.

In the current system in place people have not embraced the use of new technology and social media advertisement[4] Most people nowadays spend most of their time online, specifically on social media, embracing social media advertising increases the number of target customers. A potential customer might be roaming on the social media platform and comes across a farm product being advertised and gets interest to buy it so he/she is able to log into the system and make an order, make payment and the product is shipped to the desired destination.

## **1.3 PROBLEM STATEMENT**

The challenges encountered by the existing system serve as a major drawback to the realization of efficiency and seller and buyer satisfaction. The experience of walking to market places for customers and gathering all the products needed and for the farmers to transport their farm produce to the market it might be tiring and expensive.

Some of the problems occurred in the current system are: -

1. Expensive cost of transporting goods to the market by the farmers.
2. Customers being exploited by the middle man(brokers) who might double the price of the farm produce.
3. Cumbersome and tiring busy market places when customers are trying to locate and converge different products from different sections of the physical market.
4. Its time consuming since the sellers and buyers have to travel to the physical market place.
5. Due to low knowledge of customers about a new product in market, the product might go bad due to lack of customers and the owner will incur losses.
6. High chances of many perishables going bad due to enough preservation facilities in the open-air market.

7. Less demand of a certain product in the open-air market due to large production of the similar product hence farmers lack customers and sell in a throw away price and incur more losses.

## **1.4 OBJECTIVES**

### **1.4.1 GENERAL OBJECTIVES**

1. To improve and maximize the market scope of goods and service delivery between farmers and customer of farm produce through better applications of technology and increase on the user interface and experience.
2. To enable the “*UptownFresh*” farm produce sales company to stand out from all other competitors in the market and popularity to attract more farmers to sell us their farm produce and customers to buy from the industry.

### **1.4.2 SPECIAL OBJECTIVES**

1. To save on time and increase the efficiency of the farmers(seller) and customers(buyers) to ensure they don't get tired easily.
2. To enable customer to track the farm products available in the website at a certain period.
3. To enable the customers to have visual confirmation of the type of farm products availability and if the product is available to order and be delivered.
4. Eliminates brokers and increases the level of accuracy in finance management.

## **1.5 JUSTIFICATION**

1. To reduce time wastage by eliminating the middle man.
2. The system will be user friendly and provide a high efficiency in the ordering of products
3. To increase job customer satisfaction by speeding up the buying and delivering process of goods once a customer places an order.

## 1.6 PROJECT SCOPE

The main aim of the project is to ensure simplicity and improve the efficiency of goods movement process for both the farmers and customers in need of the products. Local farmers will benefit from it since they will be able to have a ready market to sell their farm produce and make profits instead of going looking for customer in the physical market[5] It will also minimize the manual way of farmers taking their products to open air market and customers travelling long distances to come and buy preferred products. The project also focuses on ensuring time management and expenses of going to purchase manually in the open-air market.

The ecommerce system will be a web-based application whose main language of programming will be PHP 8.0, JAVA SCRIPT, HTML 5 and CSS. The database version will be MYSQL 5.7. Customers will have the ability to view available products uploaded by the admin and the HR will be able to manage all the customers, their orders and manage the entire system.

# CHAPTER 2

## 2.1 PROPOSED SYSTEM

*“UptownFresh”* is the proposed system which will enable customers to register online and create accounts which enable them to make orders, purchase and make payments online and remotely. The customers details will be saved in the customers database details database. The admin will authenticate and validate the customers details before activating the customers’ account. He/she can deactivate a customer’s account when the account has suspicious activities, this is to secure the system from potential attacks since it has a financial sector which could fall prey to malicious attackers.

The system admin will get a detailed query of all the customers who log in and make orders through the website. From customers reviews about their customer service management the company and system administrator may get tips on how to improve the site’s efficiency, accuracy, responsiveness and security. The proposed system will have a section where customers will be able to search for a particular product if they don’t find in the products display, which will help them use the limited time to access the product and the customer save on energy used.

The system will also have a virtual shopping cart which will help the customers track the number of products to be purchased and also calculate the cost and even the shipping cost and give the total and final figure for the customer to pay online either through M-Pesa, PayPal or Bank. The system will also have fields for a customer to fill in the shipping details and address, the system can’t allow the customer to checkout and pay without filling in the shipping address This will ensure there is an efficient supply chain with clear logistics for exquisite service delivery to the customer.

The admin has the mandate and full control of the system and he/she is the only one who can upload the available products in the warehouse. He/she manages customer orders and will forward the order to the warehouse for shipping and logistics once the order has been validated and authenticated that the customer has paid the required amount.

## **2.2 FEASIBILITY STUDY**

This study assesses the operational, technical and economic metrics of the proposed project. The feasibility study is intended to view the facts if it is worthy of proceeding to the analysis stage[6] From the system analyst perspective, feasibility analysis is the primary tool for recommending whether to proceed to the next phase or terminate the project.

## **2.3 TECHNICAL FEASIBILITY**

This assessment is focused on gaining an understanding of the present technical resources of the organization and expanding the needs of the proposed system. It is an evaluation of the hardware and software and how it meets the needs of the proposed system. The systems project is considered technically feasible if the internal technical capability is sufficient to support the project requirements. Some of the essential questions that help in testing the technical feasibility are the following: -

1. Is the project feasible within the limits of current technology?
2. Is it a practical proposition?
3. Is it available within given resource constraints?
4. Does technology exist at all?
5. Does the required software and hardware exist?
6. Does the technology have the capacity to handle the solution?
7. Do we currently possess the necessary technology?

### **2.3.1 OPERATIONAL FEASIBILITY**

Is the measure of how well the project will support the customers and the service provider during the operational phase. It is dependent on human resources available for the project and involves projecting whether the system will be used if it is developed and implemented. Some of the essential questions that will help in testing the operation feasibility are the following: -

1. Is the system feasible to operate or not?

2. Could there be a reduction in cost or an increase in benefits?
3. Does the current mode of operation offer effective controls to protect against fraud and guarantee accuracy and security of data and information?
4. If the system is developed, will it be used?
5. Does it agree with the government regulations?
6. Will the proposed system really benefit the organization?
7. How will the end user feel about their role in the new system?

### **2.3.2 ECONOMIC FEASIBILITY**

This assessment aims to determine the positive impact that the proposed system will provide to the organization. It mainly involves the cost benefits analysis and its mostly used for evaluating the effectiveness of a new proposed system. Possible questions to be raised are:

1. Is the system cost effective?
2. The cost of doing the full system study?
3. Do benefits outweigh the cost?
4. Estimated cost of the hardware?
5. Estimated cost of the software development?
6. Is the project possible given the resources constrains?
7. Cost for customer time to study and learn how to use the software?
8. Cost of packaged software/software development?
9. The cost of doing full system study?

### **2.3.3 SCHEDULE FEASIBILITY**

Is the measure of how much time it will take to complete the project. Can the project be completed within a given period of time?

## 2.4 SYSTEM DESIGN AND ANALYSIS

The analysis specifies the system's objectives and constraints to which designers have to comply. The purpose of doing analysis is to transform the system's major input into structured specification.

### 2.4.1 CONCEPTUAL FRAMEWORK

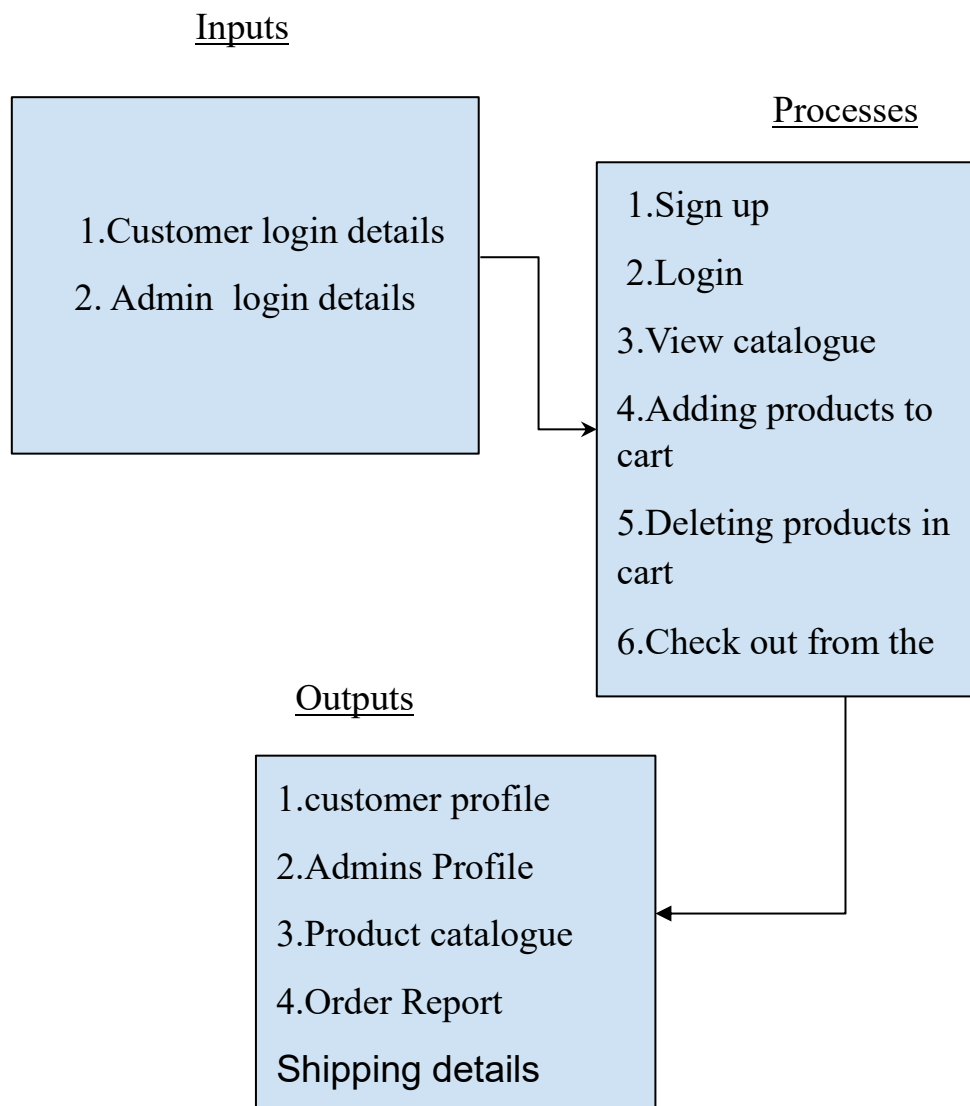


Figure 1: Conceptual framework of the system design

## 2.4.2 CONTEXT DIAGRAM

This is a brief structure which provides the environment in which a software system exists and helps in communicating about what lies outside the system boundary.

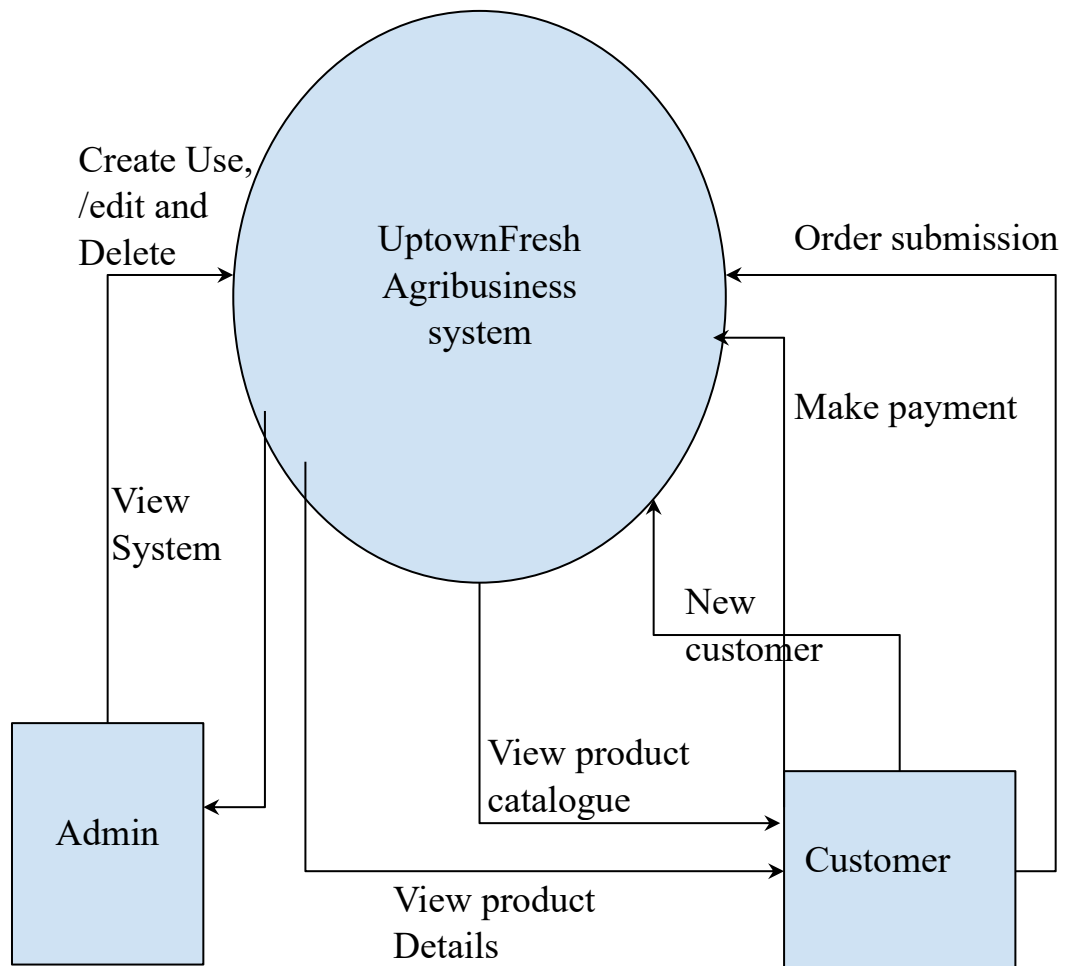


Figure 2: A context diagram to show various system functionalities



## **2.5 FUNCTIONAL REQUIREMENTS**

They define functions and functionality within and from the software system and along with the data operated on by the functions. They focus mainly on what the system should do.

1. Register farmers and customers.
2. Login farmers and customers.
3. Updating information that is products that the farmers are selling.
4. Information validation
5. User authentication.
6. Administration control.
7. View products in the catalogue
8. Submit an order
9. Make payments

## **2.6 NON-FUNCTIONAL REQUIREMENTS**

They address actions of the system other than the specific function it performs. They focus on how the system works or how the system should behave by providing its quality attributes. The aspects include system performance, cost and such general system characteristics such as reliability, security and portability.

The system will be of high performance with very high security features. The pages of the system are responsive and the entire system content can be contained in different screen size of different users hence making it easy and user friendly.

## **2.7 PERFORMANCE REQUIREMENTS**

Since User Experience (UX) is critical to the success or failure of our system in the market and performance is UX, we should put a strict requirement on our system's performance which includes: -

1. The system should support more than 50 users to checkout at the same time.
2. The response time of HTTP interfaces should be less than 5 seconds.
3. When the user requests data, the result shall be presented on the screen within no more than 5 seconds.

## **2.8 SECURITY REQUIREMENTS**

User's personal information like phone number and passwords should be encrypted before storing in databases. In this system Message-Digest hash function algorithm (MD 5 HASH) which is a type of cryptography protocol for digital signatures will be used to encrypt the log in passwords of the customers and of the admins.

## **2.9 SOFTWARE REQUIREMENTS ATTRIBUTES**

1. The system should be available 24/7.
2. The server shall be capable of supporting an arbitrary number of available products, that is, no product uploaded by the admin shall be lost under any circumstances.
3. Operating system-Windows XP/Windows 7,8,10.
4. Technology used PhP, Java Script, HTML, CSS and Bootstrap.
5. Database- MYSQL.
6. Backup and Data Recovery Software.

## 2.10 HARDWARE REQUIREMENTS

1. **Processor:** Intel dual core or above
2. **Processor Speed:** 1.0GHZ or above
3. **RAM:** 1 GB RAM or above
4. **Hard Disk:** 20 GB hard disk or above
5. Printer for printing reports
6. Uninterruptible power supply to ensure a constant access of data.
7. USB flash disk (At least 2GB)

## 2.11 SOFTWARE TESTING

Upon the completion of the development phase the system will be tested if it meets all the requirements it was designed for. This includes ensuring the customers are able to create an account, log in successfully, be able to view available products, add them to the cart and check out successfully and other functionalities.

## 2.12 SOFTWARE DEPLOYMENT

When the testing phase has been completed and the system approved that it meets all the requirements, then the deployment phase begins where it is now released to the public for use. The system will be hosted online so that the potential customers can access it and make their purchases using the website.

## CHAPTER 3

The final results of the whole project are to be able to come up with a system that will help the farmers to have a ready market and sell their products while still fresh to minimize losses after production.

The system will also help customers to easily buy the farm products at a cheaper price and minimizing time wastage and efforts of going to the physical open-air market to buy the products. It will also enable the company keep track of its financial status and be able to evaluate if it's making profit or losses.

# CHAPTER 4

During the development of the system some of the challenges, obstacles and risks faced are: -

1. High cost of the hardware like the laptops
2. Expensive internet connectivity since I have to buy internet data bundles so as I can do research
3. Shortage of power during the system development.
4. Lack of enough information from the locals because of language barriers and uncooperative individuals.
5. Absence of enough man power to see through the completion of the software development.

# CHAPTER 5

## 5.1 SCHEDULING

The system is scheduled to be developed, designed, tested and deployed in a period of 7 months. Within this time the system will be set for full functioning and deployment in the market.

**Table 1: Scheduling of the system design period**

Stage	period	October ---- November	November ---- December	January ---- February	February ---- march	March- --- April
1	Project definition & proposal					
2	SRS and SDD documentation					
3	Front end coding					
4	Backend coding					
5	testing					

## 5.2 BUDGETTING

The table below gives a brief summary of the total cost of the development and maintenance of the proposed system.

**Table 2: Approximated Budget for developing and designing the system**

	ITEM DESCRIPTION	AMOUNT (KSH)
1	PHP SOFTWARE	FREE
2	LAPTOP/ DESKTOP	60000
3	DATA COLLECTION	1000
4	FLASH DISK	500
5	STATIONERY AND PRINTING	400
6	INTERNET AND HOSTING CHARGES	100,000 per year
7	MISCELLANEOUS COST	4000

# **CHAPTER 6**

## **6.1 CONCLUSION**

After a thorough analysis of the manual system, the problems identified could not be improved or reduced manually. It was therefore seen as necessary to introduce a new user-friendly computerized system. The analyst waits for the approval from the management to proceed with the project. Upon approval, it is now necessary to make careful planning for the resources needed, project duration and its feasibility therefore the next chapter.

## **6.2 FUTURE WORKS**

The system will be basically be used in Kenyan region and future improvements will be made to accommodate international purchasing of products and shipping to other countries and continents. The system has only 2 users, that is the customer who their main work is to purchase and order good through the system and the second user is the system administrator whose work is to manage the whole system and its users.

Future improvements will be done to accommodate more system user with different specified roles like registering farmers so that they can post their produce directly into the system without relying on the system administrator. We will also introduce more payment methods like paying using cards like the debit cards, paypal and pay on delivery mode.

Since this is a web-based application system we will also come up with a mobile application for the same so that users who have mobile phones can also enjoy our services too.



# REFERENCES

- [1] J. G. Beierlein, K. C. Schneeberger, and D. D. Osburn, *Principles of agribusiness management*. Waveland Press, 2013.
- [2] D. K. Fisher, J. Norvell, S. Sonka, and M. J. Nelson, "Understanding technology adoption through system dynamics modeling: implications for agribusiness management," *The International Food and Agribusiness Management Review*, vol. 3, no. 3, pp. 281-296, 2000.
- [3] W. Ma, C. Gan, P. Vatsa, W. Yang, and H. Zheng, "Factors affecting online shopping frequency: lessons from New Zealand," (in eng), *SN Bus Econ*, vol. 2, no. 6, p. 41, 2022, doi: 10.1007/s43546-022-00214-5.
- [4] A. Semkin, A. Alpatov, E. Voronin, and K. Belyakova, "Areas of development strategy for the regional agribusiness management system," in *IOP Conference Series: Earth and Environmental Science*, 2019, vol. 403, no. 1: IOP Publishing, p. 012122.
- [5] L. Dubé, P. Webb, N. K. Arora, and P. Pingali, "Agriculture, health, and wealth convergence: bridging traditional food systems and modern agribusiness solutions," (in eng), *Ann N Y Acad Sci*, vol. 1331, pp. 1-14, Dec 2014, doi: 10.1111/nyas.12602.
- [6] C. A. Jennissen, S. Sweat, K. Wetjen, P. Hoogerwerf, and G. M. Denning, "Engaging Agribusinesses: Feasibility and Cost of an ATV Safety Poster Project," (in eng), *J Agromedicine*, vol. 22, no. 4, pp. 364-375, 2017, doi: 10.1080/1059924x.2017.1358228.

# DOCUMENT II

## SOFTWARE REQUIREMENT SPECIFICATION(SRS)

### 2.1 INTRODUCTION

The System Requirements Specifications (SRS) for an online agribusiness system describes the design decisions, architectural designs and the detailed design to implement a functional and well working ecommerce system. The SRS document is an essential part of the software development process, as it serves as a blueprint for the system design, development and testing phases.

The SRS document simplifies the complexities in the implementations of the ecommerce system so that the customers and the administrators can easily understand the functionalities and capabilities of the system. Furthermore, this SRS gives a detailed explanation of the high-level structure of the ecommerce system in terms of functions, performance, external interfaces and design.

The SRS for an online agribusiness system is a crucial document that ensures the successful development and deployment of a fully functioning and efficient ecommerce system that meets the customers' and service providers' needs, expectations and requirements.

### 2.2 OVERALL DESCRIPTION

This section describes the general interpretation of the online agribusiness system in terms of the product perspective, product features, user problem statement, user objectives, operating environment, design and implementation constraints, user documentation, assumptions and dependencies and user constraints.

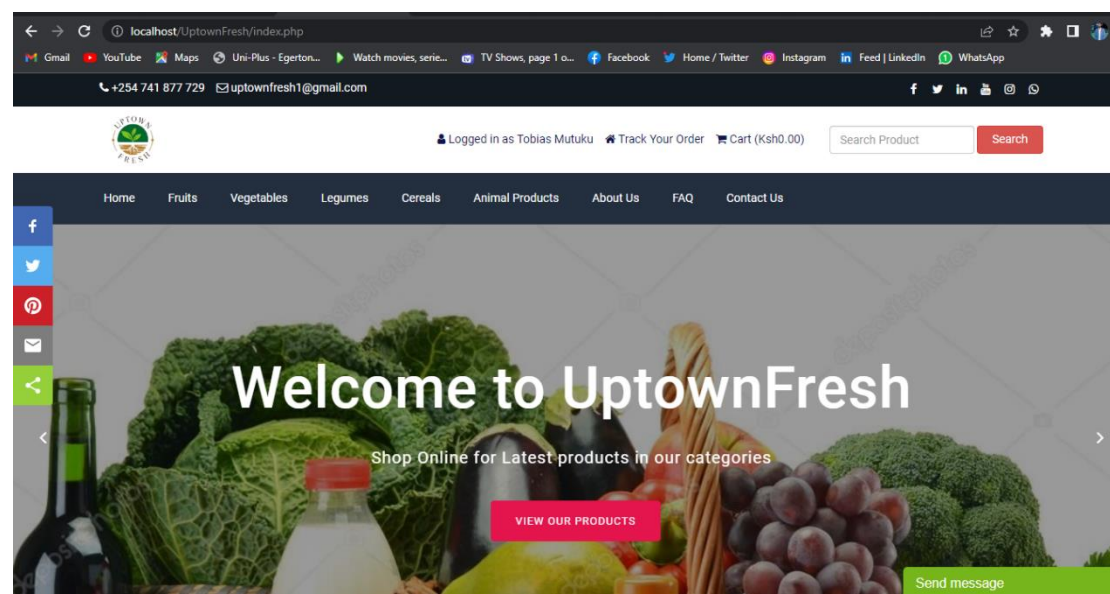
#### 2.2.1 PRODUCT PERSPECTIVE

*“UptownFresh”* is an advancement of the current system used in purchasing farm products from the physical open-air market where record keeping is done mostly by paper work and farmers who are the sellers and customers who are the buyers present themselves to the open-air market physically to sell and buy farm products. It's of much importance to produce such a system since it will increase the productivity of each

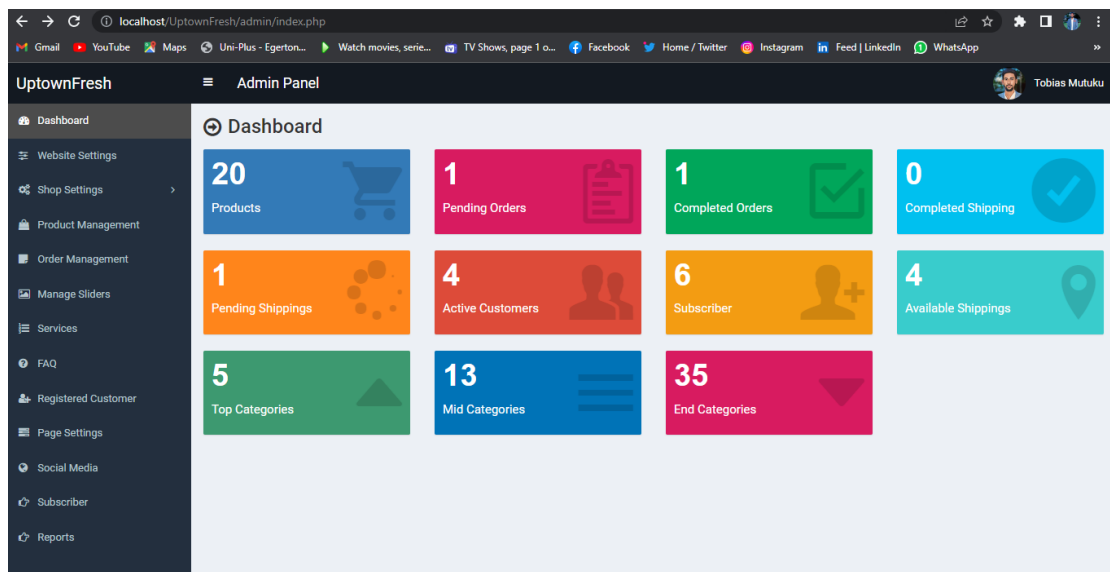
person involved in the selling and buying processes since time wastage will be reduced since physical movement will be eliminated and also money management since there will be elimination of the middle men who are the brokers who exploit both farmers and the customers. This system is therefore steered by the goodwill to help more farmers and customers country wide more so those in the town to locate fresh and high-quality farm produce easily. **“UptownFresh”** is a web-based application system that consists of three parts: the client/customer side application which runs on the web browsers and back-office part which is used by the system administrator to manage the system, view details of customers, their orders, products available for uploads, verify customer payments, activate customers and authorize shipping of purchased goods.

**“UptownFresh”** requires access to the internet to be able to register, log in and place an order and also check the order progress and shipping. This system also needs somewhere to store the data, that is the database. The data stored in the database includes the customers details, system administrators’ details, products details, order details and other system information.

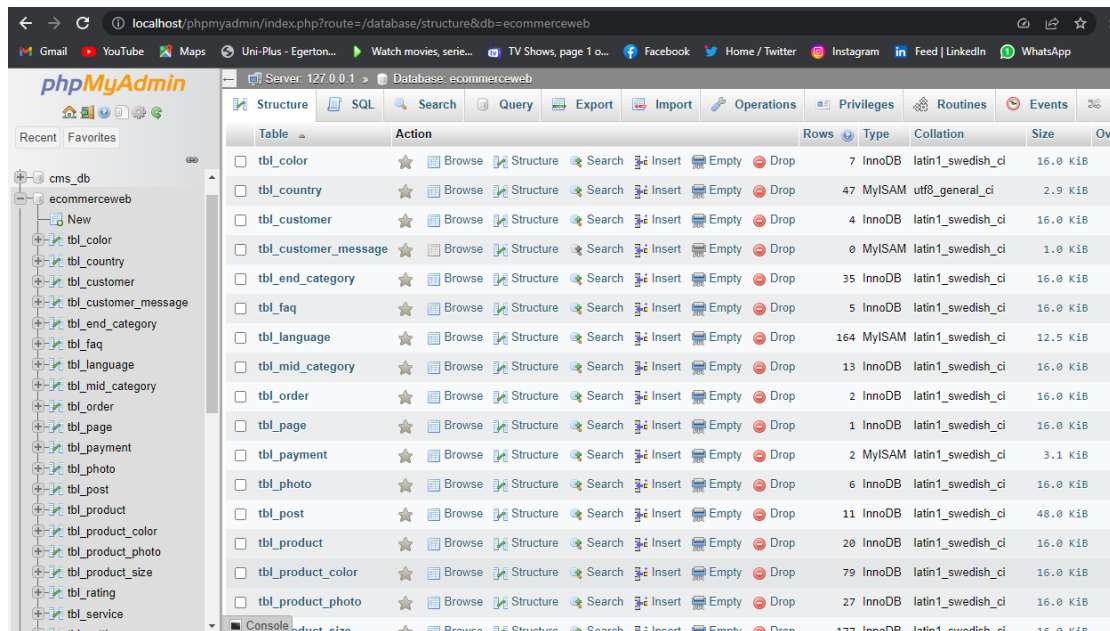
The database used in this system is the SQL database mainly because it allows flexibility of access for the users. Figure 1, 2 and 3 shows the specified version of the two major parts that make up the **“UptownFresh”** application system: the web application and the database which are connected together using internet.



**Figure 3: The user interface for customers uses**



**Figure 4: Administrators' panel for official use only.**



**Figure 5: MySQL database connected to the web application**

## 2.2.2 PRODUCT FEATURES

Product features is a description of the characteristics, appearance, components and functionalities of the system. The following list shows a brief outline of the major functionalities and features of this online agribusiness system.

- 1) User registration and authentication.
  - ✓ Registration page for one to register and create account where the user clicks the “register” button and submits the details where the administrator confirms and activates the users account so that the user can be able to log in.
  - ✓ The log in page accessed through the “log in” button whereby after the users account has been activated, he/she can be able to log in successfully.
- 2) View products catalogue and placing an order.
  - ✓ After the customer has logged into the account, he /she can be able to view the available products and add them to the cart and check out.
  - ✓ When the customer shops the needed products, they are able to place and order by proceeding to check out where they make payments and places an order successfully.
- 3) Update profile and shipping details.
  - ✓ Customers are able to update their accounts profile and change passwords in case of any need to.
  - ✓ They are also able to change and update their billing and shipping details in case they change their recent location.
- 4) View orders and print receipt.
  - ✓ Customers after making payment, they are able to view their order and confirm if the system administrator has confirmed their payment successfully.
  - ✓ They can also download print their purchase receipts after the payment has been confirmed.
- 5) Update products and system information.
  - ✓ The system administrator is able to manage the whole system by uploading goods, activating and deactivating customers’ accounts, validating and verifying customers purchases and authorizes shipping.

### **2.2.3 USER PROBLEM STATEMENT**

In modern days, there has been an increase in demand for fresh and high-quality agricultural products by the consumers, however the traditional way of shipping products from the farms by the farmers have had several challenges that has led to customer dissatisfaction including long wait time for products to reach the physical open-air market, invasions by the middle men, unclear pricing models, complexity in transporting the goods from the market to the desired destinations, spoiling of goods in the stores due to poor storage facilities and high production with consequently low demand. To address these challenges, there is a need to come up with a digital way to provide a seamless and efficient application system to smooth user experience.

### **2.2.4 USER OBJECTIVES**

The primary objective of the online agribusiness system is to provide a user-friendly platform by enhancing customer experience. A system that enables customers to purchase and order goods quickly and easily at the comfort of their home or work place. The system should also have the following specific objectives: -

- 1) To provide streamlined services to the customers in shortest time possible: The system should automate the purchasing process, eliminate brokers, fast and efficient shipping of goods and reduce wait time.
- 2) Provide transparent pricing whereby the system displays prices of products for every user to see clearly.
- 3) Ensure availability of products which are in demand and should be accessed in real time and customers purchase and make orders instantly.

### **2.2.5 ASSUMPTIONS**

While developing the system requirements specifications for this ecommerce system, the following assumptions were made: -

- 1) The application system is designed for all mobile handsets that has browsers with access to internet connection, desktops and portable laptops that enough memory capacity and reliable processor.
- 2) The system will be developed using the most modern versions of software development methodologies.

- 3) The system will be developed using the state-of-the-art technologies to ensure optimum performance, scalability and security.
- 4) The system will have a database that will store customers details, order details, products and other system information.

### **2.2.6 DEPENDENCIES**

The following dependencies should be considered while in the process of developing this ecommerce system: -

- 1) The system development will depend on the availability of skilled software developer and project managers.
- 2) The systems success will depend on the cooperation and support of stakeholders including farmers, customers and internet service providers.
- 3) The system development will depend on the availability of relevant development tools and technologies.
- 4) The system will also require specified hardware for deployment and functionality.

## **2.3 SYSTEM FEATURES**

### **2.3.1 USER REGISTRATION AND AUTHENTICATAION**

Once the users access the application, they can view different categories of products but cannot make an order without creating an account through the “register” button. After the administrator has activated their account, the customer can now log in the system successfully and make his/her orders. When the user wants to log in their prompted to enter their email address and the password, they set during account creation.

The credentials are sent to the database for verification and validation. Once they have been ascertained to be correct the system logs in the user and taken to the dashboard which has different action buttons like customer update profile, update billing and shipping information, view orders, print receipt and track order progress.

## **2.4 EXTERNAL INTERFACE REQUIREMENTS**

- 1) The system should be designed to handle a large number of users at ago. It should be optimized for speed and efficiency to ensure a seamless user experience.
- 2) The system should be supported by all browser.
- 3) The system should comply with accessibility standards such as web content accessibility guidelines to ensure that all users can access all the systems functionalities without failure.
- 4) The system should also be compatible with most operating systems.
- 5) The system should comply with legal requirements including customer data protection regulations, privacy and consumer protection laws.
- 6) The system should also be able to run on various hardware configuration devices like desktops, laptops and smartphones.

## **2.5 FUNCTIONAL REQUIREMENTS**

### **2.5.1 INPUT SPECIFICATIONS**

- 1) First time customer must provide their details for registration and account creation.
- 2) Customers must use the set credentials to log into the system that is the email address and password.
- 3) Customers can enter their billing and shipping information and update them when need be.
- 4) Users can also update their profile and change passwords when need be.
- 5) Users can send messages to administrators by using the contact form.
- 6) Customers can order and purchase goods through the virtual cart and make payments.
- 7) The users can send email to the admin through the companies email address or call, SMS them directly through the companies contact number.
- 8) The user can also subscribe to company's newsletter.

### **2.5.2 PROCESSING SPECIFICATIONS**

- 1) The system can process the number of registered customers and subscribers.
- 2) The system can enable the admin to review customer orders, payments and can confirm or cancel them.
- 3) Can also removes subscribers and inactivate customers' accounts.



### **2.5.3 OUTPUT SPECIFICATIONS**

- 1) Display customers' profile.
- 2) Display customers billing and shipping information.
- 3) Display a list of all the customers' orders.
- 4) Display products available in the store/warehouse for purchase and shipping.
- 5) Display the number of all registered customers and subscribers.
- 6) Display the messages sent by the customers.
- 7) Display the list of all the products available.
- 8) Display all shipping available.

### **2.5.4 SOFTWARE SPECIFICATIONS**

1. The application system can be supported and run at any known operating systems.
2. The application system can also be used in available web browsers like google chrome, Mozilla etc.

### **2.5.5 HARDWARE SPECIFICATIONS**

1. A computer with 2GB RAM memory and a minimum of 300Mhz processing speed.
2. A hard drive of 5GB of storage memory.

### **2.5.6 USER REQUIREMENTS**

The application system will provide online purchase of agricultural products, make payments and follow ups on if the payment was verified and shipping of the goods to the desired location.

### **2.5.7 SYSTEM REQUIREMENTS**

Customers must have been registered by creating an account and the system administrator has activated the account. The admin will verify the details which identifies the customer before activating the account. Once the account is active, the customer can now be able to make orders and make payments online.

The system users should be privileged with the following functionalities: -

1. They should be able to manage their accounts i.e., update profile, change password, update billing and shopping details.

2. Can be able to log in the platform.

The system should provide the following functionalities to user: -

1. Allow full navigation of the system.
2. View and edit profile.
3. View the agricultural products available.
4. Register to the portal.
5. Subscribe to the system information.
6. Add products to the virtual cart.
7. View the products added to the cart and total amount.
8. Checkout from the cart to make payment and place an order.
9. View the list of the orders made.
10. Send messages to the admin through the email.
11. Contact the admin through the contact message form.
12. Print receipts after making an order.

The system should provide the following functionalities to the admin: -

1. Access the admin dashboard.
2. Add and change the products information.
3. View, activate and deactivate customers' accounts.
4. Delete subscribers.
5. View the number of the registered customers and subscribers.
6. View the list of available products.
7. View customer orders.
8. Update customers payment information.
9. Authorize customers' orders for shipping.
10. View and print reports.

Other requirements include the following.

- 1) The system should be able to display user profile details that can be edited.
- 2) The system should capture the users' credentials.
- 3) It should have advanced monitoring and accountability of information.
- 4) It should be able to give customers notification after making actions.

## **2.6 NON-FUNCTIONAL REQUIREMENTS**

### **2.6.1 PRODUCT REQUIREMENTS**

1. The system should be available and fully operational throughout.
2. Accessibility clearance of the system information may only be granted to the system administrator.
3. During password entry the credentials should not be visible by hashing them.
4. The system should provide user-friendly interface for easier navigation.
5. Response to view information shall not take longer than 5 seconds to display.
6. System response to user's request shall not take less than 4 seconds.
7. Constant daily refreshment of the information system despite the absence of updates.
8. Update failure must be able to be rolled back to default.

### **2.6.2 ORGANIZATON REQUIREMENTS**

1. The system database that will be used is MySQL.
2. The backend application will be created using PHP.

## **2.7 PRELIMINARY OBJECT-ORIENTED DOMAIN ANALYSIS**

1. Farm products, this is the primary object in the ecommerce system. its attributes include product name, product size, product color, product price. The behaviors associated with the product include confirming or cancelling customers' orders.
2. Customers, this object represents the person who makes the purchase. Its attribute include name, company name, county, street, zip code, address, phone number, email address. The behaviors involved with customer include adding products in the cart, checkout, make payment, update personal information and updating billing and shipping information.

3. Purchasing products, this object represents customers request to purchase a product and make payment. Its attributes include adding interested products to the virtual cart, view the products in the cart, check the total summed up cost, check the shipping cost, proceed to checkout, make payment, check order details, print receipts, track order. The behaviors involved include cancelling orders, cancelling payments and logging out of the system.

# DOCUMENT III

## SOFTWARE DESIGN DOCUMENTATION(SDD)

### 3.1 INTRODUCTION

#### 3.1.1 REVIEW SCOPE

The System Design Documentation (SDD) for an online agribusiness system is used to provide a detailed system design. The system will allow customers buy goods online in bulky, view availability and manage their orders. The system will also manage a backend management interface for the online agribusiness administrator to manage the inventory, orders and customers data and information.

#### 3.1.2 EXECUTIVE SUMMARY

The agribusiness is an online platform that provides a convenient way for visitors to view products and register an account and become a customer. As a customer you can be able to make orders in bulky and online make online payment through M-Pesa or Bank transfer. The system is designed to be user friendly and easy to navigate with features such as search button to help customers find the desired product easily.

Overall, the online agribusiness is designed to stream line the buying and selling process of farm products and a well-defined administrator platform to manage the entire system.

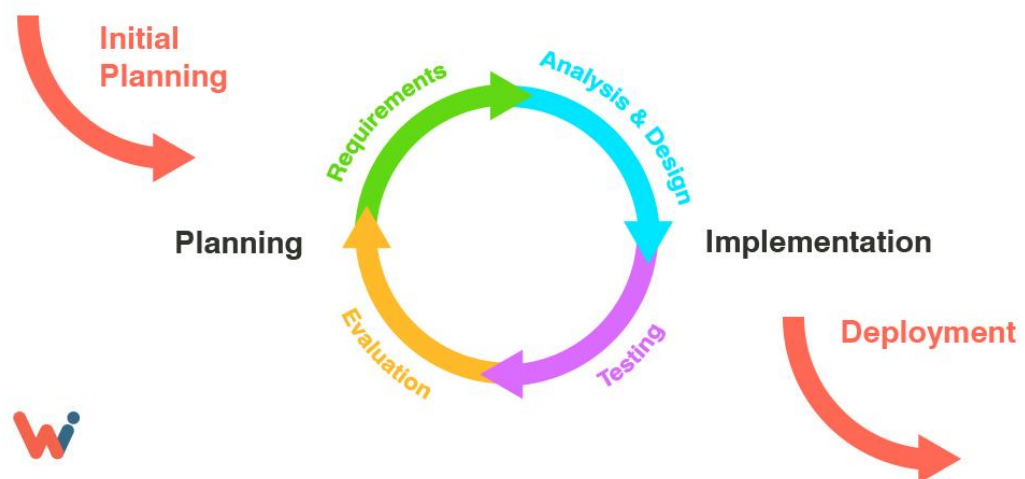
#### 3.1.3 GLOSSARY

1. Online Agribusiness system: An online platform that allows customers to make orders from the products available in the catalogue.
2. Backend management interface: A secure platform that allows the online agribusiness administrators to manage inventory, orders, and customer data.
3. Order history: A record of a customer's past orders.
4. Inventory: A list of available products in the catalogue.
5. orders: A record of customer orders
6. Data Privacy - The protection of personal data and ensuring that data is only accessed by authorized individuals or systems.

7. Performance and Reliability - The ability of the car rental system to perform efficiently and reliably, including speed, uptime, and response time.
  1. User Interface - The graphical user interface (GUI) or user interface (UI) is the front-end component of the system that the user interacts with, allowing them to input data, retrieve data, and interact with the system.
8. Pricing Calculation - The process of calculating the product price for a specific product based on factors such as how fresh it is, quality and demand.

### 3.1.4 DEVELOPMENT METHODOLOGY

The method used in the designing and development of this system was the iteration design and development model as shown below. This is because I had to repeat a number of design stages to make sure they were working properly.



**Figure 6: waterfall model diagram used for system design**

1. Requirement analysis and gathering: in this phase we analyzed the requirements to develop the system and implement it.
2. System design: in this phase we designed the system from the analysis made initially.
3. Implementation: here the system is implemented from the system design.
4. Testing: the units implemented in the system are tested to check for any faults.
5. Deployment: on testing the system was deployed to be used.
6. Maintenance: the system undergoes constant maintenance and updates to make it more efficient.

## **3.2 SYSTEM ARCHITECTURE**

The system architecture in online agribusiness refers to the overall design and structure of the software system that facilitates the online buying and selling process. This includes the various software components, database, user interface and external systems that work together to provide the required functionality.

A typical online agribusiness system architecture would include the following components: -

1. **User Interface:** This is the front-end of the system that the users interact with. It includes the web interface or mobile app where users can search for available farm products, make orders, and view their order history.
2. **Order Management System:** This component manages the entire ordering process, including product availability, ordering date and pricing.
3. **Inventory Management System:** This component manages the inventory of available products. It keeps track of the goods and their availability.

### **3.2.1 TOOLS USED**

The following tools were used to design and develop the system: -

1. HTML
2. JavaScript
3. PHP
4. CSS
5. Bootstrap
6. MySQL
7. XAMPP

## **3.3 FILE AND DATABASE DESIGN**

### **3.3.1 FILE DESIGN: -**

1. User file: This file contains information about the customers of the online agribusiness system. It includes fields such as full name, address, phone number, email address, county, street, company name, postal code and city.
2. Product file: This file contains information about the products available for purchasing and ordering in the system. It includes fields such as product name, product description, purchase price, discount, return policy and other details.
3. Ordering file: This file contains information about the orders made by the customers. It includes fields such as customer details, order dates, shipping location, amount paid and other details.

### **3.3.2 DATABASE DESIGN: -**

- Users table: This table contains information about the customers of the online agribusiness system. It includes fields such as customer ID (primary key), name, address, phone number, email address, county name, city, street, postal code, company name and other details.
- Products table: This table contains information about the products available for purchase in the system. It includes fields such as product ID (primary key), product name, featured photo, current price, old price, description, quantity, return policy and other details.
- Admin table. Contains user name, photo, email, status and password.
- Order table. Contains user ID, Product ID, product name, quantity, size, unit price and payment ID.
- Payment table. It contains user id, customer name, customer email address, payment date, payment amount, payment method, payment ID and payment and shipping status.
- Subscribers table. It contains the subscriber email and posting date.
- Shipping cost table: It contains shipping cost ID, county ID and shipping amount.



- Contact us info table. The table contains address email and contact number.
- County table: It contains county ID and county name.

## 3.4 HUMAN MACHINE INTERFACE

### 3.4.1 USE CASE DIAGRAM

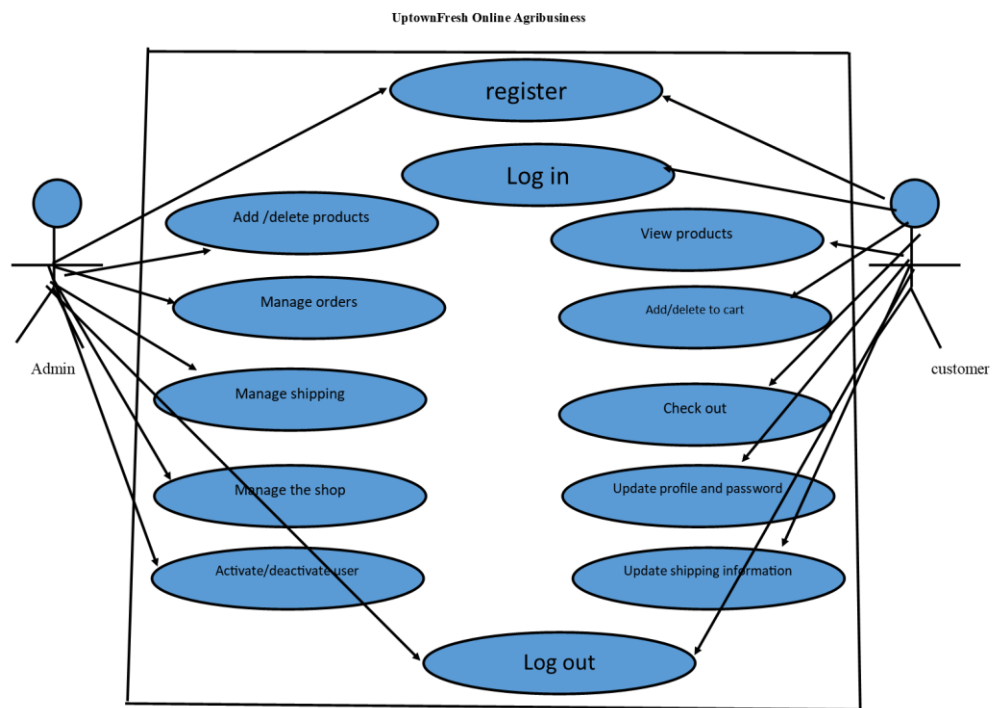


Figure 7: Use Case Diagram

#### 3.4.1.0 USE CASE TEMPLATES

Table 3: Register Template

<b>Use case:</b>	Register
<b>Actors:</b>	Customer
<b>Short description:</b>	The customer registers himself/herself into the platform for the first time.
<b>Pre-conditions:</b>	Must not be a registered user.
<b>Post-conditions:</b>	The user is registered in to the platform.

<b>Main flow:</b>	1) The customer opens the website. 2) Click the ‘register’ button. 3) Fill the form provided and click the “register” button.
<b>Alternative flow:</b>	None.
<b>Exception flow:</b>	The users’ email already exists in the system then the use case is terminated.

**Table 4: Log in Template**

<b>Use case:</b>	Log in
<b>Actors:</b>	Admin, Customer
<b>Short description:</b>	The users log in to the platform
<b>Pre-conditions:</b>	Must be already registered into the system.
<b>Post-conditions:</b>	The user is logged in to the platform.
<b>Main flow:</b>	1) The customer/admin enters their personal credentials i.e., user enters email and password. 2) The entered credentials are verified. 3) After verification the user is logged in to the system.
<b>Alternative flow:</b>	None.
<b>Exception flow:</b>	The users’ credentials are invalid-the use case is terminated.

**Table 5: View Product Template**

<b>Use case:</b>	View product
<b>Actors:</b>	Customer
<b>Short description:</b>	The customer views the products available in the catalogue

<b>Pre-conditions:</b>	Must have been able to access the system. The customer can either be logged into the system or not.
<b>Post-conditions:</b>	The user is just a visitor or customer in to the platform.
<b>Main flow:</b>	<ol style="list-style-type: none"> <li>1) The customer must not be logged in to the website.</li> <li>2) Views the products available</li> <li>3) Can be able to see the prices, colors and quantities of the products</li> </ol>
<b>Alternative flow:</b>	None
<b>Exception flow:</b>	None

**Table 6: Add product Template**

<b>Use case:</b>	Add product
<b>Actors:</b>	Customer
<b>Short description:</b>	The customer adds the product he/she desires.
<b>Pre-conditions:</b>	Must have a registered account and also must be logged in to the system.
<b>Post-conditions:</b>	The user is registered in to the platform.
<b>Main flow:</b>	<ol style="list-style-type: none"> <li>4) The customer logs in to the website.</li> <li>5) Select the product of their choice.</li> <li>6) On the page opened he sets the product specifications like quantity, size and color</li> <li>7) Click on the add to cart button.</li> </ol>
<b>Alternative flow:</b>	None
<b>Exception flow:</b>	If the user clicks on a product that is not available the system gives him a “Out of stock” error message and the page returns to the home page to select other products.

**Table 7: Check Out Template**

<b>Use case:</b>	Check out
<b>Actors:</b>	Customer
<b>Short description:</b>	The customer after confirming her products is well, he/she proceeds to check out and make payments
<b>Pre-conditions:</b>	Must have a registered account, must be logged in to the system and also must have updated the shipping and billing information.
<b>Post-conditions:</b>	The user is registered in to the platform and provided shipping information.
<b>Main flow:</b>	<ol style="list-style-type: none"> <li>1) The customer logs in to the website.</li> <li>2) Select the products of their choice.</li> <li>3) On the page opened he sets the product specifications like quantity, size and color</li> <li>4) Click on the proceed to check out button.</li> </ol>
<b>Alternative flow:</b>	If the user has not provided the shipping and billing information, he/she is provided with a link to the page to input the information
<b>Exception flow:</b>	The user can provide the shipping information before trying to check out.

**Table 8: Order Payment**

<b>Use case:</b>	Order payment
<b>Actors:</b>	Customer
<b>Short description:</b>	The customer after checking out and confirming the product price and shipping cost he/she makes payment to place his order
<b>Pre-conditions:</b>	Must have a provided the shipping information in to the system.
<b>Post-conditions:</b>	The user must have money to pay for the products.
<b>Main flow:</b>	<ol style="list-style-type: none"> <li>1) The customer logs in to the website.</li> </ol>

	2) Select the product of their choice. 3) Confirm the total price and the shipping cost 4) Click on the proceed to checkout and then select payment method 5) Click on the pay button
<b>Alternative flow:</b>	If the customer does not have money in their wallet the order is not success.
<b>Exception flow:</b>	None

**Table 9: Add Product Template**

<b>Use case:</b>	Add product
<b>Actors:</b>	Admin
<b>Short description:</b>	New products are added in to the system.
<b>Pre-conditions:</b>	Must be logged in to the dashboard.
<b>Post-conditions:</b>	None.
<b>Main flow:</b>	1) The admin logs in to the dashboard. 2) Select the products management tab on the left panel. 3) Select 'add product'. 4) Fill out the form provided. 5) Click submit changes.
<b>Alternative flow:</b>	None.
<b>Exception flow:</b>	None.

**Table 10: Update password**

<b>Use case:</b>	Update password
<b>Actors:</b>	Admin, customer
<b>Short description:</b>	Changing the old password of the user.
<b>Pre-conditions:</b>	Must be logged in to the system.
<b>Post-conditions:</b>	None.
<b>Main flow:</b>	<ol style="list-style-type: none"> <li>1) The user logs in to the system.</li> <li>2) Select the update password option.</li> <li>3) Enter your old password.</li> <li>4) Enter new password and repeat it again.</li> <li>5) Click save changes.</li> </ol>
<b>Alternative flow:</b>	<p>If user enters the wrong old password a prompt is indicated.</p> <p>If the new password entered in the first field does not match the password entered in the second field the user is prompted.</p>
<b>Exception flow:</b>	None.

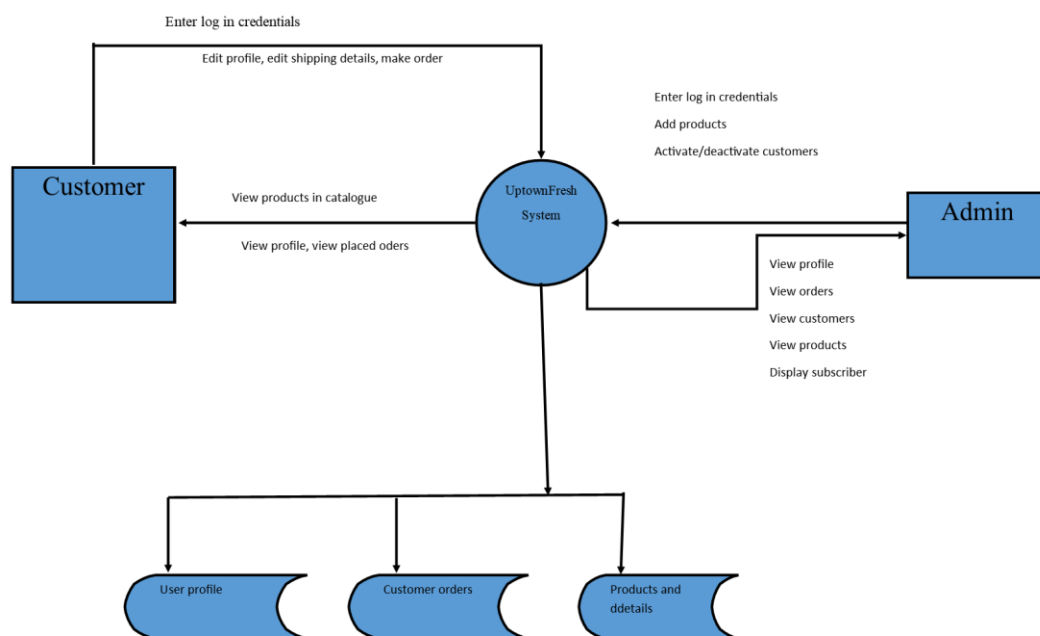
**Table 11: Add shipping cost**

<b>Use case:</b>	Add shipping cost
<b>Actors:</b>	Admin
<b>Short description:</b>	New shipping cost added in to the system.
<b>Pre-conditions:</b>	Must be logged in to the dashboard.
<b>Post-conditions:</b>	None.
<b>Main flow:</b>	<ol style="list-style-type: none"> <li>1) The admin logs in to the dashboard.</li> <li>2) Select the shop settings on the left panel.</li> <li>3) Select 'shipping cost'.</li> </ol>

	4) Enter the county name and the new shipping cost. 5) Click submit.
<b>Alternative flow:</b>	None.
<b>Exception flow:</b>	None.

## 3.5 DETAILED DESIGN

### 3.5.1 Data Flow Diagram (DFD)



**Figure 8: Detailed Data Flow Diagram level 1**

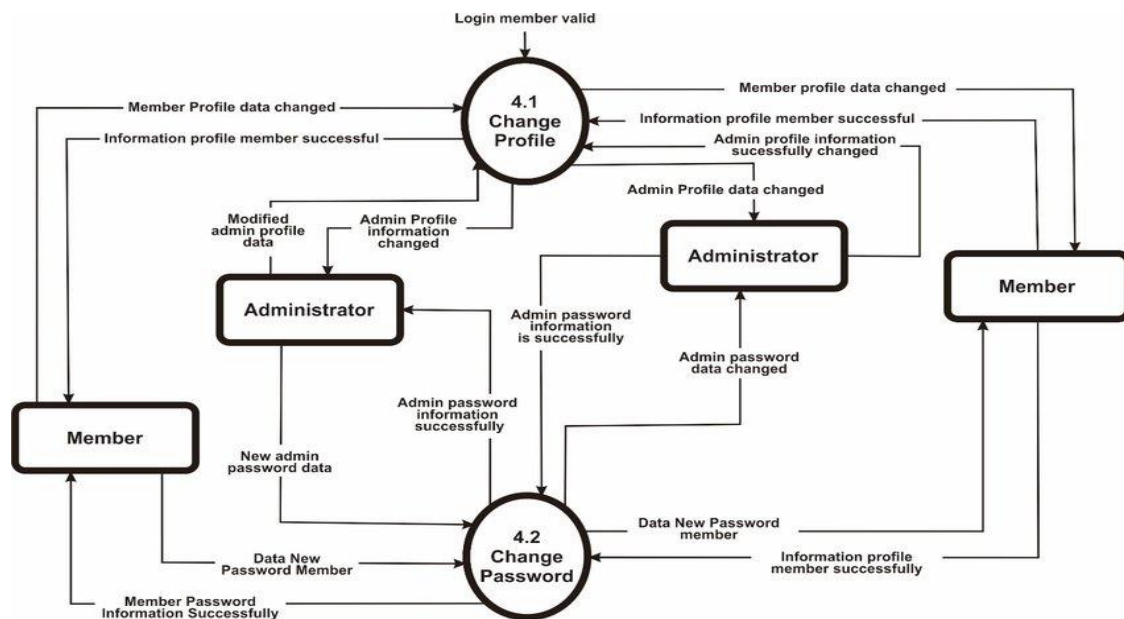


Figure 9: DFD Level 2 diagram

### 3.5.2 SEQUENCE DIAGRAM

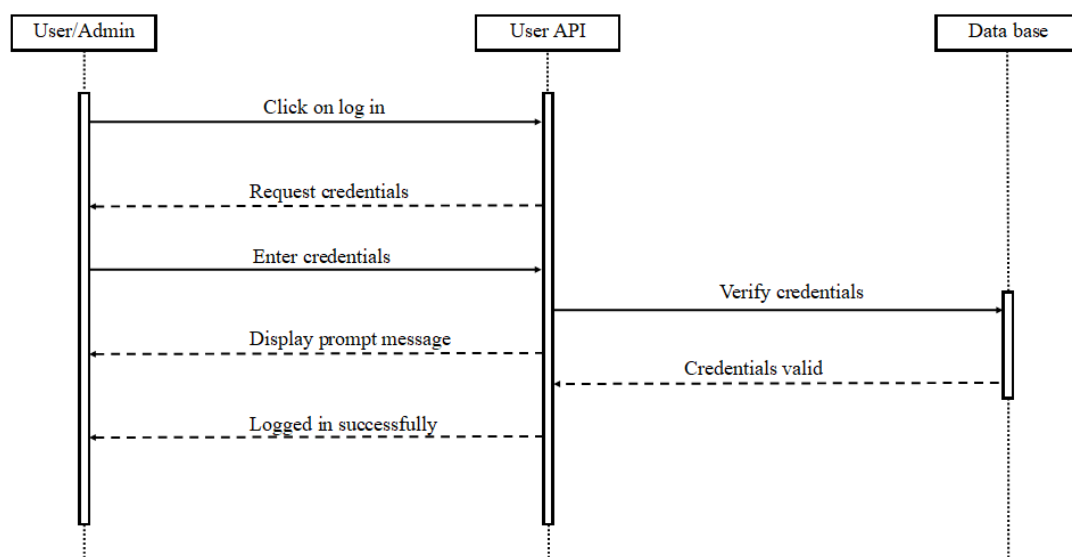


Figure 10: Sequence diagram for user/admin log in into the system



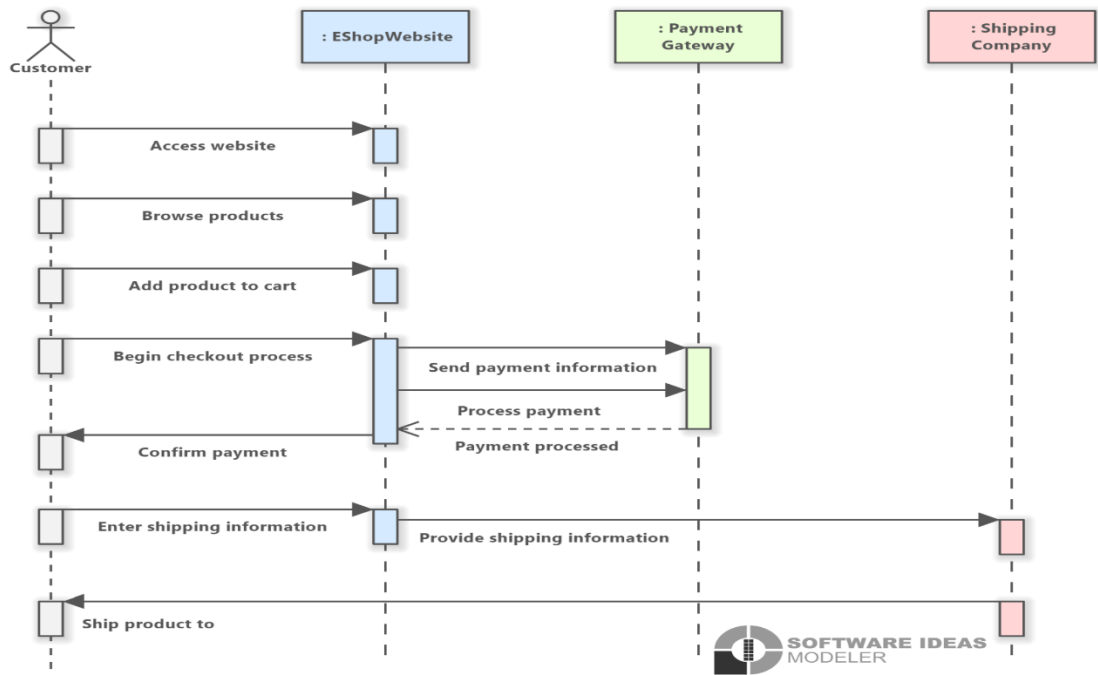


Figure 11: sequence diagram for user process of making an order

### 3.5.3 STATE CHART DIAGRAM

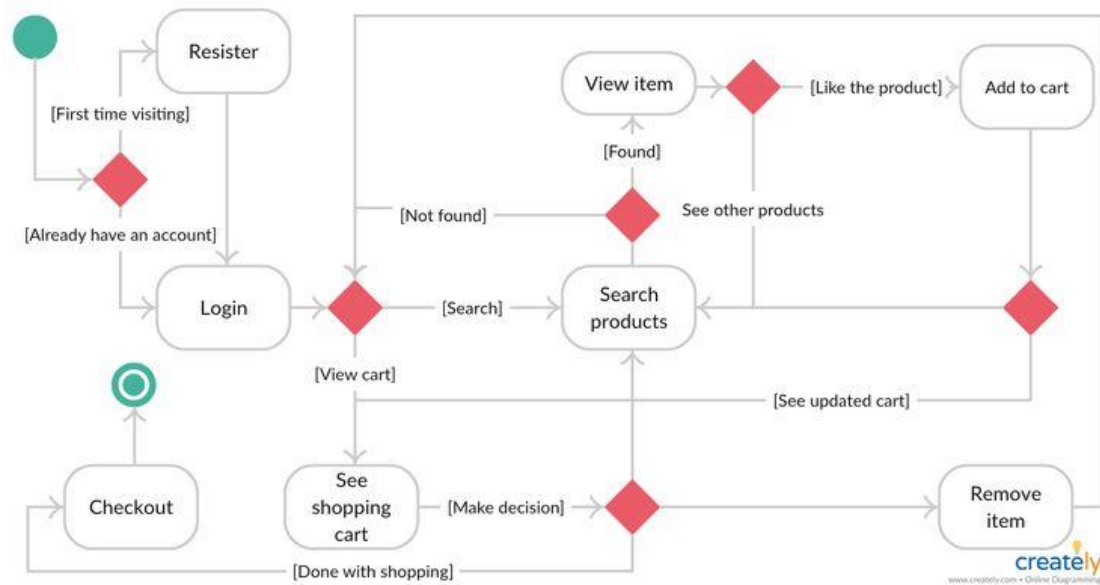


Figure 12: State Chart Diagram of the agribusiness system

### 3.5.4 ACTIVITY DIAGRAM

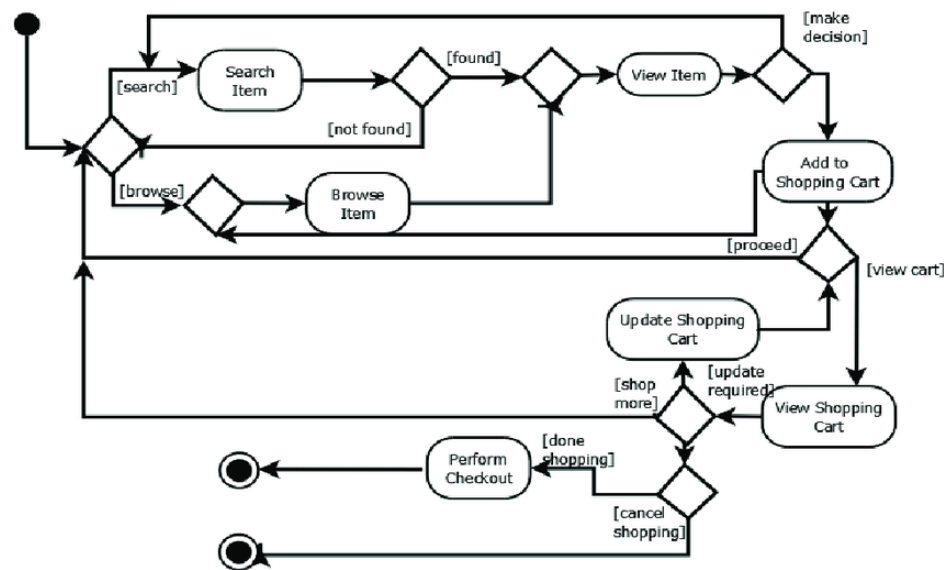


Figure 13: Activity Diagram of the ecommerce system

### 3.5.4 ENTITY RELATIONSHIP DIAGRAMS

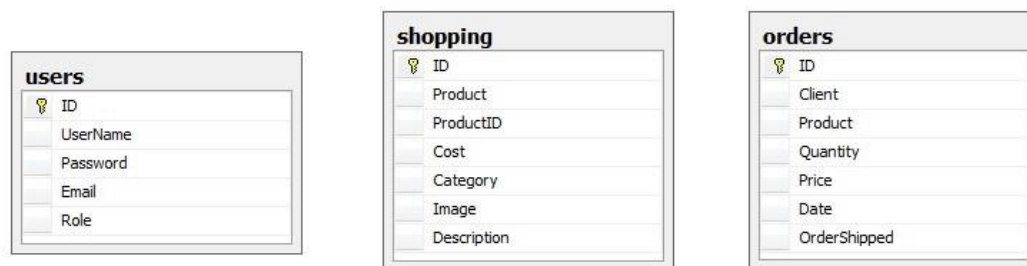


Figure 14: Entity relationship diagram to show primary keys

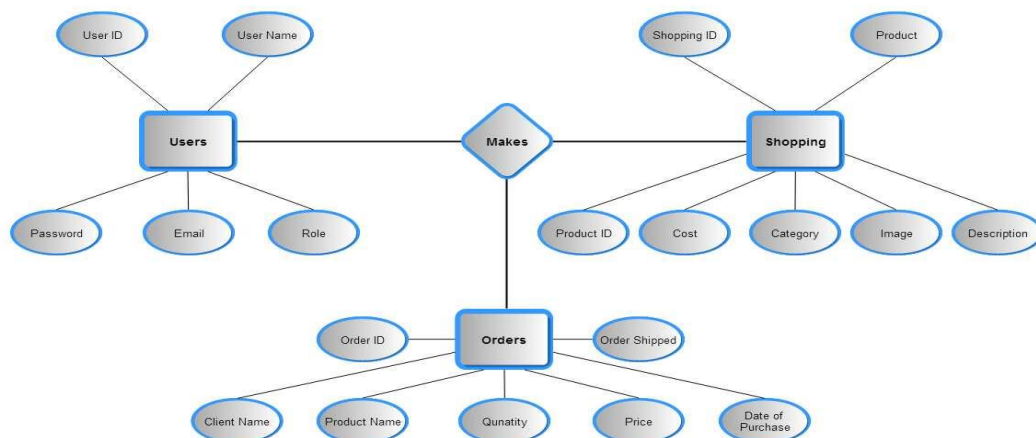


Figure 15: Entity relationship diagram

## **3.6 SYSTEM INTEGRITY CONTROLS**

### **3.6.1 INTRODUCTION**

System integrity controls are a critical aspect of any system design and development. These controls ensure that the system operates effectively, efficiently, and securely. In this document, I will outline the system integrity controls that will be implemented in this online agribusiness system design.

### **3.6.2 ACCES CONTROLS**

Access control is essential in ensuring the security of the online agribusiness system. The system should have access controls in place to limit user access to only the information and functionalities they need to carry out their job functions. The access controls should be role-based, meaning that each user is granted access based on their role in the organization. In this case the customers and the system administrator are assigned different roles and privileges. Additionally, the system should implement a strong password policy to prevent unauthorized access.

### **3.6.3 AUDIT TRAIL**

An audit trail is a record of all system activities that are related to security, such as user login attempts, password changes, and access requests. The system should have an audit trail that can track any suspicious activity or unauthorized access attempts. This audit trail can help identify any security breaches, and it can also be used to investigate any security incidents that occur in the system.

### **3.6.4 DATA INTEGRITY**

Data integrity refers to the accuracy and consistency of data. The online agribusiness system should have data integrity controls in place to ensure that the data entered into the system is accurate and consistent. These controls can include data validation, which checks that data meets specific criteria, and data normalization, which eliminates redundant data

### **3.6.5 BACKUP AND RECOVERY**

The online agribusiness system should have a backup and recovery plan in place to ensure that data is not lost in the event of a system failure or disaster. The backup plan should include regular backups of the system, including data and configuration settings. Additionally, the recovery plan should include procedures for restoring the system from the backups in the event of system or attack failure.

### **3.6.6 SECURITY MONITORING**

The online agribusiness system should have security monitoring controls in place to detect any security breaches or suspicious activity. These controls can include intrusion detection systems, which monitor the system for unauthorized access attempts and log analysis, which looks for anomalies in the system logs that may indicate any security breach.

# DOCUMENT IV

## TEST PLAN

### 4.1 INTRODUCTION

The purpose of this test plan is to ensure the functionality, reliability, and efficiency of the online agribusiness system. The primary goal of this plan is to provide a comprehensive testing process that will identify and eliminate any defects or issues within the system before it is released to the public.

#### 4.1.0 GOALS AND OBJECTIVES

1. To verify that the online agribusiness system is functioning correctly, meeting the requirements of the stakeholders, and is user-friendly.
2. To ensure that all data entered into the system is accurate and stored securely.
3. To validate that the system is performing efficiently and responsively, without any delays or errors in processing.
4. To test the system's compatibility with different browsers, operating systems, and devices.
5. To check the system's ability to handle different volumes of data.
6. To evaluate the system's ability to recover from failures, errors, and disruptions.

#### 4.1.1 SCOPE

This test plan will cover all aspects of the online agribusiness system, including the frontend, backend, and database. It will include functional, performance, compatibility, and security testing. The testing process will be carried out in a controlled environment using both manual and automated techniques. The scope of this plan does not include any testing related to hardware or network infrastructure.

## **4.2 TEST PLAN**

### **4.2.1 TESTING STRATEGY**

The testing strategy will involve a combination of manual and automated testing techniques. The testing will be conducted in a controlled environment to ensure consistent results. The testing process will be divided into different phases, including functional testing, performance testing, compatibility testing, and security testing.

### **4.2.2 INTERFACES TO BE TESTED**

The interfaces to be tested include the following:

1. User interface – to test the usability and user-friendliness of the system.
2. Database interface – to verify data integrity and accuracy.

### **4.2.3 BUG TRIAGE TESTING**

During testing, any bugs or issues found will be documented and triaged according to their severity and priority. The triage process will involve assigning a severity level to each bug based on its impact on the system and the severity of the issue. The priority of the bug will be determined based on its impact on the system and its urgency for resolution. The bug triage process will involve the following steps:

1. Identify the bug and document it.
2. Assign a severity level to the bug.
3. Assign a priority level to the bug.
4. Assign the bug to the appropriate team member for resolution.
5. Re-test the system to verify the bug has been resolved.

## 4.3 TEST PROCEDURE

### 4.3.1 CRITERIA

1. **Functional requirements:** This test criteria are based on the expected behavior of the system. For instance, a requirement specifies that the system should allow users to search for products by names and price, the test criteria verifies that the search function works correctly and returns accurate results.
2. **Performance requirements:** Performance requirements define the expected response time, throughput, and resource utilization of the system. The test criteria verify that the system meets these requirements under expected and peak load conditions.
3. **Security requirements:** Security requirements specify the measures that the system should take to protect user data and prevent unauthorized access. The test criteria verifies that the system enforces security measures to prevent vulnerabilities to common security threats.
4. **Usability requirements:** Usability requirements define the ease of use of the system, including navigation, interface design, and user assistance. The test criteria verify that the system is intuitive, user-friendly, and meets the needs of the target users.
5. **Compatibility requirements:** Compatibility requirements define the system's ability to work with other software, hardware, and network environments. The test criteria verify that the system works correctly with the specified hardware and software.
6. **Regulatory requirements:** Regulatory requirements define the legal and industry standards that the system must comply with. The test criteria verify that the system meets these requirements and is certified to operate in the intended jurisdiction.

### 4.3.2 UNIT TESTING

The purpose of unit testing is to validate that each unit/component of the system is working as expected and to identify and fix any defects early in the development cycle.

**Test Cases:** A set of test cases is developed to test each unit/component. Test cases is designed to test the functionality of each unit/component and should cover all possible use cases.

#### Examples

##### Test Case 1: User Registration

##### Test Steps:

1. Open the main page and click on register.
2. Enter valid user information in all the required fields.
3. Click on the "Submit" button.
4. Verify that the user has provided all the data and validated by the admin
5. Verify that the user account is created successfully by checking the database.
6. Give the user notification message that the account was created.

##### Expected Results:

- The registration page should load successfully.
- All required fields should accept valid user information.
- The system checks if the email entered is already registered.
- The user is able to submit the form successfully.
- The user is redirected to the login page.
- The user's account is created successfully and stored in the database.
- The user's account should be activated by the admin.

##### Test Case 2: User Login

##### Test Steps:

- Navigate to the homepage page and click on the "login" button.
- Enter the correct email and password.
- Click on the "Login" button.



- Verify that the user is redirected to the homepage.
- Verify that the user's login details are stored in the database.

#### Expected Results:

- The login page should load successfully.
- The user should enter the correct login credentials.
- The user is able to log in successfully.
- The user is redirected to the homepage.
- The user's login details are stored in the database.

#### Test Case 3: add product to cart

#### Test Steps:

- Select any preferred product after logging in.
- Enter the specification like quantity and the unit size.
- Click on the "add to cart" button.
- Verify that the booking results are displayed correctly.
- Click on “my bookings” tab on the navigation bar to view to booked vehicles.
- Verify that the booked vehicles and status are displayed correctly.

#### Expected Results:

- The system should update the cart once the users add another product.
- The system should prompt the user incase the product is out of stock.
- The user should be able to view their orders once they finish to checkout
- The user is able to view the available products in the catalogue.

**Test Data:** Test data is created to test the unit/component. The test data includes both valid and invalid data to test the robustness of the unit/component.

### **Example**

Valid data:

Email: [mutukutobias98@gmail.com](mailto:mutukutobias98@gmail.com)

Password: Tobias@2018\*

Invalid data:

Email: mutukutobias.com

Password: 12345

**Test Environment:** The environment is similar to the production environment to ensure that the test results are accurate. This is achieved by use of local server (XAMPP) and web browser (Google Chrome)

**Test Execution:** The unit tests are executed using the test cases and test data in the test environment.

**Defect Resolution:** Any defects encountered during the unit testing is resolved. The defects are logged in a defect tracking tool and prioritized based on severity.

### **4.3.3 INTERGRATION TESTING**

Integration testing focuses on verifying that individual software modules work together correctly as a larger integrated system. The main objective of integration testing is to uncover defects in the interaction between different software modules and ensure that the system functions as expected.

1. **Integration Test Objective:** This section provides an overview of the integration testing objectives, including the software components to be tested, the type of integration testing to be conducted, and the expected results.
2. **Integration Test Approach:** This section describes the approach that is taken to carry out integration testing. The approach includes a detailed description of the sequence in which the software components is integrated, and the

criteria for determining when to move from one integration phase to the next.

3. **Integration Test Risks:** This section identifies potential risks associated with integration testing, such as conflicts between different software components or unforeseen compatibility issues and steps to mitigate the issues.

#### **4.3.4 SYSTEM TESTING**

**System Test:** The System Test verifies the overall functionality, usability, reliability, and performance of the Online Library System. The following tests is performed during the System Test:

1. **Functionality Test:** This test verifies that all the functions and features of the system work as intended. The test covers all the use cases and scenarios defined in the Use Case document.
2. **Usability Test:** This test evaluates the ease of use and user-friendliness of the system. The test involves a group of representative end-users who perform typical tasks using the system. The goal is to identify any issues related to the user interface, navigation, or any other usability concerns.
3. **Reliability Test:** This test evaluates the system's ability to perform consistently and reliably over time. The test involves running the system continuously for an extended period to identify any potential stability or reliability issues.
4. **Performance Test:** This test evaluates the system's ability to handle the expected workload and traffic. The test involves simulating the expected user traffic and monitoring the system's response time, resource utilization, and throughput.

**Security Test:** This test evaluates the system's ability to maintain the confidentiality, integrity, and availability of the data. The test involves attempting to bypass security measures, perform unauthorized actions, and exploit vulnerabilities in the system.

### **4.4 TESTING RESOURCE PLANNING**

#### **4.4.1 SYSTEM RESOURCES REQUIRED**

The following system resources are required for testing the online agribusiness: -

1. **Testing environment** – a dedicated testing environment with the necessary hardware and software required for testing.

2. Test management software – a tool to manage test cases, track bugs, and generate test reports.
3. Test automation tools – tools required for automated testing of the car rental system.
4. Load testing tools – tools required for testing the system's performance under different load conditions.
5. Security testing tools – tools required for testing the system's security.

#### **4.4.2 HUMAN RESOURCES REQUIRED**

The following human resources are required for testing the online agribusiness

1. Test Manager – responsible for overall test planning and execution.
2. Test Lead – responsible for coordinating testing activities, assigning tasks, and tracking progress.
3. Test Engineers – responsible for carrying out testing activities, including manual and automated testing.
4. Performance Engineer – responsible for load testing and performance testing of the system.
5. Security Engineer – responsible for security testing of the system.
6. Test Analyst – responsible for analyzing and documenting test results.
7. Test Coordinator – responsible for coordinating testing activities with external stakeholders.

### **4.5 TEST ENVIRONMENT**

#### **4.5.1 HARDWARE REQUIREMENTS**

The following hardware is required for the test environment:

1. Dedicated server – a dedicated server is required to host the agribusiness system and database.

2. Workstations – workstations are required for the testing team to perform manual testing.
3. Mobile devices – mobile devices are required to test the system's compatibility with different devices and operating system.

#### **4.5.2 SOFTWARE REQUIREMENTS**

The following software is required for the test environment:

1. Operating system – a compatible operating system is required for the server and workstations.
2. Web server – a web server is required to host the online agribusiness system.
3. Database server – a database server is required to store and manage data for the online agribusiness system.
4. Test management software – a tool is required to manage test cases, track bugs, and generate test reports.
5. Test automation tools – tools are required for automated testing of the online agribusiness system.
6. Load testing tools – tools are required for testing the system's performance under different load conditions.
7. Security testing tools – tools are required for testing the system's security.

#### **4.5.3 NETWORK REQUIREMENTS**

The following network requirements are necessary for the test environment:

1. Internet access – internet access is required for testing the system's integration with external services such as payment gateways and order tracking services.
2. Firewalls – firewalls are required to protect the test environment from external threats.

## 4.6 TEST ESCHEDULE AND ESTIMATION

### 4.6.1 TESTING PHASES

The testing phases included in this Test Plan are Unit Testing, Integration Testing, and System Testing. Each testing phase have a specific timeline and estimated completion date.

### 4.6.2 TESTING SCHEDULE

The following table outlines the estimated start and end dates for each testing phase:

**Table 12: Test schedule**

Test	Testing Phase	Estimated Start Date	Estimated End Date
1	Unit Testing	15/03/2023	20/03/2023
2	Integration Testing	21/03/2023	05/04/2023
3	System Testing	08/04/2023	20/04/2023

# DOCUMENT IV

## USER MANUAL

### 5.1 SYSTEM OVERVIEW

The “*UptownFresh*” online agribusiness system is a web-based application that enables customers to purchase fresh farm products at cheaper prices. This system aims to provide and simplify the buying and selling process by providing an easy-to-use platform that allows users to search their desired farm products, place orders and make payments online. The system is designed to be user-friendly and efficient, providing a seamless experience for customers customer and stakeholders.

The system includes features such as order management, product management and customer management. Customers are able to view details about the available farm products, select the desired products and choose the method of payment online. The system provides the admin with an interface to manage the customer orders, products and customers. The online agribusiness system project aims to provide a convenient and efficient solution for customers to buy products online and for the “*UptownFresh*” company to manage their operations.

### 5.2 SYSTEM REQUIREMENTS

#### 5.2.1 HARDWARE REQUIREMENTS

1. Computer: The system requires a computer with a minimum of 2GB of RAM and a dual-core processor. The system can run on both Windows and Mac OS.
2. Internet Connection: The system requires a stable internet connection to access the online library resources. A minimum speed of 2 Mbps is recommended.
3. Mobile Device: The system can also be accessed through a mobile device, such as a smartphone or tablet. The mobile device should have a minimum screen size of 5 inches and an internet connection.

## 5.2.2 SOFTWARE REQUIREMENTS

To access the online agribusiness system, the following software must be installed on your computer:

1. Web browser: The Car rental System is accessed through a web browser. The system supports the following browsers:
  - Google Chrome (latest version)
  - Mozilla Firefox (latest version)
  - Microsoft Edge (latest version)
  - Safari (latest version)
2. Operating System: The Car rental System is compatible with the following operating systems:
  - Windows 7 or later
  - MacOS
  - Linux

## 5.3 SYSTEM FEATURES

### 5.3.1 HOME PAGE AND NAVIGATION

The “*UptownFresh*” online agribusiness system provides an intuitive and user-friendly homepage and navigation that allows users to easily search and access different farm products. The homepage is the first page that users see when he/she visits the system, and it serves as a hub for all major functions and features.

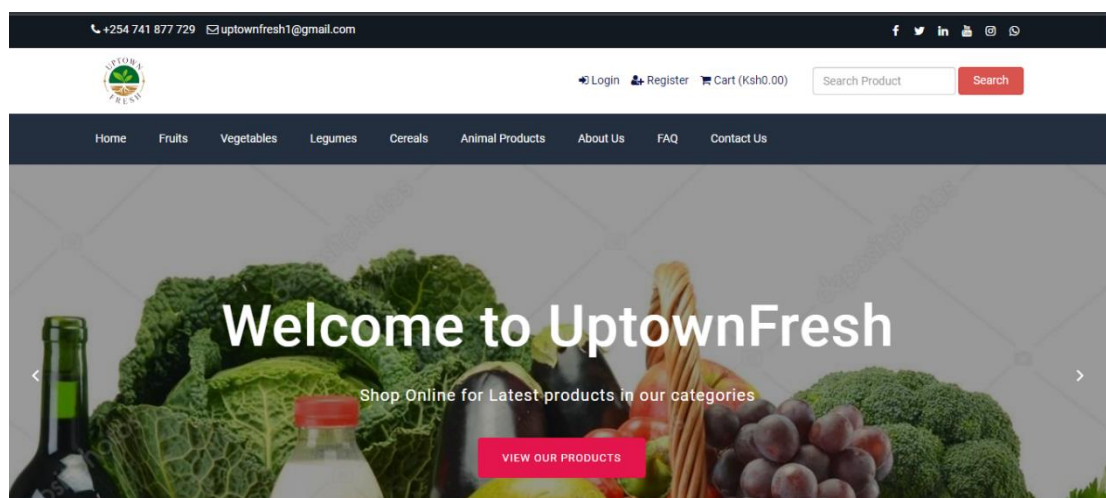


Figure 16: User interface interactive to the user



## 5.3.2 NAVIGATION BAR

The navigation bar has the navigation menu which provides quick links to various categories of the products and other various parts of the system including the home, about us, contact us etc.

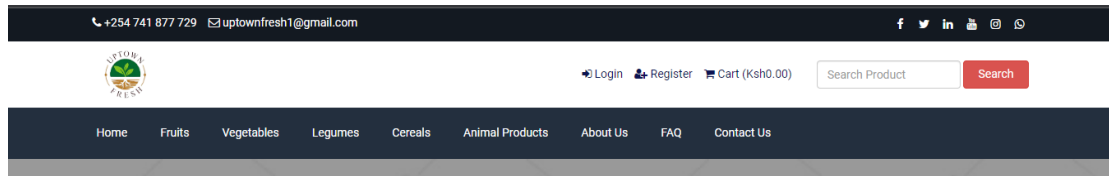


Figure 17: User navigation bar and other important action buttons

## 5.3.3 PRODUCTS CATALOGUE

This is the panel that has all the available farm products uploaded by the admin from the backend. The system visitors and active registered customers are able to view the products catalogue containing all the information about the products.

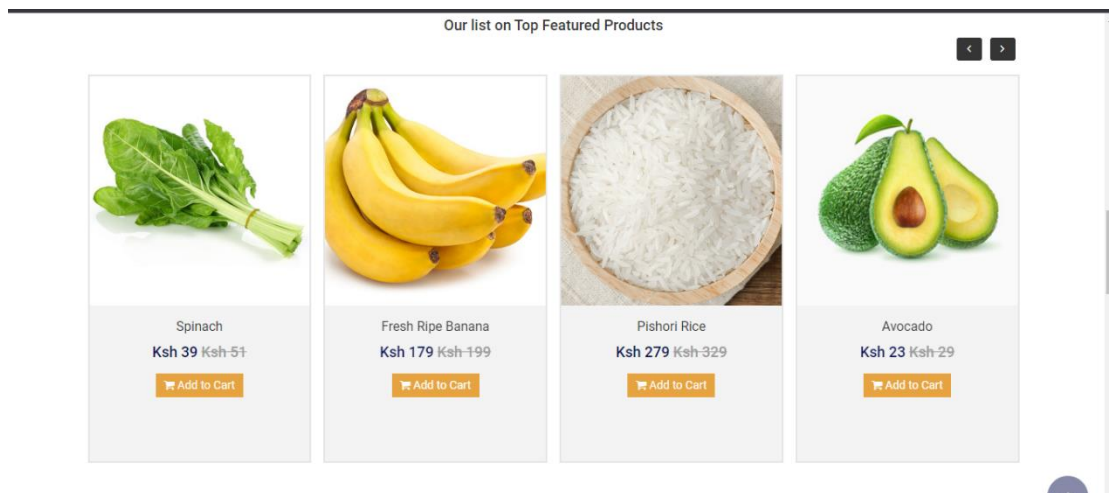
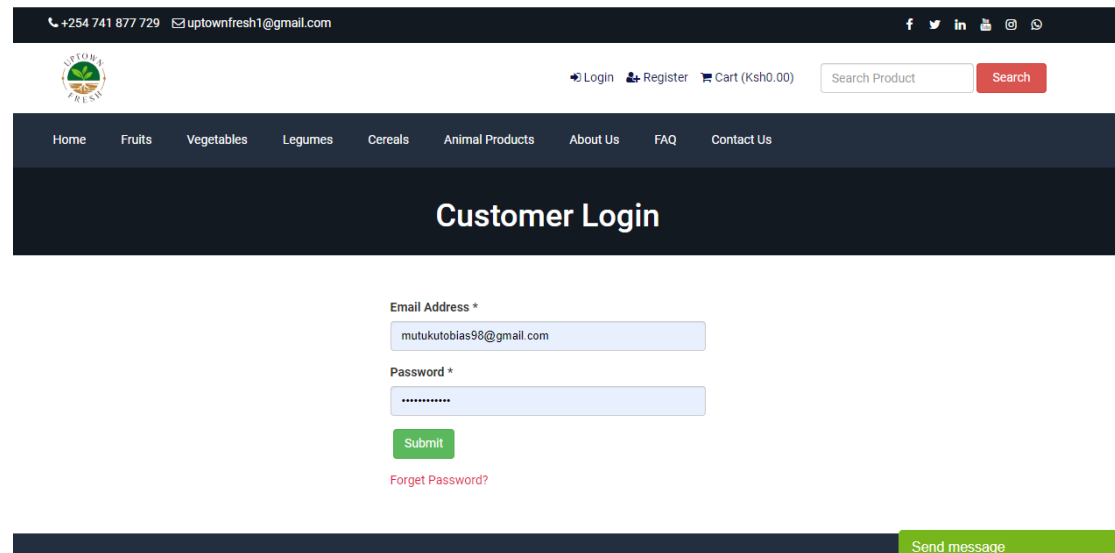


Figure 18: product catalogue

### 5.3.4 LOG IN FORM

This is the form used to enter log in credentials to be given access to the system. Only verified customers with registered email and correct password are allowed to access the system

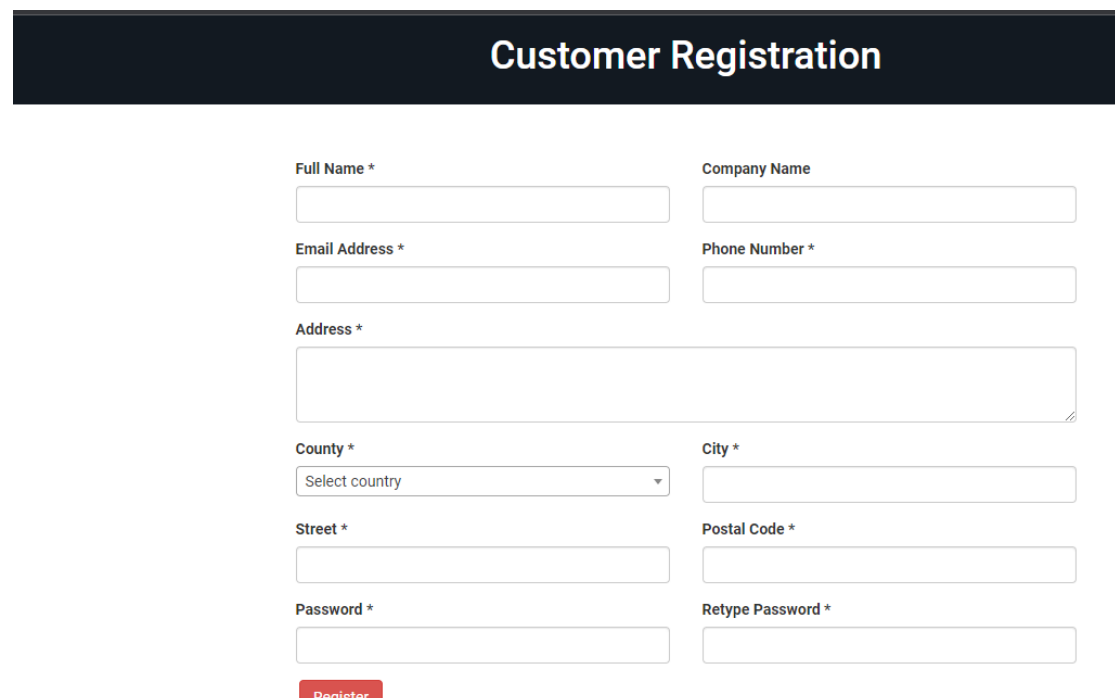


The screenshot shows the top navigation bar with contact information (+254 741 877 729, uptownfresh1@gmail.com) and social media icons. Below is a header with the 'UPTOWN FRESH' logo, a 'Login' button, a 'Register' button, a 'Cart (Ksh0.00)' button, and a search bar. A main menu lists categories: Home, Fruits, Vegetables, Legumes, Cereals, Animal Products, About Us, FAQ, and Contact Us. The central section is titled 'Customer Login' and contains a form with fields for 'Email Address \*' (filled with 'mutukutobias98@gmail.com') and 'Password \*' (filled with '\*\*\*\*\*'). A green 'Submit' button is below the password field, and a red 'Forgot Password?' link is at the bottom. A green 'Send message' button is in the bottom right corner.

Figure 19: Customer Log In form interface

### 5.3.5 REGISTRATION FORM

This is the registration form where customers are prompted to provide their details to be activated by the admin so that they can access the services and purchase products in the site.



The screenshot shows the 'Customer Registration' form. It has a title 'Customer Registration' at the top. The form fields are arranged in two columns. The left column contains: 'Full Name \*' (text input), 'Email Address \*' (text input), 'Address \*' (text input), 'County \*' (dropdown menu with 'Select country' selected), 'Street \*' (text input), 'Password \*' (text input), and a red 'Register' button at the bottom. The right column contains: 'Company Name' (text input), 'Phone Number \*' (text input), 'City \*' (text input), 'Postal Code \*' (text input), and 'Retype Password \*' (text input).

Figure 20: Customer registration form interface

### 5.3.6 CUSTOMER DASHBOARD

The customers' dashboard is where the customer land after successfully logs into the system. The dashboard has different action buttons like order to view a list of the orders the customer made, update profile, update shipping information and log out button.

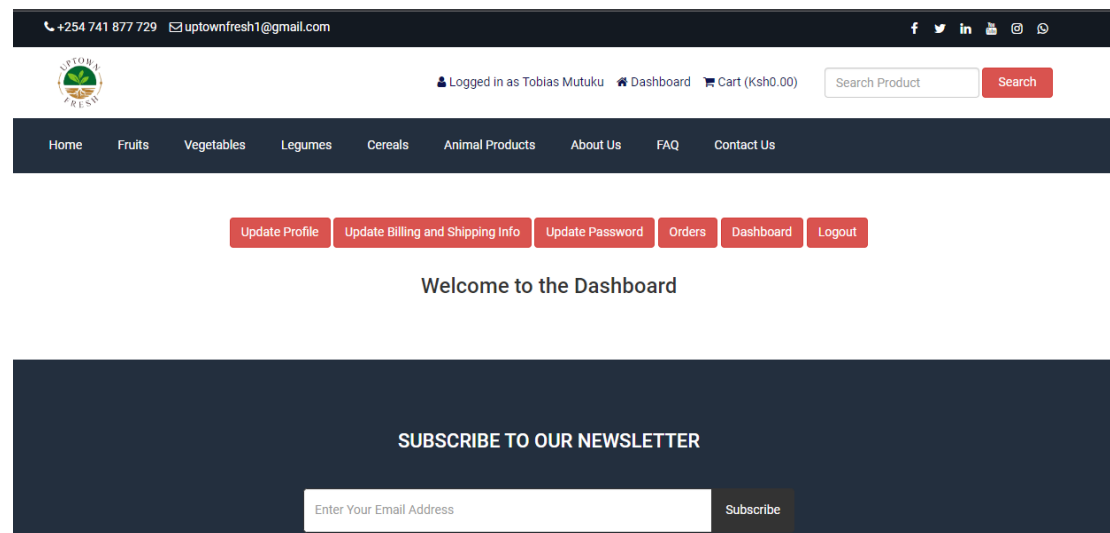


Figure 21: Customer Dashboard with different action button

### 5.3.7 ADD TO CART

After a customer has selected the desired product, he/she proceeds to adding the product in the cart. The customer can set the size and the color where necessary and click on add to cart. The product will be sent to the online cart.

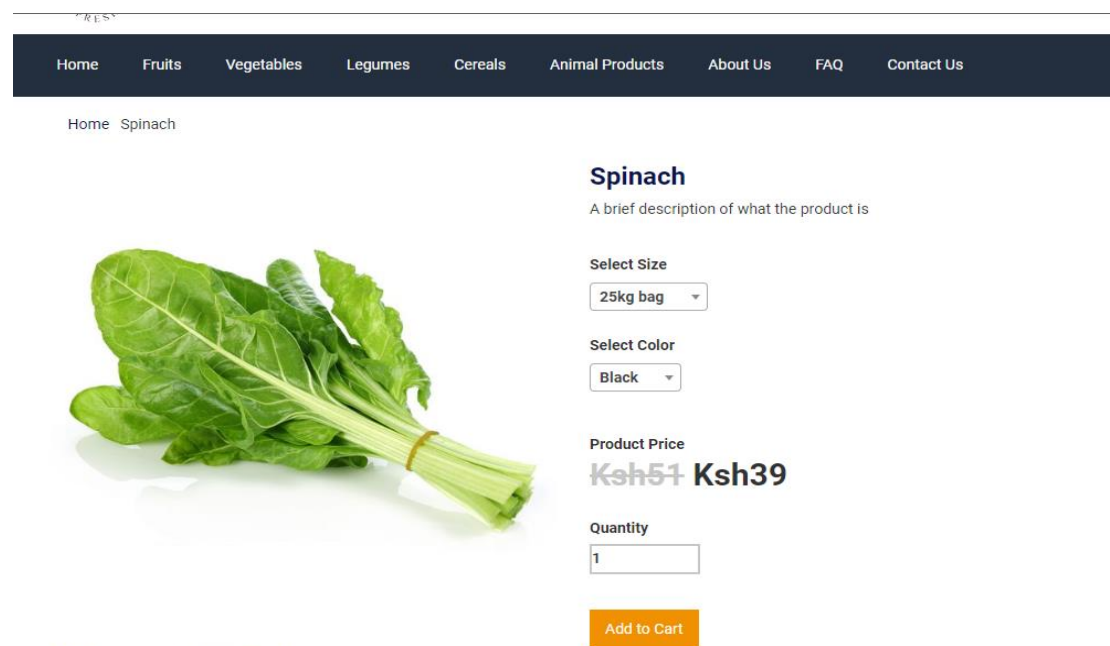



Figure 22: Customer Add to cart page

### 5.3.8 VIEW CART ITEMS

The customer can view the items on the cart and maybe delete some if not required. He /she can see the total amount of money of the products.

+254 741 877 729uptownfresh1@gmail.com

f t in @

Logged in as Tobias MutukuDashboardCart (Ksh39)

Search ProductSearch

HomeFruitsVegetablesLegumesCerealsAnimal ProductsAbout UsFAQContact Us

Cart





#	Photo	Product Name	Size	Color	Price	Quantity	Total	Action
1		Spinach	25kg bag	Black	Ksh39	<input type="text" value="1"/>	Ksh39	
Total							Ksh39	

Figure 23: Customer virtual cart

### 5.3.9 CHECKOUT

Once the customer is satisfied with his/her shopping they can proceed to check out by clicking the checkout button. After clicking the button, they will be able to see the final charges of both the products and the shipping fee.

Cart

#	Photo	Product Name	Size	Color	Price	Quantity	Total	Action
1		Spinach	25kg bag	Black	Ksh39	<input type="text" value="1"/>	Ksh39	
Total							Ksh39	

Update CartContinue ShoppingProceed to Checkout

Figure 24: Customer Check Out option

### 5.3.1.0 SHIPPING DETAILS AND PAYMENT OPTIONS

The customer is able to see the shipping details and can change them on the customer dashboard in case of relocation of business. After checking everything the customer can now pay for the products either through M-Pesa or through Bank Transfer.

---

Billing Address		Shipping Address	
Full Name	Tobias Mutuku	Full Name	Tobias Mutuku
Company Name	UptownFresh	Company Name	UptownFresh
Phone Number	0741877729	Phone Number	0741877729
County	Kilifi	County	Kilifi
Address	50100	Address	50100
City	Nakuru	City	Nakuru
Street	Ngala phase 2	Street	Ngala phase 2
Postal Code	10100	Postal Code	10100

Back to Cart

Figure 25: Customer Billing and Shipping information interface

### 5.3.1.1 PAY

After choosing the payment method the customer can now click on pay button and the transaction will be successful. His or her order will be placed on the admins side.

---

Back to Cart

#### Payment Section

Select Payment Method \*

Mpesa

Pay Now

SUBSCRIBE TO OUR NEWSLETTER

Copyright © 2023 - UptownFresh - Developed By Tobias Mutuku

Figure 26: Customer payment option interface

### 5.3.1.2 ADMIN DASHBOARD

In this page the administrator can view all the entities and reports of the entire system.

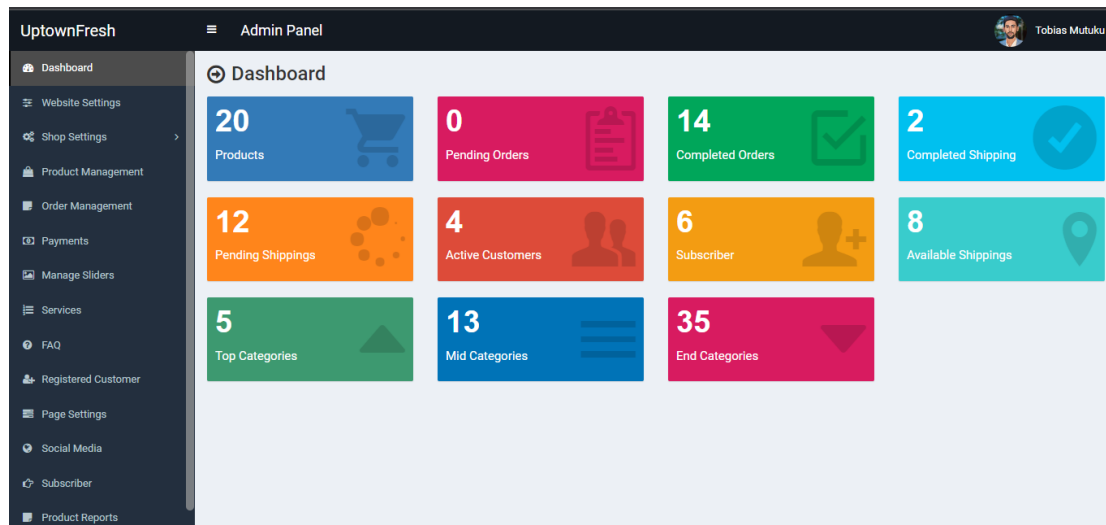


Figure 27: System Administrator User Interface (Dashboard)

### 5.3.1.3 ORDER MANAGEMENT

When customers make a purchase on the front end, the admin is able to view the orders and process them by confirming the customer payment and the orders so that the admin can authorize release and shipping of the customers farm products.

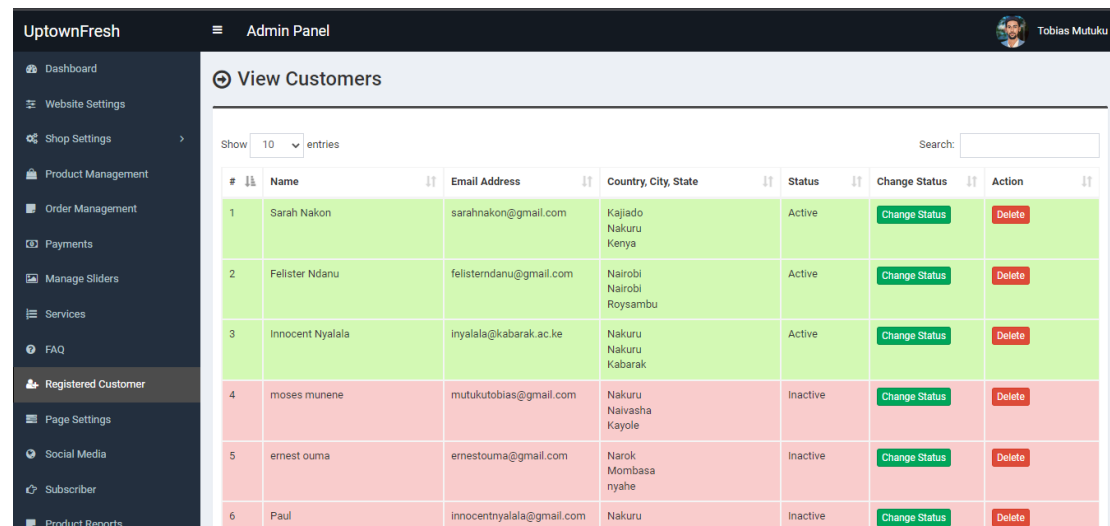
The screenshot shows the UptownFresh Admin Order Management page. The table displays the following data:

#	Customer	Product Details	Payment Information	Paid Amount	Payment Status	Shipping Status	Action
1	<b>ID:</b> 15 <b>Name:</b> Innocent Nyalala <b>Email:</b> inyalala@ikabarak.ac.ke <a href="#">Send Message</a>	<b>Product:</b> Green sagga (Size: 5kg bag, Color: Green) (Quantity: 1, Unit Price: 87) <b>Product:</b> Beef Meat (Size: 1 box, Color: Red) (Quantity: 1, Unit Price: 700) <b>Product:</b> Yellow oranges (Size: crate, Color: Yellow) (Quantity: 1, Unit Price: 120)	<b>Payment Method:</b> M-Pesa <b>Payment Id:</b> 1681723865 <b>Date:</b> 2023-04-17 02:31:05	Ksh5907	Completed	Completed	<a href="#">Delete</a>
2	<b>ID:</b> 1 <b>Name:</b> Tobias Mutuku <b>Email:</b> mutukutobias98@gmail.com <a href="#">Send Message</a>	<b>Product:</b> Avocado (Size: 50kg bag, Color: Green) (Quantity: 1, Unit Price: 23) <b>Product:</b> Spinach (Size: 25kg bag, Color: Black) (Quantity: 1, Unit Price: 39)	<b>Payment Method:</b> M-Pesa <b>Payment Id:</b> 1681116389 <b>Date:</b> 2023-04-10 01:46:29	Ksh62	Completed	Pending <a href="#">Mark Complete</a>	<a href="#">Delete</a>

Figure 28: Admins Order management page

### 5.3.1.4 CUSTOMER MANAGEMENT

After new customers register their accounts, the admin has a page to manage their accounts. He/she can delete customer's account, activate or deactivate the status of any customer's account. If customers account is deactivated, he/she cannot be able to log into the system.



#	Name	Email Address	Country, City, State	Status	Change Status	Action
1	Sarah Nakon	sarahnakon@gmail.com	Kajiado Nakuru Kenya	Active	Change Status	Delete
2	Felister Ndanu	felisterndanu@gmail.com	Nairobi Nairobi Roysambu	Active	Change Status	Delete
3	Innocent Nyalala	inyalala@ikabarak.ac.ke	Nakuru Nakuru Kabarok	Active	Change Status	Delete
4	moses munene	mutukutobias@gmail.com	Nakuru Naivasha Kayole	Inactive	Change Status	Delete
5	ernest ouma	ernestouma@gmail.com	Narok Mombasa nyahe	Inactive	Change Status	Delete
6	Paul	innocentnyalala@gmail.com	Nakuru Kajiado	Inactive	Change Status	Delete

Figure 29: Admins customer management page

## 5.4 SYSTEM SUPPORT

### 5.4.1 SYSTEM INSTALLATION

- Get the zipped project source code
- Extract the files using any file extractor e.g winrar
- Download a local hot server and install e.g XAMPP so as to be able to run the system live on a browser
- Store the project files on the folder htdocs of the system folder of XAMPP.
- Turn on the local server and run the project folder on your browser.
- The database for the project in in the database folder you can import it to your myphp admin and work with it.
- Log in credentials for user1 and the admin are provided on the read me file on the project files.

## **5.4.2 TROUBLESHOOTING COMMON ISSUES**

### **Problem: Unable to log in to the system**

#### **Solution:**

- Confirm that the credentials entered (email and password) are correct. Passwords are case sensitive, so make sure you are using the correct case.
- Make sure that the system administrator has activated your account
- Make sure that you have an active internet connection.
- Clear your browser cache and cookies, and try logging in again.

### **Problem: Unable to checkout**

#### **Solution:**

- Confirm that you have logged in to the account.
- Make sure that you have entered the shipping and billing information.

### **Problem: Unable to register**

#### **Solution:**

- Ensure that you have filled all the required fields.
- Make sure you use the right password format as stated in the register form
- Do not use an already registered email address when filling in the form.

### **Problem: Unable to log in**

#### **Solution:**

- Ensure that you have registered your account successfully
- Contact the admin to verify your account if still unable to log in.
- Reset password and set a new one in case you forgot the previous one.