Agricultural Crop Exchange System

Software Requirements Specification

Version 1.2

11-18-2023

Amin Elhag
Joiner Software Engineer

Revision History

Date	Description	Author	Comments
11-18-2023	Version 1.0	Amin Elhag	Initial Version
11-24-2023	Version 1.1	Amin Elhag	Complete the first edition

1. Introduction

This document will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate, and how the system will react to external stimuli. This document is intended for the stakeholders and the developers of the system.

1.1 Purpose

This system works to facilitate the process of selling and exchanging agricultural crops and protects the farmer and the consumers from brokers controlling prices.

1.2 Scope

The project is to develop an agricultural crop exchange platform for Farmers and consumers. The system should allow Farmers to register and create profiles with their personal and farm information, such as name, location, contact details, farm size and crop types.
The system should allow Farmer s to post offers for selling or exchanging their crops with other Farmer s or consumers. Each offer should include the crop name, quantity, quality, price, and delivery method. The system should allow Farmer s to browse and search for offers posted by other Farmer s or consumers. The system should also provide filters and sorting options based on criteria such as crop name, location and price. The system should allow Farmer s to initiate and negotiate transactions with other Farmer s or consumers. The system should also facilitate the payment and delivery of the crops using secure and reliable methods. The system should allow Farmer s to rate and review their transactions with other Farmer s or consumers. The system should also display the ratings and reviews of each Farmer or consumers on their profile.

1.3 Definitions, Acronyms, and Abbreviations

Table 1-Definition

Term	*Full Description*
Manager	*The person responsible for managing the support team.*
Support team	*The persons responsible for following up and solving problems that appear in the system.*
Farmer	*The person how has* farm and want to sell, buy or exchanging his crop.
Consumers	*The person how has a business name and want to buy crop.*

1.4 References

The document refers to the following assignments submitted:
(1) Agricultural Crop Exchange System chat with chatGPT.

1.5 Overview

The software product is a web-based platform that allows farmers and consumers to exchange crops directly without intermediaries. The product is called CropX and its purpose is to provide a fair and efficient way to trade agricultural products.
The intended audience of the product are farmers who want to sell their crops and consumers who want to buy fresh and organic crops. The intended use of the product is to enable users to create and browse crop listings, negotiate and agree on prices and quantities, and arrange and confirm delivery and payment. The product scope is limited to facilitating the exchange of crops between users, and does not include any quality assurance, dispute resolution, or legal services.

2. General Description

- *Product perspective: The product is a consumer's platform that allows farmers and consumers to exchange crops directly without intermediaries. The product is a standalone system that does not integrate with any existing systems. However, the product relies on some external interfaces, such as a third-party delivery service provider and a third-party payment service provider, to facilitate the crop exchange process. The product also uses some standard protocols and formats, such as HTTPS, JSON, and QR codes, to ensure secure and reliable communication and data exchange.*
- *Product functions: The product provides the following main functions for the users:*
    - Create and browse crop listings: The product allows users to create crop listings that include the name, description, quantity, location, and price of the crops they want to sell. The product also allows users to browse crop listings that match their preferences, such as location, category, or rating. The product can also suggest some relevant or popular crop listings for the users to choose from.
    - Negotiate and agree on crop exchange terms: The product allows users to communicate with each other through a chat system and negotiate the terms of the crop exchange, such as price, quantity, delivery method, and payment method. The product also allows users to accept or reject the offers made by other users. The product can also generate and send a crop exchange contract that summarizes the agreed terms and conditions for both parties to sign.
    - *Arrange and confirm crop exchange delivery and payment: The product allows users to arrange and confirm the delivery of the crops through a third-party delivery service provider. The product also allows users to make and receive payments through a third-party payment service provider. The product can also track and update the status of the crop exchange, such as shipped, delivered, paid, or completed.*
- *User characteristics: The product has two types of users: farmers and consumers. The farmers are the users who want to sell their crops and the consumers are the users who want to buy fresh and organic crops. The users are expected to have some basic knowledge and skills, such as:*
    - *How to use a mobile device and access the internet.*
    - *How to create and manage an account and a profile on the product platform.*

  - *How to create and edit a crop listing and upload an image of the crop.*
  - *How to search and filter crop listings and view the details and ratings of the crops.*
  - *How to initiate and respond to a chat message and negotiate the terms of the crop exchange.*
  - *How to sign a crop exchange contract and confirm the agreement.*
  - *How to arrange and confirm the delivery and payment of the crops through the third-party service providers.*
  - *How to rate and review the crop exchange experience and provide feedback to the product platform.*

- *Constraints: The product is subject to some constraints that limit or restrict its design, development, or operation. Some of the constraints are:*
  - *Legal and regulatory constraints: The product must comply with the laws and regulations of the country where it is used, such as the agricultural, trade, tax, and consumer protection laws. The product must also respect the intellectual property rights and privacy rights of the users and the third-party service providers.*
  - *Hardware and software constraints: The product must run on a web server that can handle the requests and responses between the users and the database. The product must also use HTTPS protocol for secure communication and JSON format for data exchange. The product must also use QR codes for verification and confirmation of the crop exchange.*
  - *Quality and performance constraints: The product must meet some quality and performance standards, such as usability, reliability, availability, security, scalability, and efficiency. The product must also meet some quality and performance metrics, such as response time, error rate, availability rate, and user satisfaction rate.*

2.1 Product Perspective

At the end of each agricultural season, most agricultural crops are purchased by brokers at low prices and then sold at very high prices, which harms the farmer and the consumers. At the same time, there is no accurate information available about the crops that were sold, such as quality, place of cultivation, and place of purchase. The availability of this the information may benefit parties interested in agriculture in the future.

2.2 Product Functions

- *The system should allow farmers to register and create profiles with their personal and farm information, such as name, location, contact details, farm size, crop types, etc.*
- *The system should allow farmers to post offers for selling or exchanging their crops with other farmers or consumers. Each offer should include the crop name, quantity, quality, price, and delivery method.*
- *The system should allow farmers to browse and search for offers posted by other farmers or consumers.*

- *The system should also provide filters and sorting options based on criteria such as crop name, location, price, etc.*
- *The system should allow farmers to initiate and negotiate transactions with other farmers or consumers.*
- *The system should also facilitate the payment and delivery of the crops using secure and reliable methods.*
- *The system should allow farmers to rate and review their transactions with other farmers or consumers.*
- *The system should also display the ratings and reviews of each farmer or consumers on their profile.*

2.3 User Characteristics

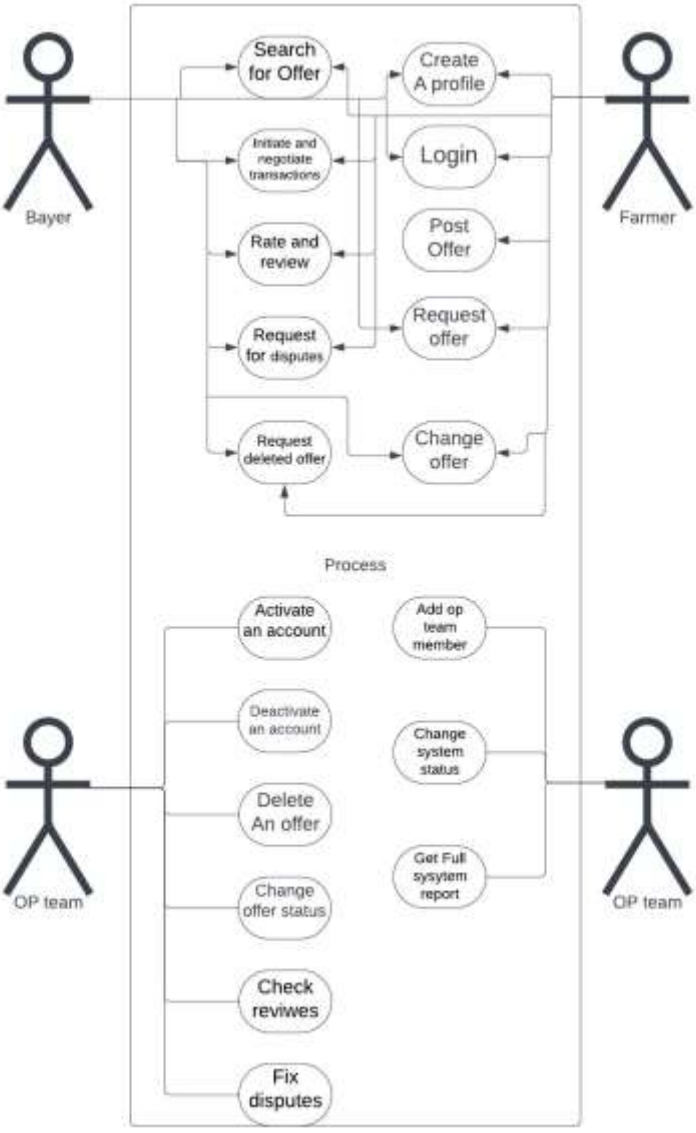

Figure 1-Use-case Diagram

The actors and use cases are clearly shown in above USE-CASE diagram.

The following are the Stakeholders/ Actors who perform the functions/ use cases as stated above:

Table 2-Use cases

S. No	*Actor name*	*Description/role*
01	*Manager*	• *Change system status (Open and close the system).* • *Management support team.*
02	*Support team*	• *Active farmer account.* • *Management farmer account.* • *Change offer status (delete, update, cancel, reopen).*
03	*Farmer*	• *Registration.* • *Update profile.* • *Post offer.* • *Change his offer status.* • *Rating.*
04	*Consumers*	• *Registration.* • *Update profile.* • *Post offer request.* • *Management his offer request.*

2.4 General Constraints

- *Internet constraint: Developer should know the famer have poor and unstable internet connect.*
- *Power usage constraint: System should not use high power.*
- *Age gap constraint: Developer should know famer, consumers and manager maybe old and have problems with dealing with electronic systems.*

2.5 Assumptions and Dependencies

1. *Assumption*
    a. *The system assumes that the users have access to the internet and a web browser that supports the system's features.*
    b. *The system assumes that the users have valid and accurate information about their crops, such as quantity, quality, price, location, etc.*
    c. *The system assumes that the users are honest and trustworthy, and will not engage in fraudulent or malicious activities on the platform.*

d. *The system assumes that the users are familiar with the basic concepts and terminology of agriculture and crop exchange.*
e. *The system assumes that the users will follow the rules and regulations of the platform, and respect the rights and obligations of other users.*

2. *Dependencies*
   a. *The system depends on a reliable and secure database to store and retrieve the data of the users and the crops.*
   b. *The system depends on a robust and scalable web server to handle the requests and responses of the users and the system.*
   c. *The system depends on a third-party payment service to facilitate the transactions between the consumers and the sellers of the crops.*
   d. *The system depends on a third-party logistics service to arrange the delivery and pickup of the crops.*
   e. *The system depends on a third-party weather service to provide the weather information and forecasts for the crop locations.*

3. Specific Requirements

- *Farmer profile*

  - *Description and Priority:*
    The Farmer can create profiles with their personal and farm information, such as name, location, contact details, farm size, crop types and manage their accounts, log in and log out securely.
    This feature is high priority, we do this to make sure the farmer has corrected information and make sure he is a real farmer to prevent fraud.
    This feature can make the registration process slow and complicated and reduce the registration of new farmers.

  - *Stimulus/Response Sequences:*
    - *Created a new profile: Farmer open the app and click on the "Create new account" button on the log in screen. Response: The app open create new account screen with all data required for registration. The crate account will never complete until all required, corrected data filled. App should not allow incorrect data to be pass.*
    - *Login: Farmer open log in screen and filled phone number and otp corrected then click on the "Login" button. Response: The app will show the main screen if login successful or show error message to tell the farmer what is going wrong.*
    - *Log out: Farmer click on the "log out" button in profile details screen. Response: System will show dialog box to confirm the logout.*

  - *Functional Requirement*
    REQ-1: The system should allow the farmer to fill all information.
    REQ-2: The system should validation all farmer inputs.
    REQ-3: The system should not send request if any data is not correct.
    REQ-4: The system should show that filled have not correct.
    REQ-5: The system should not allow farmer to login if account not exit.
    REQ-6: The system should not allow farmer to login if phone number or otp is not correct.
    REQ-7: The system should cash all information has been filled.

REQ-8: The system should delete all cash if farmer has logout.
REQ-9: The system should logout form previous session when farmer relogging.

- *Farmer Offer*
  - *Description and Priority:*
    The farmer can create, edit and completed his own offer, specifying the location, payment and delivery methods, then deciding whether to post it or save it for another time, or request to deleted is offer.
    This feature is high priority.

  - *Stimulus/Response Sequences:*
    - *Created a new offer: Farmer open the app and click on the "Create new offer" button on the main screen. Response: The app open create new offer screen with all data required for offer. The crate offer will never complete until all required, corrected data filled. App should not allow incorrect data to be pass.*
    - *Edited offer: Farmer open the app and click on the offer list screen. Response: The app open offer details screen. Then farmer click on options. Response: The app open dialog menu with options. Then farmer click on "Edit" button. Response: The app open edit offer screen with all data of offer. The farmer edits the data and click of "save edit" button. Response: request send.*
    - *Completed offer: Farmer open the app and click on the offer list screen. Response: The app open offer details screen. Then farmer click on "Completed offer". Response: request send.*
    - *Deleted offer: Farmer open the app and click on the offer list screen. Response: The app open offer details screen. Then farmer click on options. Response: The app open dialog menu with options. Then farmer click on "Deleted" button. Response: request send.*

  - *Functional Requirement*
    REQ-10: The system should allow the farmer to fill all offer data.
    REQ-11: The system should validation all farmer inputs.
    REQ-12: The system should not send create request if any data is not correct.
    REQ-13: The system should show that filled have not correct.
    REQ-14: The system should confirm before create offer.
    REQ-15: The system should send notification when offer create to anyone who has previously bought or been interested in the farms offer.
    REQ-16: The system should allow the farmer to edited his active offer data.
    REQ-17: The system should validation all farmer edit data.
    REQ-18: The system should not send edit request if any data is not correct.
    REQ-19: The system should show that filled have not correct.
    REQ-20: The system should confirm before edit offer.
    REQ-21: The system should send notification when offer create to anyone who has been interested in the offer.
    REQ-22: The system should allow the farmer to completed his active offer.
    REQ-23: The system should confirm before completed active offer.
    REQ-24: The system should cancel all discussion sessions or interest in the offer.
    REQ-25: The system should send notification to other side of offer.
    REQ-26: The system should change is status of offer to complete.
    REQ-27: The system should allow the one who buy the offer to rating the farmer, offer, method and app.
    REQ-28: The system should allow farmer to request to deleted the active offer.

- *Consumer's profile*
  - *Description and Priority:*
    The consumers can create profiles with their personal and business information and manage their accounts, log in and log out securely.
    This feature is high priority, we do this to make sure the consumers have corrected information and make sure he is a real consumer to prevent fraud.
    This feature can make the registration process slow and complicated and reduce the registration of new consumers.

  - *Stimulus/Response Sequences:*
    - *Created a new profile: Consumers open the app and click on the "Create new account" button on the log in screen. Response: The app open create new account screen with all data required for registration. The crate account will never complete until all required, corrected data filled. App should not allow incorrect data to be pass.*
    - *Login: Consumers open log in screen and filled phone number and otp corrected then click on the "Login" button. Response: The app will show the main screen if login successful or show error message to tell the farmer what is going wrong.*
    - *Log out: Consumers click on the "log out" button in profile details screen. Response: System will show dialog box to confirm the logout.*

  - *Functional Requirement*

REQ-29: The system should allow the consumers to fill all information.
REQ-30: The system should validation all consumers inputs.
REQ-31: The system should not send request if any data is not correct.
REQ-32: The system should show that filled have not correct.
REQ-33: The system should not allow consumers to login if account not exit.
REQ-34: The system should not allow consumers to login if phone number or otp is not correct.
REQ-35: The system should cash all information has been filled.
REQ-36: The system should delete all cash if consumers have logout.
REQ-7: The system should logout form previous session when consumers relogging.

- *Consumers buys*
  - *Description and Priority:*
    The consumers can make interest in any offer and start discussions with owner of offer or create a request some offer or search in available offer.
    This feature is high priority, we do this to make consumers found or buy what he wants.
    This feature maybe makes a consumer not found is correct or best offer.

  - *Stimulus/Response Sequences:*
    - *Search of offer: Consumers open offer list screen and type in search bar. Response: Filter is offer base on the search.*
    - *Request offer: Consumers open offer list screen and click on flat button. Response: The app open request offer screen. Consumers filled the form and click "Create offer request". Response: System will create offer and send offer to all farmer who grow the crop.*
    - *Interest offers: Consumers open offer list and click on "interest" button. Repones: System send notify to farmer who own the offer and open dialog. Consumers click on "start Negotiation". Response: System will start chatting between farmer and consumers.*

- *Complete offer: Consumers click "Complete" on offer details screen. Response: notify the framer.*
- *Cancelling: Consumers click on "Cancel" button on offer details screen. Response: System will send cancelling request.*
- *Ratting offer: Consumers will click on "ratting" on complete offer screen. Response: System will show dialog to ratting offer.*

- *Functional Requirement*

REQ-38: The system should allow the consumers to search for offer.
REQ-39: The system should allow the consumers to filter offers.
REQ-40: The system should allow the consumers to follow offer.
REQ-41: The system should notify the consumers in change status of following offer.
REQ-42: The system should not allow consumers to change is status of complete offer.
REQ-43: The system should not allow consumers to follow complete or deleted offer.
REQ-44: The system should allow to ratting alone one time and the offer was be completed by him.

- *Operation*
  - *Description and Priority:*
    The support team can manage and control all users of system, monitoring and control all offer, change is status of offer and get daily, week and month report.
    This feature is medium priority, we do this to make system under control and fix issue and help users.
    This feature add cost to system.
  - *Stimulus/Response Sequences:*
    - *Activate account: Support team click on "Activate" on accounts details screen. Response: Activate account.*
    - *Block account: Support team click on "Block" on accounts details screen. Response: Account block.*
    - *Change offer status: Support team click one status "change" button on offer details screen. Response: System will change offer status and send notify to farmer and follower of offer.*
    - *Accept of deleting offer request: Support team click on "deleted offer" on deleted request screen. Response: app show dialog to confirm deleted. Operation team click on "ok" button. Response: system deleted offer and send notify to followers of offer.*
    - *Follow up offer: Support team open offer details. Response: Show offer details.*
  - *Functional Requirement*
    REQ-45: The system should allow operation team to activate account.
    REQ-46: The system should allow operation team to block account.
    REQ-47: The system should allow operation team to management offer.
    REQ-48: The system should save operation event.

- *Manger*
  - *Description and Priority:*
    The manger can manage and monitoring the system and operation team.
    This feature is medium priority, we do this to make system under control and fix issue and help operation team.
    This feature add complexity to system.

- *Stimulus/Response Sequences:*
  - *See system status: Manger open dashboard. Response: System get all information about the system.*
  - *Add member to operation team: Manger click on add "new member" button on dashboard. Response: Open dialog with email filed. Manger add new member email. Response: System send email to the email.*
  - *Deleted Member from operation team: Manger click on add "deleted member" button on member details screen. Response: member account will deactivate.*
  - *Get report: Manger click on "Create a report" on main screen. Response: Open dialog with report type option. Manger select report type. Response: System will create a report.*

- *Functional Requirement*
  REQ-49: The system should allow manger to manage operation team.
  REQ-50: The system should allow manger to manage the system.
  REQ-51: The system should allow manger to create a report.

3.1 External Interface Requirements

3.1.1 User Interfaces

The User Interface section defines the way the various stakeholders interact with the System. The manager and support system screens will be developed to work on a PC/Laptop. The client system screens will be developed to work on a mobile device Android/iOS. Error messages will appear at the bottom and shall be self-descriptive - The maximum size of error messages will be 80 Characters. The design system will flow will the KMM design guideline.

3.1.2 Hardware Interfaces

The System shall be deployed on Heroku Platform. Manager and Support team are supposed to login into desktop applications. The clint (Farmer and consumers) will login into mobile (android/iOS) app.
Hardware Requirements for desktop applications:

- Pentium 4 processor or higher
- Approximately 100 MB of free hard drive space
- Minimum 128 MB RAM

Hardware Requirements for android device:

- Android 9 and up.

Hardware Requirement for iOS device:

- iOS 12 and up.

Hardware Requirement for hosting:

- Minimum 1GB database space
- Minimum 2GB RAM

3.1.3 Software Interfaces

The System is self-contained and data will be shared with government. All chatting will be in the app. They will be an interface with some standard Payment Gateway.
Software Requirements for Hosting:

- *Minimum java 11.*
- *MySQL.*

3.1.4 Communications Interfaces

The System will be self-contained and connected with back-end with API's with encrypted connection.

3.2 Functional Requirements

This section describes specific features of the software project. If desired, some requirements may be specified in the use-case format and listed in the Use Cases Section.

3.3 Use Cases

3.3.1 Use Case farmer want to Sells him produce

Farmer John is a small-scale farmer who grows tomatoes, cucumbers, and peppers on his farm. He has been struggling to sell his produce at the local market, as he faces stiff competition from bigger and cheaper suppliers. He also has to deal with middlemen who take a large cut of his profits and often cheat him on the quality and quantity of his produce. One day, he hears about the agricultural crop exchange system from his friend, Farmer Bob. Farmer Bob tells him that he has been using the system to trade his crops with other farmers or consumers online. He says that the system is easy to use, secure, and fair. He also says that he has been able to get better prices and quality for his crops, as well as to save time and money on transportation and delivery. Farmer John decides to give the system a try. He downloads the app on his smartphone and registers his profile with his personal and farm information. He also takes some photos of his crops and uploads them on the app. He then posts an offer for selling 100 kg of tomatoes at $1 per kg, with cash on delivery as the payment method and pickup at his farm as the delivery method. Soon, he receives a notification from the app that a consumers named Alice has shown interest in his offer. He opens the app and sees Alice's profile. She is a restaurant owner who is looking for fresh and organic tomatoes for her dishes. She has a high rating and positive reviews from other users. She also lives nearby, which makes the delivery easier. Farmer John accepts Alice's transaction request and initiates a chat with her. They agree on the details of the transaction, such as the date, time, and location of the pickup. They also exchange their contact details and confirm their agreement on the app. On the day of the pickup, Farmer John prepares his tomatoes and packs them in crates. He also prints out a QR code that contains the transaction information and attaches it to one of the crates. He then waits for Alice to arrive at his farm. Alice arrives on time with her truck. She greets Farmer John and scans the QR code with her smartphone. The app verifies the transaction and prompts her to pay Farmer John $100 in cash. She hands over the money to Farmer John and receives a receipt from him. She then loads the crates of tomatoes onto her truck and thanks Farmer John for his produce. Farmer John thanks Alice for her business and bids her farewell. He then opens the app and rates Alice 5 stars on her communication, payment, and delivery. He also writes positive feedback for her, saying that she

is a reliable and friendly consumer. He then submits his rating and feedback on the app. Alice does the same thing for Farmer John. She rates him 5 stars on his crop quality, price, and delivery. She also writes positive feedback for him, saying that he is a trustworthy and helpful seller. She then submits her rating and feedback on the app. The app notifies both Farmer John and Alice that their transaction is completed successfully and that their ratings and feedback are recorded on their profiles. The app also rewards them with some points that they can use to redeem discounts or bonuses on their next transactions. Farmer John is very happy with his experience using the system. He feels that he has made a good deal with Alice and that he has earned some extra income from his produce. He also feels that he has gained a new customer and a new friend in Alice. He decides to use the system more often to sell or exchange his crops with other farmers or consumers online. Alice is also very happy with her experience using the system. She feels that she has found a good source of fresh and organic tomatoes for her restaurant. She also feels that she has saved some time and money on transportation and delivery. She decides to use the system more often to buy or exchange crops with other farmers or sellers online.

3.3.2 Use Case farmer want to register

Farmer Ali is a young and ambitious farmer who grows maize, beans, and sorghum on his farm. He has been looking for new ways to market and sell his crops, as he is not satisfied with the traditional channels. He wants to reach more customers and get better prices for his crops. One day, he sees an advertisement on the internet about the agricultural crop exchange system. The advertisement claims that the system is a revolutionary platform that allows farmers to trade their crops with other farmers or consumers online. The advertisement also says that the system is easy to use, secure, and fair. Farmer Ali is intrigued by the advertisement and decides to check out the system. He clicks on the link and lands on the homepage of the system. He sees a brief introduction and a video tutorial about the system. He also sees some testimonials and success stories from other farmers who have used the system. Farmer Ali is impressed by what he sees and decides to join the system. He downloads the app and clicks on the “Register” button and fills out a simple registration form with his name, national No, phone number. He also agrees to the terms and conditions of the system. He then submits the form and receives an otp from the system. The system also asks him to complete his profile with his personal and farm information, such as location, contact details, farm size, crop types, etc. Farmer Ali follows the instructions and completes his profile. He receives notification the information has be vitrificated. He also uploads some photos of his farm and his crops on his profile. He then browses through the features and functions of the system. He learns how to post offers, search for offers, initiate transactions, rate transactions, etc. Farmer Ali is excited to use the system and start trading his crops online. He thinks that the system will help him to grow his business and achieve his goals. He decides to post his first offer for selling 50 kg of maize at $0.5 per kg, with mobile money as the payment method and delivery by courier as the delivery method. He hopes that someone will be interested in his offer and contact him soon. He also plans to look for other offers that might suit his needs and preferences. He thinks that he might find some good deals or opportunities on the system.

3.3.3 Use Case consumers want to buy produce

Consumers Mary is a health-conscious and environmentally-friendly consumer who likes to buy fresh and organic produce for her business. She has been dissatisfied with the quality and variety

of the produce available at the supermarkets, as they are often imported, processed, or genetically modified. She also has to pay high prices and fees for the produce and the delivery. One day, she reads an article on the internet about the agricultural crop exchange system. The article claims that the system is a revolutionary platform that allows consumers to trade crops with farmers online. The article also says that the system is easy to use, secure, and fair. Consumers Mary is curious about the system and decides to check out the system. She clicks on the link and lands on the homepage of the system. She sees a brief introduction and a video tutorial about the system. She also sees some testimonials and success stories from other consumers who have used the system. Consumers Mary is impressed by what she sees and decides to join the system. She He downloads the app and clicks on the "Register" button and fills out a simple registration form with her name, national No, phone number. She also agrees to the terms and conditions of the system. She then submits the form and receives a confirmation otp from the system. The system also asks her to complete her profile with her personal and delivery information, such as location, contact details, delivery address, delivery preferences, etc. Consumers Mary follows the instructions and completes her profile. She then browses through the features and functions of the system. She learns how to search for offers, post requests, initiate transactions, rate transactions, etc. Consumers Mary is eager to use the system and start buying crops online. She thinks that the system will help her to find and connect with local farmers who grow fresh and organic crops. She decides to search for some offers that match her needs and preferences. She enters "carrots" in the search box and clicks on the "Search" button. The system displays a list of offers from different farmers who are selling carrots. The system also provides filters and sorting options based on criteria such as location, price, quality, etc. Consumers Mary applies some filters and sorts the offers by price. She sees an offer from a farmer named Sam who is selling 10 kg of carrots at $0.3 per kg, with mobile money as the payment method and delivery by courier as the delivery method. She opens Sam's profile and sees that he has a high rating and positive reviews from other users. He also lives nearby, which makes the delivery faster. Consumers Mary likes Sam's offer and decides to buy his carrots. She clicks on the "Buy" button and sends a transaction request to Sam. She also initiates a chat with him to confirm his availability and ask some questions about his carrots. Sam replies to Consumers Mary's chat message and accepts her transaction request. He answers her questions and assures her that his carrots are fresh, organic, and delicious. He also agrees to deliver his carrots to Consumers Mary's address within two hours. Consumers Mary thanks Sam for his prompt response and confirms her agreement on the app. She also shares her contact details and delivery address with him. Two hours later, Sam arrives at Consumers Mary's address with his carrots in a bag. He greets Consumers Mary and scans a QR code that contains the transaction information with his smartphone. The app verifies the transaction and prompts him to deliver his carrots to Consumers Mary. Consumers Mary receives Sam's carrots and checks their quality and quantity. She is satisfied with what she sees and thanks Sam for his produce. Sam thanks Consumers Mary for her business and bids her farewell. He then opens the app and rates Consumers Mary 5 stars on her communication, payment, and delivery. He also writes positive feedback for her, saying that she is a loyal and friendly consumer. He then submits his rating and feedback on the app. Consumers Mary does the same thing for Sam. She rates him 5 stars on his crop quality, price, and delivery. She also writes positive feedback for him, saying that he is honest and helpful seller. She then submits her rating and feedback on the app. The app notifies both Consumers Mary and Sam that their transaction is completed successfully and that their ratings and feedback are recorded on their profiles. The app also rewards them with some points

that they can use to redeem discounts or bonuses on their next transactions. Consumers Mary is very happy with her experience using the system. She feels that she has found a good source of fresh and organic carrots for her family. She also feels that she has saved some time and money on transportation and delivery. She decides to use the system more often to buy or exchange crops with other farmers or sellers online. Sam is also very happy with his experience using the system. He feels that he has sold his carrots at a fair price and that he has earned some extra income from his produce. He also feels that he has gained a new customer and a new friend in Consumers Mary. He decides to use the system more often to sell or exchange his crops with other consumers or farmers online.

3.3.4 Use Case consumers who wants to register

Consumers Tom is a grocery store owner who sells various kinds of produce to his customers. He has been facing some challenges in sourcing and supplying his produce, as he has to deal with unreliable and expensive suppliers, fluctuating market prices, and changing customer demands. He also has to pay high fees and taxes for importing his produce from other countries. One day, he receives a flyer in the mail about the agricultural crop exchange system. The flyer claims that the system is a revolutionary platform that allows consumers to trade crops with farmers online. The flyer also says that the system is easy to use, secure, and fair. Consumers Tom is interested in the system and decides to check out the system. He scans the QR code on the flyer and lands on the homepage of the system. He sees a brief introduction and a video tutorial about the system. He also sees some testimonials and success stories from other consumers who have used the system. Consumers Tom is impressed by what he sees and decides to join the system. He downloads the app and clicks on the “Register” button and fills out a simple registration form with his name, national No, phone number. He also agrees to the terms and conditions of the system. He then submits the form and receives a confirmation otp from the system. The system verifies his email address and welcomes him to the platform. The system also asks him to complete his profile with his personal and delivery information, such as location, contact details, store name, store address, delivery preferences, etc. Consumers Tom follows the instructions and completes his profile. He then browses through the features and functions of the system. He learns how to search for offers, post requests, initiate transactions, rate transactions, etc. Consumers Tom is eager to use the system and start buying crops online. He thinks that the system will help him to find and connect with local farmers who grow fresh and quality crops. He decides to post his first request for buying 20 kg of potatoes at $0.4 per kg, with mobile money as the payment method and delivery by courier as the delivery method. He hopes that someone will be interested in his request and contact him soon. He also plans to look for other requests that might suit his needs and preferences. He thinks that he might find some good deals or opportunities on the system.

3.4 Non-Functional Requirements

3.5.1 Performance

At the peak, system should be able to scale to 10,000+ users concurrently. The following processes are critical and must respond as per below:

- *Farmer registration <= 4 seconds.*
- *Consumer’s registration <= 4 seconds.*
- *Searching <= 2 seconds.*

- *Filtering <= 2 seconds.*
- *Buying <= 2 seconds.*
- *Creating offer <= 4 seconds.*

3.5.2 Reliability

The software should have a reliability of 99.99%, meaning that it should fail no more than once in 10,000 uses. The software should handle any exceptions or errors gracefully and provide informative messages to the users. The software should have a self-diagnostic and self-repair mechanism that can detect and correct any faults within 5 minutes. The software should log any failures and their causes and report them to the administrators.

3.5.3 Availability

The software should be available 24/7, except for scheduled maintenance periods of no more than 4 hours per month. The software should have a minimum availability of 99.9%.

3.5.4 Security

Transactions and financial transfer information must be kept confidential and not disclosed except to support staff after requesting permission from both parties of the transaction.

3.5.5 Maintainability

The software should follow the SOLID principles of object-oriented design to ensure high cohesion, low coupling, and easy extensibility1. The software should use consistent coding standards and conventions throughout the project. The software should be well-documented with clear and concise comments, diagrams, and user manuals. The software should have a comprehensive test suite that covers all the functional and non-functional requirements, as well as any edge cases or exceptions. The software should have a version control system that tracks all the changes made to the source code and allows for easy rollback and recovery. The software should have a continuous integration and delivery pipeline that automates the building, testing, and deployment of the software. The software should have a bug tracking and reporting system that allows users and developers to report, prioritize, and resolve any issues or defects. The software should have a maintenance team that is responsible for providing regular updates, enhancements, and fixes to the software. The software should have a feedback mechanism that allows users to suggest improvements or new features for the software.

3.5.6 Portability

The software should be portable across different devices and operating systems. The software should support Windows, iOS, and Android platforms. The software should use KMM for the mobile and desktop development, and Kotlin and Spring boot for the back-end development. The software should use JSON as the data exchange format, and HTTPS as the communication protocol.

3.6 Inverse Requirements

The system should not have an internal money transfer system. The system should not allow price comparison by default. The system should not allow talking to another person outside the transaction.

3.7 Design Constraints

- *The system must follow the state laws.*
- *The system must comply with ISO/IEC 25010:2011 standard for software quality.*
- *The system must use HTTPS protocol for secure communication with server.*
- *The software system must have a user interface that is compatible with screen readers and other assistive technologies.*

3.8 Logical Database Requirements

- *The software system should store the information of the customers, products, orders, and payments in a relational database. The database should have tables for each entity, with primary and foreign keys to link them. The database should enforce referential integrity and uniqueness constraints on the data. The database should keep the historical records of the orders and payments for at least one year.*

3.9 Other Requirements

- *The software system must provide a user manual, a troubleshooting guide, and a FAQ section for the end-users. The software system must also provide a technical documentation, a testing report, and a deployment guide for the developers and administrators.*
- *The software system must be installed on a dedicated server with a minimum of 8 GB of RAM, 500 GB of disk space, and a quad-core processor. The software system must also have a backup server and a disaster recovery plan in case of any failure.*
- *The software system must handle any invalid or unexpected inputs, network interruptions, or power outages gracefully and without causing any data loss or corruption. The software system must also log any errors or exceptions and notify the users and the administrators accordingly.*

4. Analysis Models

List all analysis models used in developing specific requirements previously given in this SRS. Each model should include an introduction and a narrative description. Furthermore, each model should be traceable the SRS's requirements.

4.1 Sequence Diagrams

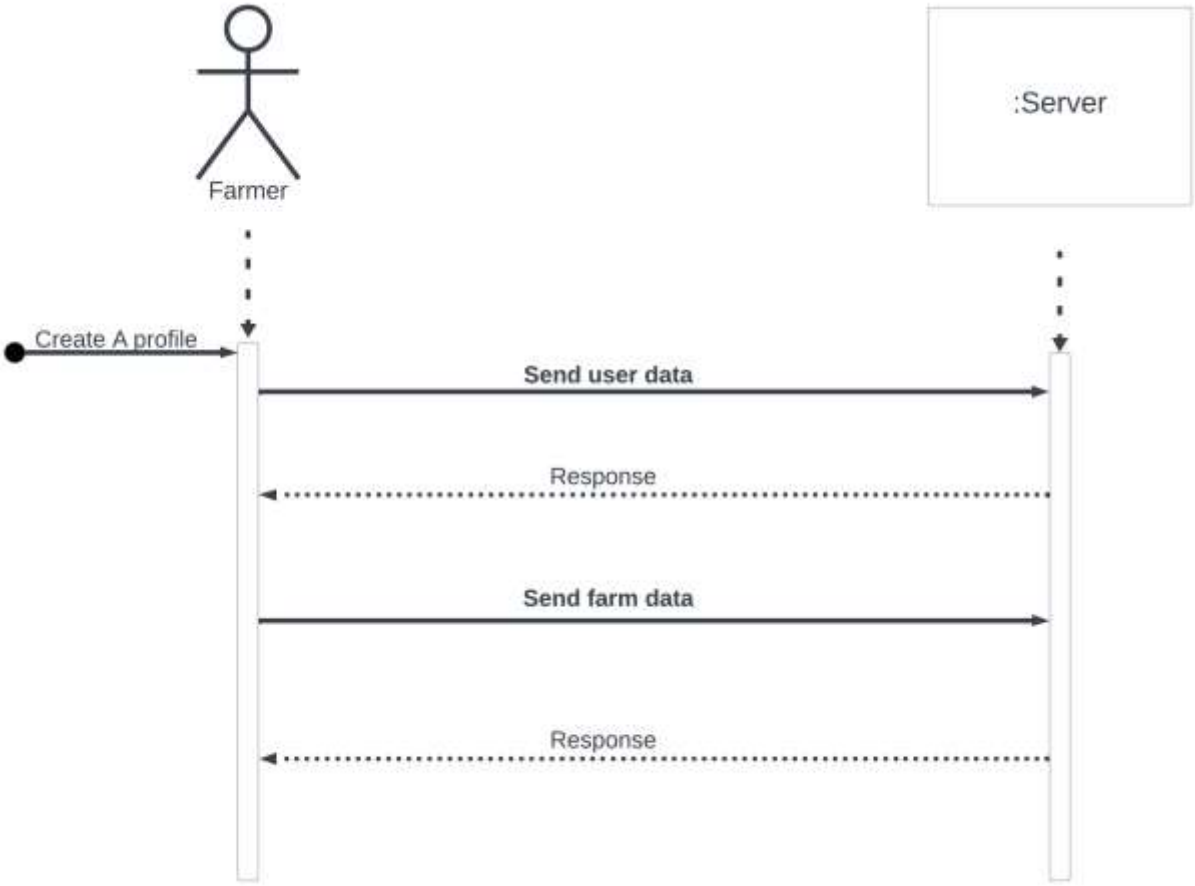

Figure 2-Farmer Create profile Sequence Diagram

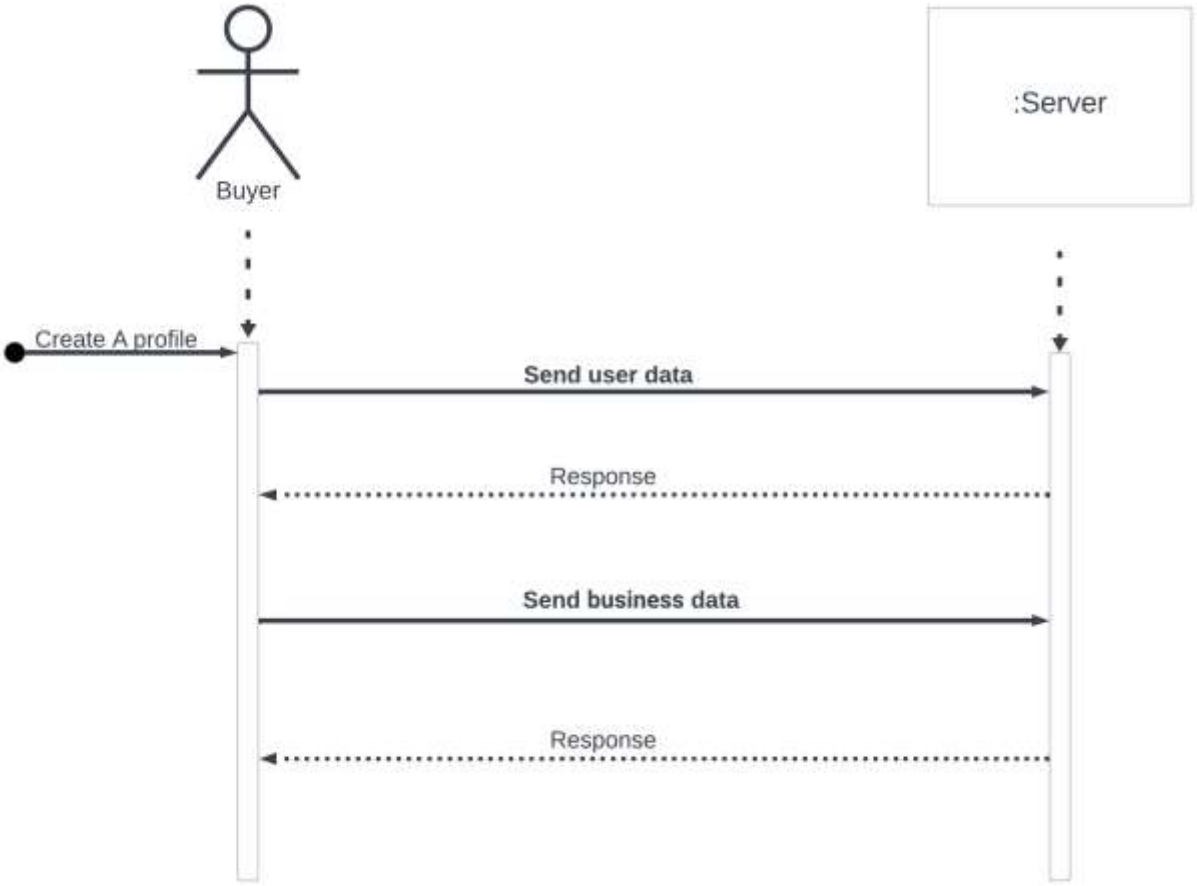

Figure 3-Consumers Create profile Sequence Diagram

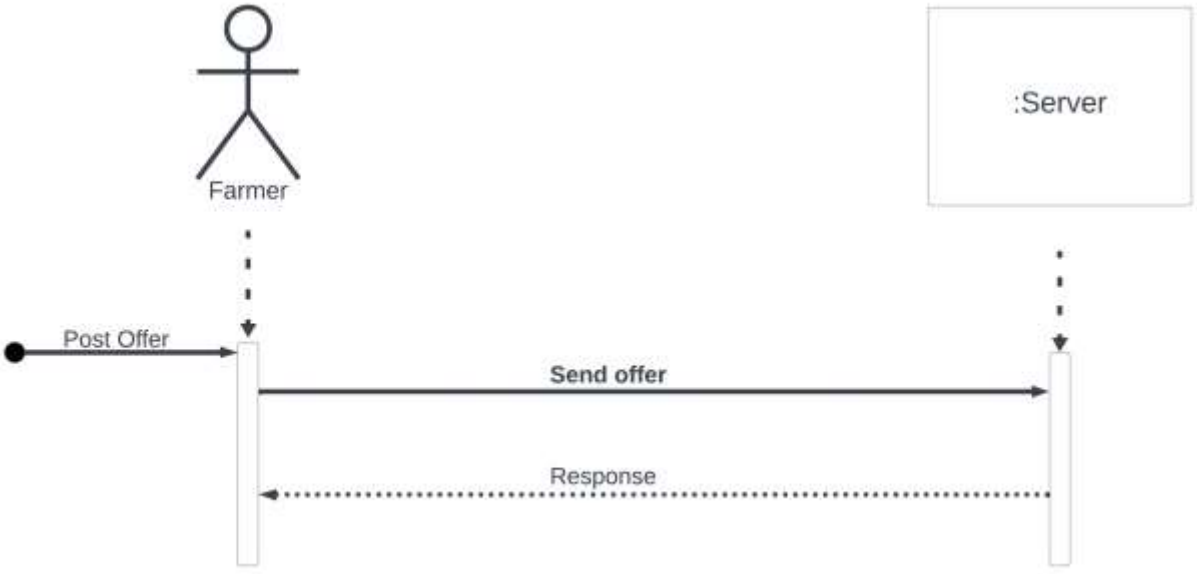

Figure 4-Farmer Create offer Sequence Diagram

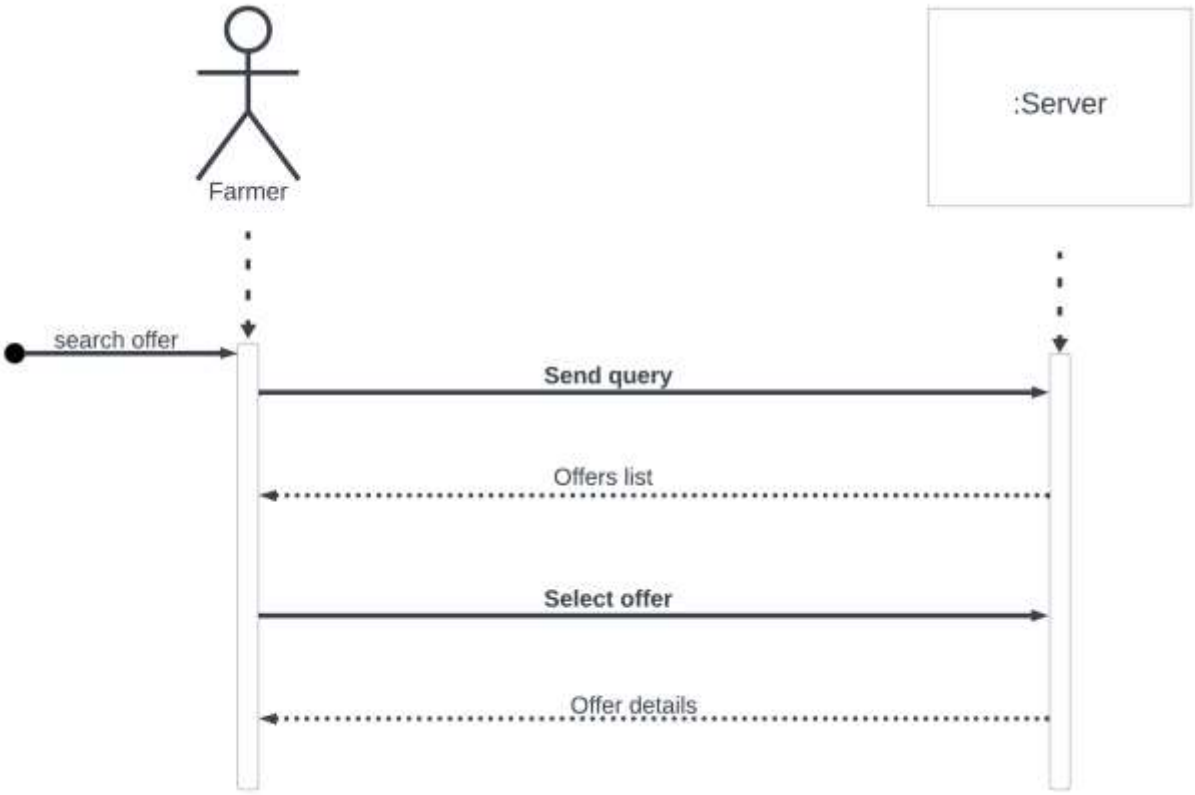

Figure 5-Farmer Search Offer Sequence Diagram

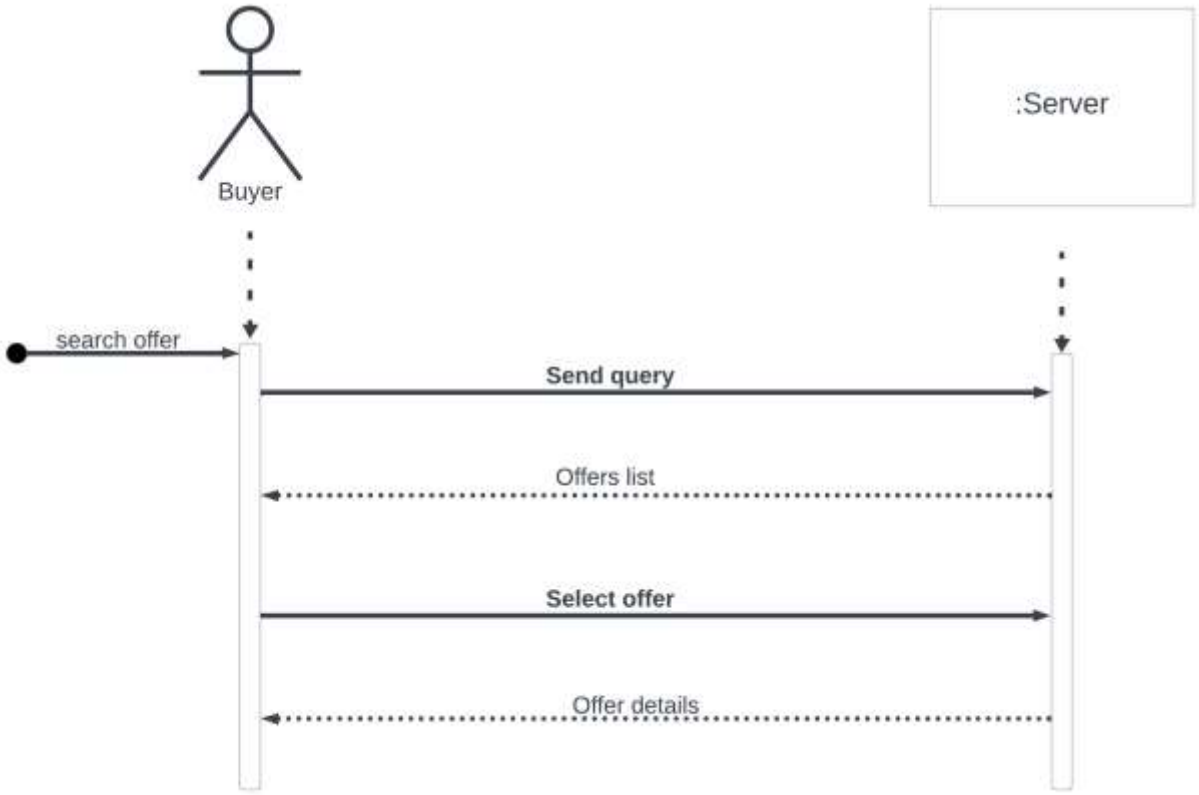

Figure 6-Consumers Search offer Sequence Diagram

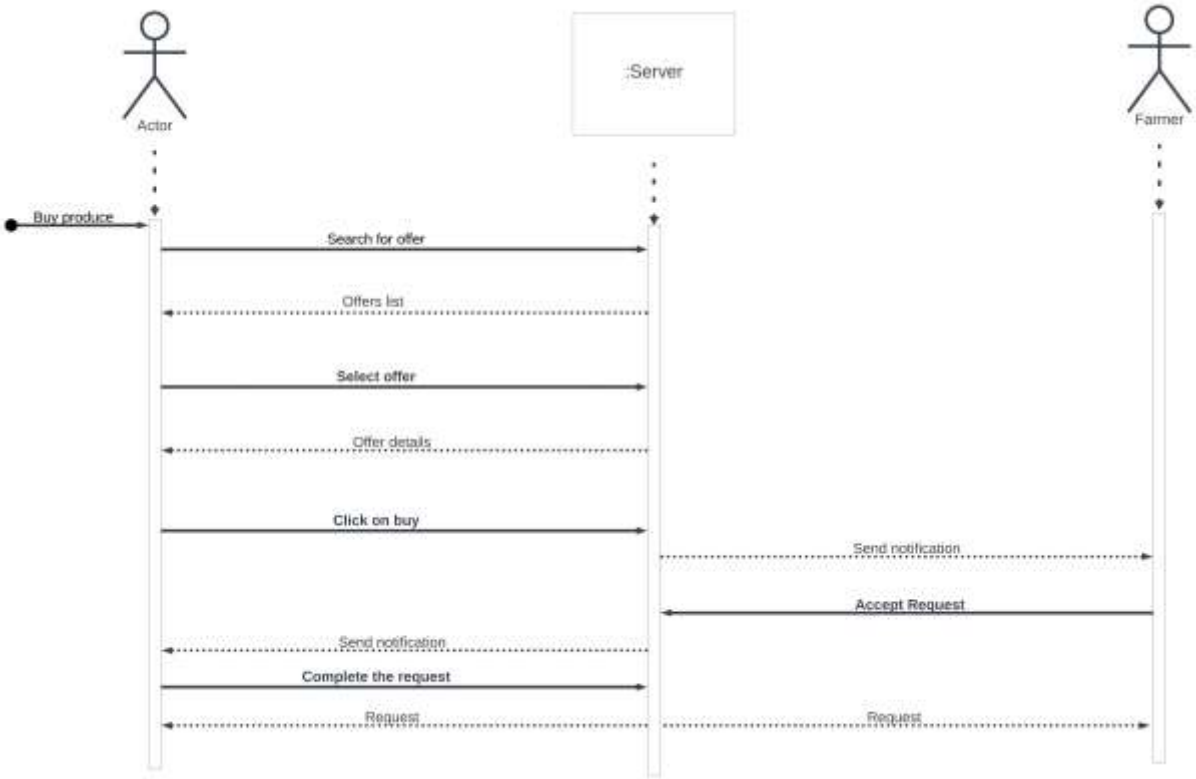

Figure 7-Actor Buy Sequence Diagram

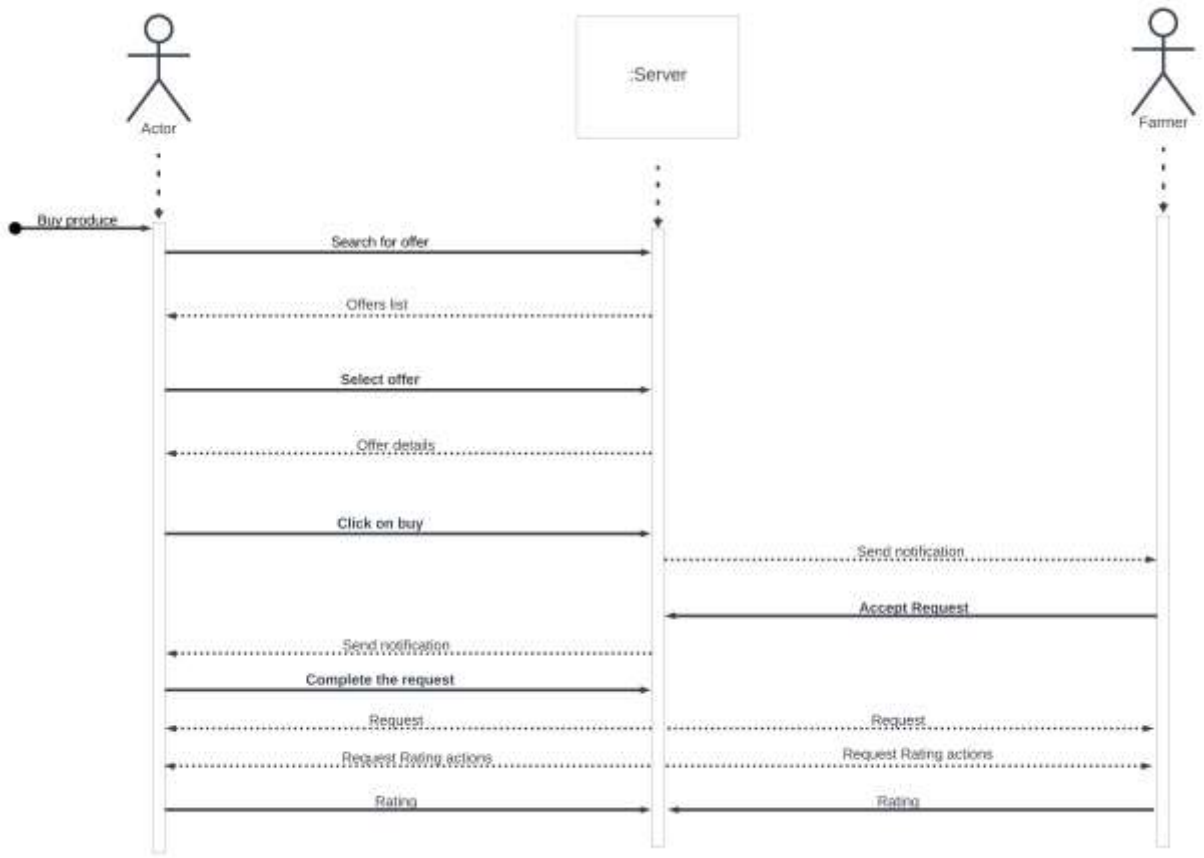

Figure 8-Actors Rating Sequence Diagram

5. Change Management Process

- *The change management process defines how changes to the software requirements specification will be handled, controlled, and tracked throughout the software development lifecycle1.*
- *The change management process involves the following roles and responsibilities2:*
    - *Change requester: The person who identifies the need for a change and submits a change request form that describes the change, its rationale, its priority, and its impact.*
    - *Change reviewer: The person who evaluates the change request and determines its feasibility, cost, benefits, and risks. The change reviewer may approve, reject, or defer the change request, or request more information from the change requester.*
    - *Change approver: The person who has the authority to approve or reject the change request based on the change reviewer's recommendation. The change approver may also consult with other stakeholders, such as the project manager, the customer, or the end-users, before making the final decision.*
    - *Change implementer: The person who implements the approved change in the software system and updates the SRS document accordingly. The change implementer may also perform testing and verification to ensure the change meets the requirements and does not introduce any errors or defects.*

- *The change management process follows these steps2:*
    - *Initiate: The change requester fills out a change request form and submits it to the change reviewer.*
    - *Evaluate: The change reviewer analyzes the change request and assesses its feasibility, cost, benefits, and risks. The change reviewer may also request more information from the change requester or consult with other experts or stakeholders. The change reviewer then makes a recommendation to approve, reject, or defer the change request and sends it to the change approver.*
    - *Approve: The change approver reviews the change request and the change reviewer's recommendation and makes the final decision to approve or reject the change request. The change approver may also consult with other stakeholders, such as the project manager, the customer, or the end-users, before making the final decision. The change approver then notifies the change requester, the change reviewer, and the change implementer of the decision and the reasons behind it.*
    - *Implement: The change implementer implements the approved change in the software system and updates the SRS document accordingly. The change implementer may also perform testing and verification to ensure the change meets the requirements and does not introduce any errors or defects. The change implementer then reports the completion and the results of the change to the change requester, the change reviewer, and the change approver.*
    - *Track: The change management process uses a change log to record and track all the changes made to the SRS document. The change log includes the following information for each change2:*
        - *Change ID: A unique identifier for the change.*
        - *Change description: A brief summary of the change and its rationale.*
        - *Change requester: The name and role of the person who initiated the change request.*
        - *Change reviewer: The name and role of the person who evaluated the change request.*
        - *Change approver: The name and role of the person who approved or rejected the change request.*
        - *Change implementer: The name and role of the person who implemented the change.*
        - *Change date: The date when the change was requested, evaluated, approved, and implemented.*
        - *Change status: The current status of the change, such as pending, approved, rejected, deferred, or completed.*
        - *Change impact: The impact of the change on the software system, the SRS document, and other related documents or artifacts.*