System Design of A Food Donation Platform

PREPARED FOR

Dr. Nkemeni Valery

CEF440 - Internet programming and Mobile programming

PREPARED BY GROUP 15

Randy Susung Nesinyu Kwalar - FE20A101

Niba Godfaith Cedric Fuh - FE20A081

Achale Ebot Oma - FE20A002

Agyingi Jan Royal - FE20A005

Ataba Emmanuel Junior - FE20A013

Table of Contents

INTRODUCTION

The purpose of this document is to provide the design of a food donation platform. The platform is designed to allow people and organisations to donate excess food for free or at a discount to those who are in need. The platform aims to provide an efficient and streamlined process for food donation, thereby increasing the accessibility of food for those that need it.

1.1 Overview

The problem that this project aims to solve is the waste of food in our society. There are many people who have excess food and are willing to donate it, but the lack of a suitable platform to facilitate these donations often results in the food going to waste. This project seeks to address this issue by creating a food waste management app that serves as a platform for food donations. The app will allow people to upload details and pictures of available food items that they are willing to donate. These food items can be listed at a discounted price or for free. Other users can then browse the app for available food items and claim them. The app will be designed to be user-friendly and accessible to both donors and recipients, with a focus on ensuring that all donations are safe and edible. Overall, the goal of this project is to create a sustainable solution to food waste while also providing a means of helping those in need.

1.2 Project Scope

The project scope for addressing the food donation problem includes the following:

- Identifying food donors: The project will identify individuals or organisations that have excess food and are willing to donate it.
- Partnering with food banks or organisations: The project will partner with food banks or organisations that address hunger and food insecurity, to facilitate the distribution of donated food to those in need.
- Ensuring food safety: The project will ensure that all donated food is safe and of good quality by adhering to safety regulations and guidelines.

- Developing a communication plan: The project will develop a communication plan to effectively communicate the benefits of food donation, how the system works, and how to get involved.
- Setting up an escrow payment system: The project will set up a reliable and efficient payment system to deliver the money to the donor when the food is gotten.
- Monitoring and evaluation: To ensure the project's success, there will be continuous monitoring and evaluation to track progress towards project goals, and identify areas that may need improvement.
- Raising awareness: The project will raise public awareness about the issue, to encourage more people to donate food and reduce food waste.
- Sustainability plan: The project will develop a sustainability plan to ensure the viability of the project in the long term. This may include securing funding sources, optimising operations, and creating partnerships with businesses, and other organisations.

Overall, the project aims to create a robust and sustainable system for food donation and reduce food waste while helping those in need.

1.3 System Requirements

Functional Requirements

Here are some functional requirements for the food donation platform:

- User registration: The platform should allow users to create an account and register as a donor, requester, or guest. The registration process should collect the user's name, email, password and verify the user's email address.
- Donor upload: The platform should allow donors to upload details of excess food items such as name, short description, expiry date, availability, and discounted price or free.
- Search and browse: The platform should allow users to search for available food items by name or keyword, filter food items by categories such as location, availability, and discounted price or free. The platform should also display a list of available food items that match the search criteria.
- Request for food item: The platform should allow users to request a food item, and the request will be sent to the donor for approval.

- Approval of request: The platform should allow donors to approve or reject a request made by a user. The donor should provide feedback on the reason for the rejection of a request.
- Payment: The platform should have a payment gateway where users can make payment for food items that are not free.
- Transaction history: The platform should keep a record of all transactions, including the donor, requester's name, date of transaction, food ID and amount paid.
- Expiry date management: The platform should manage the expiry date of each item uploaded by donors and mark them as unavailable when expired.
- User feedback: The platform should allow users to provide feedback on the platform's functionality, usability, and user experience.
- User account management: The platform should allow users to modify their account information, including their name, email and password.
- Admin controls: The platform should have an administrator account that can manage user accounts, review transactions, and handle disputes between users.
- Messaging: The platform should allow users to communicate with each other via messages to find out more about the food
- Notification system: The platform notifies donors when a user requests an item on discount and also notifies users when donors around them upload new food items.

Overall, the functional requirements will ensure that the platform provides a comprehensive platform for managing food waste and encourages food donation for the betterment of the society.

Non-Functional Requirements

To ensure users and donors have a seamless experience while using this platform, the following properties are considered

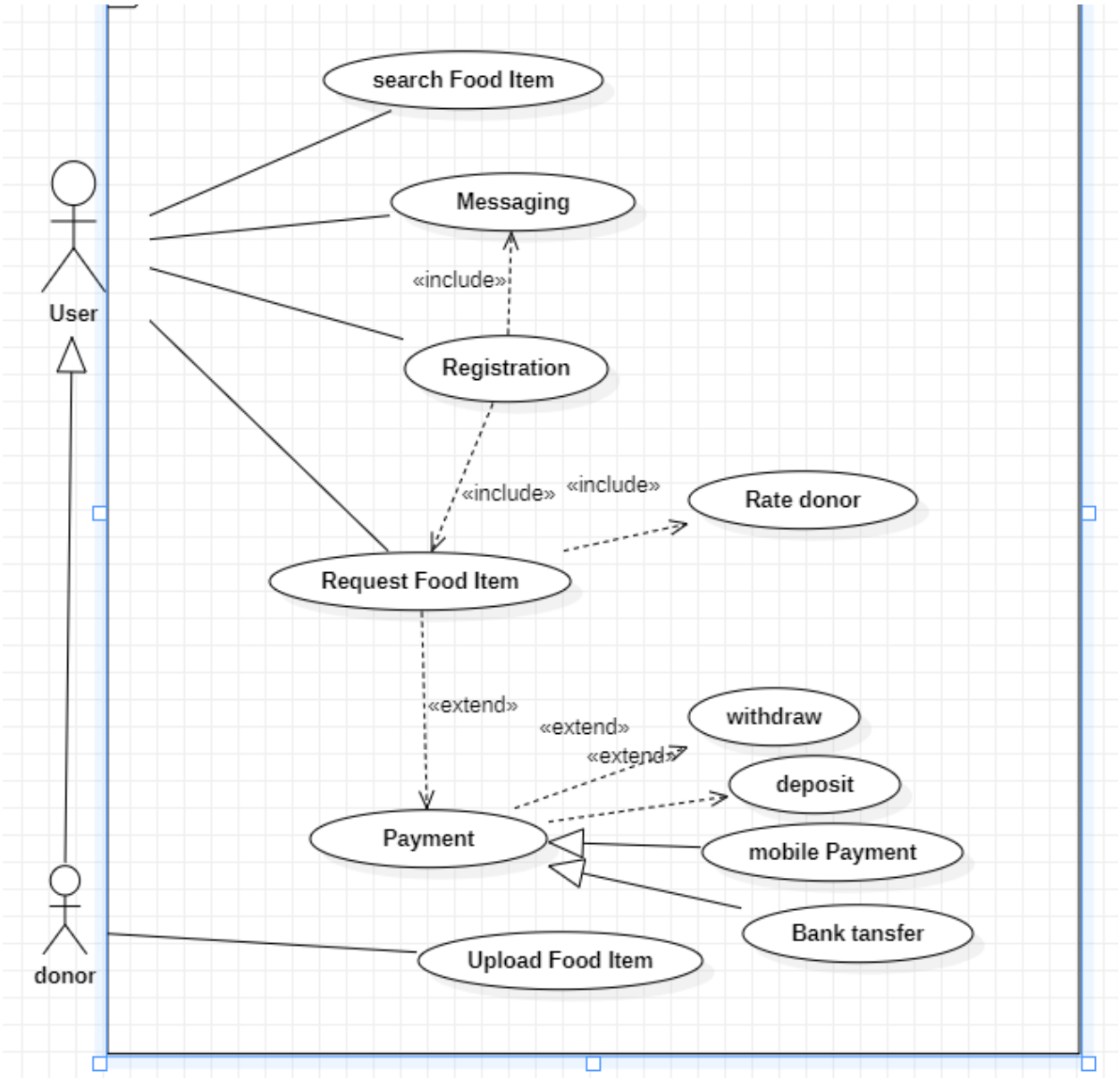
- Usability: The platform should be easy to use and understand for all users, including those who are not tech-savvy.
- Performance: The platform should be fast and responsive, and not lag or crash during use.
- Security: The platform should ensure the privacy and security of user data by using encryption and other security measures.
- Reliability: The platform should be reliable and available 24/7, with minimal downtime or maintenance required.
- Accessibility: The platform should be accessible to all users, including those with disabilities, by implementing features such as text-to-speech and screen reader support.

- Scalability: The platform should be able to handle a large number of users and transactions simultaneously without compromising performance.
- Availability: The platform should be available on multiple platforms and devices, such as iOS and Android, and accessible through a web browser.
- Compliance: The platform should comply with all relevant laws and regulations, such as data protection laws, and adhere to industry standards for software development and data management.
- User experience: The platform should provide an excellent user experience, with an intuitive interface and clear navigation, ensuring users can complete tasks easily.
- Localization: The platform should support multiple languages and be adaptable to local cultures and customs.
- Maintenance: The software system has to be maintained over time with minimal effort to ensure its efficient and effective operation.
- Legal and regulatory requirements: Legal and regulatory requirements are non-functional requirements that refer to the legal and regulatory obligations that the food donation system should comply with.

Overall, the non-functional requirements are essential to ensure the platform is easy to use, trustworthy, and performs optimally for all users, resulting in a positive user experience.

Use Case Diagram

2.1 Diagram



2.2 Use Case Descriptions

Registration:

This use case involves a new user or client to create an account on the food donation platform to enable them to upload (donor) or buy food at discount from the platform. It involves collecting basic information about the user.

Search Food Item:

This use case involves a new user or client searching for available food stuff to get either for free or at a discount. It involves entering information about the food you desire and searching.

Request Food Item:

This use case is concerned with users requesting to get food uploaded by a donor.

Messaging:

This use case involves a user and donor exchanging messages with each other to find out more about the upload food Item.

Upload Food Item:

This use case involves donors uploading food items they want to give out either for free or at a discount. It entails entering information about the food items including name, images, and price if need be.

Payment:

This use case involves user and donors either depositing to make payments or withdrawing payments.

Rating:

This use case involves a user giving a rating out of 5 stars depending on the quality of the donors food and service.

2.3 Use Case Scenarios

Registration:

Purpose	Create user and donor account
User	User, donor
Input data	User/donor information
Output data	User/ donor account
Invariants	
Pre-conditions	Client does not have an account
Post-conditions	Client now gets account
Basic flow	Client clicks on the application, enters their information like name, email, phone, location and creates an account either as a donor or normal user

Search Food Item:

Purpose	Look for food items on the platform
User	User, donor
Input data	Search query
Output data	List of food items that match the search

Invariants	
Pre-conditions	User needs to find an item
Post-conditions	Item gets found
Basic flow	While in the application, a user can do a search for a food item and elaborate if this item is cooked or raw, free or discount

Request Food Item:

Purpose	To try to buy food or get it for free if it is.
User	user
Input data	Button click on request button
Output data	Food item and donor information
Basic flow	User finds the food item they wish to get and requests to get it.

Messaging:

Purpose	To find out more about the upload product
User	user
Input data	Messages
Output data	messages
Pre-conditions	User wants to find out more about a donor and their food item
Post-conditions	User finds out more about food item and donor
Basic flow	User can send donor a message when they find that donor's upload appealing to get more information about the food item

Upload Food Item:

Purpose	Upload a food item on the platform
User	Donor
Input data	Food item information
Output data	Food item uploaded on the platform
Invariants	
Pre-conditions	Food Item has not been uploaded
Basic flow	Donor inputs basic information about food item and takes some images of it as well. Donor also chooses to give item for free or at a discount.

Payment:

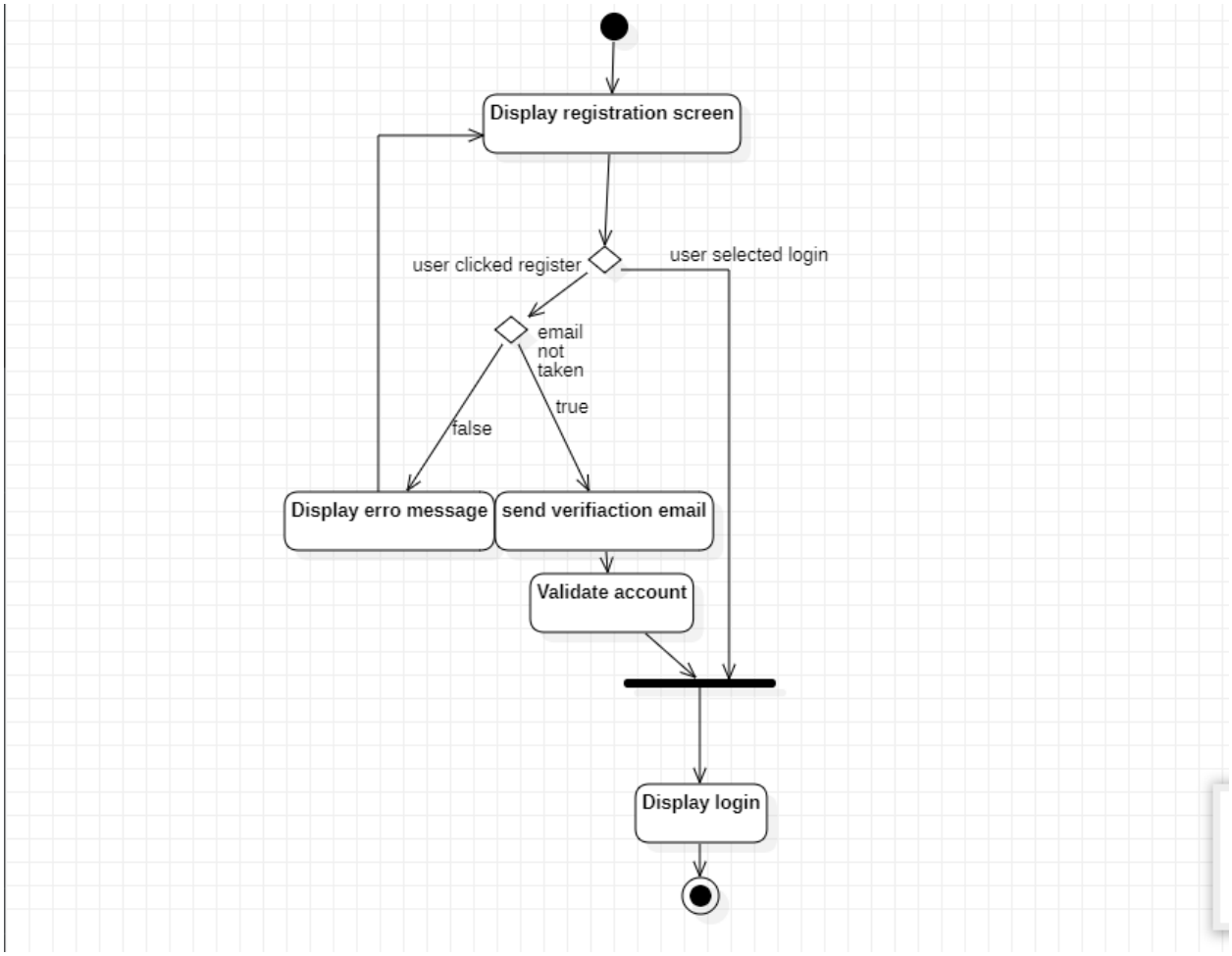
Purpose	To either withdraw or make a payment for a food item
User	User, donor
Input data	User credentials
Output data	Either a withdrawal or deposit
Basic flow	User can choose to deposit money to make a payment for a food item or withdraw money from the purchase of their food item.

Rating:

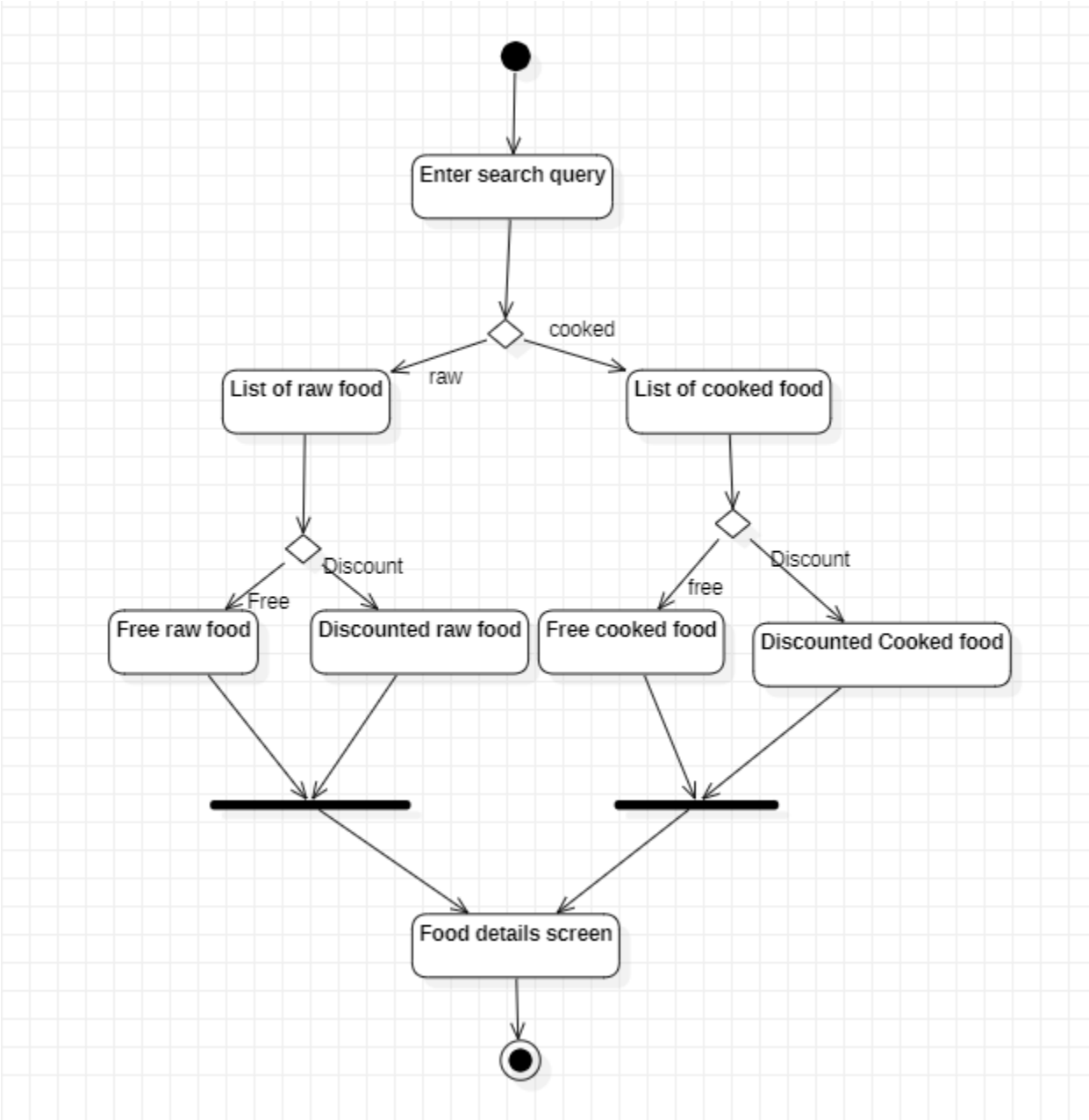
Purpose	Give merits to donors based on the quality of their food item and their service
User	users.
Input data	Rating ranging from 0-5 stars
Output data	Donor rating
Pre-conditions	User has received a food item from a donor
Basic flow	After a user has made payment and/or received their food item they can rate a donor based on the quality of the service provided by the donor.

Activity Diagrams

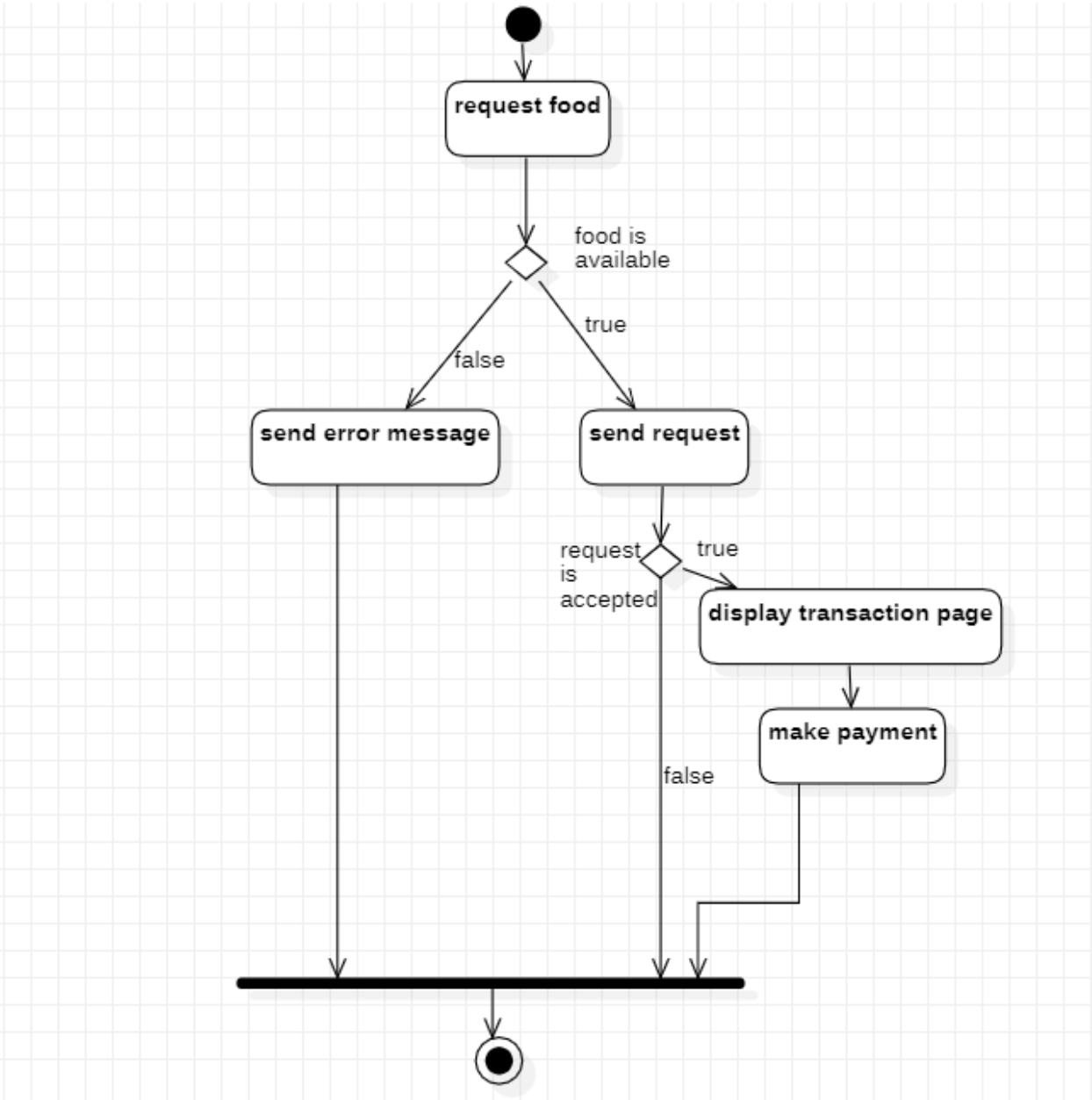
3.1 Registration



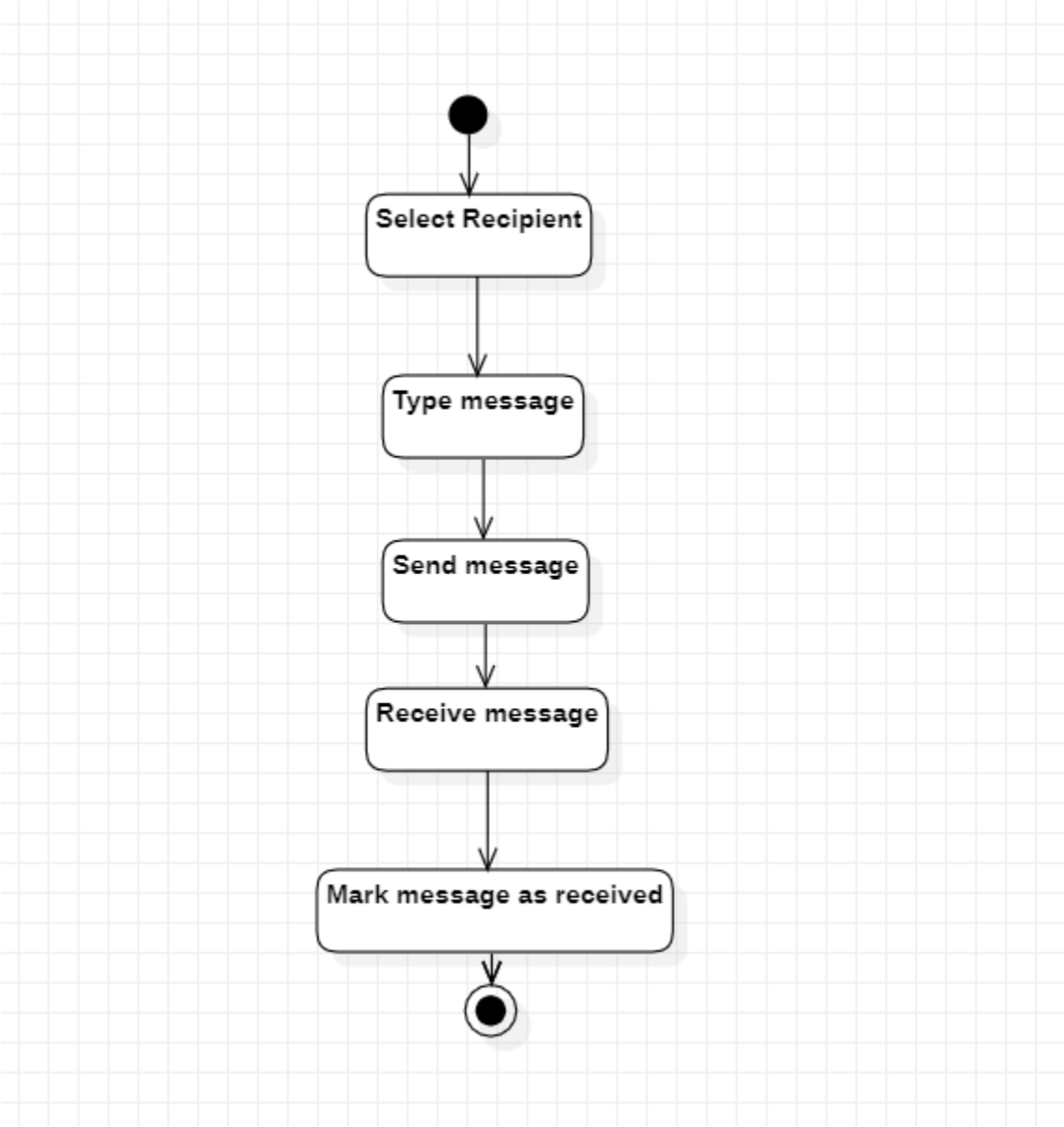
3.2 Search Food Item



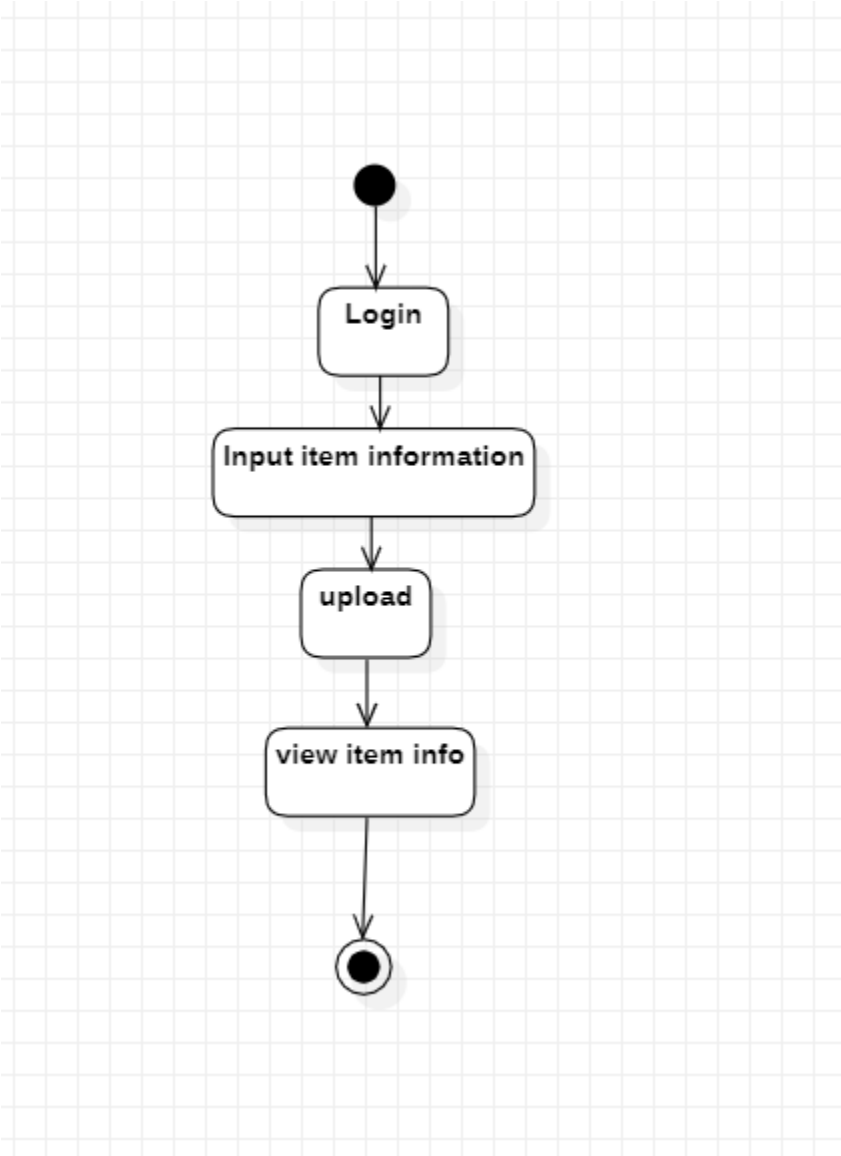
3.3 Request Food Item



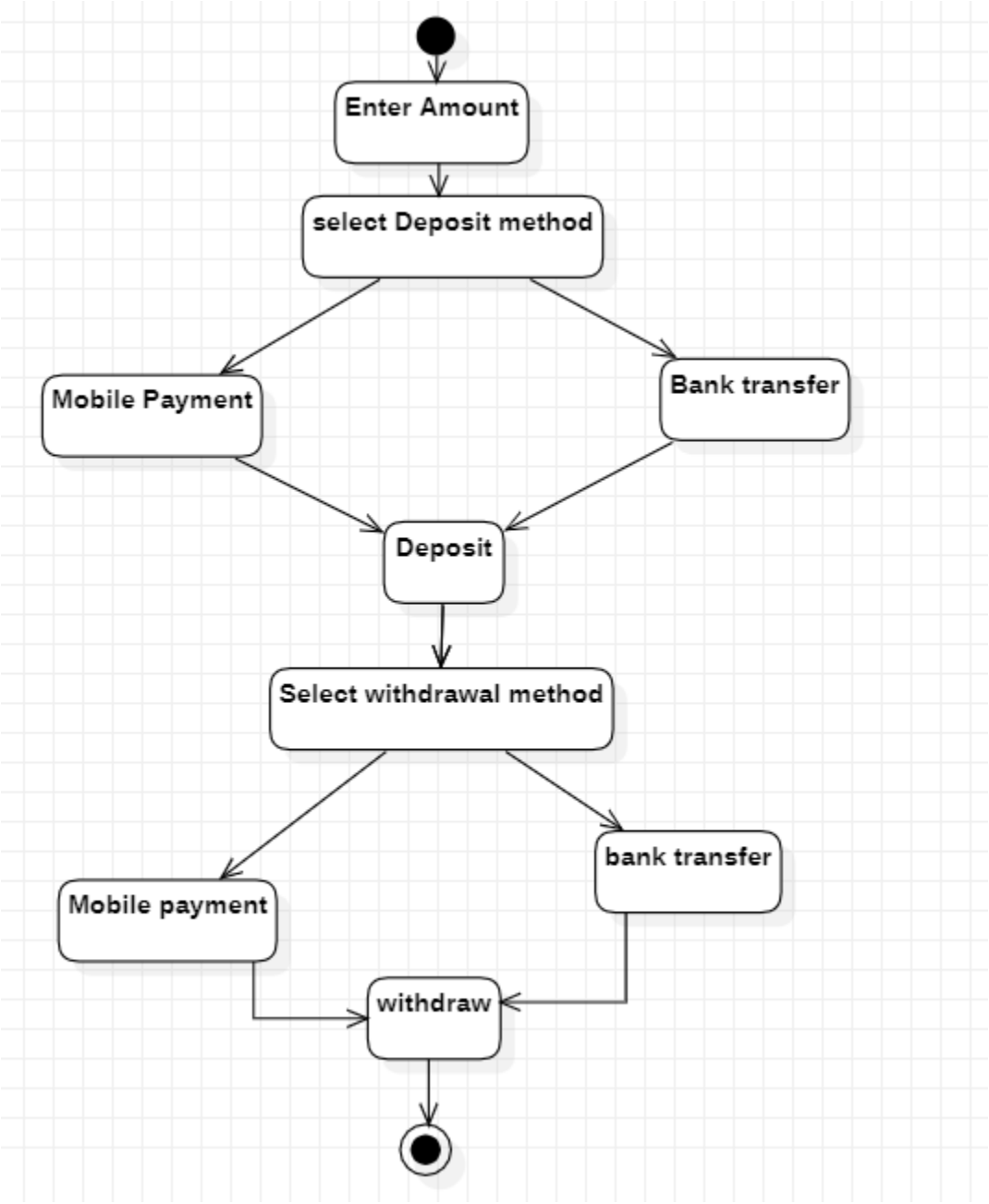
3.4 Messaging



3.5 Upload Food Item

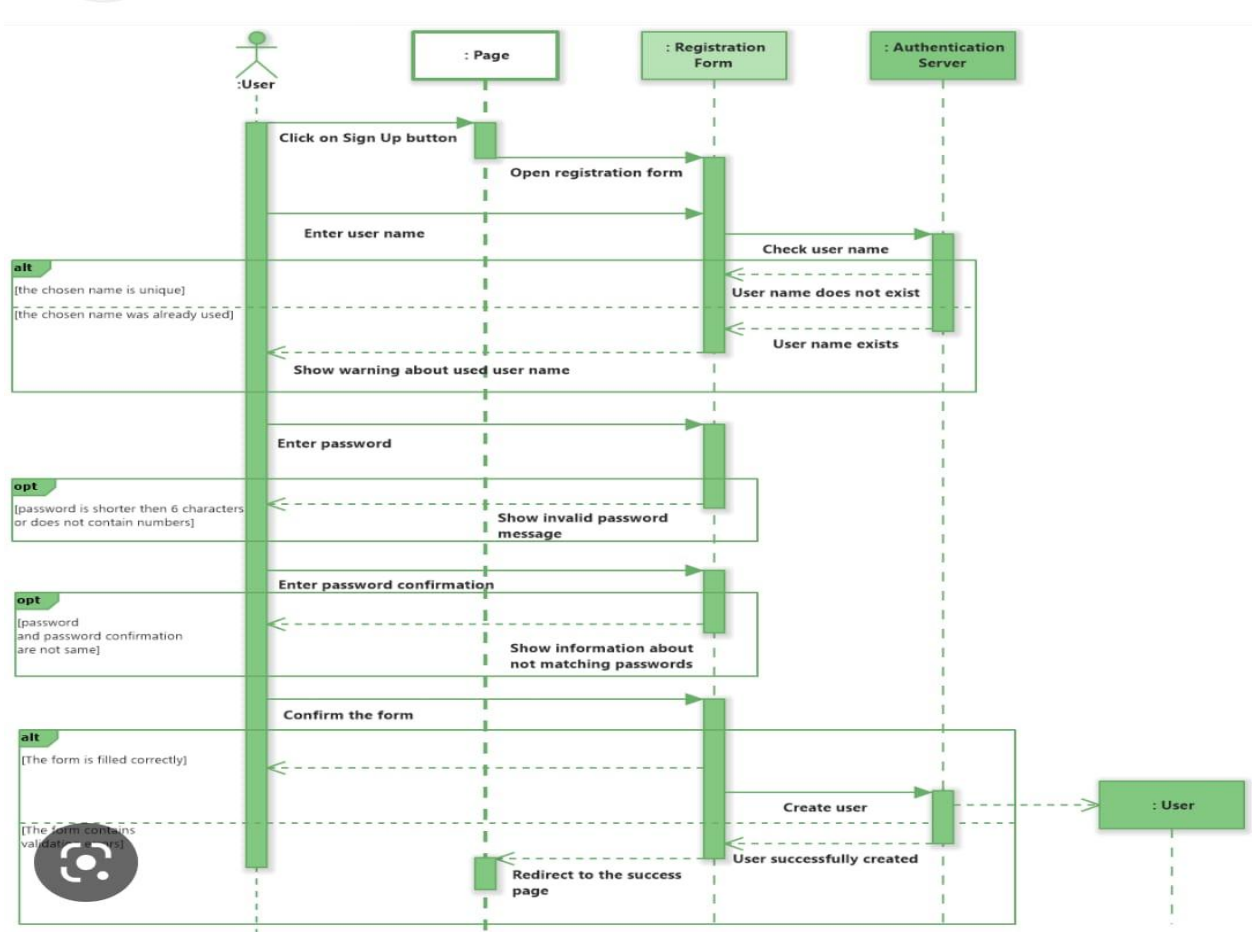


3.6 Payment

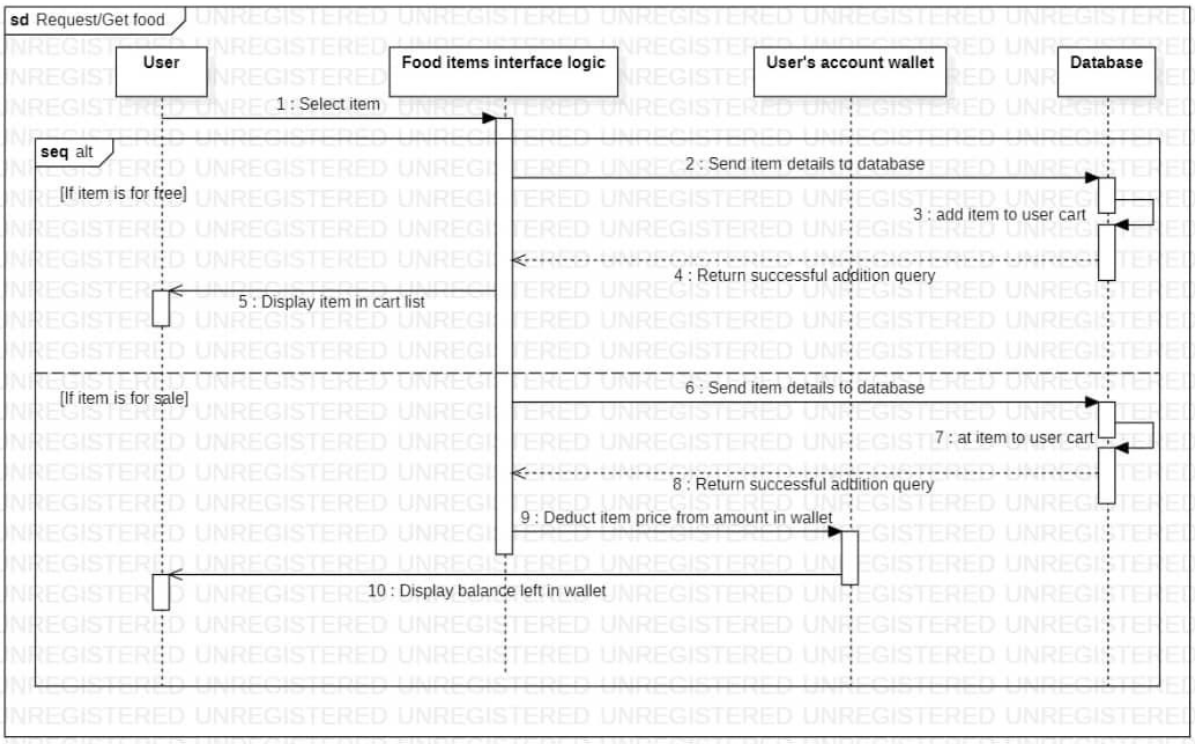

Enter Amount
select Deposit method
Mobile Payment
Bank transfer
Deposit
Select withdrawal method
bank transfer
Mobile payment
withdraw

Sequence Diagrams

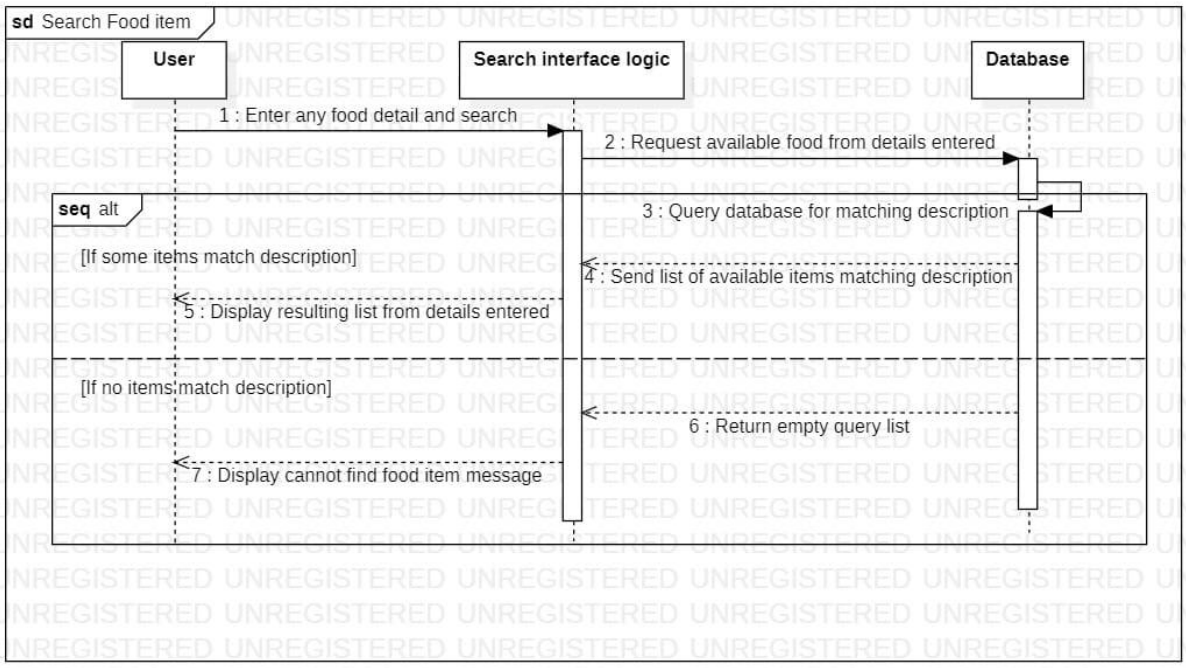
4.1 Registration



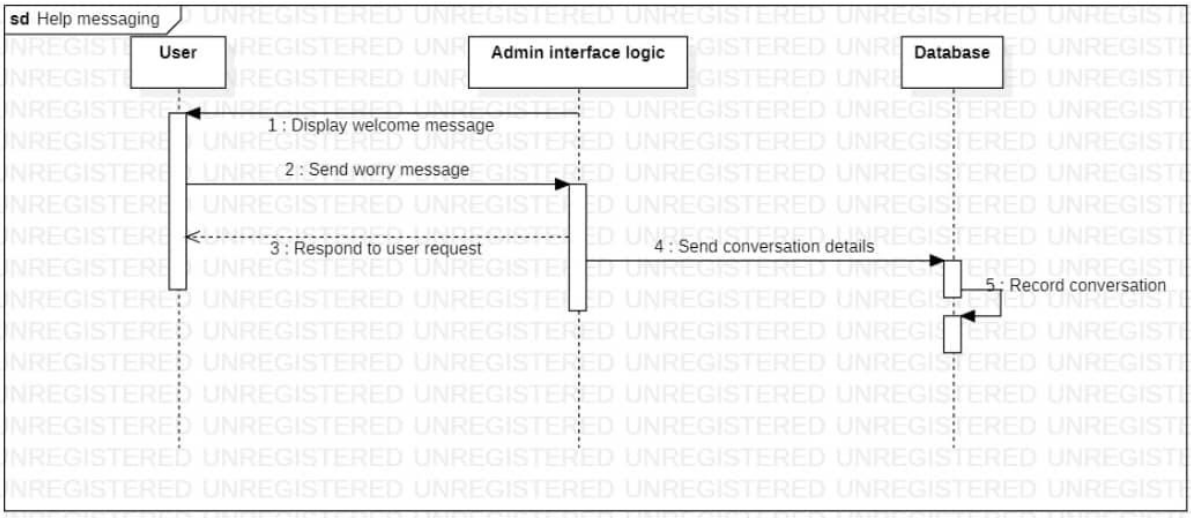
4.2 Request Food Item



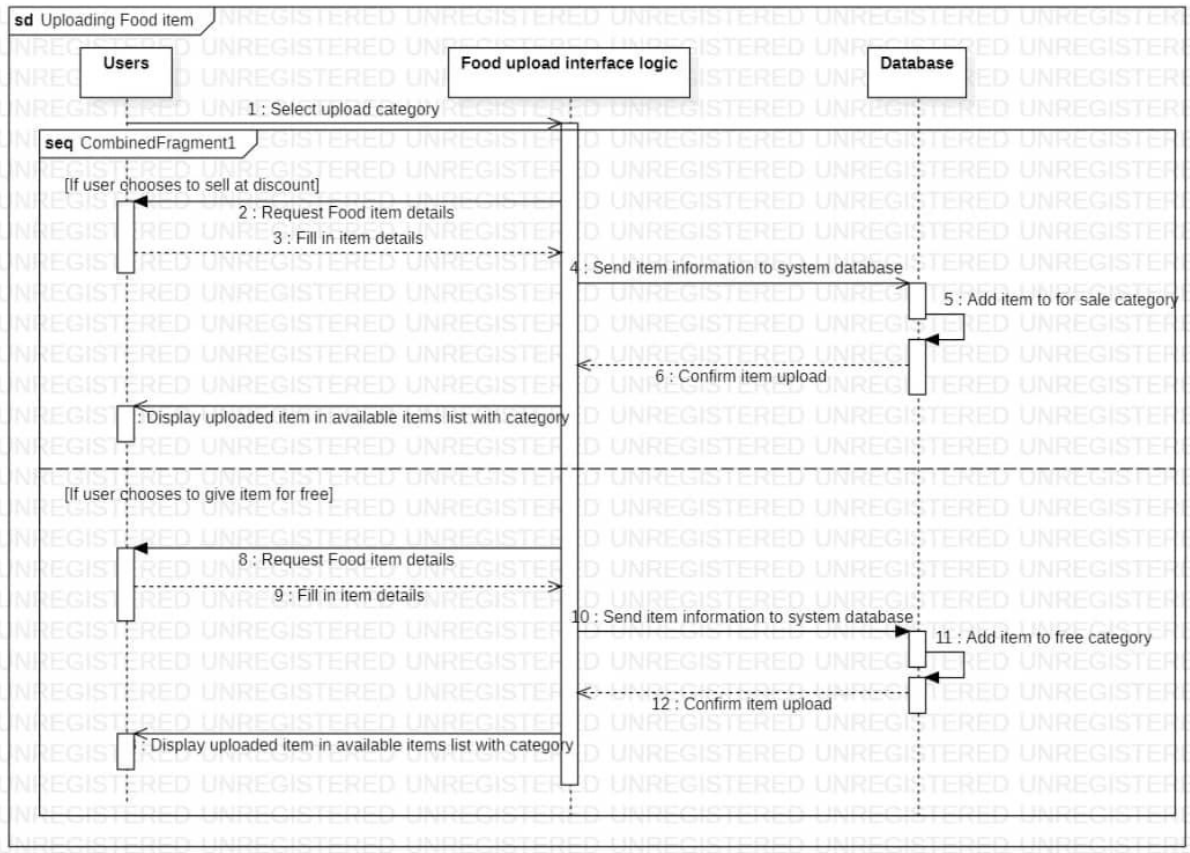
4.3 Search Food Item



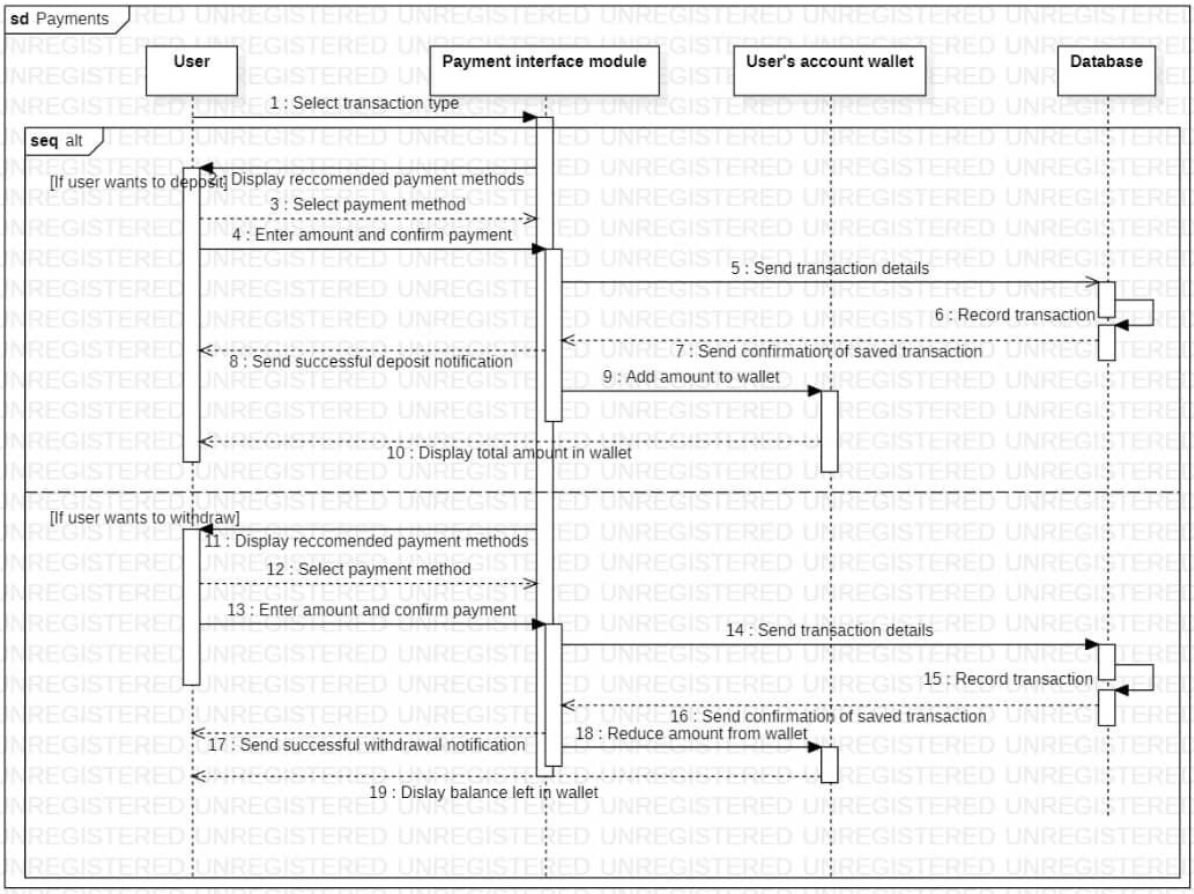
4.4 Messaging



4.5 Upload Food Item



4.6 Payment


sd Payments
User
Payment interface module
User's account wallet
Database
1 : Select transaction type
seq alt
[If user wants to deposit]
2 : Display reccomended payment methods
3 : Select payment method
4 : Enter amount and confirm payment
5 : Send transaction details
6 : Record transaction
7 : Send confirmation of saved transaction
8 : Send successful deposit notification
9 : Add amount to wallet
10 : Display total amount in wallet
[If user wants to withdraw]
11 : Display reccomended payment methods
12 : Select payment method
13 : Enter amount and confirm payment
14 : Send transaction details
15 : Record transaction
16 : Send confirmation of saved transaction
17 : Send successful withdrawal notification
18 : Reduce amount from wallet
19 : Dislay balance left in wallet

Class Diagram

5.1 Diagram