***Final Project:***

***Food Donation Management System***

***Course:***

***Software Design and Architecture***

***Members:***

***Salman Ahmed Khan (19k-1043)***

***Sadeem Sattar (19k-1102)***

***Abdullah Tilal Khan (19k-1103)***

**Dated: 22-May-2021**

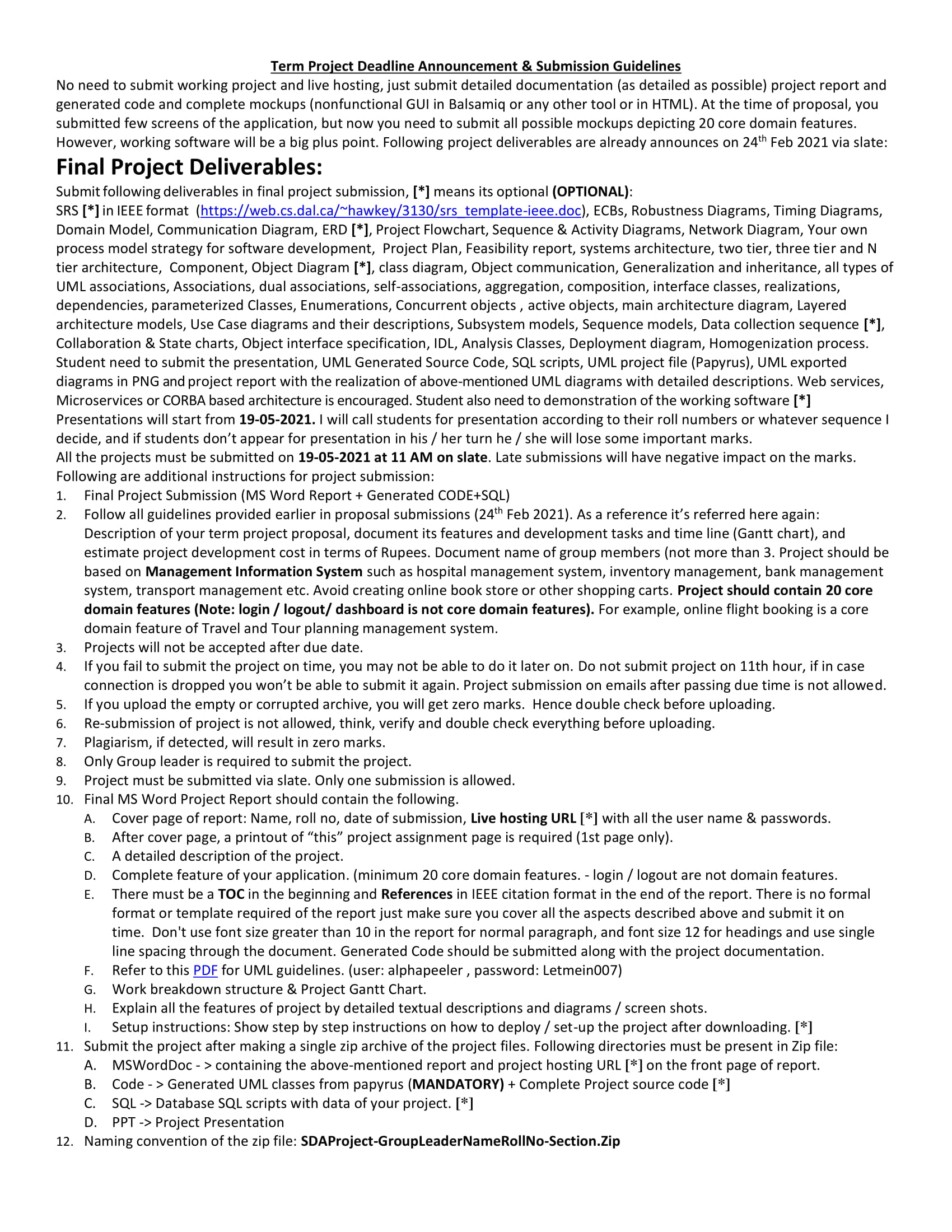
****

Table of Contents

[Project Plan 4](#_Toc72613157)

[Process Model Strategy 5](#_Toc72613158)

[Gant Chart 6](#_Toc72613159)

[Use Case Diagram 7](#_Toc72613160)

[Domain Model 11](#_Toc72613161)

[Class Diagram 12](#_Toc72613162)

[N-TIER Architecture Diagram: 13](#_Toc72613163)

[Activity Diagram 14](#_Toc72613164)

[Entity Control Boundary (ECB) 18](#_Toc72613165)

[Robustness Diagram 19](#_Toc72613166)

[Sequence Diagram 23](#_Toc72613167)

[Collaboration Diagram 27](#_Toc72613168)

[Timing Diagram 31](#_Toc72613169)

[StateChart Diagram 32](#_Toc72613170)

[ERD Diagram 34](#_Toc72613171)

[Communication Diagram 35](#_Toc72613172)

[Component Diagram 39](#_Toc72613173)

[Deployment Diagram 40](#_Toc72613174)

[Network Diagram 41](#_Toc72613175)

[MockUp Images 42](#_Toc72613176)

# Project Plan

Food Donation Management System

**Description**

The project Food Donation Management System emphasis on the saving of food in the form of donations and reduce the food waste. The project focuses on three types of the users which include Donor, Admin and NGOS. The Donor will be able to register with the system in order to publish the details of the donation. The NGOS will be required to register with the system to search for the required donation and receive it. The Admin of the system will determine the authentication of the donor and receiver according to the given registration details and control the management of the system.

**Project Core Features:**

The Project Core features are listed below:

**Donor of the System:**

1. Add food item description
2. Update food item description
3. View food item description
4. Delete food item description
5. View receipt
6. Add review to the receiver (NGO)
7. View ranking
8. View donation history
9. Add Profile
10. Update Profile
11. Submit order pickup confirmation of the receiver (NGO)
12. Submit registration details

**Receiver of the System:**

1. Search available donation
2. View the ranking of donor
3. Book the food
4. Add review to donor
5. View receipt
6. Add profile
7. Update profile
8. Submit order pickup confirmation of the donor
9. Submit registration details

**Admin of the System:**

1. Verify the details of donor
2. Verify the details of receiver (NGO)
3. Generate the receipt
4. Verify the receipts (for pickup)
5. Calculate per average
6. Update Accounts
7. Delete Accounts

**Project Cost Estimation:**

Developer Cost = Rs. 1000 per hour

Each day: 2 hours of working

Per day calculation: 1000\*2\*3 = Rs. 6000

# Process Model Strategy

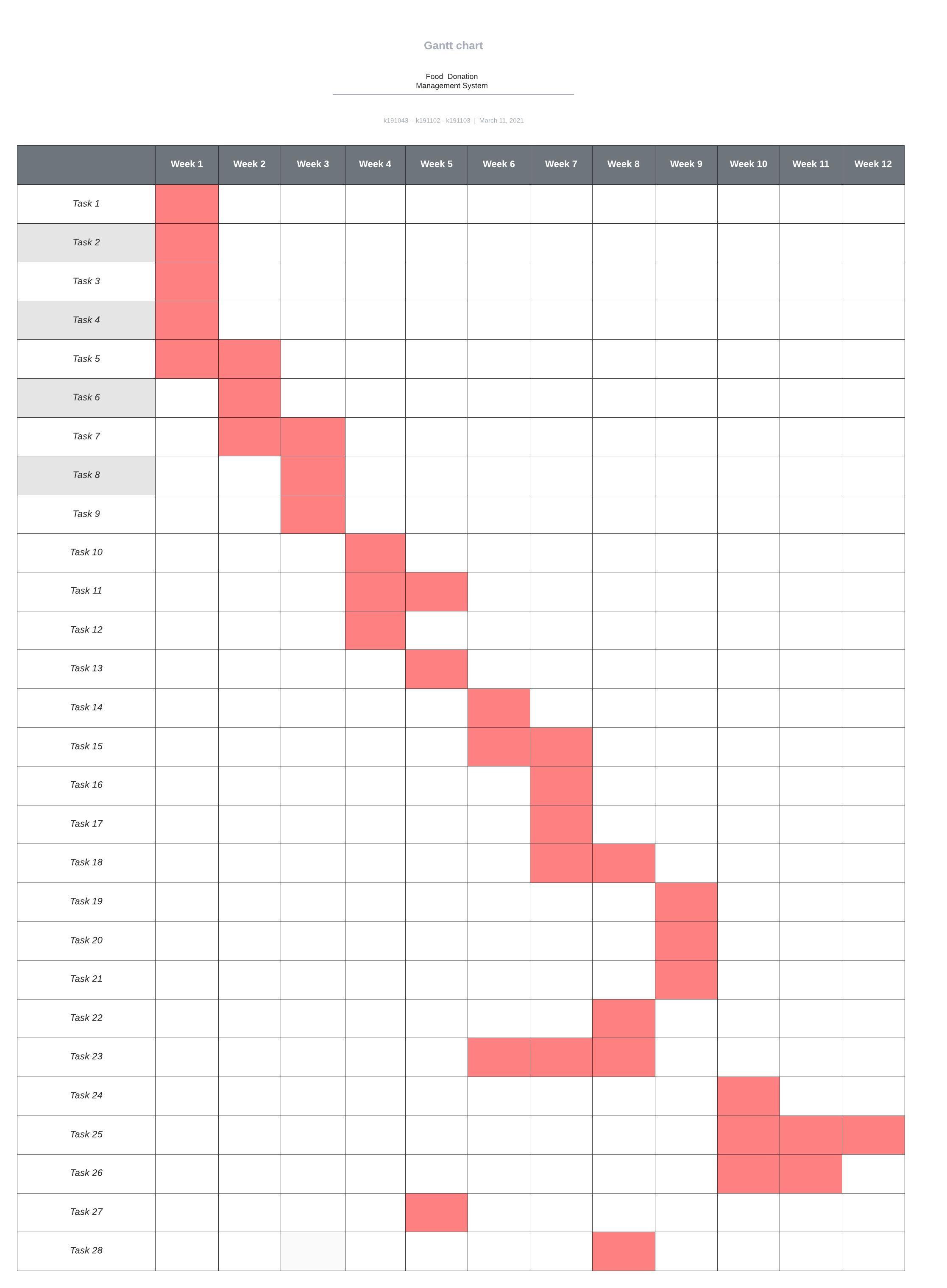
In this food donation management system we will use unified process model in our software development life cycle.

The system will be develop using Agile Development methodologies using Extreme Programming framework which is the subset of unified process model.

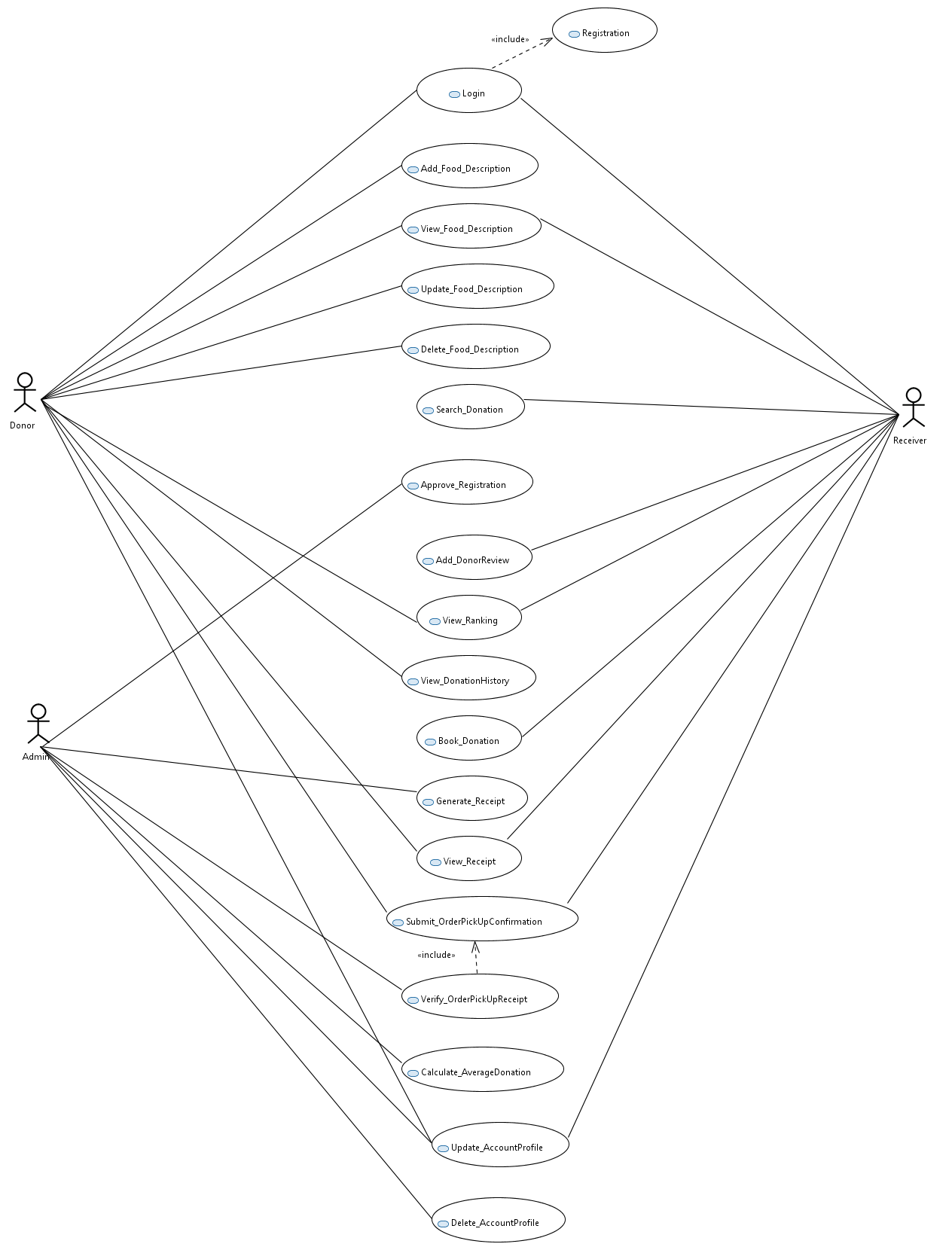
**The advantages of using Extreme Programming**

* Better for small projects.
* Stable development with continues testing
* Good for the projects with rapidly changing requirements.
* More reliable
* It minimizes time and save cost.

# Gant Chart



# Use Case Diagram



**Fully Dressed Used Case**

**Use Case 1. Login**.

* **Primary Actor:**Admin/ Receiver/ Donor.
* **Description:**This use case triggers when user want to enter into the system.  
  **Pre-condition:**User must register first.

**Flow of Events:**

* 1. User visit site add name and password.
  2. Submit it after fill login form.

**Post-condition:**User login successful.  
**Assumption:**User understand system language (English).

**Use Case 2. Registration**.

* **Primary Actor:** Receiver/ Donor.
* **Description:**This use case triggers when user want to register into the system.  
  **Pre-condition:**User must have internet connection.

**Flow of Events:**

* 1. User visit site fill click registration button.
  2. add name and password etc.
  3. Submit it after fill login form.

**Post-condition:**User registration is successful.  
**Assumption:**User understand system language (English).

**Use Case 3.**Update Profile.

* **Primary Actor:**Donor/Admin/Receiver.  
  **Description:**The profile might change according to the user wish. Profile contain parameters like user name, address, phone number etc.  
  **Pre-condition:**user must login into the system.
* **Flow of Events:**
  1. The userenters the system with some password.
  2. The user makes the required changes.
  3. The user saves the changes and logs out.

**Alternative Flow of Events:**

* 1. Some of the parameter might not need any change.
  2. User might enter invalid password and need to go back.

**Post-condition:**A Profile will be displayed when the user enters the system.  
**Assumption:**Admin is given the rights and privileges to enter the system and make the required changes.

**Use Case 4.**Add Food.

* **Primary Actor:**Donor.
* **Description:**This use case triggers when an item goes out of stock.  
  **Pre-condition:**User must login into the system.  
  **Flow of Events:**
  1. Donor Add food along with its description and quantity.
  2. Updates the data base accordingly.

**Post-condition:**User add food successfully.  
**Assumption:**The Admin is given the rights and privileges to enter the system and make the required changes.

**Use Case 5.**Book Food.

* **Primary Actor:**Receiver.
* **Description:**This use case triggers when request for the food.  
  **Pre-condition:**User must login into the system.  
  **Flow of Events:**
  1. Request food form appear on screen.
  2. Receiver fill request form along with its description and quantity.
  3. Submit request.
  4. Updates the data base accordingly.
  5. Wait for the approval of form.

**Post-condition:**Booking successful.   
**Assumption:**The Admin is given the rights and privileges to enter the system and make the required changes.

**Use Case 6.**View Ranking.

* **Primary Actor:**Donor.
* **Description:**This use case triggers user wanted to see its ranking.  
  **Pre-condition:**User must login into the system.  
  **Flow of Events:**
  1. User click view ranking button.
  2. Ranking Graph is shown to user.

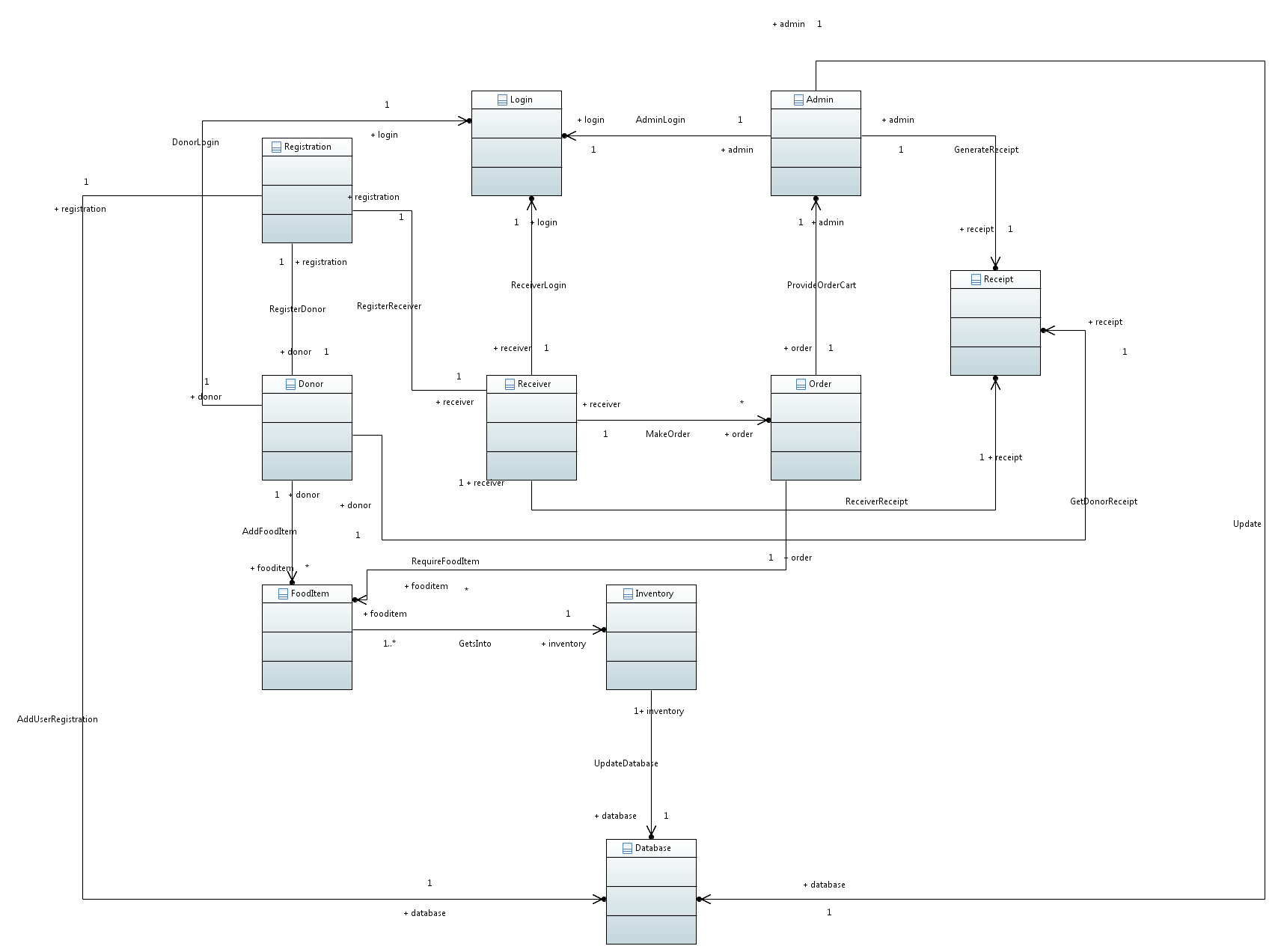
**Post-condition:**User view ranking successfully.

**Use Case 6.**Approved Registration.

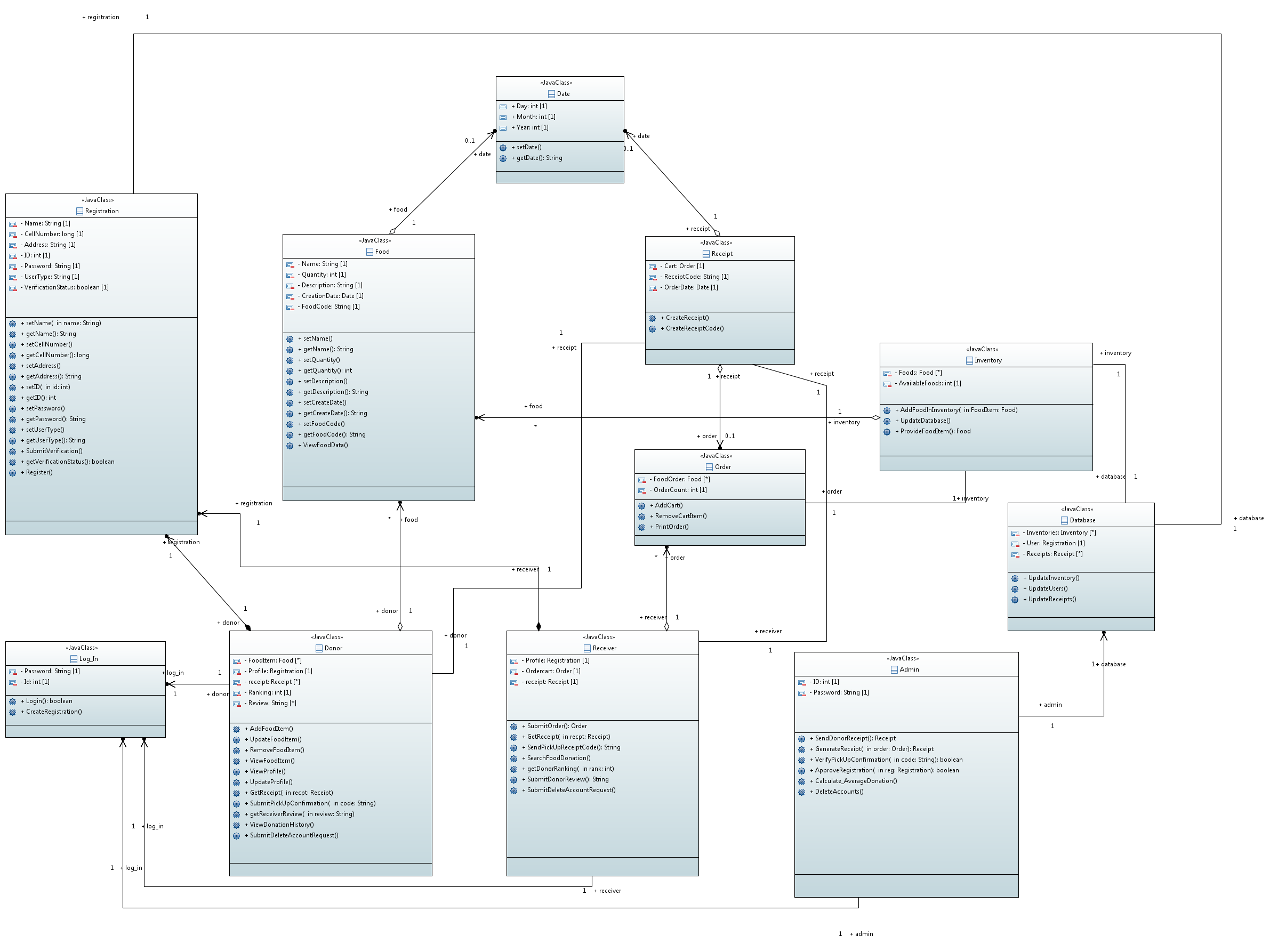
* **Primary Actor:**Admin.
* **Description:**This use case triggers when donor and receiver wanted to registration its self.  
  **Pre-condition:**Admin must login into the system.  
  **Flow of Events:**
  1. Admin click view registration list button.
  2. Check requirement and tick the text box accordingly.
  3. Save Data.
  4. Give Access to user.

**Post-condition:**User get system access successfully.

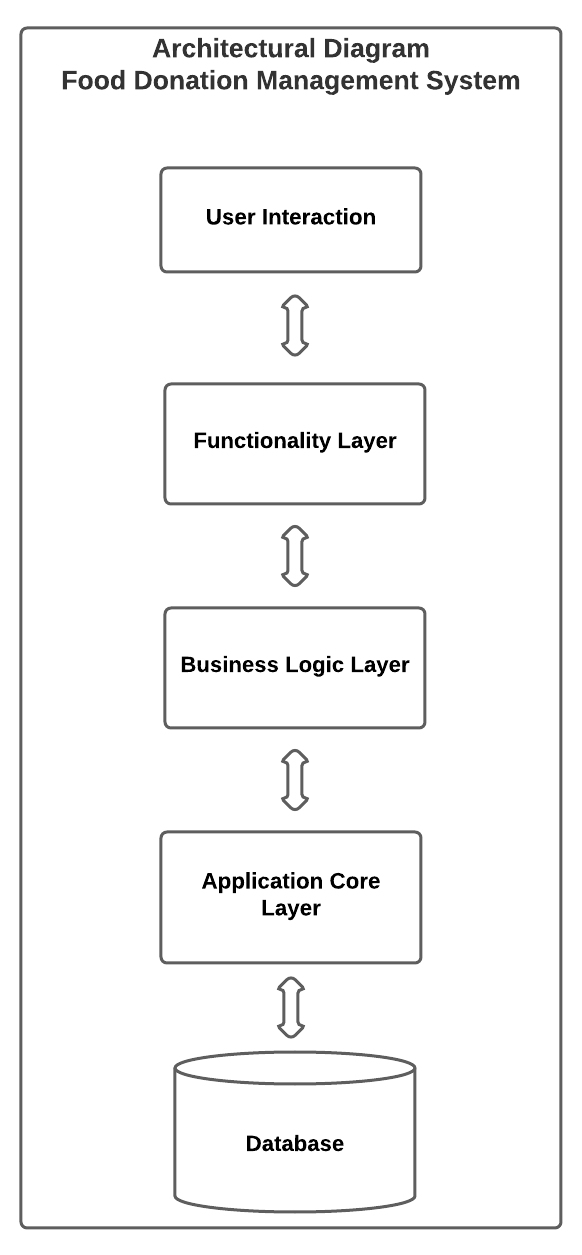
# Domain Model



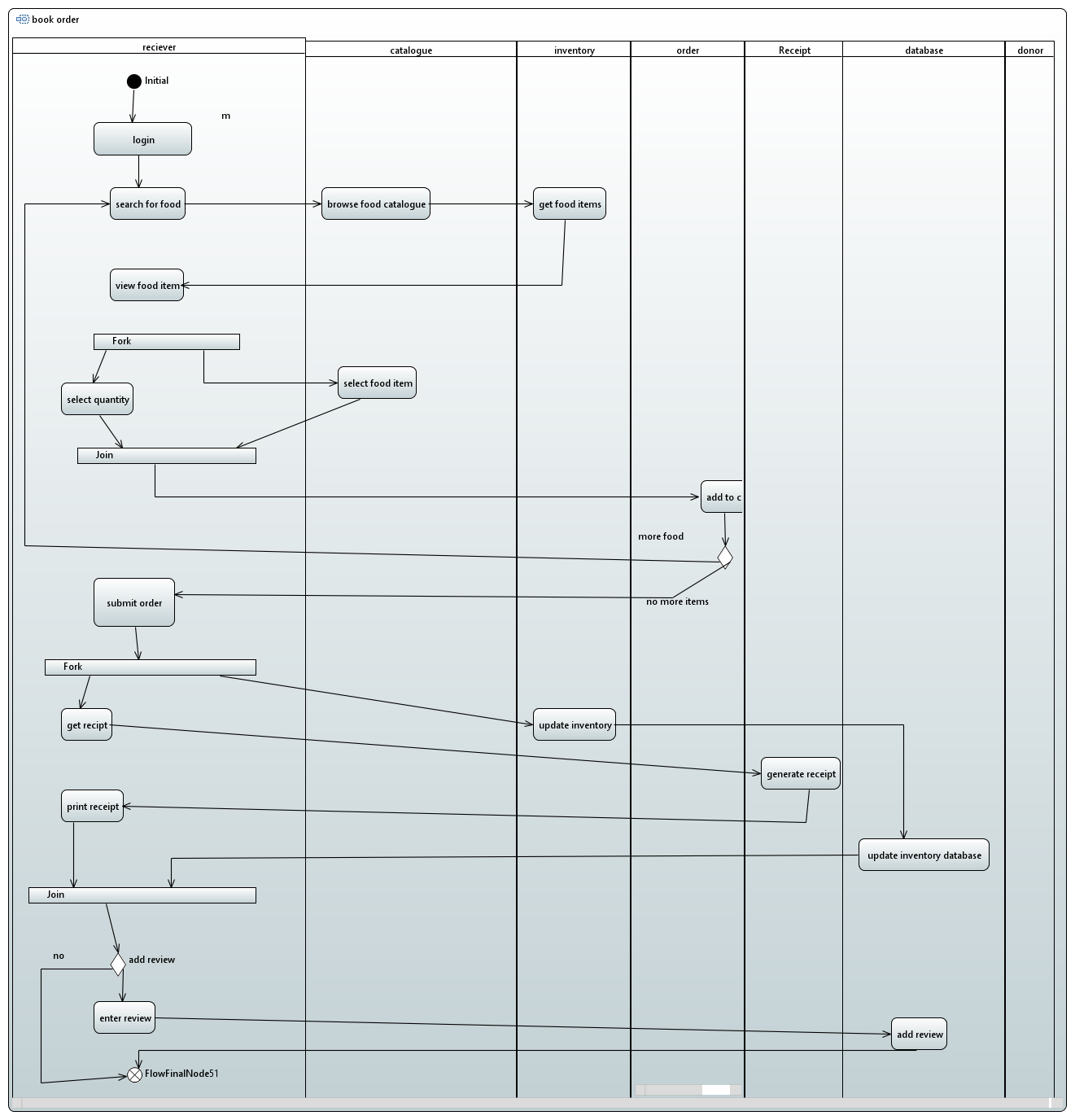
# Class Diagram

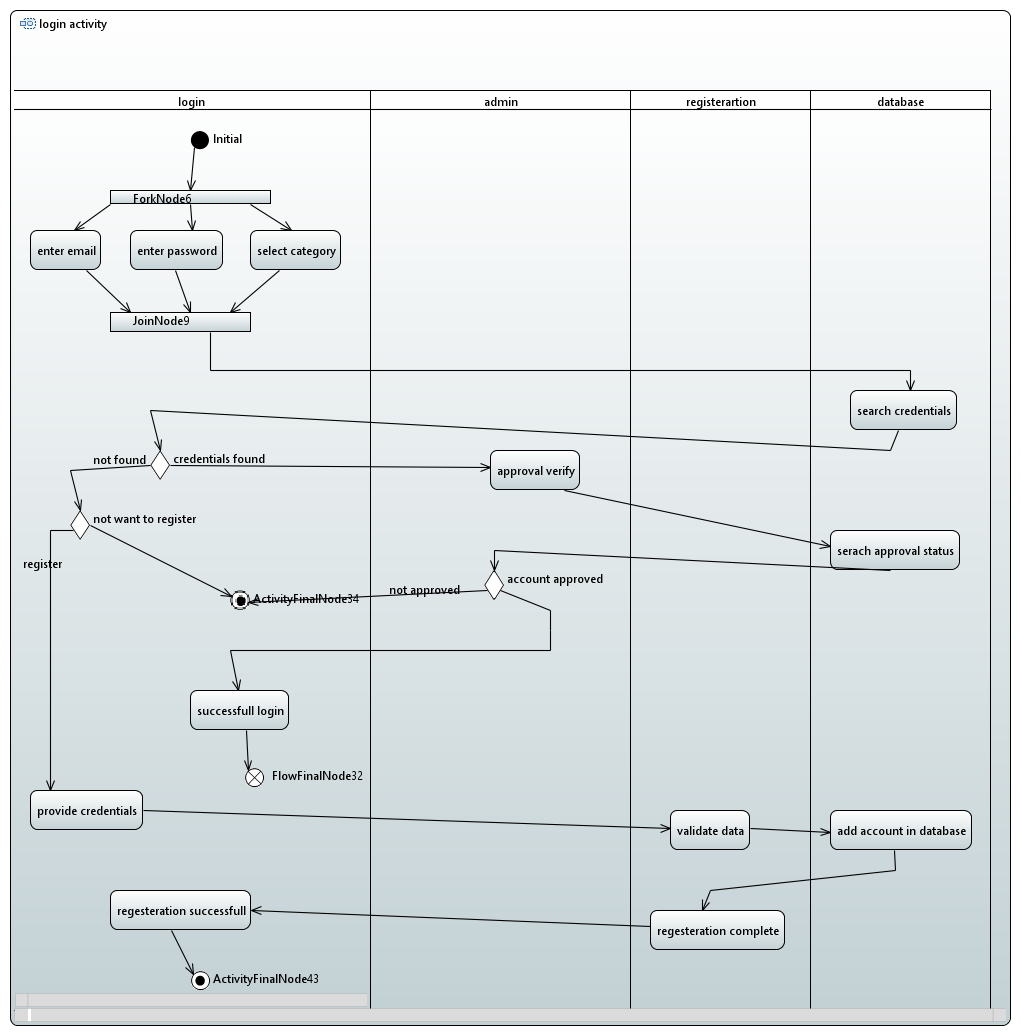


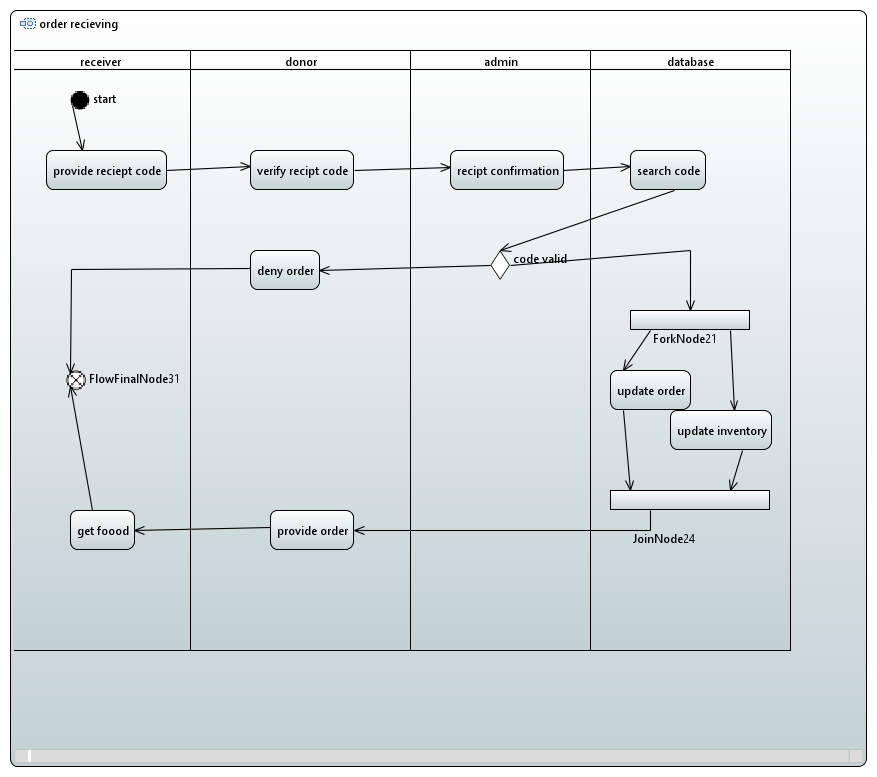
# N-TIER Architecture Diagram:

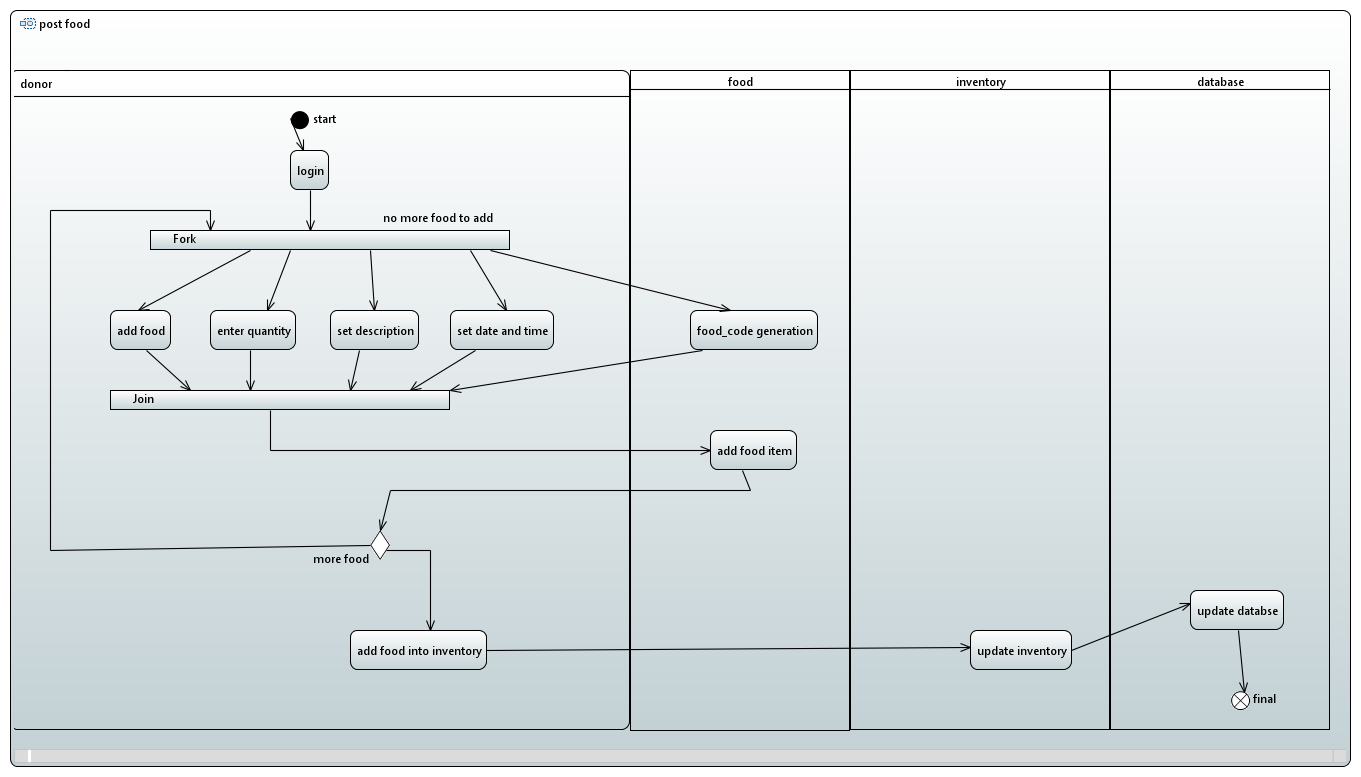


# Activity Diagram

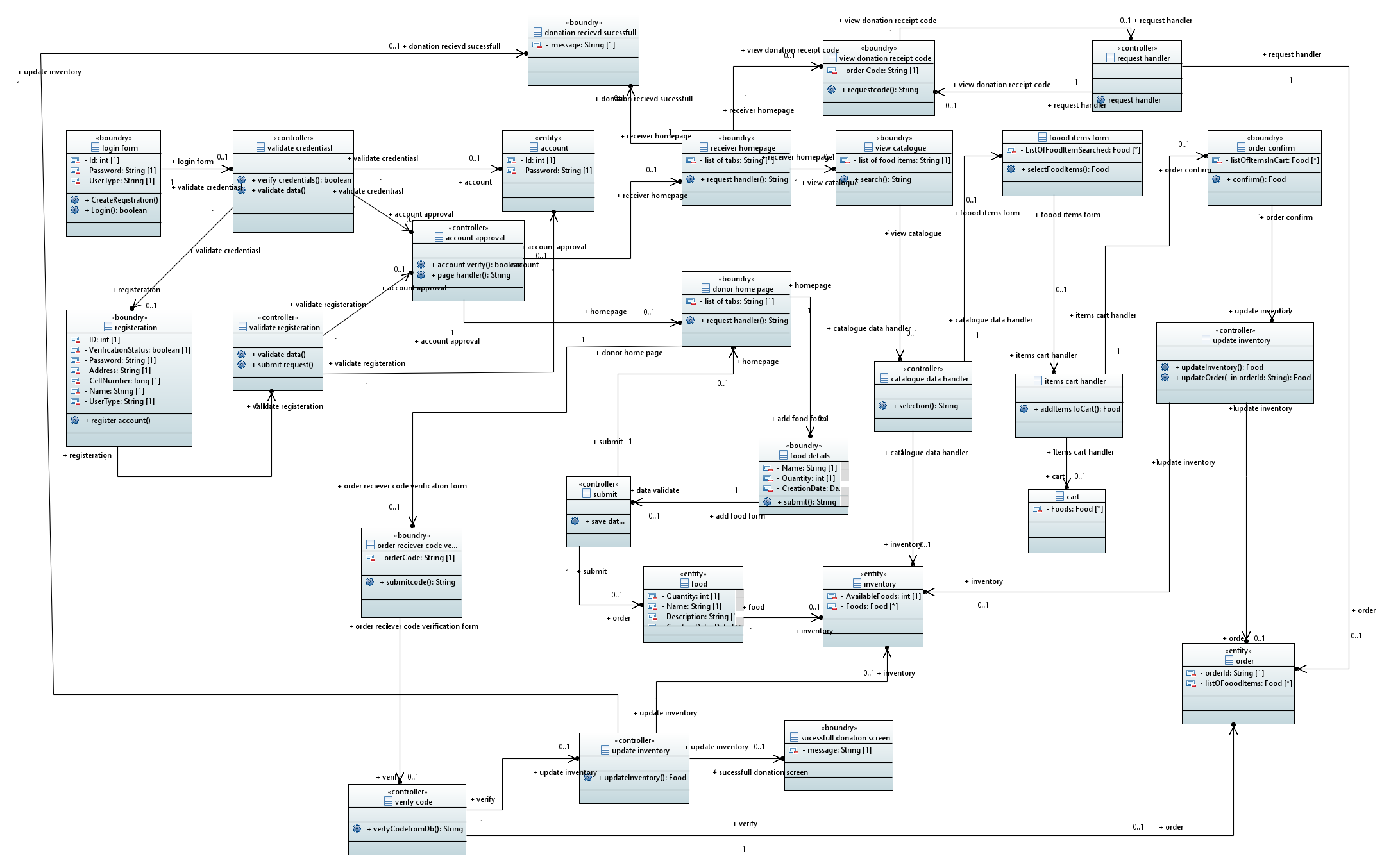
**Add food to receive donation activity**

**Login & Registration activity**

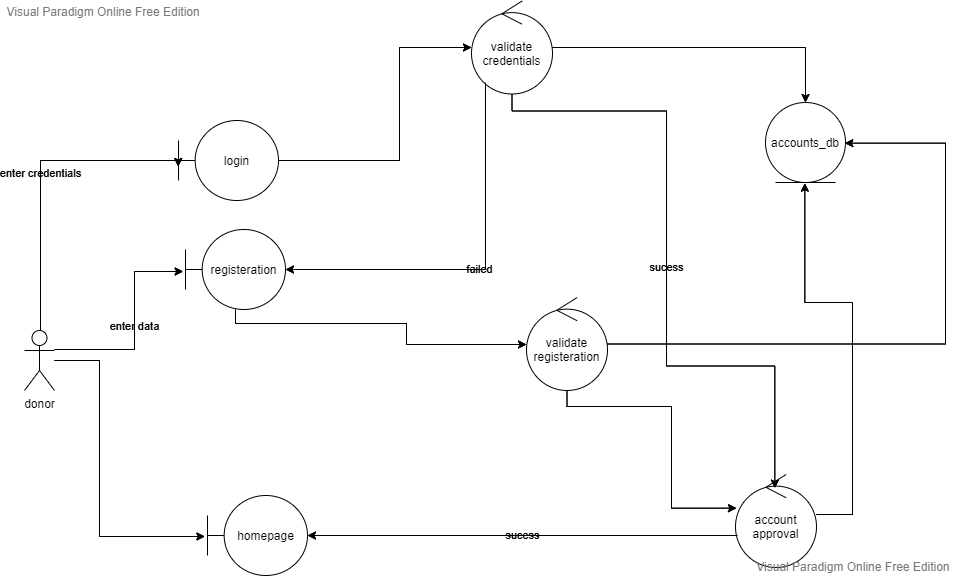
**Donation pickup/receiving activity**

**Add excess food to donate activity** 

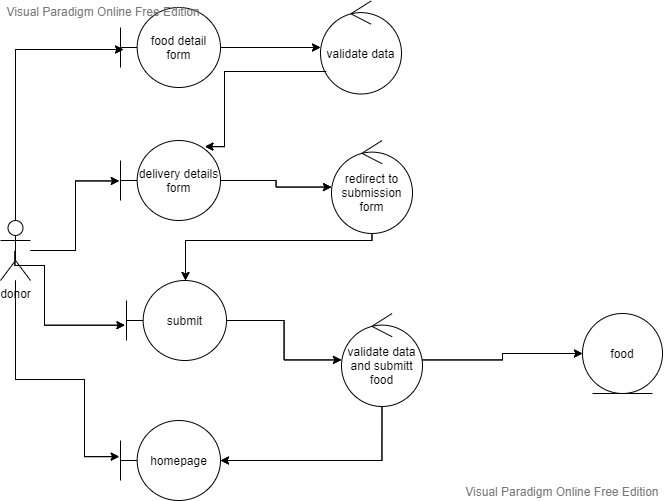
# Entity Control Boundary (ECB)

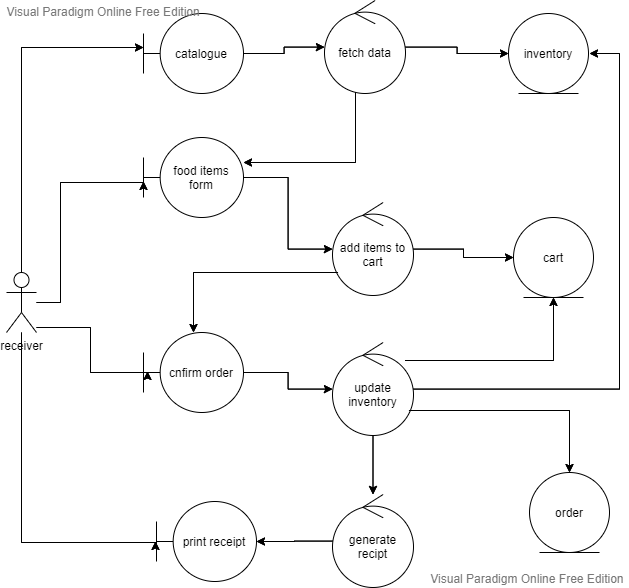


# Robustness Diagram

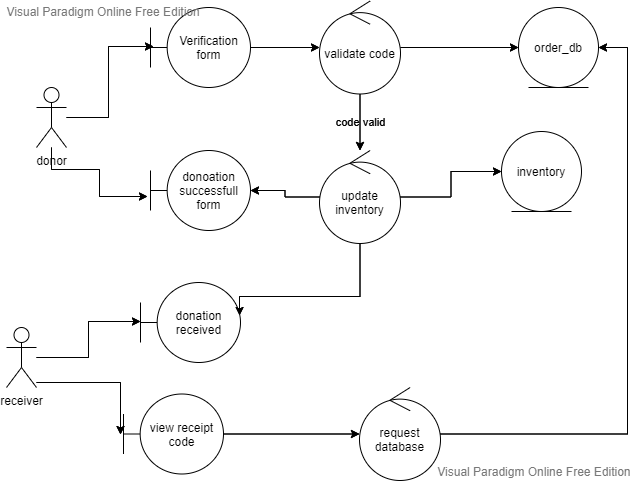
**Login-Registration**

**Post Food Item**



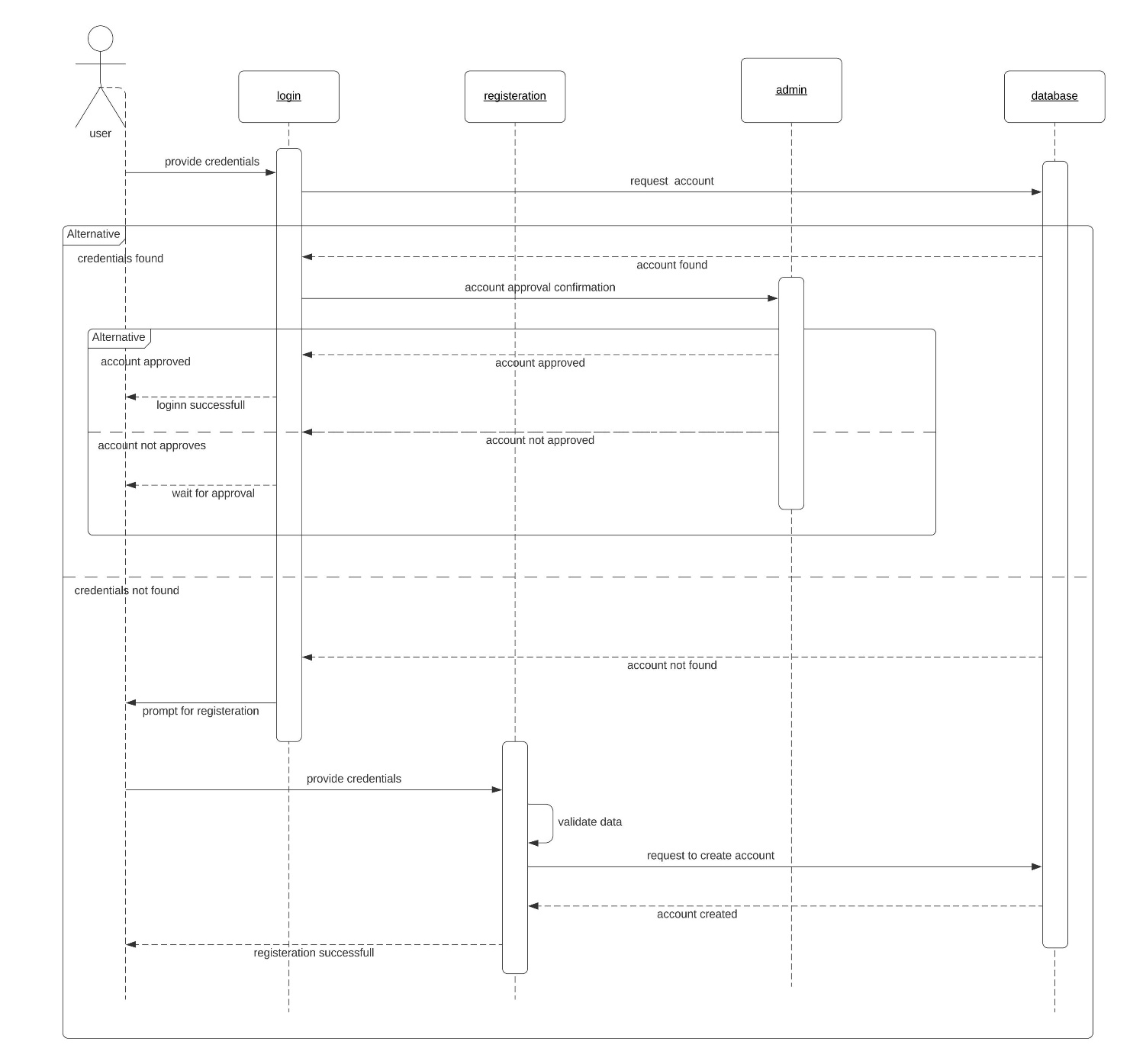
**Book Order**

**Receive Order**



# Sequence Diagram

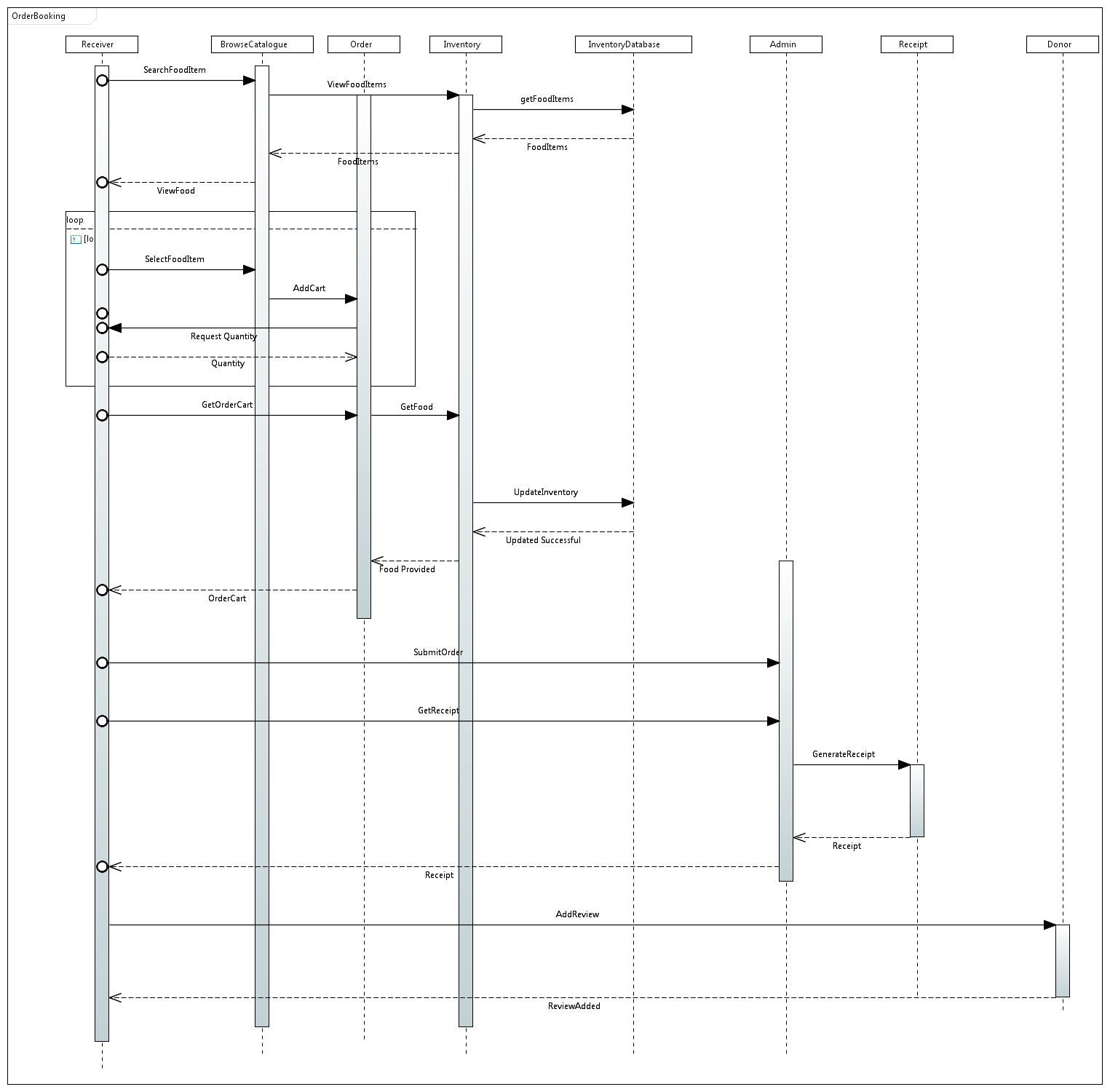
**Login-Registration**



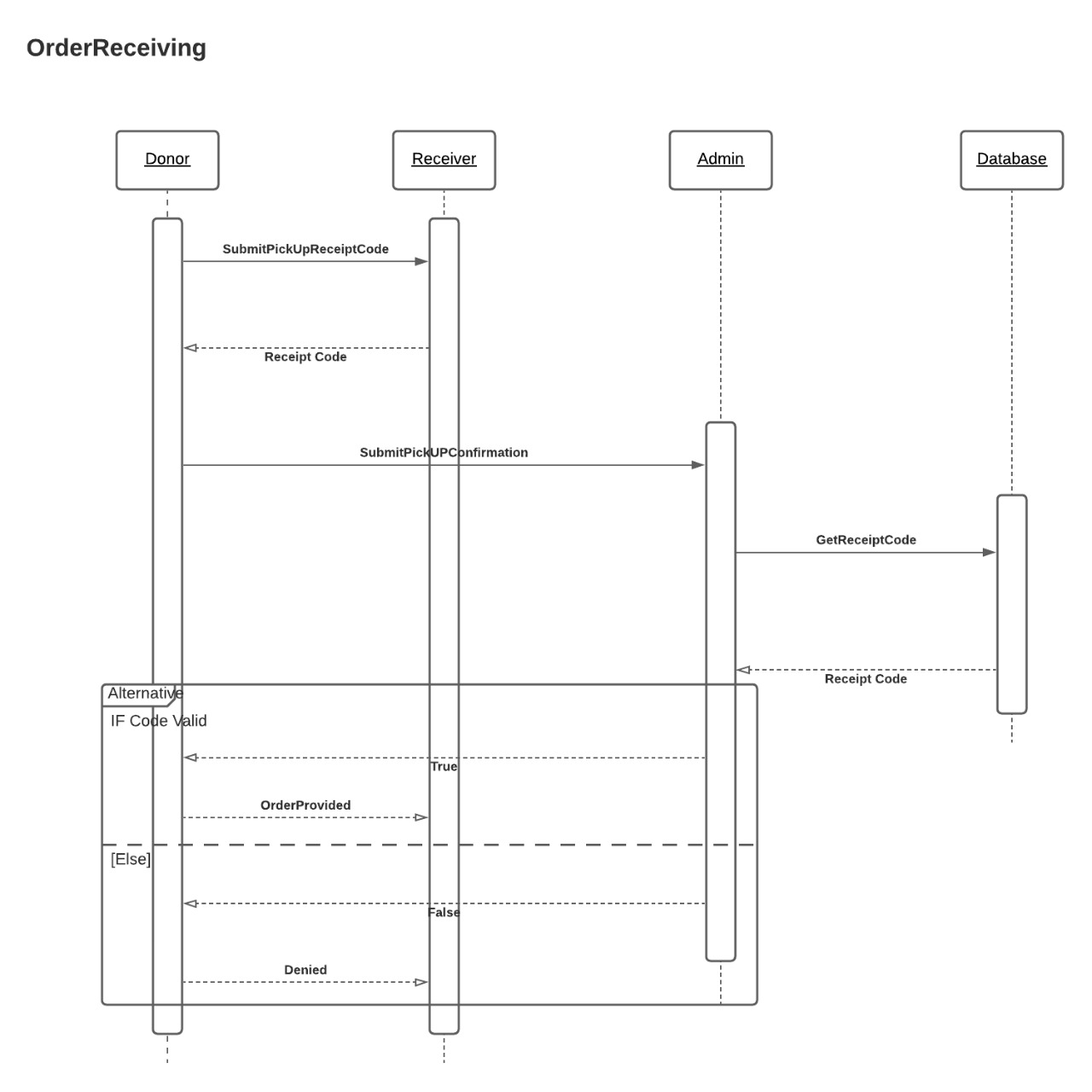
**Post Food Item**



**Book Order**

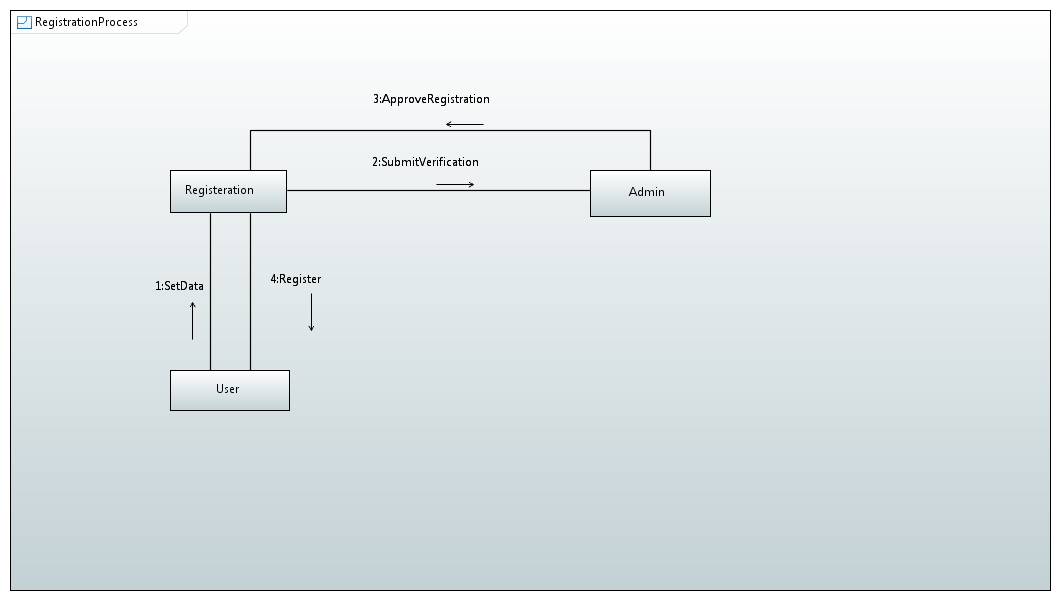


**Receive Order**

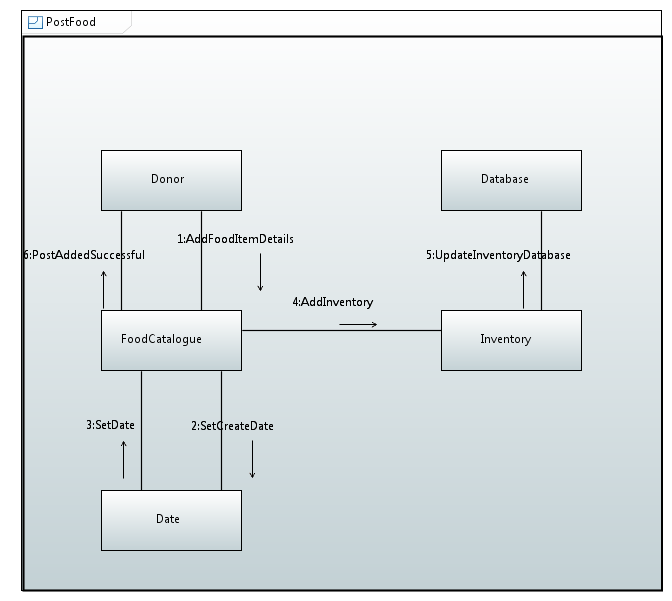


# Collaboration Diagram

**Registration**



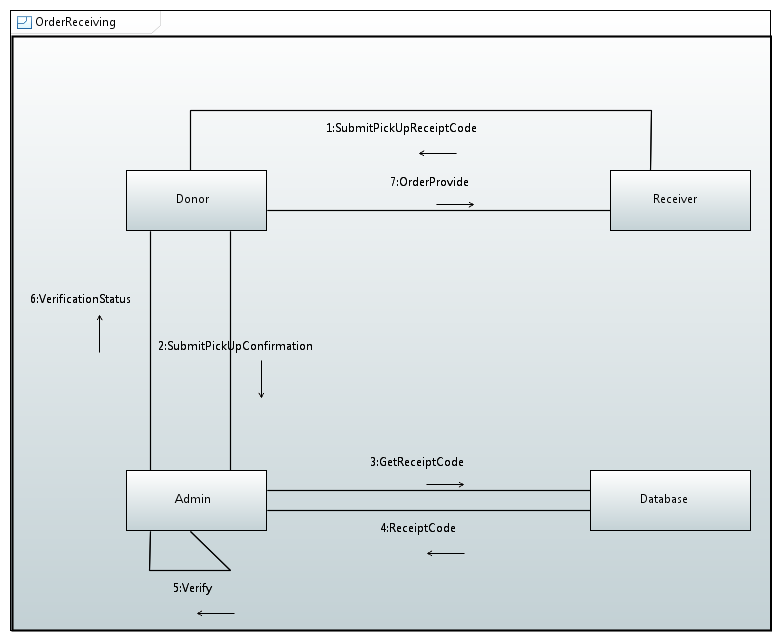
**Post Food Item**



**Book Order**

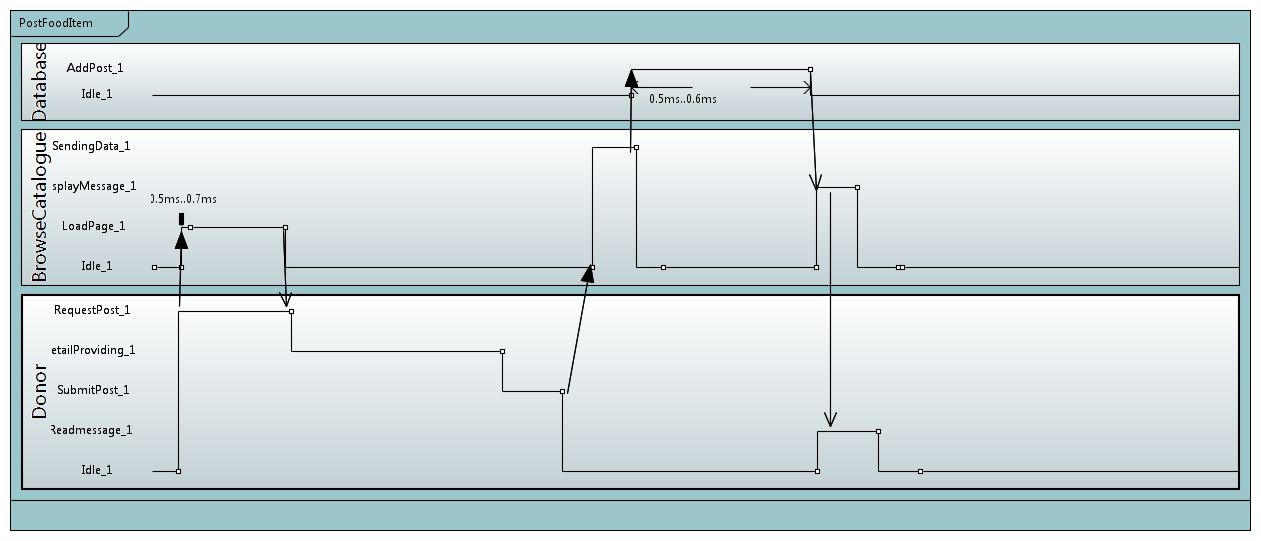


**Receive Order**

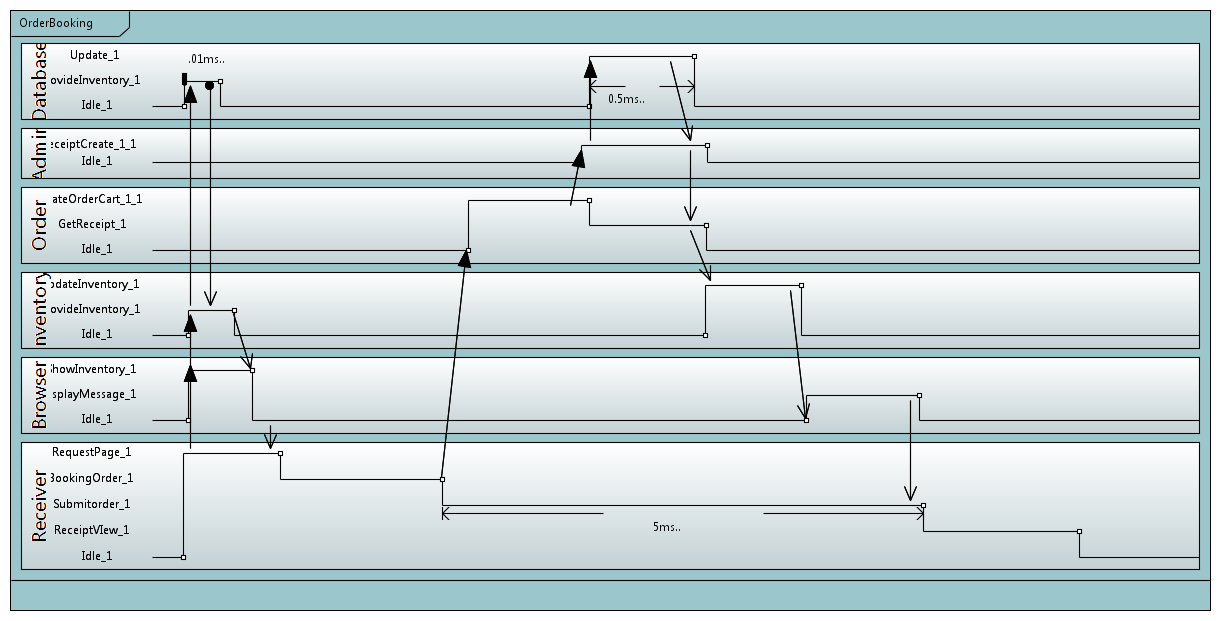


# Timing Diagram

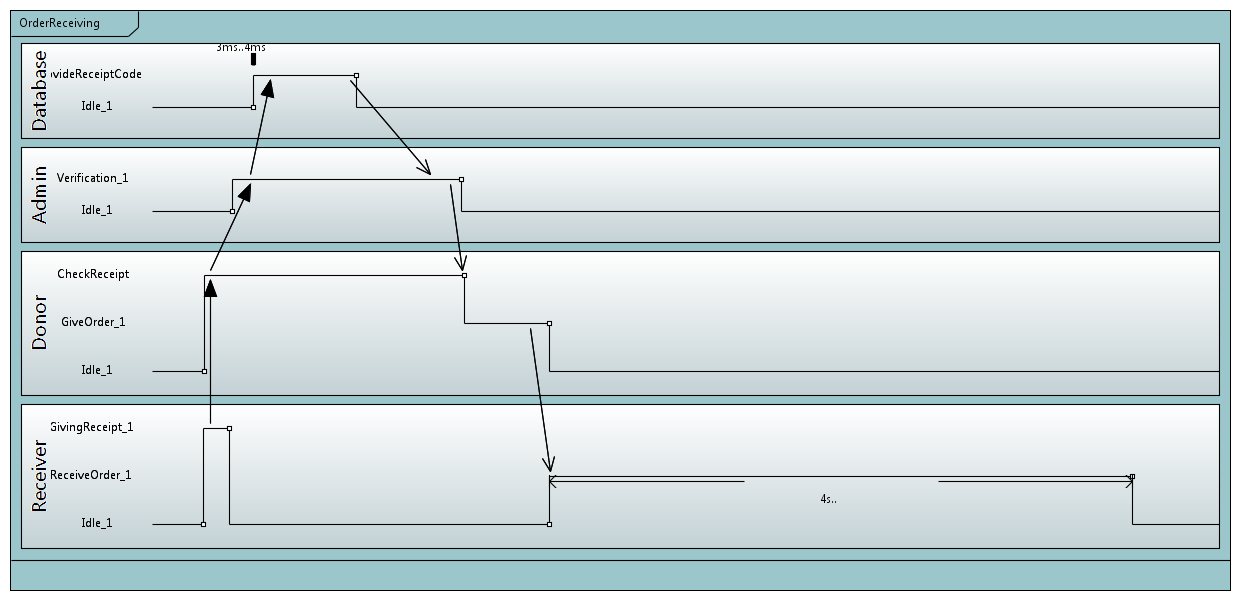
**Post Food Item**



**Book Order**

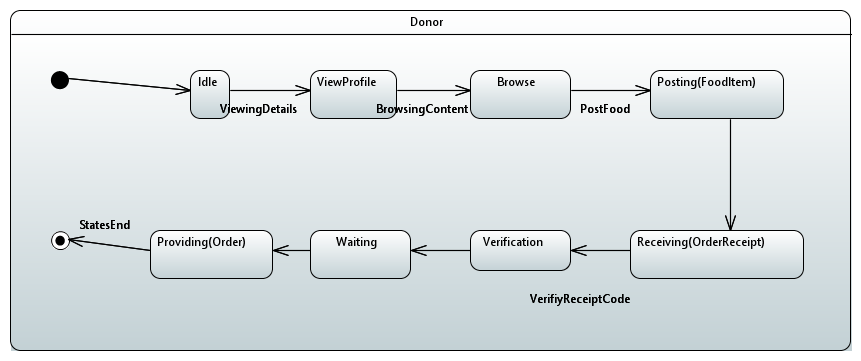


**Receive Order**

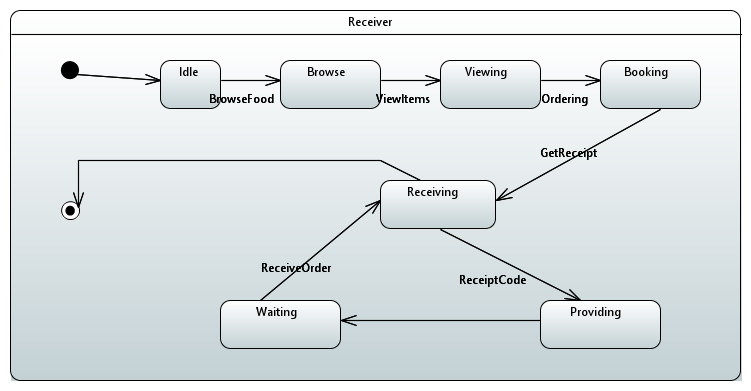


# StateChart Diagram

**Donor States**



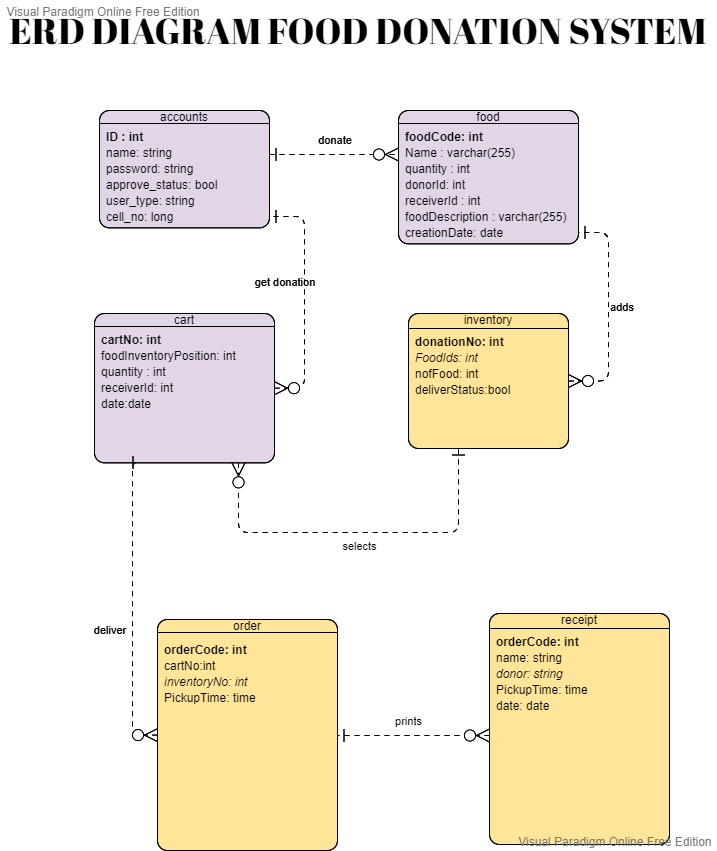
**Receiver States**



**Admin States**

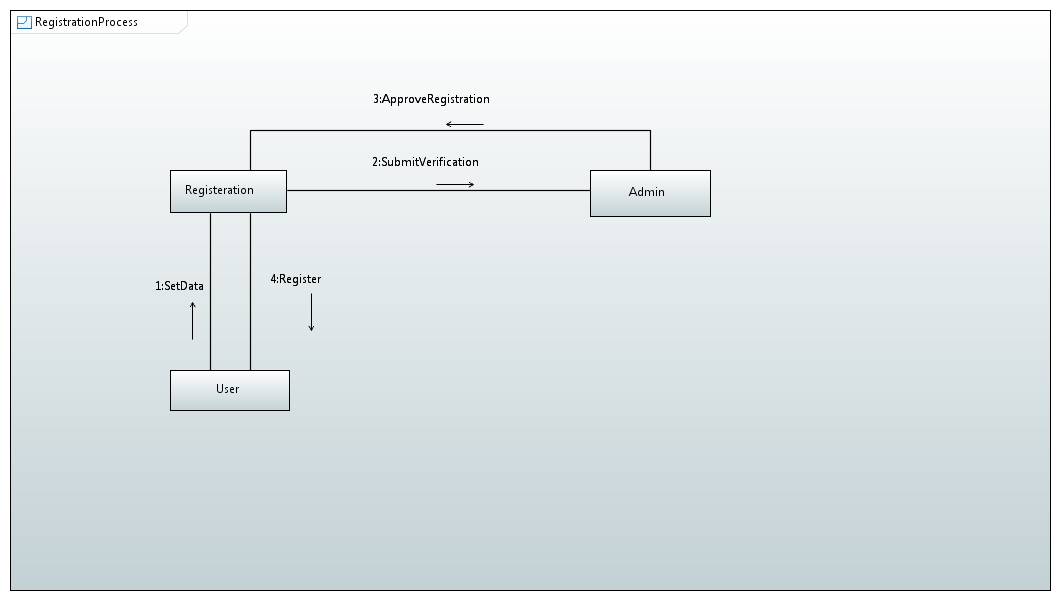


# ERD Diagram

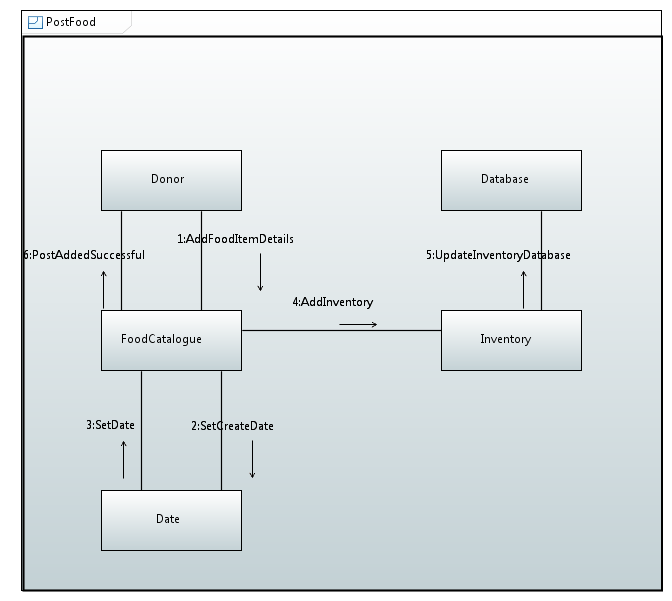


# Communication Diagram

**Registration**



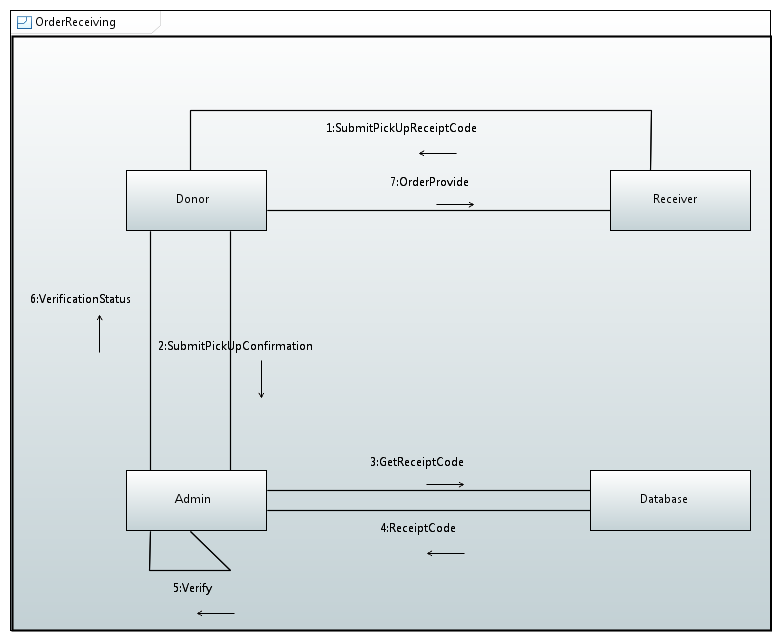
**Post Food Item**



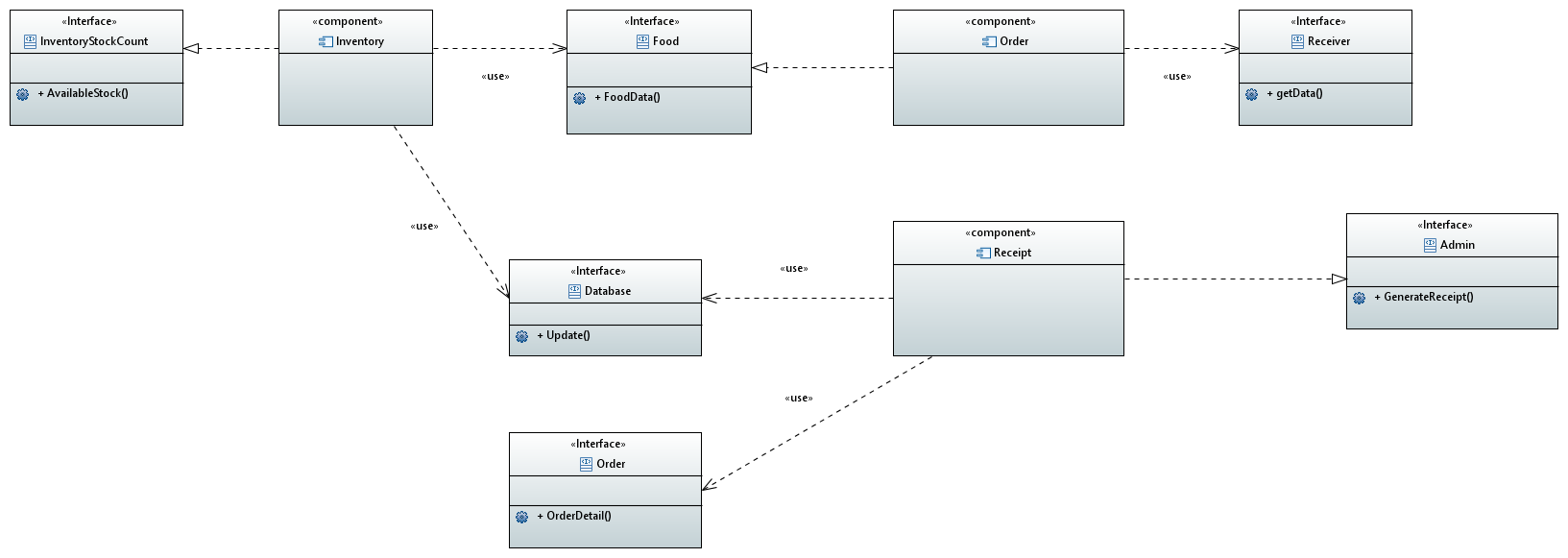
**Book Order**

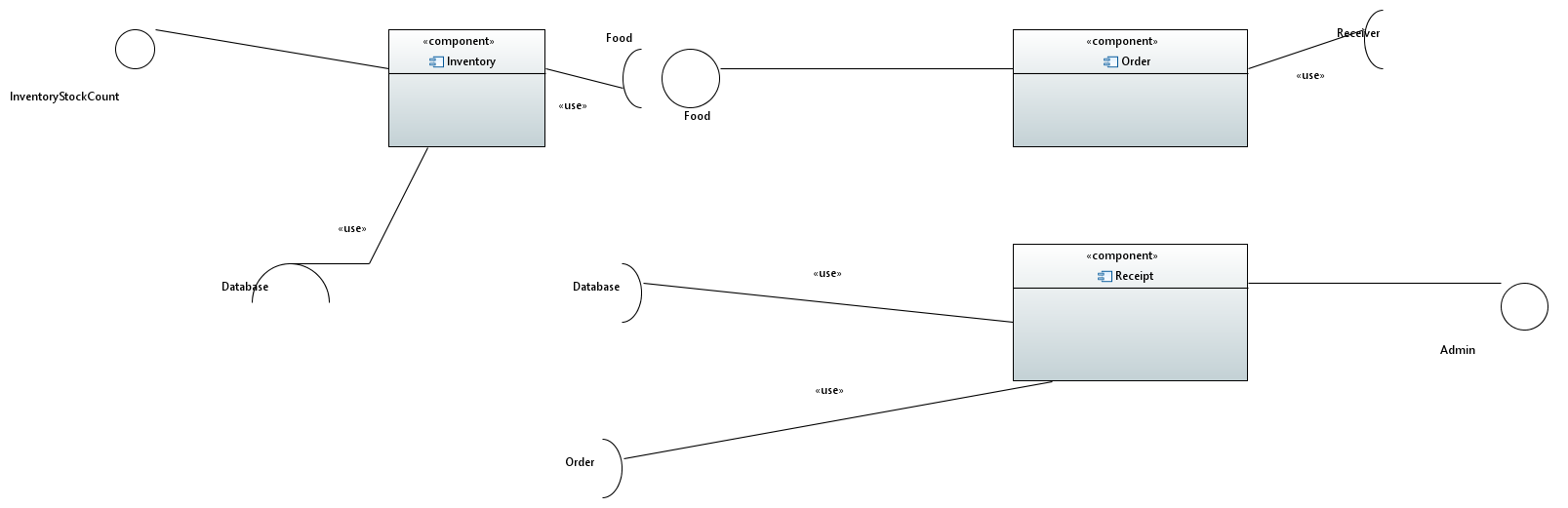


**Receive Order**

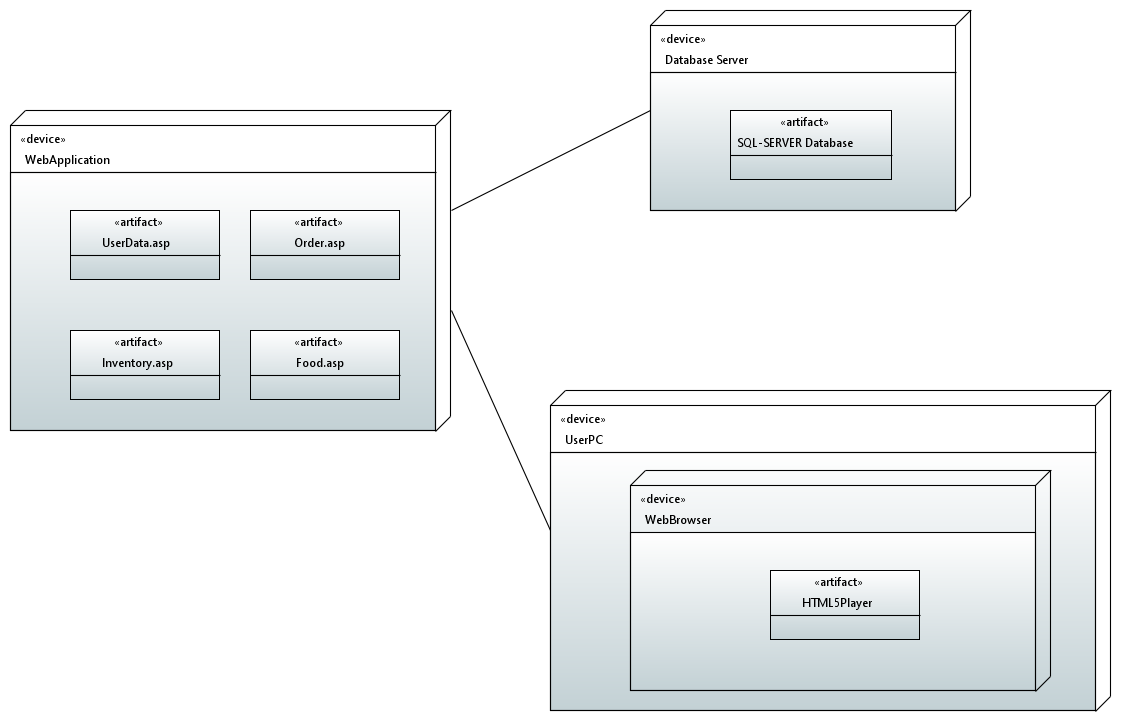


# Component Diagram

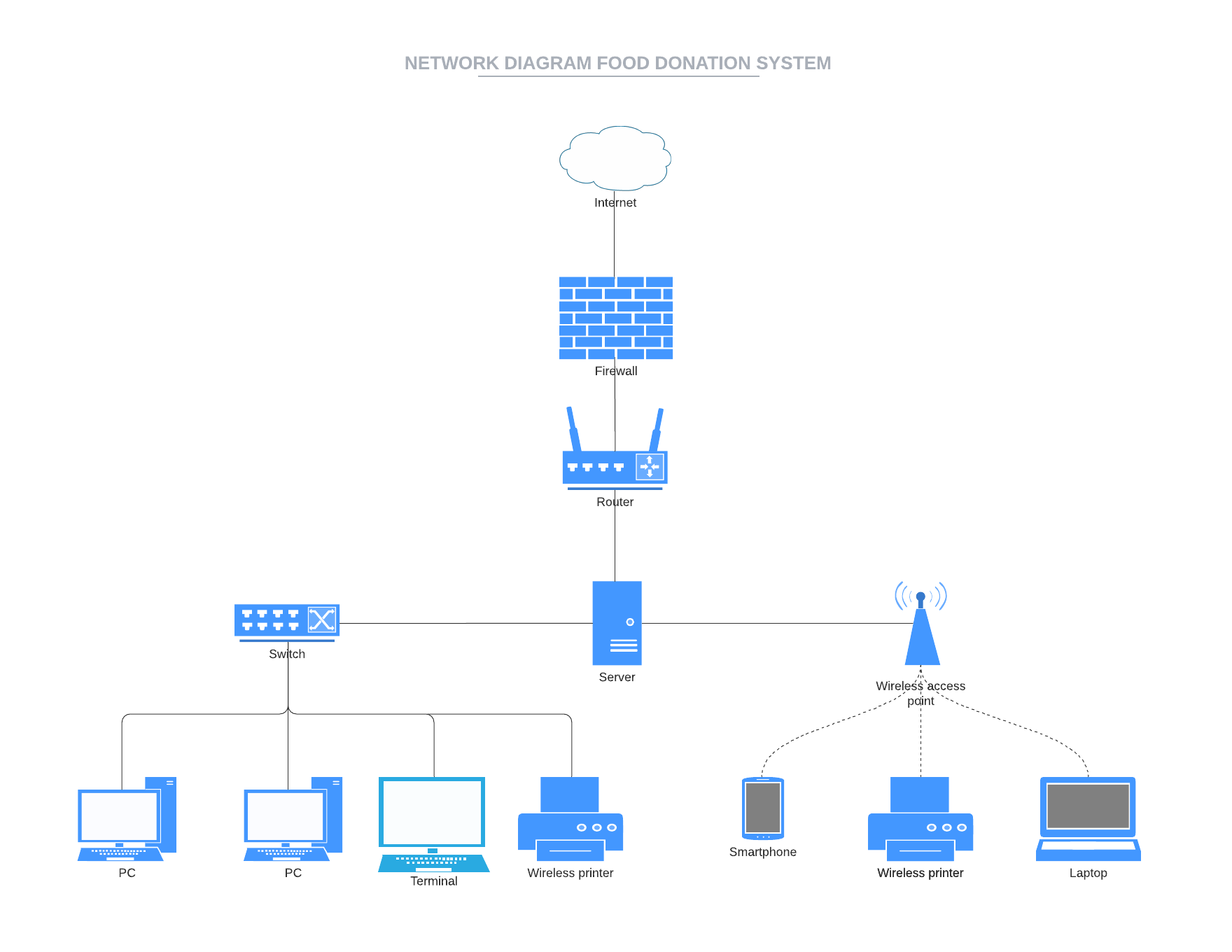




# Deployment Diagram

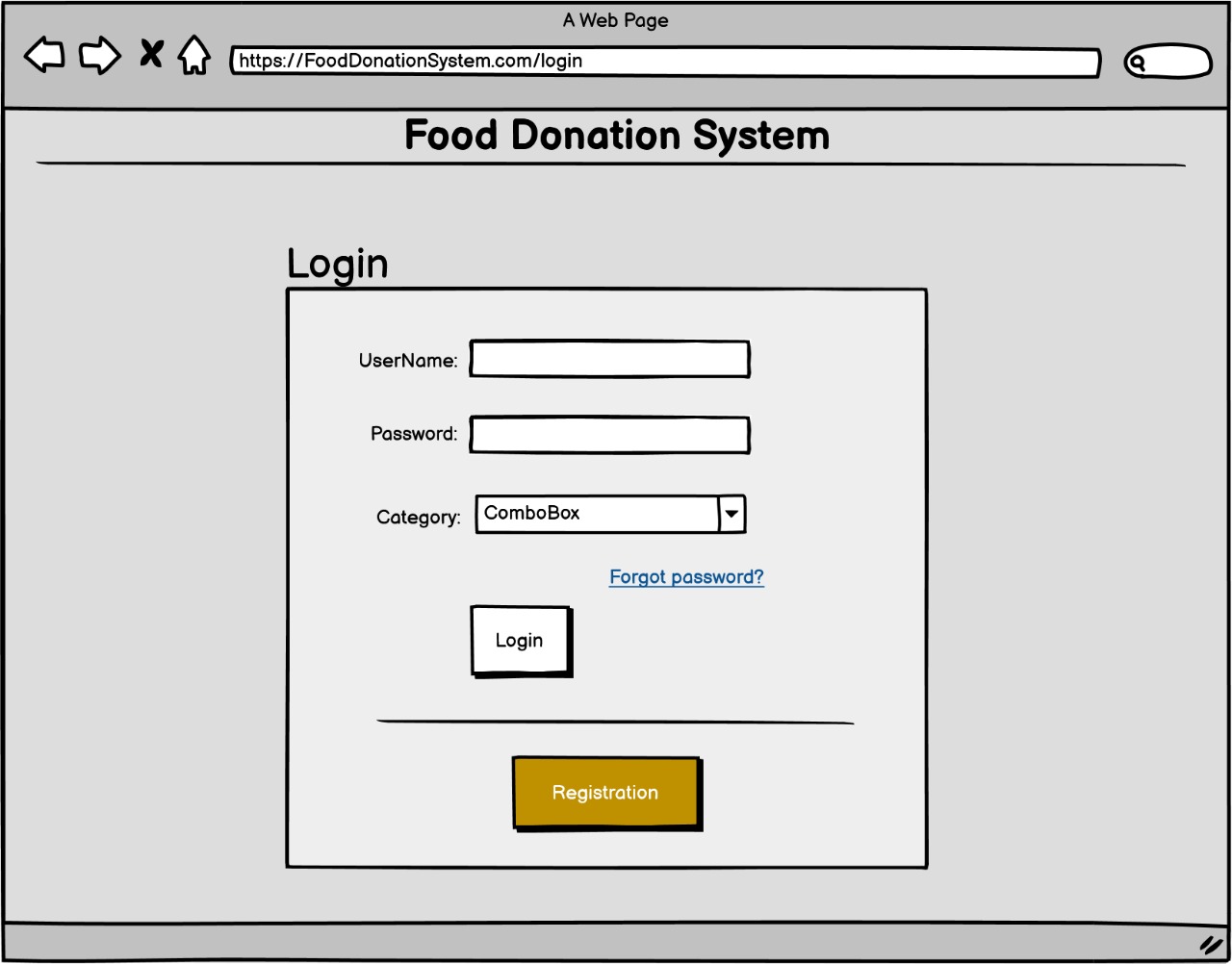


# Network Diagram

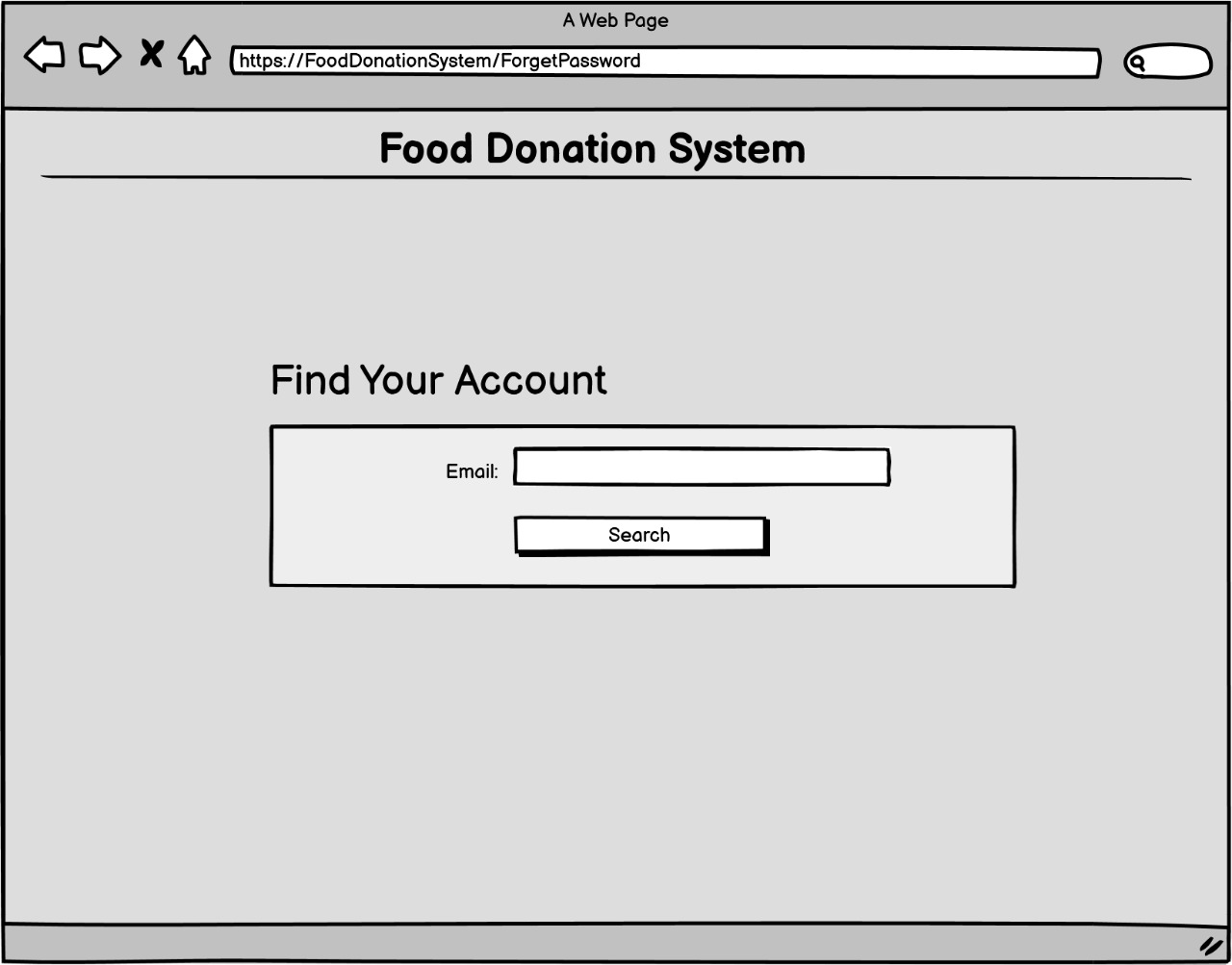


# MockUp Images

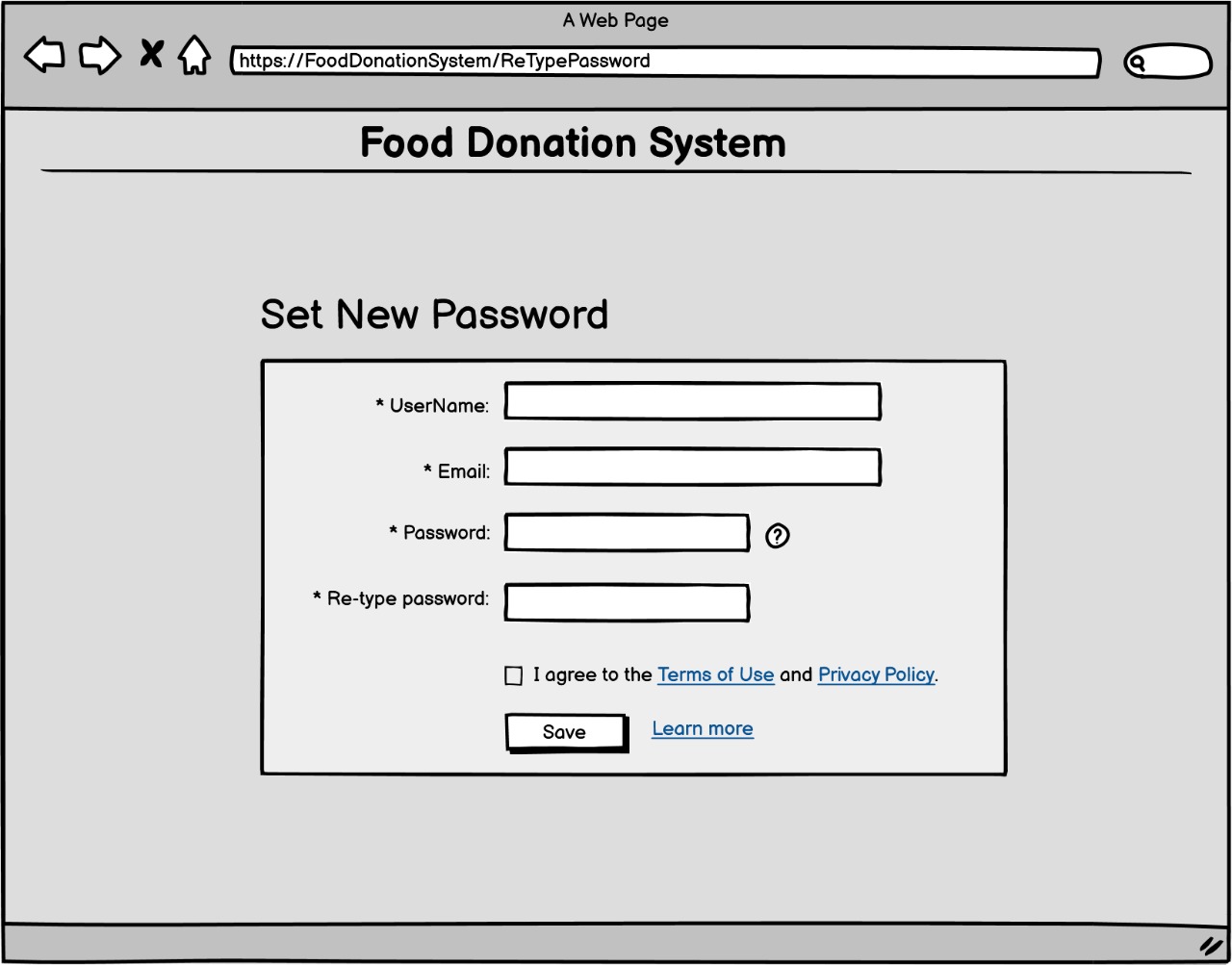
**Login System**

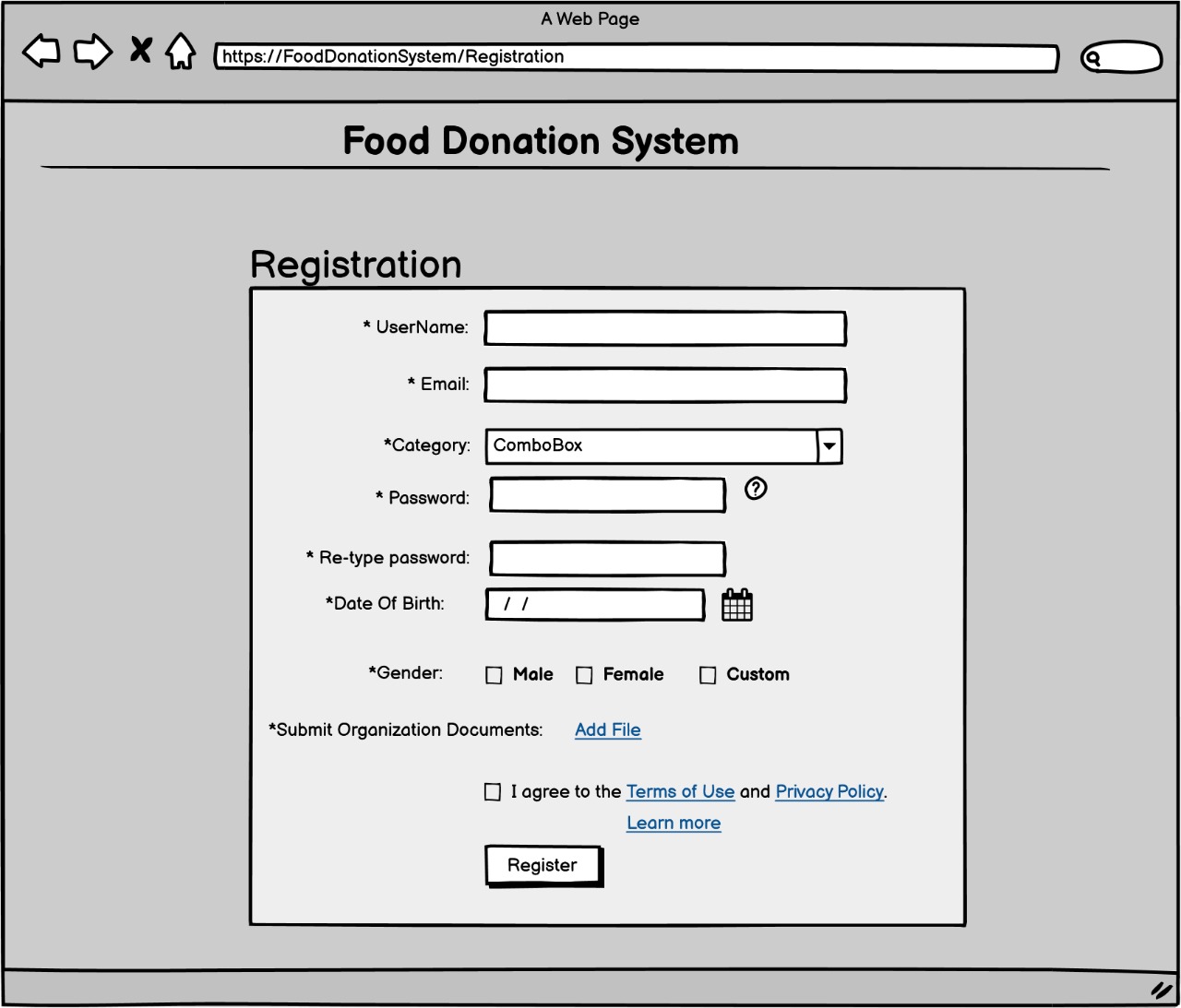


**Forget Password**

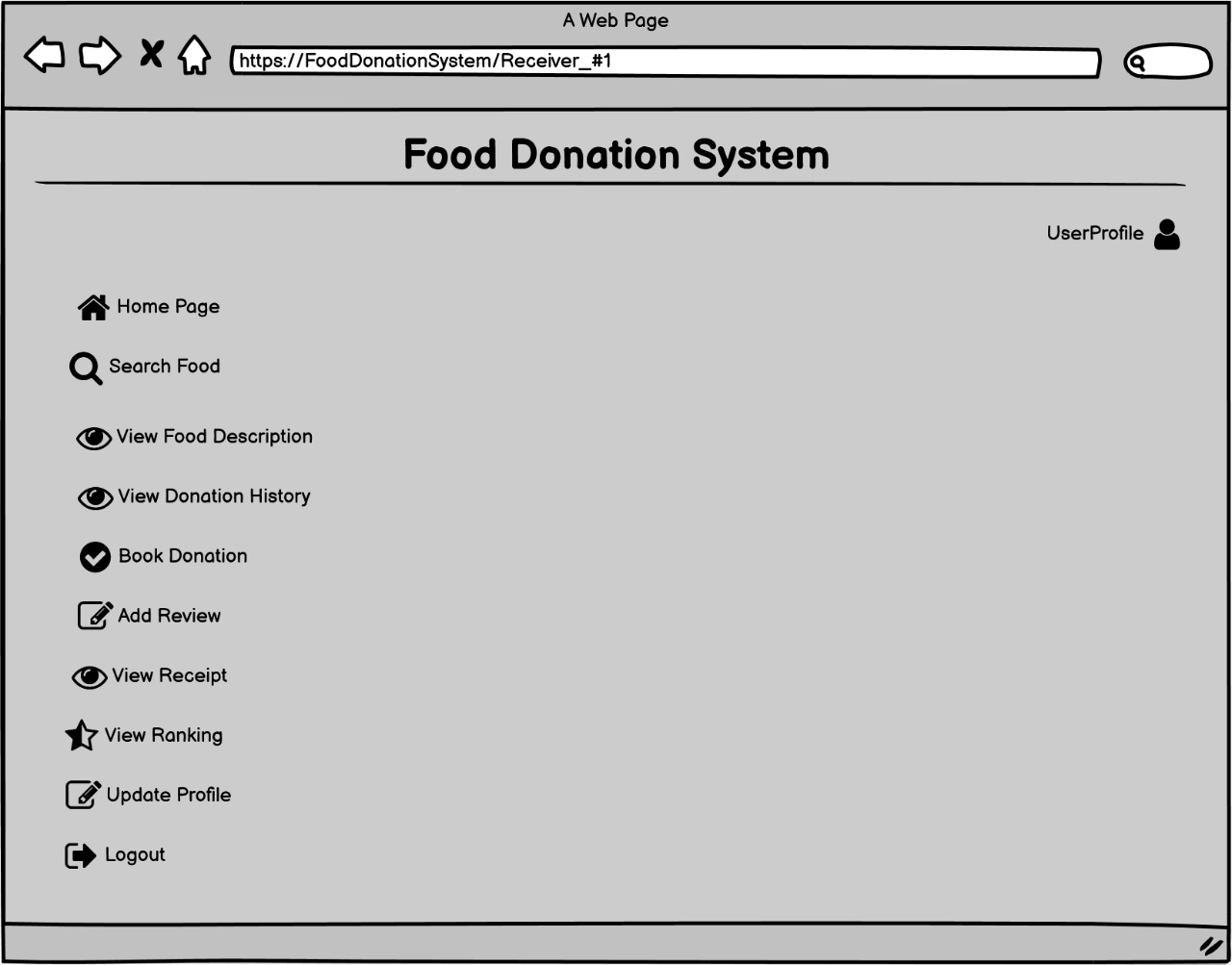


**Set New Password**

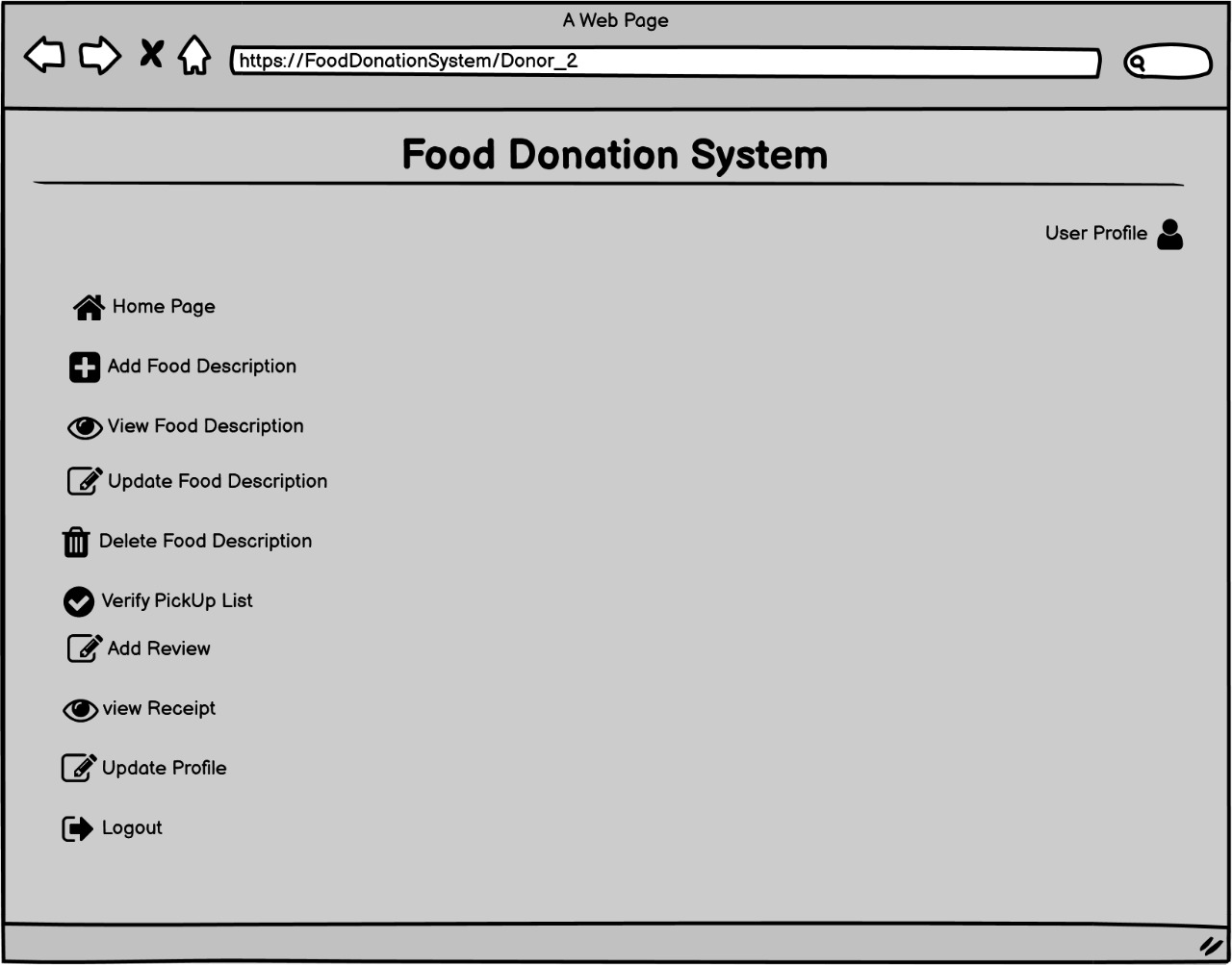


**Register User**

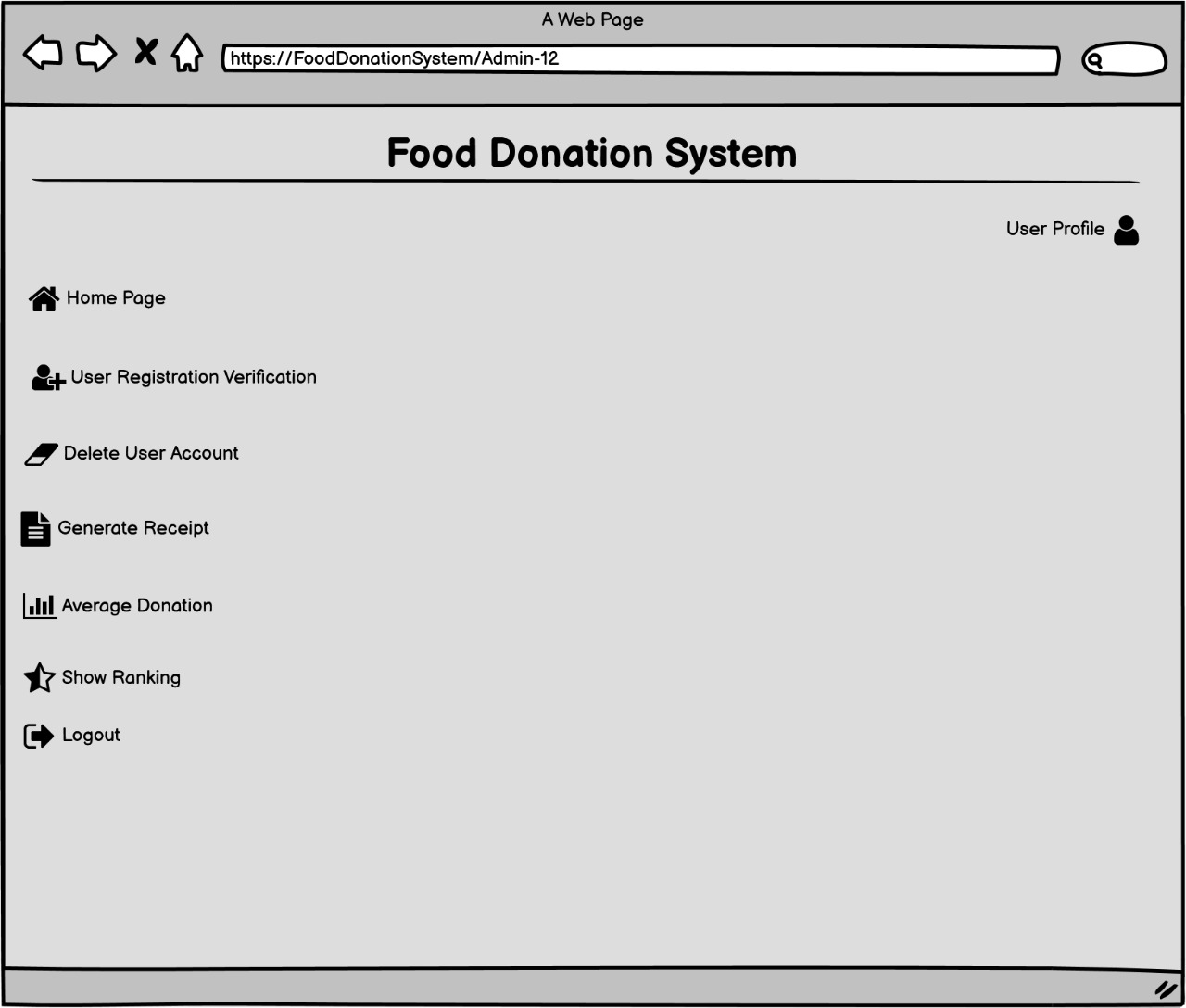
**Receiver Home Page**



**Donor Home Page**



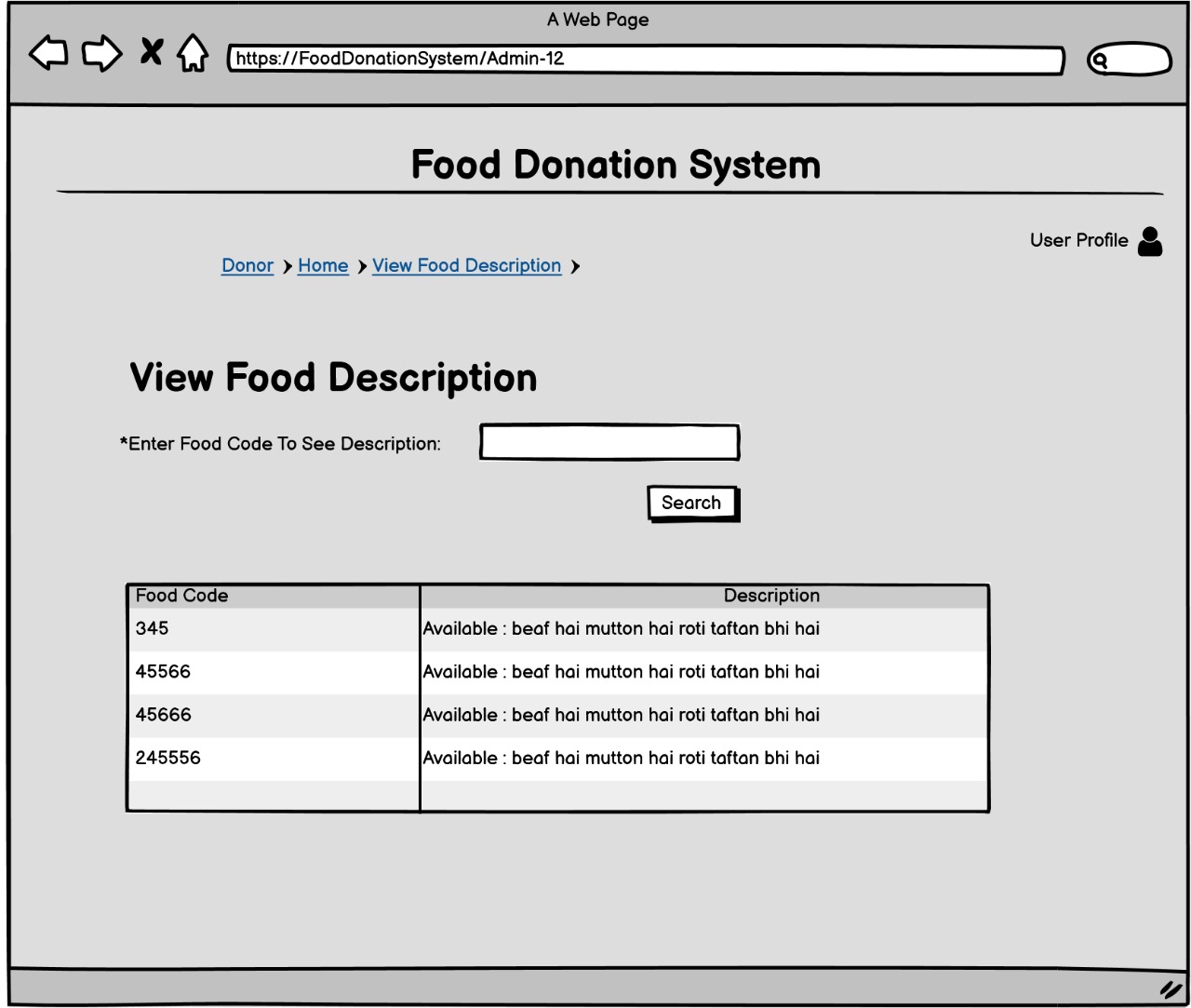
**Admin Home Page**



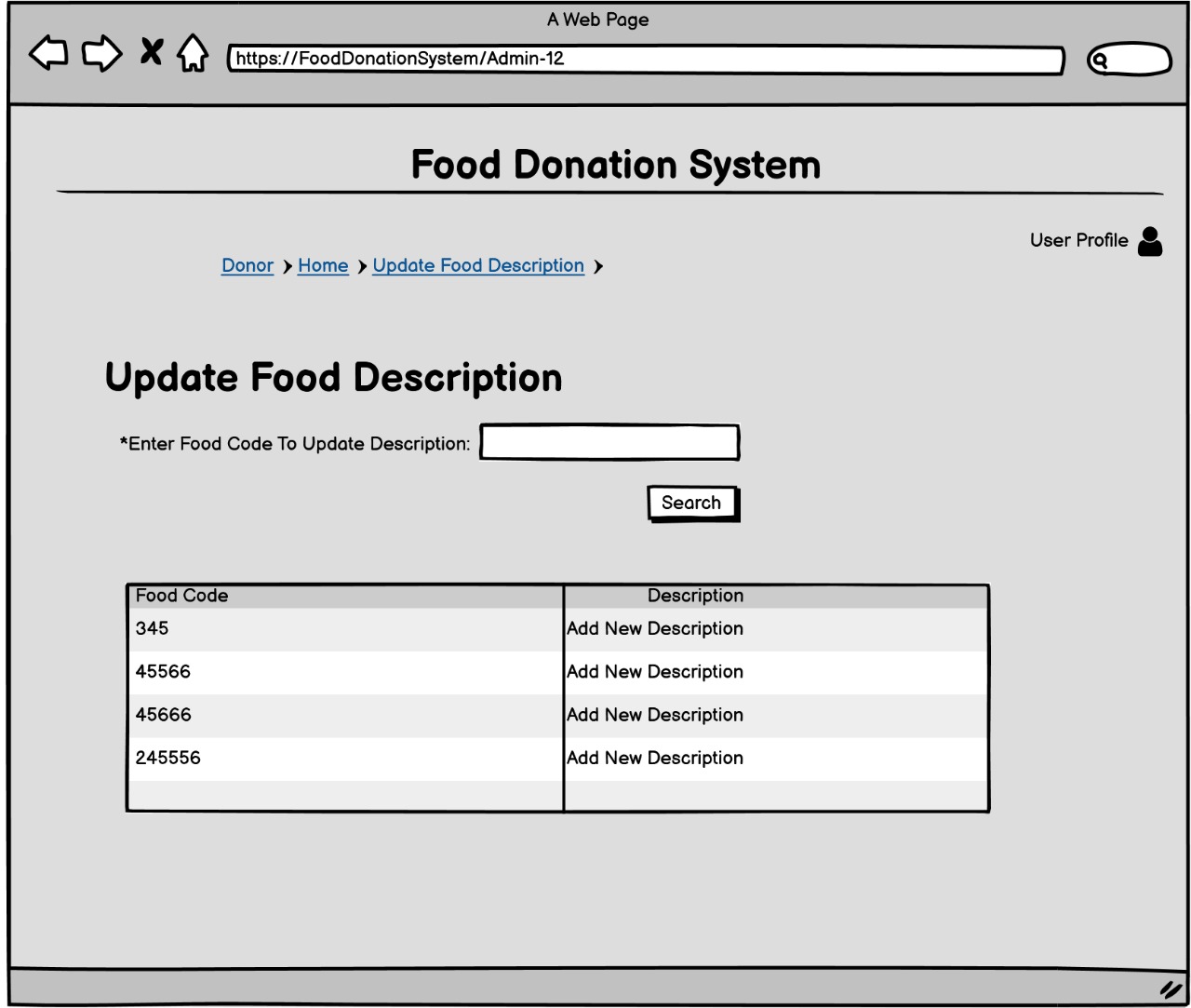
**Post Food Item**



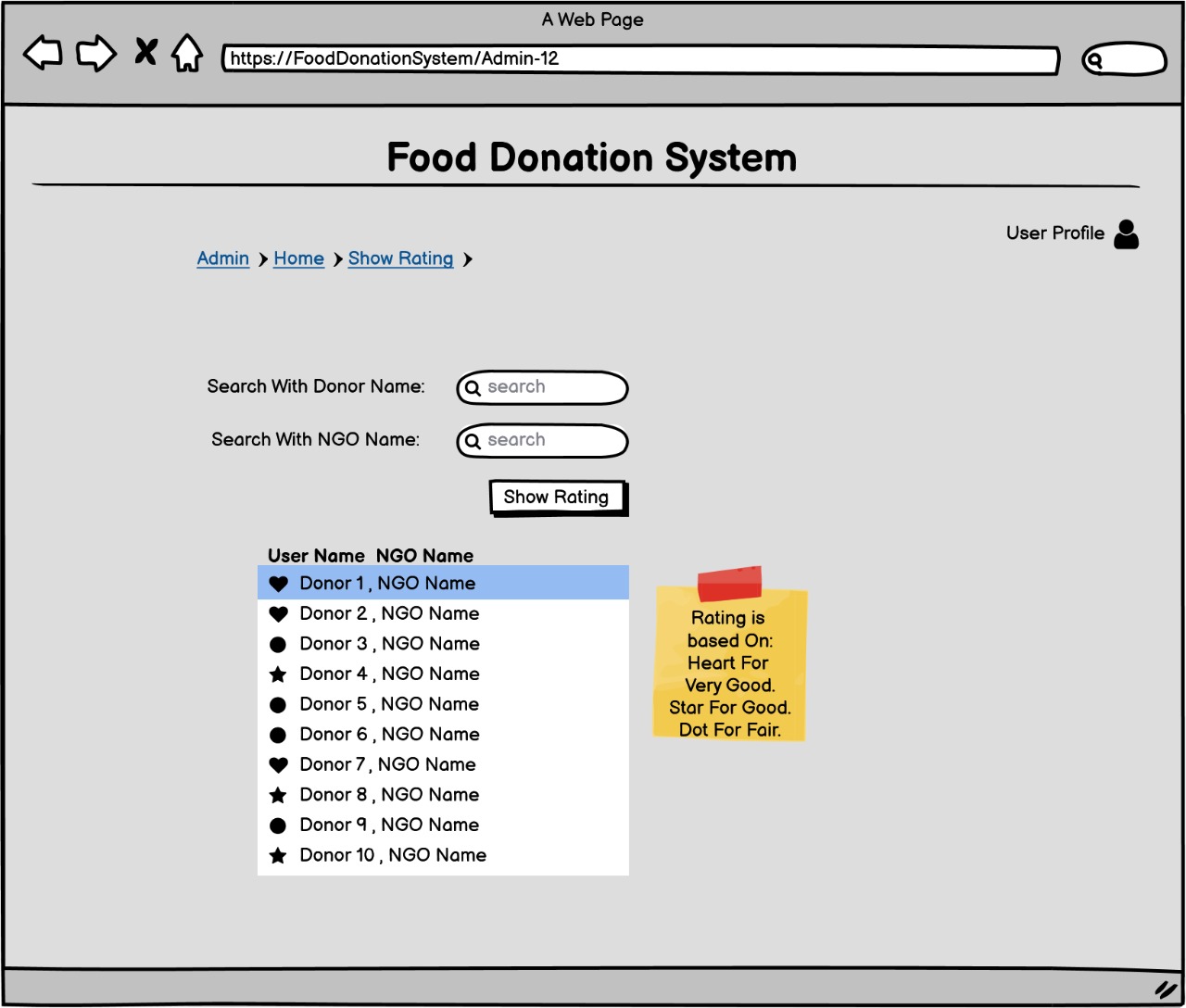
**View Food Item**



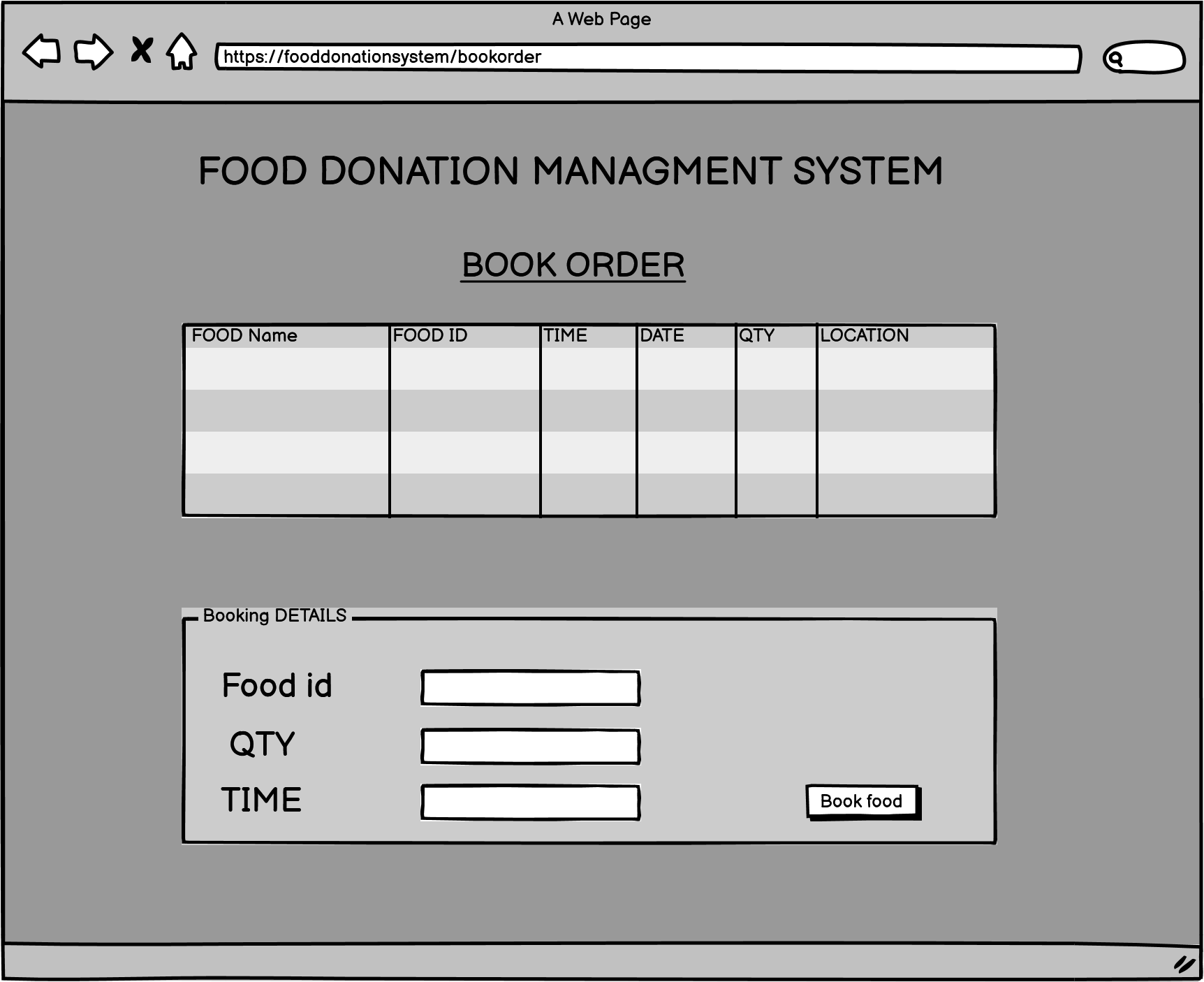
**Update Food Description**



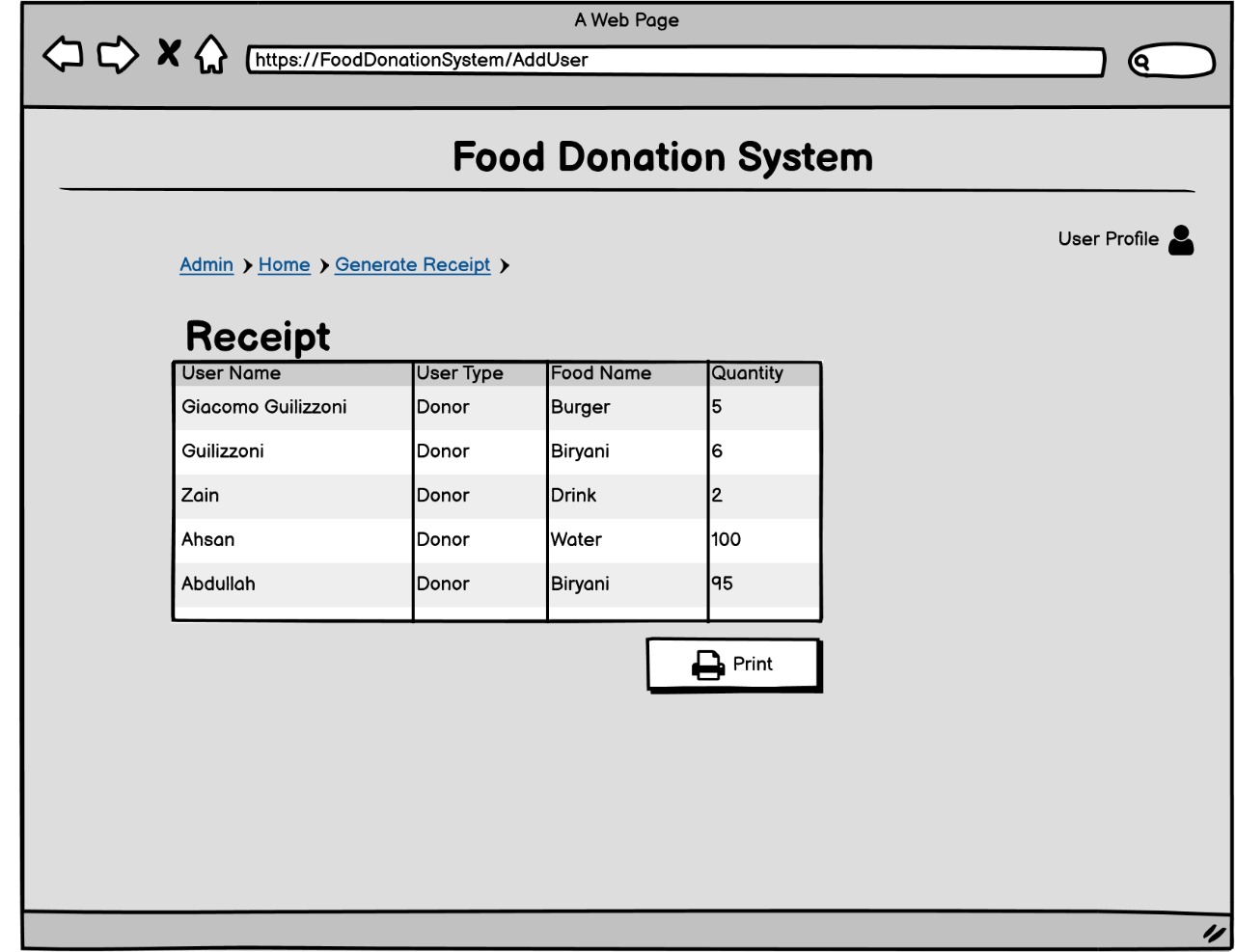
**Search Profiles**



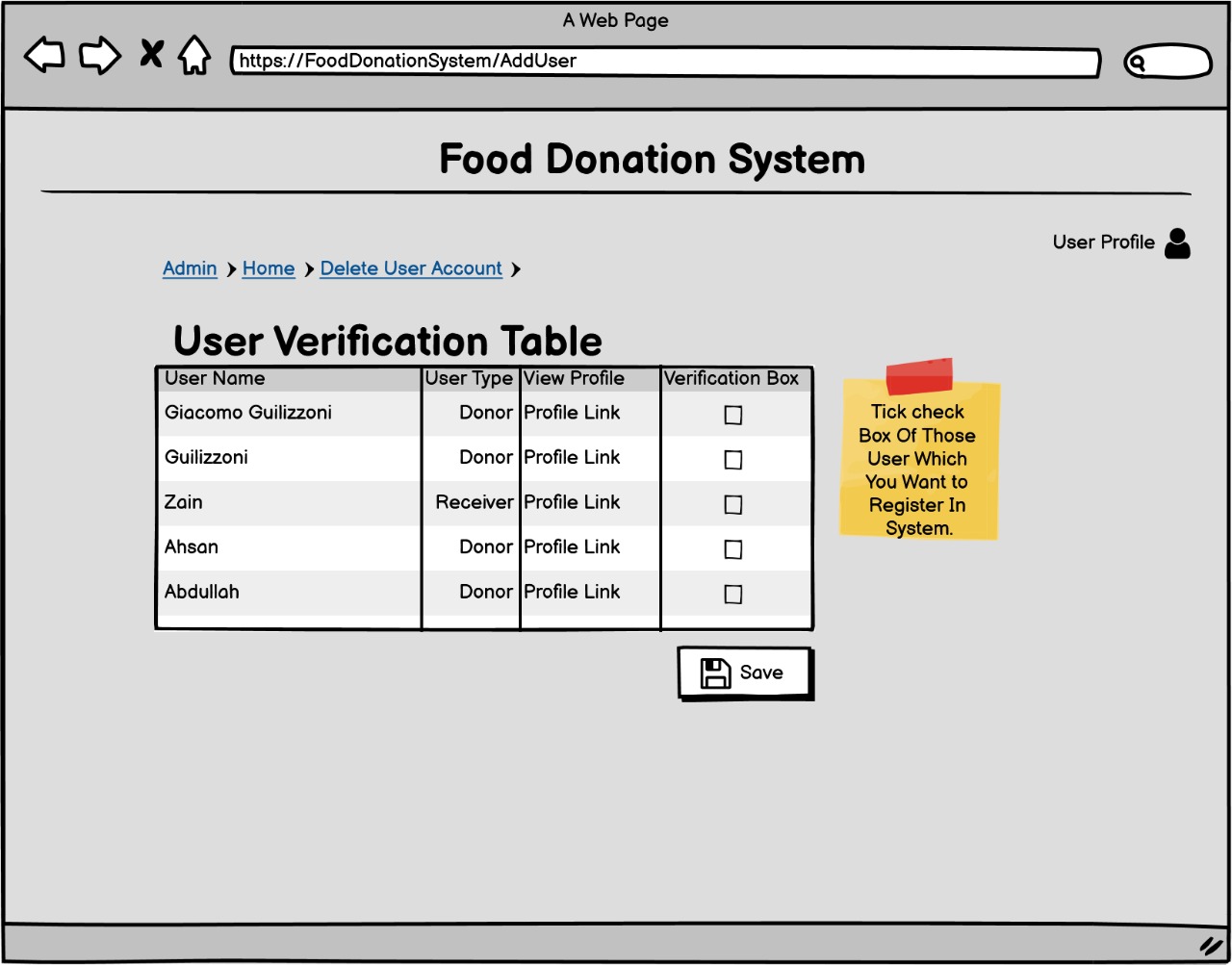
**Booking Order**

****

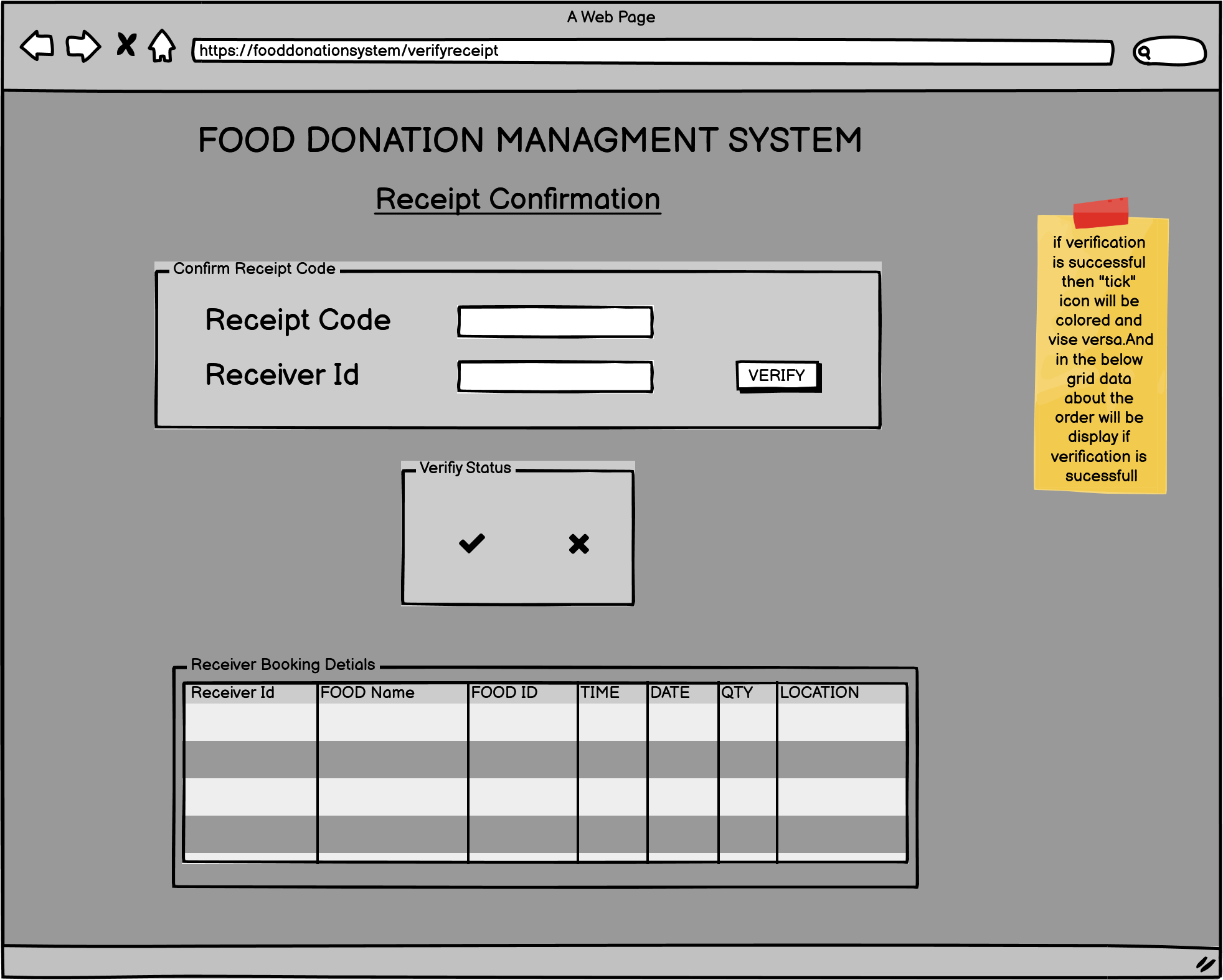
**Generating Receipt**



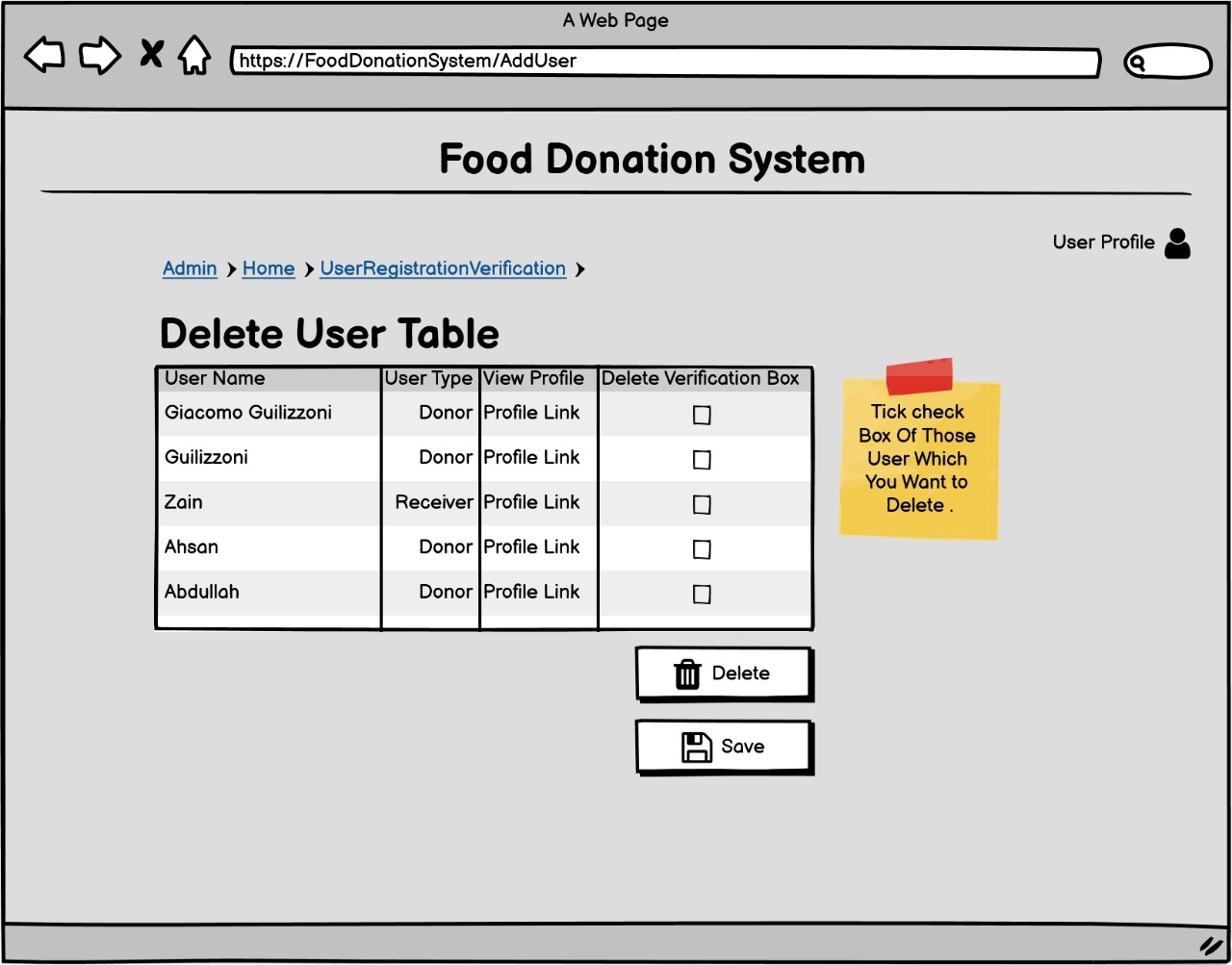
**User Verification**



**Verify Receipt**

****

**Delete User Accounts**



**Calculate Average Donation**

