American International University-Bangladesh (AIUB)

**Department of Computer Science**

**Faculty of Science &Technology (FST)**

**Fall 23-24**

# Food Cycle Management System

Software Requirement Engineering

Sec: **B**

Project submitted

By

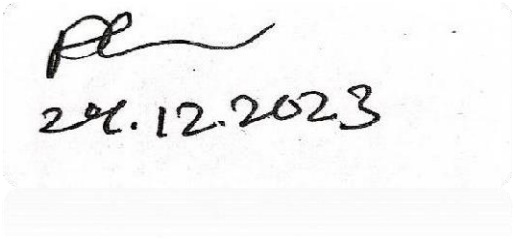
*MD NAFIJUL HOQ (20-42366-1)*

**Checked By Industry Personnel**

Name: **Palash Mallik**

Designation: **Project Manager**

Company: **DBL Group**

Sign: 

Date: 24.12.2023

# 1. PROBLEM DOMAIN

## 1.1 Background to the Problem

Food waste is the intentional discarding of edible items. Mainly by retailers and consumers about 1.3 billion tons of food is wasted globally per year. This is about one-third of all food being produced for human consumption. This waste food can fulfill 3 billion people's needs. Again, this wasted food destroys the ecological balance. So, if consumer and retailer can deliver their wasted food to needy people it can remove hunger and this problem can be solved and reduced.

The root of this problem is that the food is not eaten. It happens during production, processing, distribution retail, and food service sales and consumption. This problem is important because of needy people and keeps our environment fresh.

## 1.2 Solution to the Problem

We will build a website to solve the problem. Food is wasted all over around if we create a properly organized system then the problem might be solved. There are lots of NGOs that work with food it can save their time and also money as well. The target group of users is lots of organizers who work for food. They should benefit because it will be a smart and first system and also it will save a lot of o money and will save wasted food that makes a bad impact on the environment.

The basic functionalities are there a login function to food donor and collector on both sides. On the donor side, there will be two types of food categories one is for humans and one for animals. Two types of donor restaurants and homeowners, on the other hand admin side, there will have a food quantities calculator. If store have enough food, then informs their worker to collect food. The worker can log in and can see the order from the admin. Admin also contacts with locality informer that where the foods necessary. Locality informer can log in and informs their locality of food necessities. It will provide food for needy people and needy animals and also it will keep the environment clean.

The software being specified is a website-based Food Cycle Management System. Its purpose is to tackle the problem of food waste by providing a platform for food donors and collectors to connect and facilitate the distribution of wasted food to people and animals in need. The main objective of the software is to minimize food wastage globally by creating an organized system where food can be efficiently collected, stored, and distributed to those who require it. By implementing technology and connectivity, the software aims to bridge the gap between surplus food and hunger, while also promoting environmental sustainability.

The benefits of the software include:

1. **Addressing Hunger:** By connecting food donors with needy individuals and organizations, the software helps ensure that excess food reaches those who are in need, thus reducing hunger and food insecurity.
2. **Waste Reduction:** The system enables the collection and redistribution of wasted food, thereby reducing the amount of food that ends up in landfills. This contributes to environmental conservation and helps maintain ecological balance.
3. **Time and Cost Savings:** The software streamlines the food collection and distribution process, saving time and resources for both food donors and organizations involved in managing food assistance programs. It eliminates the need for extensive manual coordination and enables efficient utilization of available resources.
4. **Community Engagement:** The software fosters community involvement by allowing locality informers to provide information on specific food necessities in their areas. This promotes collaboration and empowers local communities to actively participate in addressing food waste and hunger.

Also, the software aligns with the following goals and objectives:

1. **Social Responsibility:** The software supports corporate social responsibility initiatives by actively addressing societal challenges such as hunger and food waste. It demonstrates a commitment to making a positive impact on the community and the environment.
2. **Operational Efficiency:** By optimizing the food collection and distribution process, the software enhances operational efficiency for both food donors and organizations involved in food assistance programs. It helps maximize resource utilization and minimize waste.
3. **Stakeholder Engagement:** The software facilitates engagement and collaboration among various stakeholders, including food donors, collectors, locality informers, and organizations working in the food assistance sector. It fosters partnerships and strengthens relationships to collectively work towards the shared goal of reducing food waste.

There are different applications, which are developed to control the wastage of food, and it gives the opportunity to send that extra food to the people who need it. There are many applications which control food waste. ‘Mobile phone Based Waste Food Supply Chain for Aurangabad Using GIS Location-Based and Google Web Services’, published in 2014, combine the client-server GIS and mobile application to make a craving-free city. The application for the client side gives the option to donate food to the people in demand. There is a lot of literature on this topic but our system will collect food from the root also it will collect dusty food for animals and distribute it to needy people where it is actually needed. It will also connect with the locality which is the main feature.

Given below the simple Flow Chart of our Food Cycle Management System:

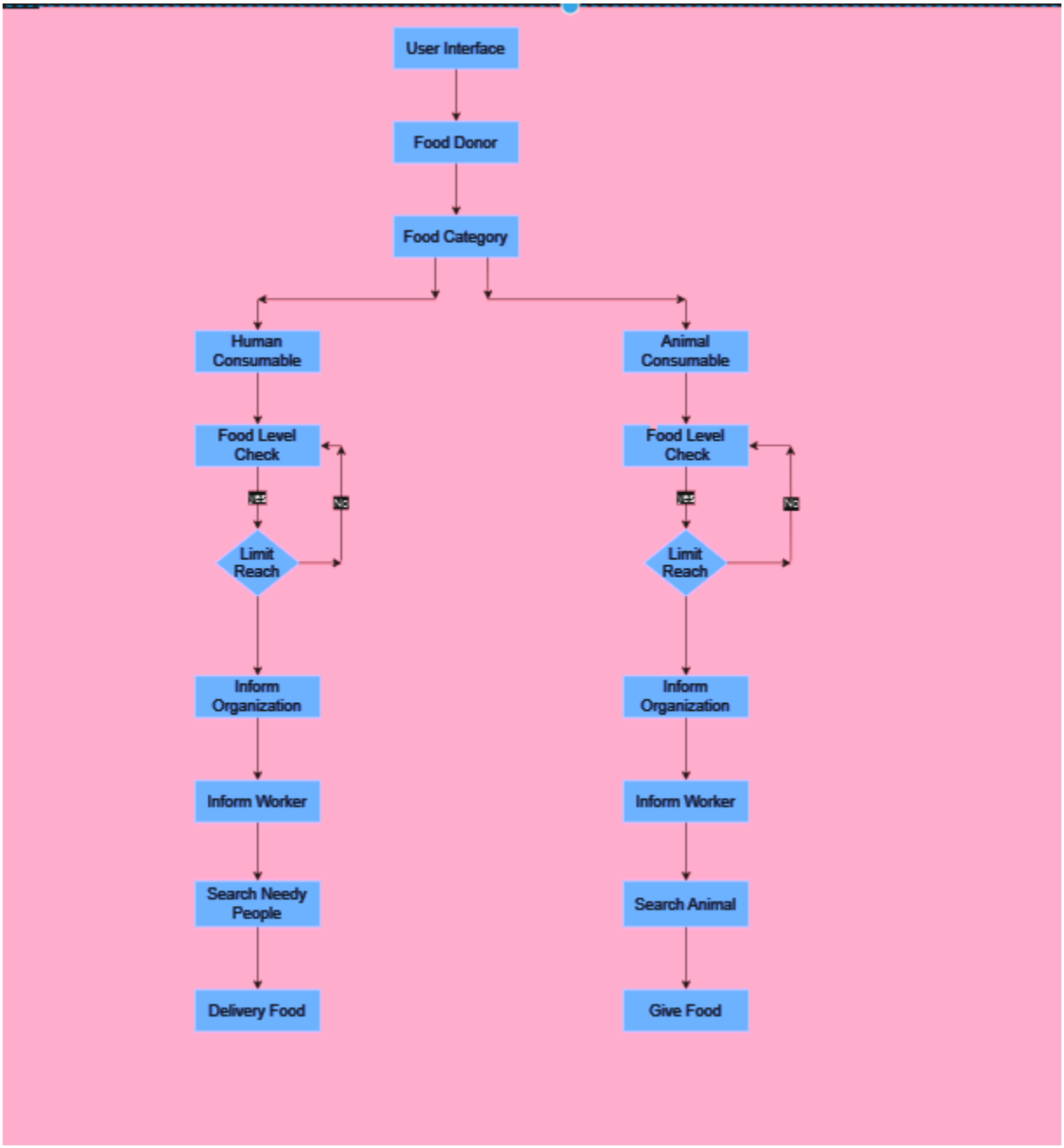



*Figure-1: Flow Chart of our Food Cycle Management System.*

# 2. SOLUTION DESCRIPTION

## 2.1 System Features

### 2.1.1. Login

**Functional Requirements**

**2.1.1.1.** The software shall allow users to log in with their given phone number and password.

**2.1.1.2.** The login credentials (phone number and password) will be verified with database records.

**2.1.1.3.** If the login is successful, the home page of the user account will be displayed.

**2.1.1.4.** If the phone number and/or password have been inserted wrong, the system will generate and send an OTP to the user's email address or phone number to retry login by resetting their password

**Priority Level:** High
**Precondition:** User has a valid user id and password
**Cross-references:** N/A

### 2.1.2. Registration

There will be given 3 options to choose the category where they want to register. The options are:

- As a Donor
- As a Worker
- As an Organizer

#### 2.1.2.1. As a Donor

**Functional Requirements**

**2.1.2.1.1.** After selecting the donor option, they will be directed to a sign-up page where they will be able to fill up their information as the donor.

**2.1.2.1.2.** After the registration they have to click submit. Then automatically they will receive an OTP to their phone (The phone number was given while registering). They have to fill up the OTP and press ok.

**2.1.2.1.3.** After successfully inserting the OTP, they will be directed to Google Maps to pinpoint their location. After they locate

their position, they are all set for that one particular option they have chosen.

**Priority Level:** High
**Precondition:** User has to log in as a donor.
**Cross-references:** 2.1.1, 2.1.2

#### 2.1.2.2. As a Worker

**Functional Requirements**

**2.1.2.2.1.** After selecting the worker option, they will be directed to a sign-up page where they will be able to fill up their information as a worker.

**2.1.2.2.2.** After the registration they have to click submit. Then automatically they will receive an OTP to their phone (The phone number was given while registering). They have to fill up the OTP and press ok.

**2.1.2.2.3.** After successfully inserting the OTP, they will be directed to Google Maps to pinpoint their location. After they locate their position, they are all set for that one particular option they have chosen.

**Priority Level:** High
**Precondition:** User has to log in as a worker.
**Cross-references:** 2.1.1, 2.1.2

#### 2.1.2.3. As an Organizer

**Functional Requirements**

**2.1.2.3.1.** After selecting the organizer option, they will be directed to a sign-up page where they will be able to fill up their information as an organizer.

**2.1.2.3.2.** After the registration they have to click submit. Then automatically they will receive an OTP to their phone (The phone number was given while registering). They have to fill up the OTP and press ok.

**2.1.2.3.3.** After successfully inserting the OTP, they will be directed to Google Maps to pinpoint their location. After they locate their position, they are all set for that one particular option they have chosen.

**Priority Level:** High
**Precondition:** User has to log in as an organizer.
**Cross-references:** 2.1.1, 2.1.2

### 2.1.3. Donor Homepage

When the user comes to this page, they will get three options. The options will be:

- ❖ Donate Food Categories.
- ❖ Suggestion Box.
- ❖ Food Donation Listing

They will be redirected to the next steps according to their selection.

**Priority Level:** High
**Precondition:** User is logged in as a donor.
**Cross-references:** 2.1.1, 2.1.2.1

#### 2.1.3.1. Donate Food Categories

**Functional Requirements**

**2.1.3.1.1.** If the first option is selected the software will present the user with a screen that includes the two food category options the user wishes to donate.

The options are:

- ❖ Human Consumable.
- ❖ Animal Consumable.

**2.1.3.1.2.** In both categories the user will be asked to provide details about the food and if there is any need for utensils there.

**2.1.3.1.3.** The software will prompt the user to submit the selection of food that they want to donate. If prompted, the user has to click submit to place the donation.

**2.1.3.1.4.** Once the user makes all of their desired food category to donate, they can proceed to the next step in using the software.

**Priority Level:** High
**Precondition:** User is logged in as a donor.
**Cross-references:** 2.1.1, 2.1.2.1

#### 2.1.3.2. Suggestion Box

**Functional Requirements**

**2.1.3.2.1.** The software shall allow users to give feedback here.

**2.1.3.2.2.** Users will be able to select a suggestion box and then can write their opinion.
**2.1.3.2.3.** If complete their write there will submit button to submit it.
**2.1.3.2.4.** If a submission is successful will display successful submission.
**2.1.3.2.5.** If any problem arises will display and try again after some time.

**Priority Level:** Medium
**Precondition:** User is logged in and on the Donor Homepage.
**Cross-references:** 2.1.1, 2.1.2.1, 2.1.3.1

#### 2.1.3.3. Food Donating Lists

**Functional Requirements**

**2.1.3.3.1.** To list the food the user has to fill up the form. First, they have to give a title and provide a description of the food.
**2.1.3.3.2.** Then they select the type of food i.e., veg, non-veg.
**2.1.3.3.3.** After choosing they will be given the option to select the quantity.
**2.1.3.3.4.** Adding photos and expiration dates are also available for the user to fill up.
**2.1.3.3.5.** After clicking the submit button a confirmation check box will pop up.

**Priority Level:** High
**Precondition:** User is logged in and on the Donor Homepage.
**Cross-references:** 2.1.1, 2.1.2.1

### 2.1.4. Organizer Homepage

If anyone registers themselves as an organizer, they will be accessing this page. Where they will be in authority of some actions –

- Food level check
- Inform workers,
- Food necessities,
- Food quantity calculation,
- Receive a suggestion

**Priority Level:** High
**Precondition:** User is logged in as an organizer.
**Cross-references:** 2.1.1, 2.1.2.3

#### 2.1.4.1. Food Level Check

**Functional Requirements**

**2.1.4.1.1.** While operating as an organizer one can push this button to check the level of food that everyone donated.
**2.1.4.1.2.** It will help the organizer to determine the quantity as well as the number of foods are on stock.

**Priority Level:** Medium
**Precondition:** User is logged in as an organizer and on the Organizer.
**Cross-references:** 2.1.1, 2.1.2.3, 2.1.4

#### 2.1.4.2. Inform worker

**Functional Requirements**

**2.1.4.2.1.** The software will notify the admin with a screen that includes that the food is ready to deliver.
**2.1.4.2.2.** The admin will then get information about the location where the storage is full also the available worker at that location.
**2.1.4.2.3.** After selecting the worker if the admin press "confirms", that particular worker will get notified.

**Priority Level:** Medium
**Precondition:** User is logged in as an organizer and on the Organizer Homepage.
**Cross-references:** 2.1.1, 2.1.2.3, 2.1.4

#### 2.1.4.3. Food Necessities

**Functional Requirements**

**2.1.4.3.1.** The software gives an option for the organizer where they will be receiving notifications from the workers.
**2.1.4.3.2.** When they push the 'notify from the worker' button they will see the location and quantity of the foods.
**2.1.4.3.3.** If the next button is pressed another window will come up where they can count the quantity of food for both categories.

**Priority Level:** Medium
**Precondition:** User is logged in as an organizer and on the Organizer.
**Cross-references:** 2.1.1, 2.1.2.3, 2.1.3.2

#### 2.1.4.4. Quantity Calculation

**Functional Requirements**

**2.1.4.4.1.** The software will present the admin with a screen that includes the option to check the collection list for both human and animal consumable foods.
**2.1.4.4.2.** Also, there will be a section where the distribution list will be registered.
**2.1.4.4.3.** The software will then calculate the total food by subtracting distributed food from collected food and let the organizer know.

**Priority Level:** Medium
**Precondition:** User is logged in as an organizer and on the Organizer Homepage.
**Cross-references:** 2.1.1, 2.1.2.3, 2.1.4

#### 2.1.4.5. Receive Suggestion

**Functional Requirements**

**2.1.4.5.1.** The organizer will receive suggestions from workers and other users. These suggestions will be checked by the organizer by typing the 'Receive suggestion' button.

**Priority Level:** Medium
**Precondition:** User is logged in as an organizer and on the Organizer Homepage.
**Cross-references:** 2.1.1, 2.1.2.3

### 2.1.5. Worker Homepage

**Functional Requirements**

**2.1.5.1.** The software will give the workers options to receive notifications from the organizer, collect the food, deliver food, and info the necessity.
**2.1.5.2.** To receive messages workers will have to press the receive message button.
**2.1.5.3.** To collect food, they first will have to select the food category option and the quantity and press confirm to collect it.
**2.1.5.4.** Similarly, to deliver food they first will have to select the food category option and the quantity and press confirm to deliver it.
**2.1.5.5.** Workers can send messages to the organizer too by clicking the necessary information.

**Priority Level:** High
**Precondition:** User is logged in as a worker.
**Cross-references:** 2.1.1, 2.1.2.2

### 2.1.6. Messages

**Functional Requirements**

**2.1.6.1.** This page will only be used to interact between the user and the organizer about donating food or collecting food.
**2.1.6.2.** Also, if there are any problems regarding delivery and collecting the food can be shared through this interface.

**Priority Level:** Low
**Precondition:** User has to login in with a valid account.
**Cross-references:** 2.1.1, 2.1.2.2, 2.1.5

### 2.1.7. My Profile

**Functional Requirements**

**2.1.7.1.** In this interface users can see their personal info, donation, rewards, and achievement.
**2.1.7.2.** They can change their settings also can reach out for help in the community.

**Priority Level:** Low
**Precondition:** User has to login in with a valid account.
**Cross-references:** 2.1.1

### 2.1.8. Sign out

**Functional Requirements**

**2.1.8.1.** If the user clicks on the logout option, the software may prompt the user to confirm if they want to log out.
**2.1.8.2.** If prompted, confirm that they want to log out by clicking on the "yes" or "confirm" button.
**2.1.8.3.** The software will log them out of their account and return them to the login page of the application.

**Priority Level:** Low
**Precondition:** User has to login in with a valid account.
**Cross-references:** 2.1.1

## 2.2. Quality Attribute

**2.2.1. Availability:** This requirement specifies a minimum level of availability for the software, which is measured over a specified time period. The system shall be available at least 90% on weekdays between 6 am to midnight local time and at least 95% available on weekdays between 10 am to 2 pm. This requirement helps to ensure that the software is highly available and reliable for users, which can contribute to a positive user experience and overall satisfaction with the software.

**Priority Level:** High
**Precondition:** User should log in to the system.
**Cross-references:** 2.2.6, 2.2.7

**2.2.2. Performance:** The system shall be able to handle a large number of users and requests simultaneously, without experiencing any significant slowdowns or crashes. The response time for the system to process requests shall not exceed two seconds.

**Priority Level:** Medium
**Precondition:** User has to log in with a valid ID.
**Cross-references:** 2.2.1, 2.2.3, 2.2.4, 2.2.8

**2.2.3. Usability:** This requirement emphasizes the importance of designing software with the user in mind and ensuring that it is easy and intuitive for users to interact with. This requirement helps to ensure that the software is user-friendly and easy to use.

**Priority Level:** Medium
**Precondition:** Software should be designed in favor of the consumers
**Cross-references:** 2.2.1, 2.2.2, 2.2.4

**2.2.4. Efficiency:** This requirement specifies a desired level of efficiency for the software, in terms of its response time to user requests. It sets both an average and a maximum response time, to allow for some variation in response times while still ensuring that the software is generally responsive to user inputs. This requirement helps to ensure that the software is efficient and effective in meeting user needs, which can contribute to a positive user experience and overall satisfaction with the software.

**Priority Level:** High
**Precondition:** Having clear goals and performance measurements analysis of the software.
**Cross-references:** 2.2.2, 2.2.5

**2.2.5. Integrity:** This requirement helps to ensure that the software's data is trustworthy and reliable, which can be critical for users who rely on the software to make important decisions or take important actions.

**Priority Level:** High
**Precondition:** Admin should log in along with a valid username and password.
**Cross-references:** 2.2.3, 2.2.4, 2.2.8

**2.2.6. Reliability:** The system shall be reliable and available for use 24/7, with minimal downtime or interruptions. The system shall have a mean time between failures of at least 1000 hours.

**Priority Level:** High
**Precondition:** Software should have fault tolerance and redundancy, Error Handling and Exception Management to perform 24/7
**Cross-references:** 2.2.3, 2.2.4, 2.2.5

**2.2.7. Robustness:** This requirement emphasizes the importance of designing software that can handle unexpected inputs and edge cases, such as invalid user inputs or unexpected system errors. By ensuring that the software is able to handle these situations gracefully, without crashing or compromising data integrity, developers can ensure that the software remains robust and resilient even in unpredictable circumstances. This requirement helps to ensure that the software is reliable and able to function effectively even in challenging conditions, which can be critical for users who depend on the software for important tasks or decisions.

**Priority Level:** High
**Precondition:** Software should have Resilient Architecture and defensive programming to be able to tackle any unexpected challenges.
**Cross-references:** 2.2.2, 2.2.3, 2.2.4, 2.2.5

**2.2.8. Testability:** This requirement emphasizes the importance of designing software that is easily testable, in order to facilitate efficient testing and debugging by developers. Additionally, by designing for testability, developers can reduce the amount of time and effort required to test and debug the software, which can lead to faster development cycles and better overall efficiency. This requirement helps to ensure that the software is reliable and of high quality, which can be critical for users who depend on the software for important tasks or decisions.

**Priority Level:** Medium
**Precondition:** Software with test environment management, testability as a design principle.
**Cross-references:** 2.2.2, 2.2.4, 2.2.5

## 2.3 UML Diagrams:

### Use Case Diagram –

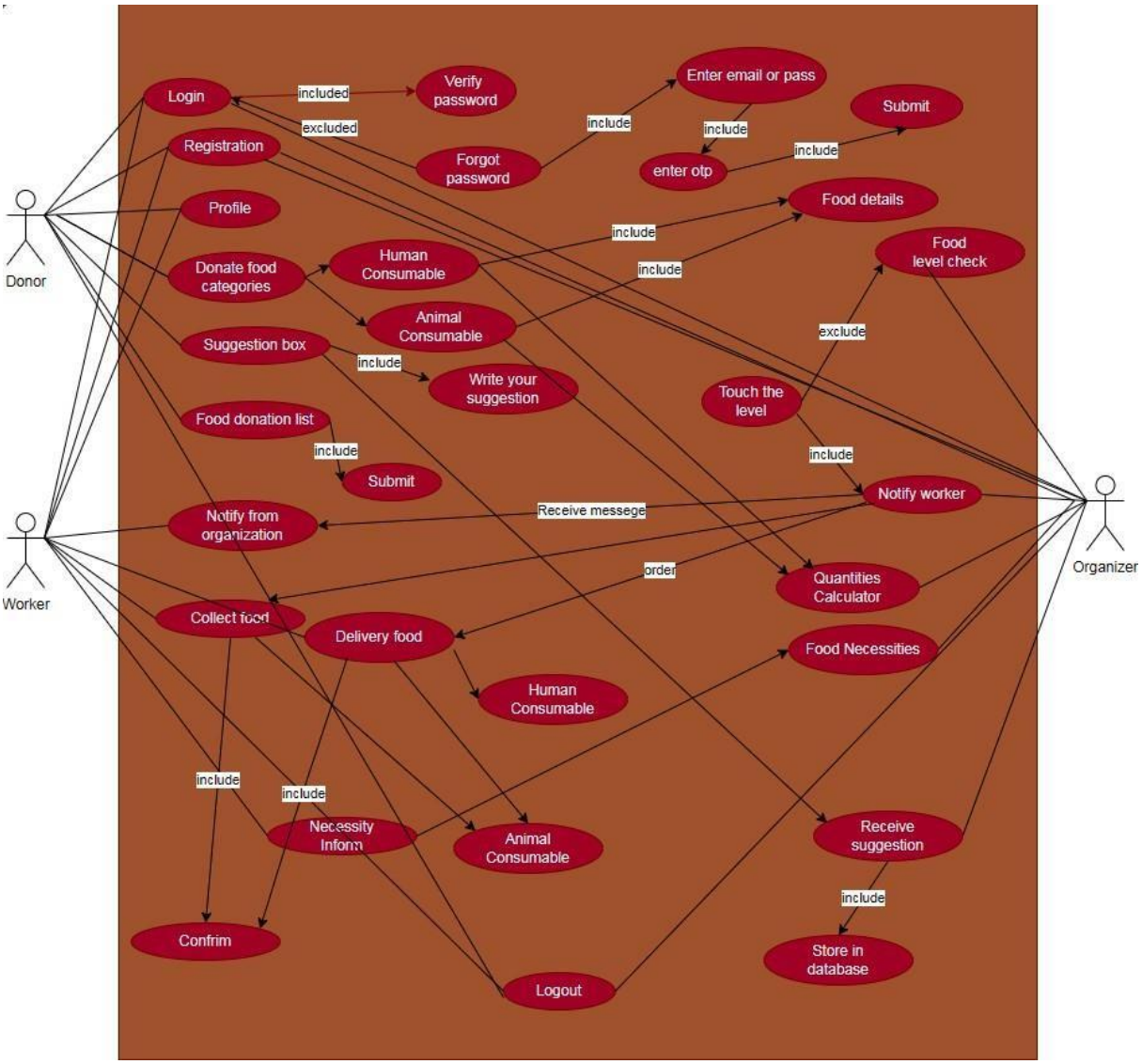



*Fig-2: Use Case Diagram*

**ER Diagram –**

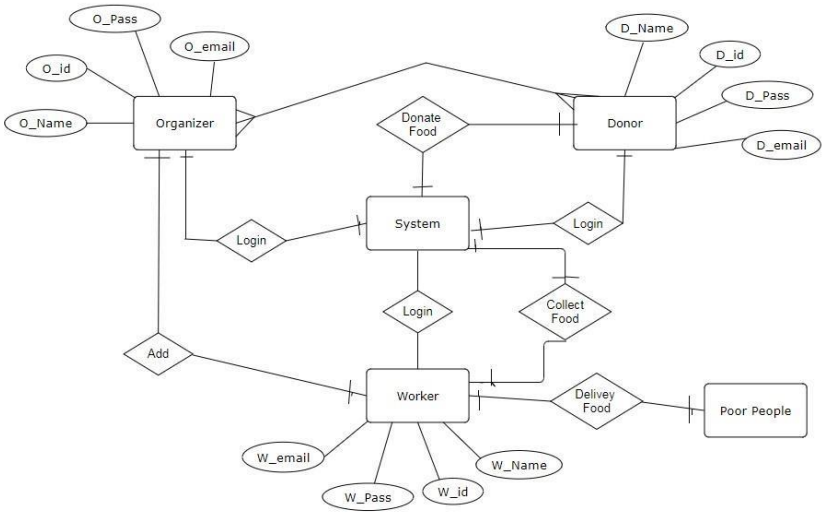



*Fig-3: ER Diagram*

**Class Diagram –**

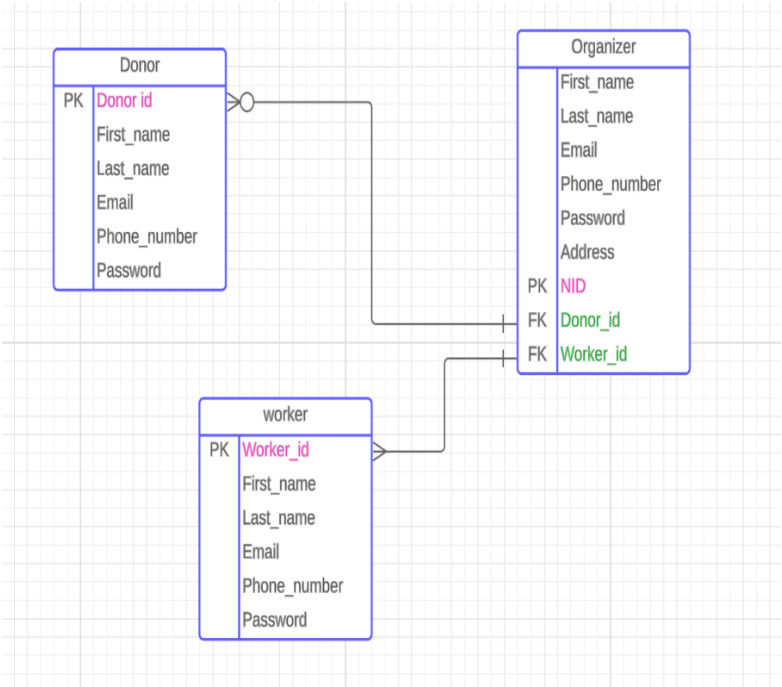



*Fig 5: Class Diagram*

## 3. Social Impact

There are several benefits that the food cycle management system offers to society.

**Reducing Food Waste:** The system effectively solves the problem of food waste by redistributing excess food from homes, eateries, and bakeries to those in need. This reduces the environmental impact of food waste while simultaneously guaranteeing that more people have access to wholesome meals.

**Reducing Hunger:** By offering a long-term way to give extra food to hungry people and families, the project directly addresses food insecurity. In addition to supporting vulnerable populations, this advances social equity.

**Community Engagement:** Empathy and a sense of community involvement are fostered by promoting cooperative efforts amongst many stakeholders, including funders, recipients, and philanthropic organizations. It establishes a platform on which people and companies can make significant contributions to the well-being of society.

**Animal Welfare:** Showing care for animal welfare is demonstrated by feeding street animals through the system. It helps make society more compassionate by attending to the needs of animal populations that are under risk.

**Resource Optimization:** By effectively redistributing extra food, resources are maximized for food use and transportation. This decreases the demand on environmental resources and lessens the financial load on groups that address hunger.

**Opportunities for Education:** The project offers chances for educational programs on community support, environmental practices, and food waste. It can act as a forum for promoting conscientious consumption practices.

All things considered, the food cycle management system has the potential to have a substantial impact on society by tackling important problems with food waste, hunger, community involvement and animal welfare.

# 4. Development Plan with Project Schedule

Project Development Plan for Food Cycle management System –

1. **Project Initiation**
   - 1.1. Define Scope and Objective
   - 1.2. Identify Stakeholder's
   - 1.3. Define Project Schedule and Budget
   - 1.4. Conduct Risk Assessment
   - 1.5. Define Project Success Criteria
2. **Requirements Gathering**
   - 2.1. Stakeholder Interviews
   - 2.2. Define Features and Functionalities
3. **System Design**
   - 3.1. Develop System Architecture
   - 3.2. Database Design
   - 3.3. User Interface Design
4. **Development**
   - 4.1. Frontend Development
   - 4.2. Backend Development
   - 4.3. Backend API Development
   - 4.4. Conduct Code Review and Debugging
5. **Testing**
   - 5.1. Unit Testing
   - 5.2. Integration Testing
   - 5.3. System Testing
   - 5.4. Acceptance Testing
6. **Marketing**
   - 6.1. Develop Marketing and Promotion Materials
   - 6.2. Implement Marketing Strategy
   - 6.3. Engage with NGOs etc.
7. **Deployment**
   - 7.1. Develop User Manuals and Documentation
   - 7.2. Software Release
   - 7.3. Post Release Monitoring
   - 7.4. Analyze Feedback

We require **Six** months to build the software.

To prepare project schedule we use **GanttPRO** project management tools.

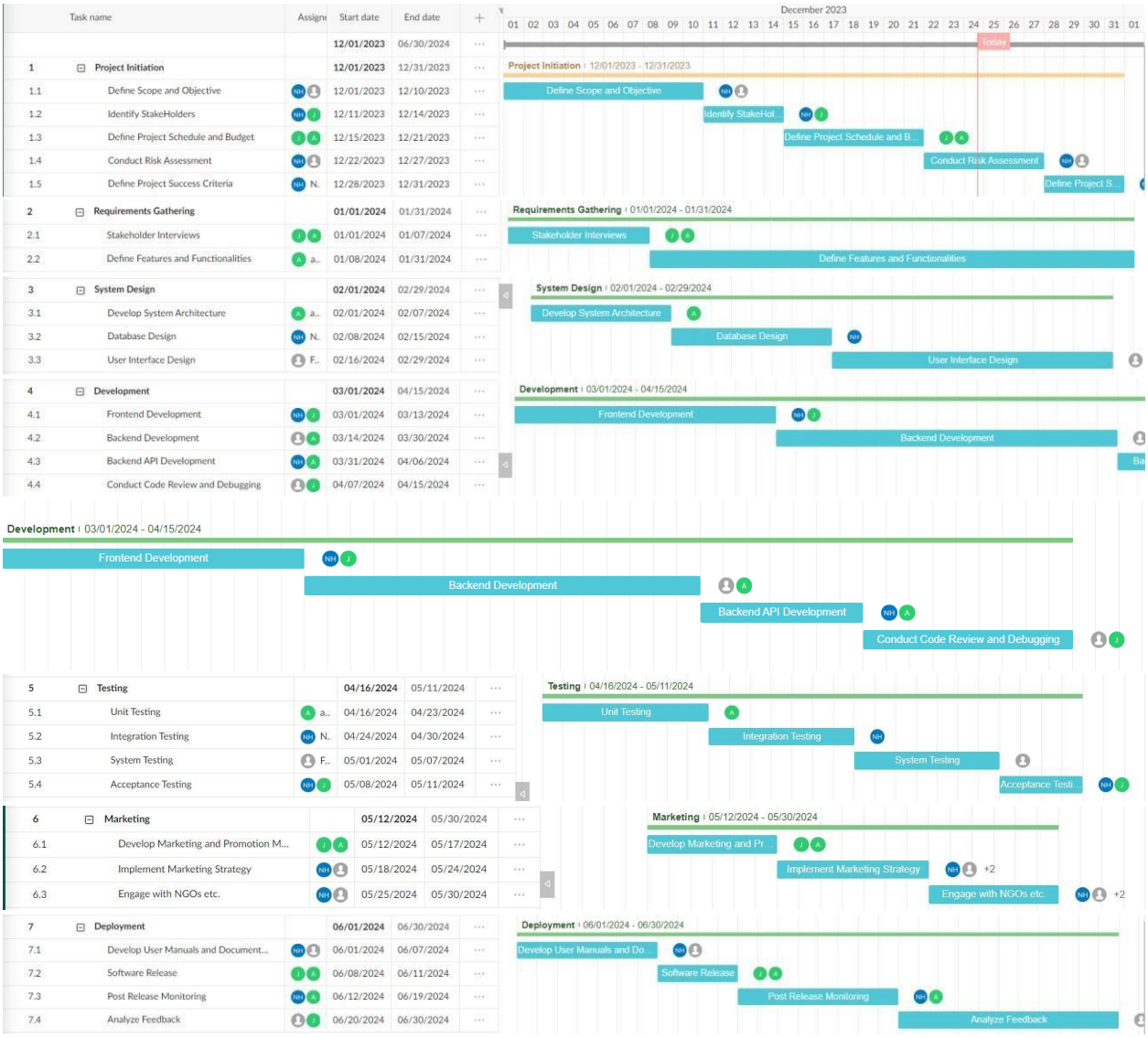



*Fig-4: Project Scheduling Gantt Chart*

**Link:** Using this link anyone can see our Project Scheduling Gantt Chart on GanttPro –

https://app.ganttpro.com/shared/token/14dbd636a397f9169ae12fb860abe3fdccee16bd58fde90721930152aa8bf121/1310357

# 5. Marketing Plan

**Short-Term Plan (0-6 months):**

1. **Online Presence and Awareness:**
    - Creating a website and social media profiles emphasizing the benefits of the system.
    - Launching targeted social media campaigns focusing on food waste reduction and social impact messaging.
    - Leverage content marketing to highlight success stories, tips for reducing food waste, and the importance of community involvement.
2. **Partnerships and Community Engagement:**
    - Collaborating with local restaurants, bakeries, and charities for initial donations and partnerships.
    - By hosting community events and workshops to raise awareness about food waste and encourage participation in the initiative.
    - Initiate local outreach programs to engage community leaders and stakeholders in supporting the cause.

**Long-Term Plan (6-12 months):**

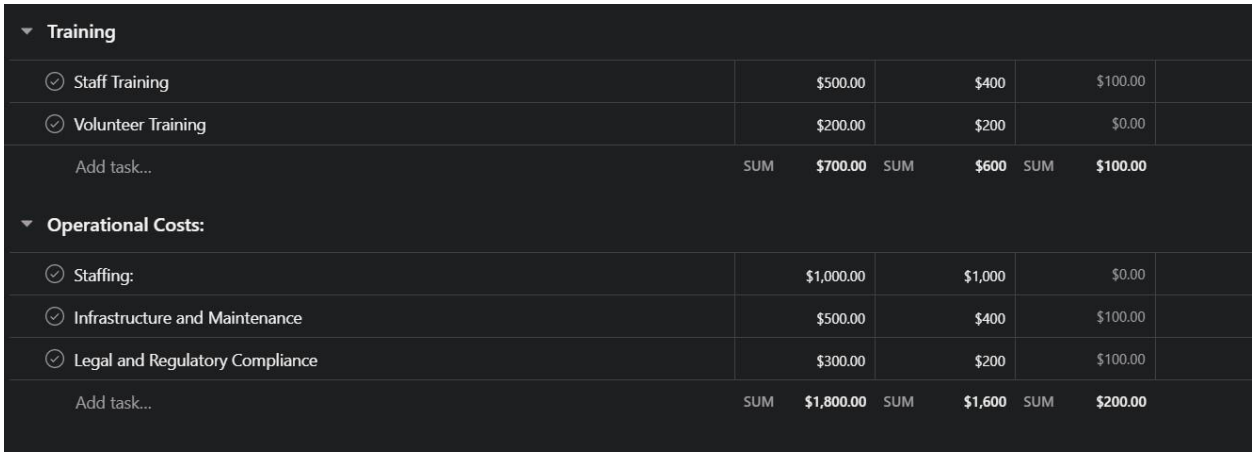
1. **Mobile App Development and Enhancement:**
    - Developing a user-friendly mobile app to facilitate easy donations and wider accessibility.
    - Continuously update the app for improved functionality, user experience, and additional features based on user feedback.
    - Implementing scheduling options and user-friendly interfaces to encourage consistent engagement.
2. **Expanded Partnerships and Outreach:**
    - Extend partnerships to grocery stores or supermarkets, implementing donation collection points.
    - Collaborate with local government bodies and non-profits to scale up the initiative's impact and reach.

- Expand community events and workshops, fostering deeper engagement and involvement.

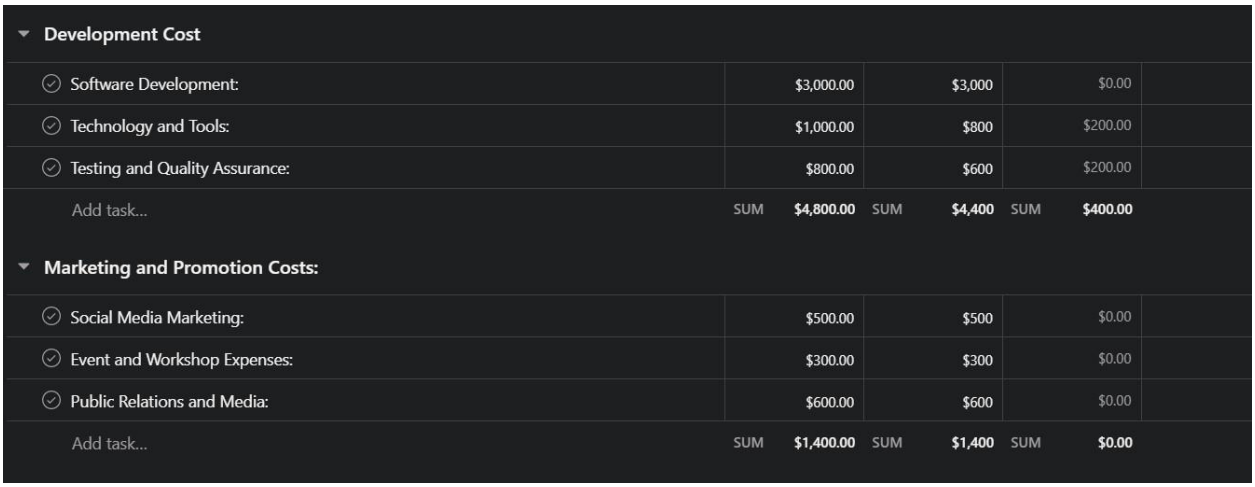
**Continuous (Ongoing and Evolving):**

1. **Sustained Online Engagement:**
    - Maintaining an active online presence with updated content, blogs, and email marketing campaigns.
    - Regularly communicate success stories, impact metrics, and updates to keep users engaged and motivated.
    - Monitoring and respond to user feedback, continuously improving online communication strategies.
2. **Partnership Strengthening and User Retention:**
    - Strengthening existing partnerships and seek new collaborations to expand reach and impact.
    - Implementing loyalty programs and incentives within the app to encourage user retention and consistent engagement.
    - Conducting regular user surveys to gather feedback and enhance the app's functionalities based on user preferences.
3. **Monitoring Impact and Amplifying Visibility:**
    - Utilizing robust analytics to track donations, measure impact, and visualize the collective effort of users.
    - Regularly publish impact reports, infographics, and success stories to showcase the initiative's tangible results.
    - Employing strategic PR and media outreach to continuously amplify the initiative's visibility and impact.

# 6. Cost and Profit Analysis



| Training | | | |
|---|---|---|---|
| Staff Training | $500.00 | $400 | $100.00 |
| Volunteer Training | $200.00 | $200 | $0.00 |
| Add task... | SUM **$700.00** | SUM **$600** | SUM **$100.00** |
| **Operational Costs:** | | | |
| Staffing: | $1,000.00 | $1,000 | $0.00 |
| Infrastructure and Maintenance | $500.00 | $400 | $100.00 |
| Legal and Regulatory Compliance | $300.00 | $200 | $100.00 |
| Add task... | SUM **$1,800.00** | SUM **$1,600** | SUM **$200.00** |



| Development Cost | | | |
|---|---|---|---|
| Software Development: | $3,000.00 | $3,000 | $0.00 |
| Technology and Tools: | $1,000.00 | $800 | $200.00 |
| Testing and Quality Assurance: | $800.00 | $600 | $200.00 |
| Add task... | SUM **$4,800.00** | SUM **$4,400** | SUM **$400.00** |
| **Marketing and Promotion Costs:** | | | |
| Social Media Marketing: | $500.00 | $500 | $0.00 |
| Event and Workshop Expenses: | $300.00 | $300 | $0.00 |
| Public Relations and Media: | $600.00 | $600 | $0.00 |
| Add task... | SUM **$1,400.00** | SUM **$1,400** | SUM **$0.00** |

**Total Expenses:**

- Development Cost = $4800
- Marketing and Promotion = $1400
- Training = $700
- Operational Cost = $1800

**Total Expenses =** $4800 + $1400 + $700 + $1800 = **$8700**

| Expected Revenue Generation | | | | |
|---|---|---|---|---|
| Revenue from Mobile app | $5,000.00 | | $5,000.00 | |
| Advertisement | $3,000.00 | | $3,000.00 | |
| Partnerships | $3,000.00 | | $3,000.00 | |
| Revenue from Website | $4,000.00 | | $4,000.00 | |
| Add task... | SUM | $15,000.00 | SUM | $15,000.00 |

**Revenue = $15,000**

**Profit =** Revenue - Total Expenses Profit = $15,000 - $8700 = **$6300**

Therefore, the profit from the software development project, considering the given expenses and assuming a revenue of $15,000, is $6300. This is the initial profit. As the popularity of this app increases, we expect the profit to increase gradually.

## 7. Reference

[1] https://www.researchgate.net/publication/336685653_Food_waste_-_a_general_overview_and_possible_solutions

[2] https://ideas.repec.org/a/zib/zbngws/v5y2021i1p17-20.html

[3] https://www.epa.gov/sustainable-management-food/united-states-2030-food-loss-and-waste-reduction-goal#goal

[4] https://www.winnowsolutions.com/