

A

Project Report

on

Waste Food Management System “FoodForAll”

Developed by

Hetavi Chaudhary – IT013

Priya Chauhan – IT015

Guided By

Prof. Ravindra Vyas

**Department of Information Technology
Dharmsinh Desai University**



**Department of Information Technology
Faculty of Technology, Dharmsinh Desai University
College Road, Nadiad – 387001**

April - 2022

**DHARMSINH DESAI UNIVERSITY
NADIAD-387001, GUJARAT**

CERTIFICATE



This is to certify that the project carried out in the subject of System Design Practice, entitled “**FoodForAll**” and recorded in this report is a bonafide report of work of

- | | | |
|---------------------|----------------|-------------------|
| 1) Hetavi Chaudhary | Roll No. IT013 | ID No: 19ITUOS002 |
| 2) Priya Chauhan | Roll No. IT015 | ID No: 19ITUBF019 |

of the Department of Information Technology, semester VI. They were involved in Project work during the academic year 2021 - 2022.

Prof. Ravindra Vyas
(Project Guide)
Department of Information Technology,
Faculty of Technology,
Dharmsinh Desai University, Nadiad Date:

Prof. (Dr.) Vipul K. Dabhi,
Head, Department of Information Technology,
DDU(Faculty of Tech., Dept. of IT)

Faculty of Technology,
Dharmsinh Desai University, Nadiad Date:

CANDIDATE'S DECLARATION



We declare that the pre-final semester report entitled "**FoodForAll – A Waste food management system**" is our own work conducted under the supervision of the guide **Prof. Ravindra Vyas**.

We further declare that to the best of our knowledge the report for B.Tech. VI semester does not contain part of the work which has been submitted either in this or any other university without proper citation.

Candidate's Signature
Hetavi Chaudhary
Student ID: 19ITUOS002

Candidate's Signature
Priya Chauhan
Student ID: 19ITUBFO19

ACKNOWLEDGMENT

On the very outset of this report, we would like to extend our sincere & heartfelt obligations towards all the personages who have directly or indirectly helped in this endeavor. Without their active guidance, help, cooperation & encouragement, we wouldn't have made the headway in the project.

We are ineffably indebted to our instructor **Prof. Ravindra Vyas** for his guidance, encouragement and conscientiousness to accomplish this project. We are extremely thankful for his support.

We would also like to express our gratitude to our Head, **Mr. Vipul Dabhi** along with Dharmsinh Desai University for giving us this opportunity.

Last but not least, we would also like to acknowledge with a deep sense of reverence, our gratitude towards our family members who have always supported us morally. Any omission in this brief acknowledgement does not mean lack of gratitude.

Thank You,
Hetavi Chaudhary
Priya Chauhan

TABLE OF CONTENT

1.	Introduction....	6
	Project Details: Broad specifications of the work entrusted to you....	6
	Purpose....	6
	Scope	6
	Objective....	6
	Technology and Literature Review....	6
2.	Project Management....	7
	Feasibility Study....	7
	Technical feasibility....	7
	Time Schedule feasibility....	7
	Operational feasibility....	7
	Implementation feasibility....	7
	Project Planning....	8
	Project Development Approach and Justification....	8
	Project Plan....	9
	Roles and Responsibilities....	9
3.	System Requirements Study....	10
	Study of Current System....	10
	Problems and Weaknesses of Current System....	10
	User Characteristics....	10
	Hardware and Software Requirements....	10
	Constraints....	10
4.	System Analysis....	11
	Requirements of New System....	11
	Functional Requirements....	11
	Non-functional Requirements....	17
5.	System Design....	20
	Use Case Diagram....	20
	Class Diagram	21

Sequence Diagram.....	22
Collaboration Diagram.....	24
State-Chart Diagram.....	25
Activity Diagram.....	26
Component Diagram.....	29
Deployment Diagram	30
 6. Implementation Plannin.....	31
Implementation Environment... ..	31
Coding Snippets... ..	31
 7. Testing.....	57
Testing Plan.....	57
Testing Strategy... ..	57
Testing Methods.....	57
Test Cases... ..	57
 8. User Manual... ..	60
 9. Limitation and Future Enhancement... ..	72
 10. Conclusion and Discussion... ..	73
Conclusions	73
Discussion.....	73
10.2.1 Self Analysis of Project Viabilities... ..	73
10.2.3 Problem Encountered and Possible Solutions... ..	73
10.3.3 Summary of Project work...	73
 11. Bibliography... ..	74

1. INTRODUCTION

PROJECT DETAILS

The waste food management system provides the functionality to users to donate their leftover food to NGO. After that NGO distribute that food to needy people. Also any user can login as consumer and made request for food to NGO.

PURPOSE

Main purpose of system is for those people who throwaway leftover food due to lack of time for going and donating food by their selves.

SCOPE

This system allows users to send request to NGO for collecting their leftover food and also allow consumer to make request for food to NGO.

OBJECTIVE

Main motto of ours is to provide a simple and efficient one stop helps a person needs for his or her to donate their leftover food and made request for food. NGO is available in between both donor and consumer to help them

TECHNOLOGY AND LITERATURE REVIEW

We used simple html, css, javascript and bootstrap for the frontend as these are all supported by any good browser. For the backend we used the Django framework which is written in python. We choose Django as it is very secure and efficient.

2. PROJECT MANAGEMENT

FEASIBILITY STUDY

Technical Feasibility

After considering our project functionalities and technical needs we decided to go with the following technologies -

Frontend - Basic html, css, javascript and bootstrap

Backend - Django framework

Database - MySQL

All the above mentioned technologies are open sources therefore it is feasible to use with the knowledge of basics in each of them it would be easy for us to manage the project too.

Time Schedule Feasibility

The planning for our project completion is as follows:

Initially we gathered & analyzed all the requirements by the beginning of January 2022. We prepared the SRS document and the GUI design right after that. This was followed by the required diagrams. For coding and unit testing 4 weeks and for system and integration testing another 2 weeks were covered. We are ready for its demonstration in the submission week along with the report. Being a 2 members team we were able to complete our project in the estimated time.

Operational Feasibility

In the current scenario where the virtual world is the new normal, our project will excel greatly along the frequent web users who want to avail services online. We have kept the website pretty simple and handy so everyone can take advantage of it and use it without any inconvenience.

Implementation Feasibility

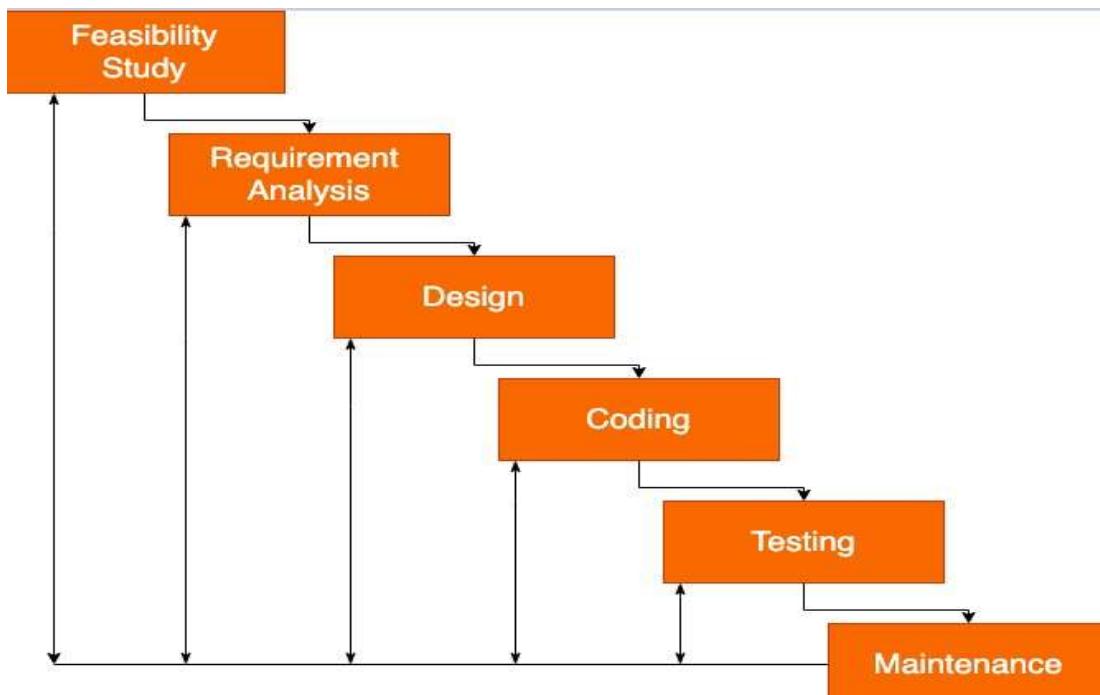
The Functional requirements of our project will be easily fulfilled by the technologies we have mentioned. All the technologies we are using are easily supported by all the browsers. Hence, the implementation is feasible.

PROJECT PLANNING

Project Development Approach and Justification

The software development approach we used for our project: Iterative Waterfall Model This includes the stages like:

1. Feasibility Study
2. Requirement Analysis
3. Design
4. Coding
5. Testing
6. Maintenance



We choose this approach because it gives us the liberty to get back to a stage if it needs some redo. Our entire approach was trial and error and learn as you need. By choosing the iterative waterfall we get the liberty of flexibility and the risk management can be done easily.

Project Plan

Sr. No	Documents	Timeline
1	Project Definition, Submission and approval	December
2	System Design	January
3	Implementation	January
4	Testing	March
5	Documentation	April

Roles and Responsibility

Name	Analysis	Design	Coding	Testing	Documentation	Maintenance
------	----------	--------	--------	---------	---------------	-------------

Hetavi	✓	✓	✓	✓	✓	✓
Priya	✓	✓	✓	✓	✓	✓

3. SYSTEM STUDY REQUIREMENT

STUDY OF CURRENT SYSTEM

Websites like feedingindia.org, is one of the examples of current system like ours. It is website by Zomato which helps restaurants to collect their leftover food and distributes it to needy poor people.

PROBLEMS AND WEAKNESS OF CURRENT SYSTEM

All the current sites working for food donation are in different fields like they are either e commerce websites or centralized websites. Our website is for local users which associated with any one local NGO.

USER CHARACTERISTICS

1. NGO- NGO is registered when website is developed. NGO can see all food donation and food request and can confirm and cancel appropriately.
2. Doner- User can register themselves as Doner for donating food.
3. Consumer- User can register themselves as consumer for making food request from NGO.

HARDWARE AND SOFTWARE REQUIREMENTS

For all the actors the basic software requirements are as follows

- Good Browser
- Good internet connection

For all the actors the basic hardware requirements are as follows

- Device to run the web application

CONSTRAINTS

- As it is a web-based application, constantly needs internet
- Response time can vary depending on the browser

4. SYSTEM ANALYSIS

REQUIREMENTS OF NEW SYSTEM

Functional requirements:

1.1.Doner Features

1.1.1. Create request for donating food

Description and Priority:

This feature provide facility to user to make request for collecting their leftover food.

Stimulations/Response Sequences:

First user has to click on donate food button. The system prompts the user to fill out details about the food. User have to fill all details about the food. User have to give all the detail about the food like type of food, quantity of food, how many days it can be fresh through filling out the form.

Functional Requirements

REQ-1: A button or link for donate food must be available.

REQ-2: A form for all details about food.

REQ-3: A table in the database must be created for all requests made by user.

1.1.2. View pervious request

Description and Priority:

In this feature users can see their previous donations.

Stimulations/Response Sequences:

First user has to click on view previous donations. The system provides the details about particular users all past donations.

Functional Requirements

REQ-1: A button or link for view previous donations must be there.

REQ-2: A table in the database must be contain all past donations of all users.

1.1.3. See request status

Description and Priority:

In this feature users can see their donation request status whether it is confirmed or pending. They notify whether NGO confirm or cancel their donation request.

Stimulations/Response Sequences:

First user has to click on view donations request status. The system provides the details about status of donation request.

Functional Requirements:

REQ-1: A button or link for view donations request status must be there.

REQ-2: A table in the database must be contain donation request of all users.

1.1.4. View gallery

Description and Priority:

In this feature users can see gallery of NGO. They can see various photos of NGO which NGO upload on website during donating food to needy people.

Stimulations/Response Sequences:

First user has to click on gallery. Then system give all the photos which NGO share are displayed on user's screen.

Functional Requirements:

REQ-1: A button or link for view gallery.

REQ-2: A table in the database must be table which have all the image which NGO share.

1.2.NGO Features

1.2.1. View donation or food request

DDU(Faculty of Tech., Dept. of IT)

Description and Priority:

This feature helps NGO to view all the requests for food donation from the doners and food request from consumers. So, they can confirm their requests.

Stimulations/Response Sequences:

First NGO have to click on view all requests button. System gives the list of all pending food donation requests.

Functional Requirements:

REQ-1: A button or link for view pending request must be available.

REQ-2: A table in the database must be maintain for all pending requests.

1.2.2. Confirm or cancel donation or food request

Description and Priority:

This feature helps NGO to confirm or cancel the requests for collecting food from the doners and food request from consumers.

Stimulations/Response Sequences:

First NGO have to click on view all requests button. After that they have to click on confirm request or decline request. If NGO click on confirm request than confirmation Mail has been send to particular user who send the request for donating food.

Functional Requirements:

REQ-1: A button for confirm request and decline request must be available.

REQ-2: A table in the database must be maintain for confirmed requests.

1.2.3. View available food

Description and Priority:

This feature provides the list of all available food to NGO.

Stimulations/Response Sequences:

First NGO have to click on view available food. After that system provide the list of all available food to NGO.

Functional Requirements:

REQ-1: A button or link must be available for view available food.

REQ-2: A table in the database must be created for storing list of all foods.

1.2.4. View list of all previous donations

Description and Priority:

This feature provides the list of all previous donations to the NGO.

Stimulations/Response Sequences:

First NGO have to click on view list of donations button or link. After that system provide the list of all the donations of the food to NGO.

Functional Requirements:

REQ-1: A button or link must be available for view list of donations.

REQ-2: A table in the database must be created for storing list of all donations.

1.2.5. Manage gallery

Description and Priority:

In this feature NGO can share the photos of donating food to needy people with users who donate the food. NGO can upload and delete the photos on their page.

Stimulations/Response Sequences:

First NGO has to click on upload photo option. After that they can upload the photos from their local device.

Functional Requirements:

REQ-1: A button or link must be available for upload photo.

REQ-2: A table in the database must be created for storing all photos.

1.2.6. Manage feedback

Description and Priority:

In this NGO can view the all feedbacks which is given by the food doners.

Stimulations/Response Sequences:

First NGO have to click on view feedback link or button. System loads all the given feedback of users on the NGO's site.

Functional Requirements:

REQ-1: A button or link must be available for view feedback.

REQ-2: A table in the database must be created for storing all feedback.

1.3.Consumer Features

1.3.1. Create food request

Description and Priority:

This feature provide facility to user to make request for food.

Stimulations/Response Sequences:

First user has to click on select food button. The system prompts the all-available food to user. User have to select food and quantity of as his/her requirements. All selected food selected food added in cart. After that user have to click on confirm order for ordering food.

Functional Requirements

REQ-1: A button or link for select food must be available.

REQ-2: A button or link for my cart must be available.

REQ-3: A button or link for confirm order must be available.

REQ-4: A table in the database must be created for manage cart.

1.3.2. View previous requests

Description and Priority:

In this feature users can see their previous orders.

Stimulations/Response Sequences:

First user has to click on view previous orders. The system provides the details about particular users all past orders.

Functional Requirements

REQ-1: A button or link for view previous orders must be there.

REQ-2: A table in the database must be contain all past orders of all users.

1.3.3. See request status

Description and Priority:

In this feature users can see their order request status whether it is confirmed or pending. They notify whether NGO confirm or cancel their order request.

Stimulations/Response Sequences:

First user has to click on view order request status. The system provides the details about status of order request.

Functional Requirements:

REQ-1: A button or link for view order request status must be there.

REQ-2: A table in the database must be contain order request of all users.

1.3.4. Give feedback

Description and Priority:

In this feature users can give their feedback about their experience.

Stimulations/Response Sequences:

First user clicks on give or share your feedback button or link. The system prompts the user to fill the form for feedback.

Functional Requirements:

REQ-1: A button or link is must be there for give feedback.

REQ-2: A table in the database must be created for store feedbacks from all users.

1.3.5. View gallery

Description and Priority:

In this feature users can see gallery of NGO. They can see various photos of NGO which NGO upload on website during donating food to needy people.

Stimulations/Response Sequences:

First user has to click on gallery. Then system give all the photos which NGO share are displayed on user's screen.

Functional Requirements:

REQ-1: A button or link for view gallery.

REQ-2: A table in the database must be table which have all the image which NGO share.

1.4. Common Features

1.4.1. Login and register

Description and Priority:

This feature creates an account for new user and give log in option for existing users.

For donating food and collecting food user have an account and must be logged into it. This feature adds the user details in the database. This is an important feature so it has high priority.

Stimulations/Response Sequences:

User first clicks on the button for registration process. The system then prompts the user to fill out his/her first name, last name, email address, address and their password. User fill all the fields. System validates the user's information and create new account for the user.

Functional Requirements:

- REQ-1: A button for register and log in must be available.
- REQ-2: A table in the database must be created for all users.

1.4.2. Manage profile

Description and Priority:

In this feature consumers and doners can manage their profiles. They can change their information like phone number, address, profile photo, etc.

Stimulations/Response Sequences:

First users have to click on my account link or button. After that they can see the various option related to managing their profile. They can view and update it.

Functional Requirements:

- REQ-1: A button or link for manage profile.
- REQ-2: A table in the database must be created for all users.

1.4.3. Change password

Description and Priority:

In this feature user can change their password for their account. consumer and donor both can change their password.

Stimulations/Response Sequences:

First user has to click on my account button or link after that he/she have to click on change password button or link. After that system provides the form for changing password to user.

Functional Requirements:

- REQ-1: A button or link for my account must be available.

REQ-2: A button or link for change password must be available.

Non-Functional Requirements

1.1.Performance Requirements

The changes if any made should be reflected automatically in the next screens. Appearance of user interface is very user friendly.

1.2.Safety Requirements

This application is password protected. Any updation is done by admin side only. User can access their account providing their id password.

1.3.Software Requirements

The necessary qualities of software products are:

1.3.1. Security

This application is password protected. Any updation is done by admin side only. User can access their account providing their id password.

1.3.2. Maintainability

The application is to be designed so that it is easily maintained. Also, it should allow incorporating new requirements in any module of the system.

1.3.3. Portability

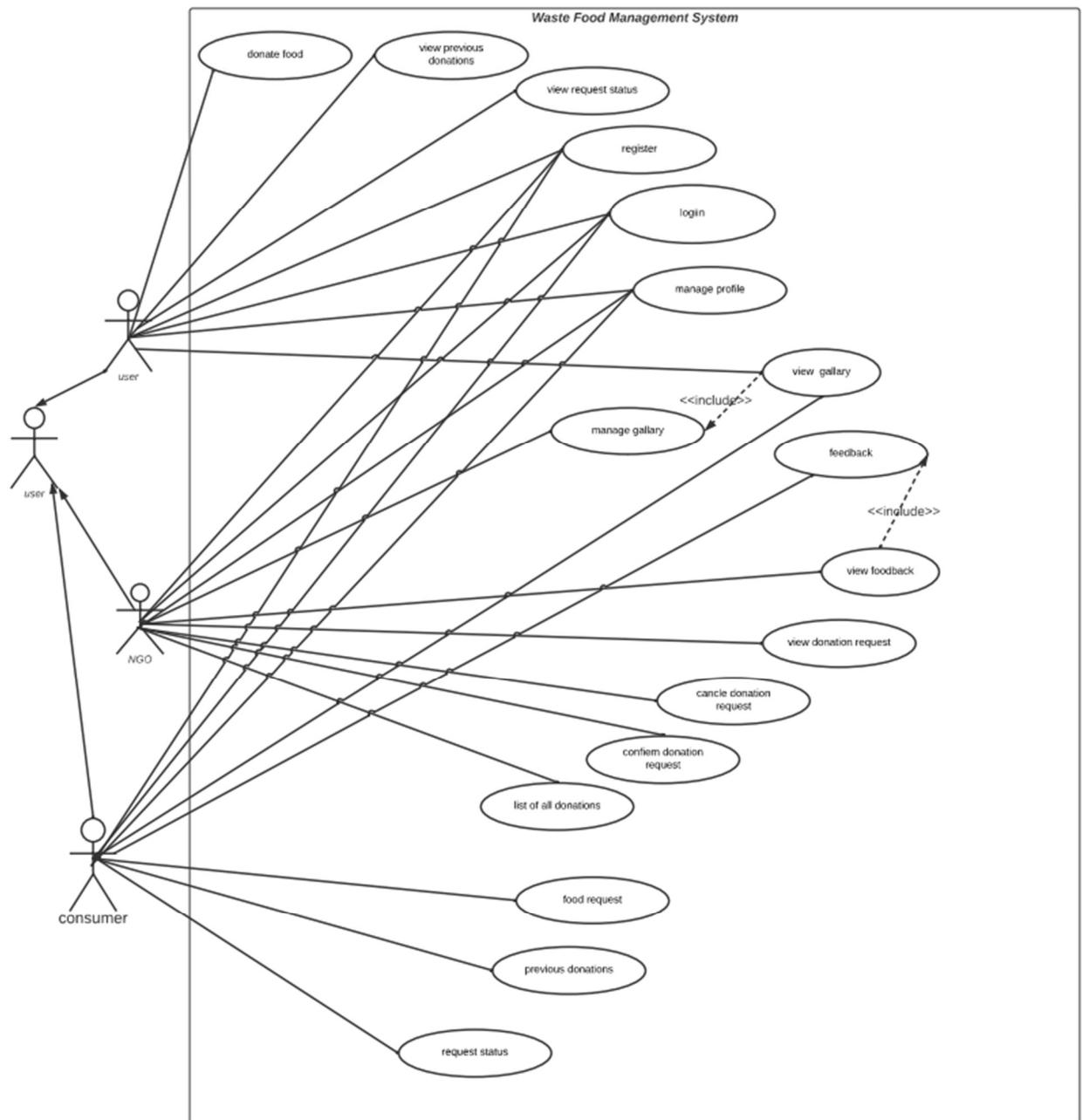
The application will be easily portable on any window-based system.

1.4.Business Rules

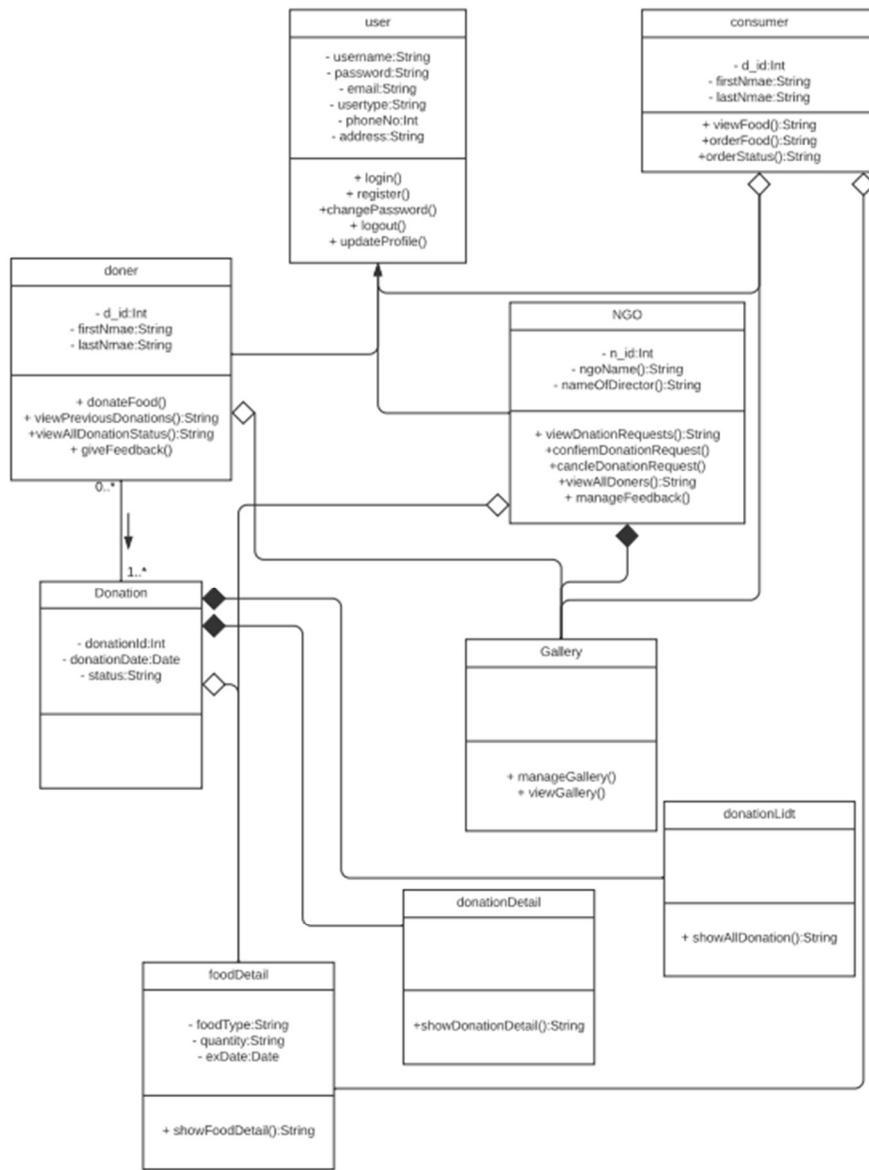
Only admin can change in database. The doner can only made request for donating food and NGO can only accept or decline request for collecting food. Admin cannot made any request for donating food or cannot confirm request for collecting food.

5. SYSTEM DESIGN

Use Case Diagram

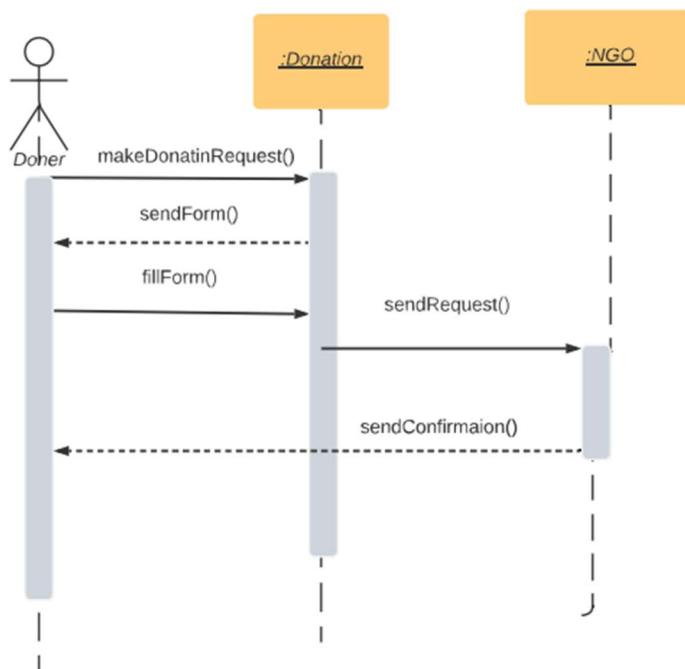


1.1. Class Diagram

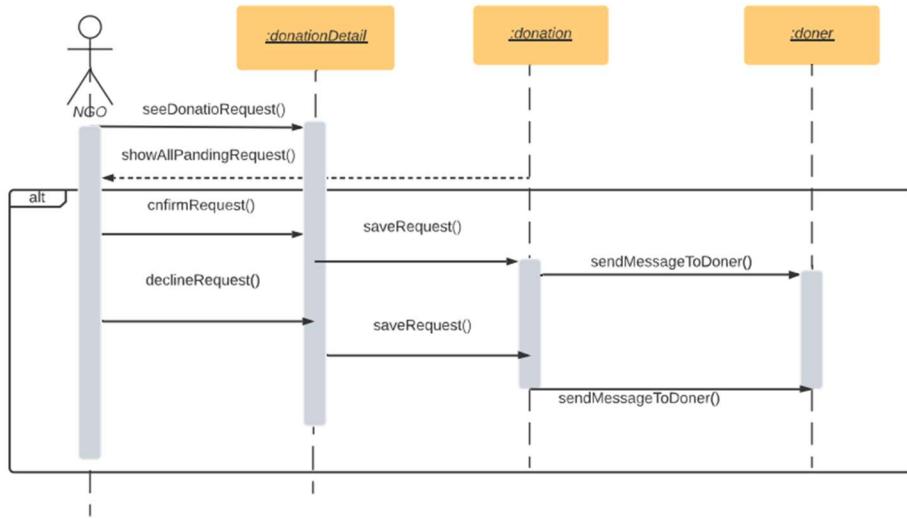


1.2.Sequence Diagram

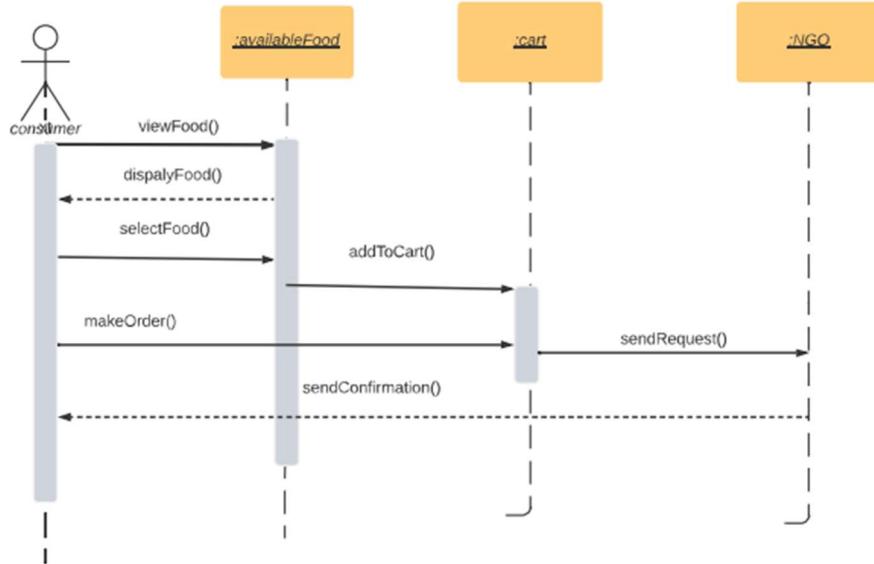
Make donation:



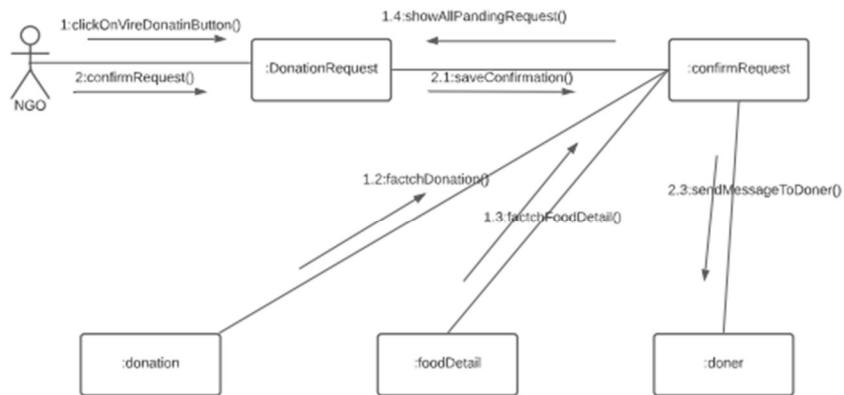
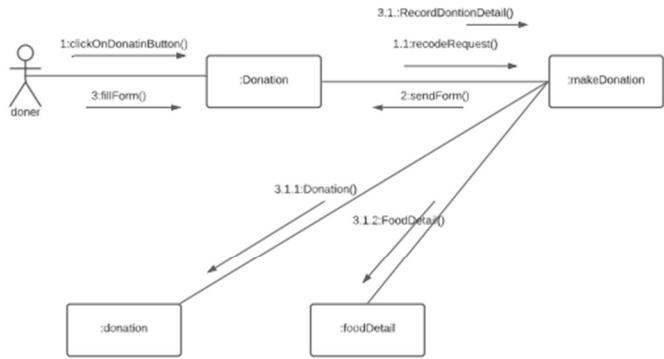
Confirm or cancel request:



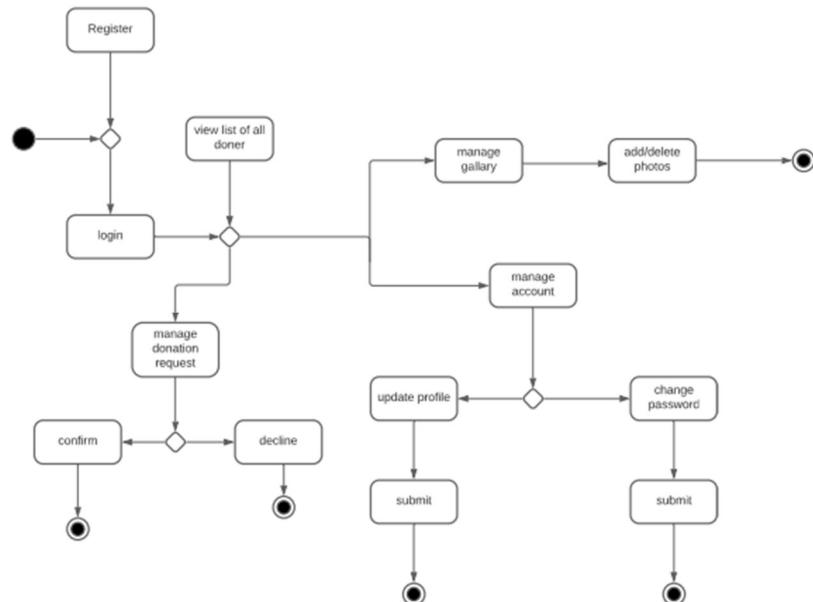
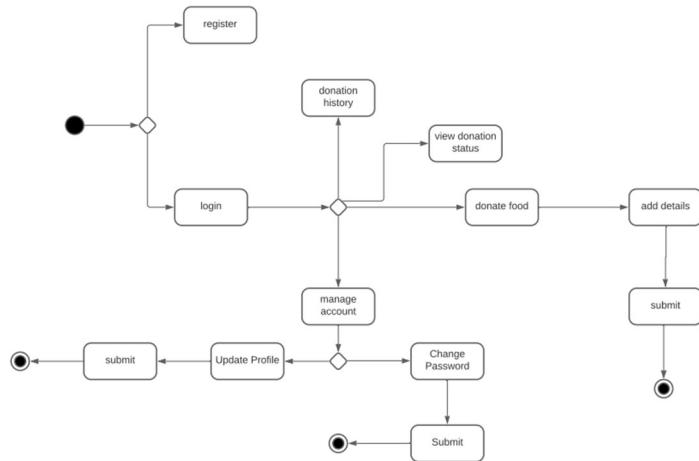
Make food request:



1.3.Collaboration Diagram

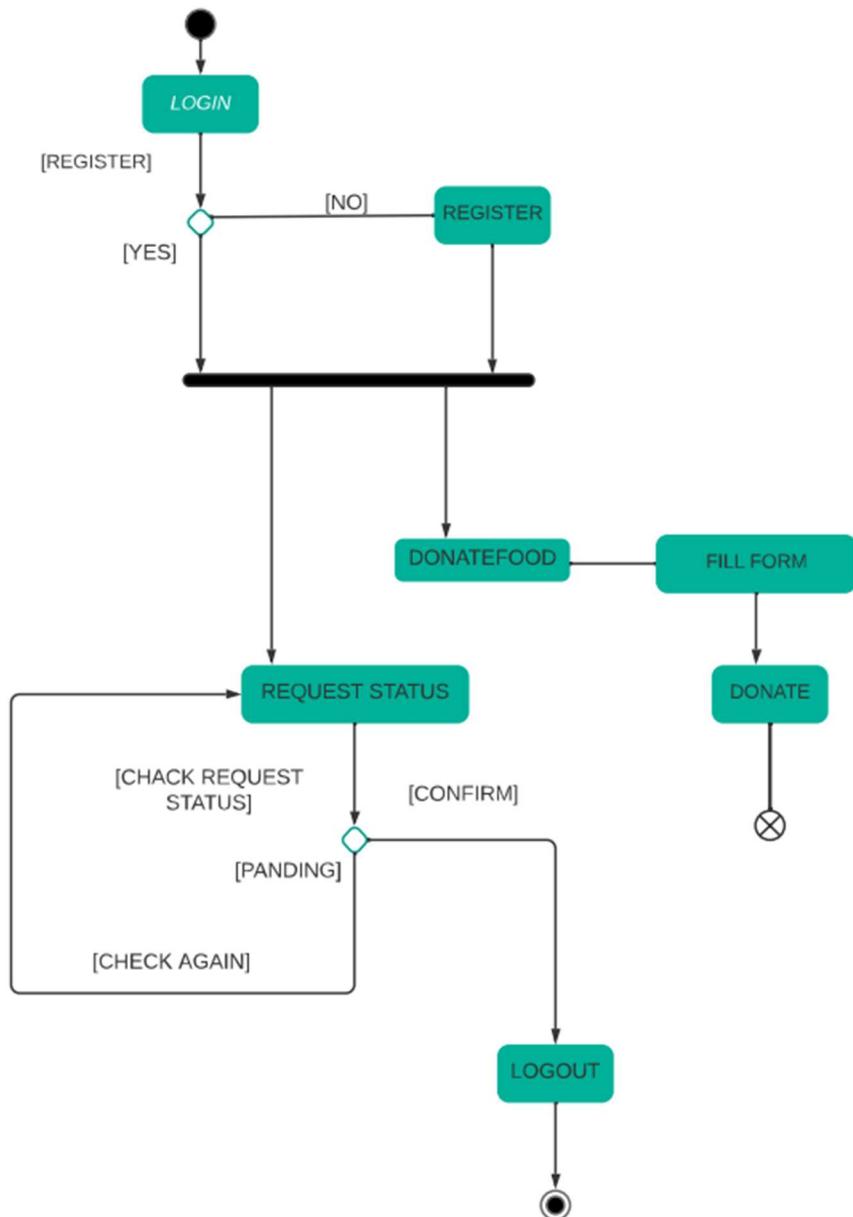


1.4.State-Chart Diagram

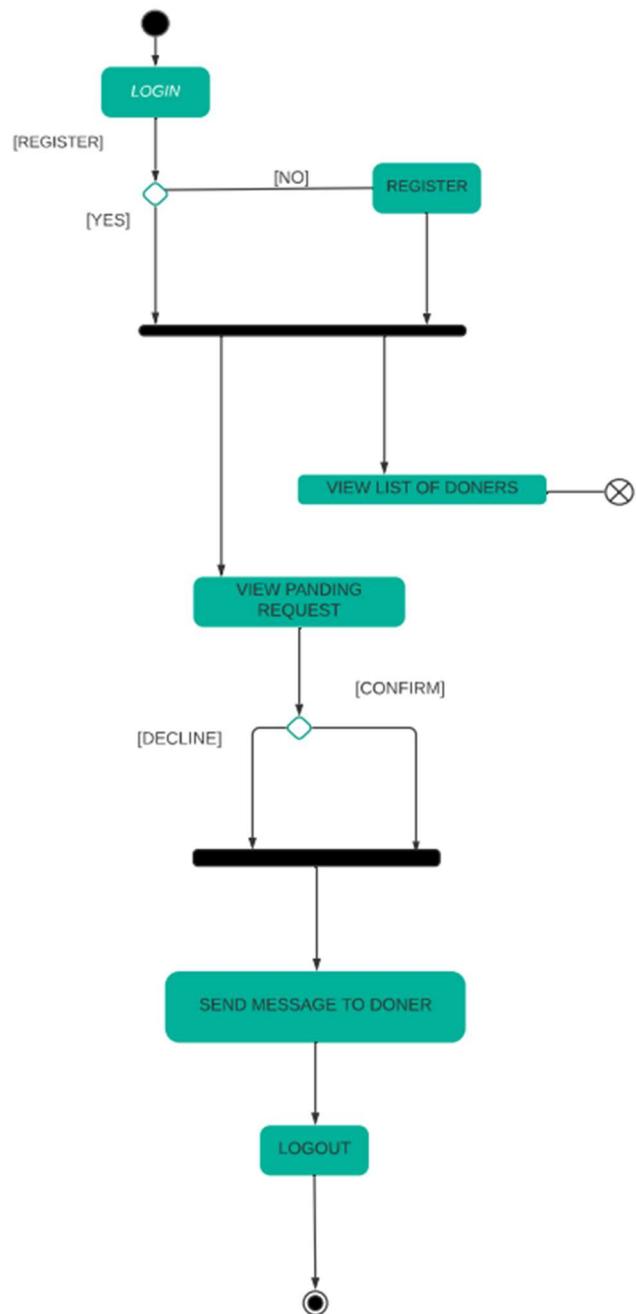


1.5.Activity Diagram

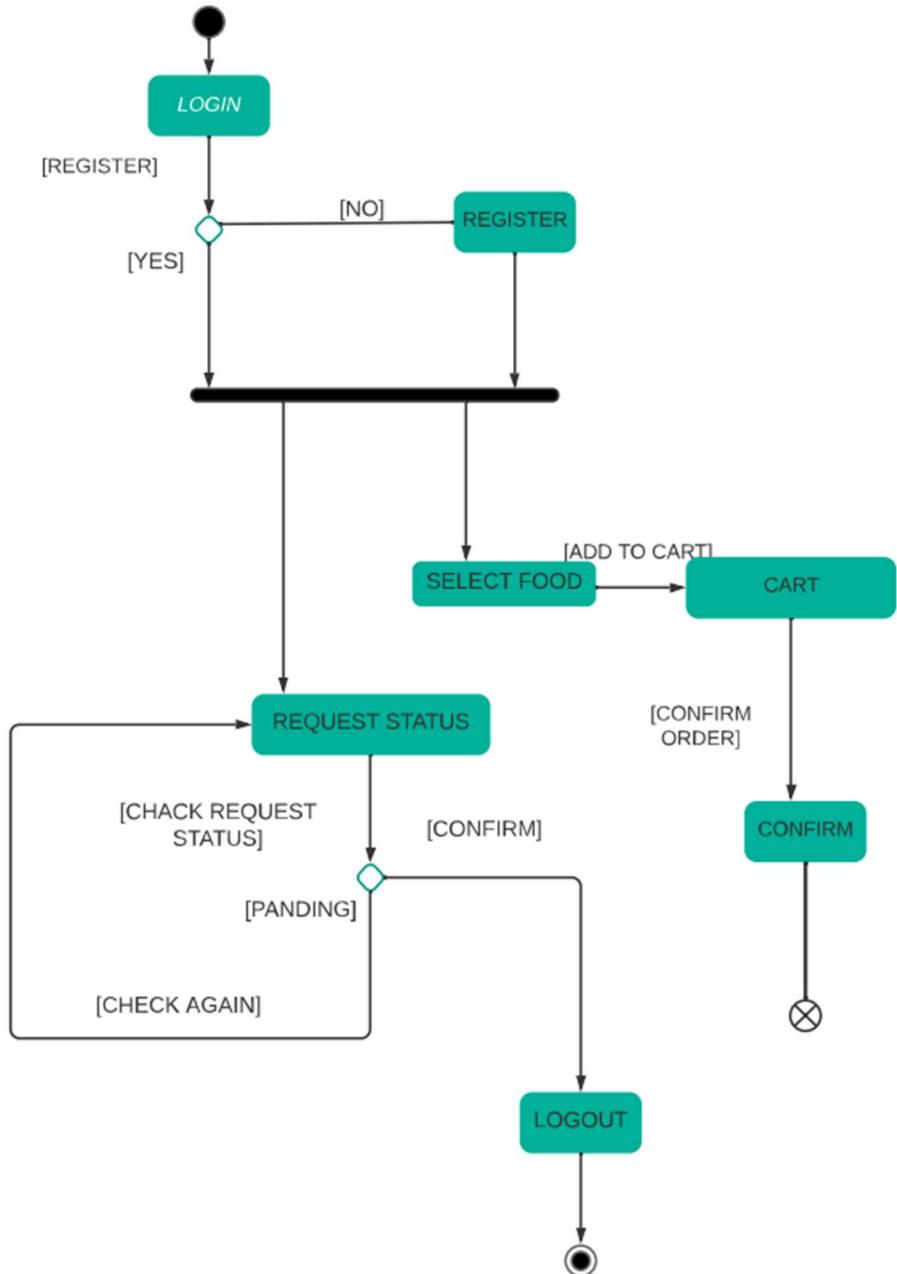
Make food donation request and check request status:



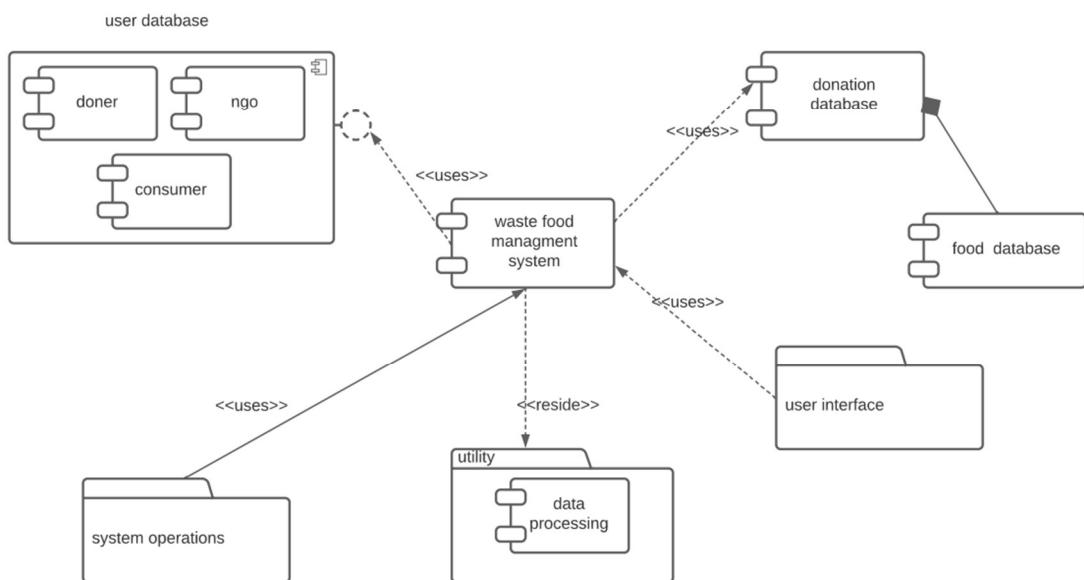
View and manage donation request:



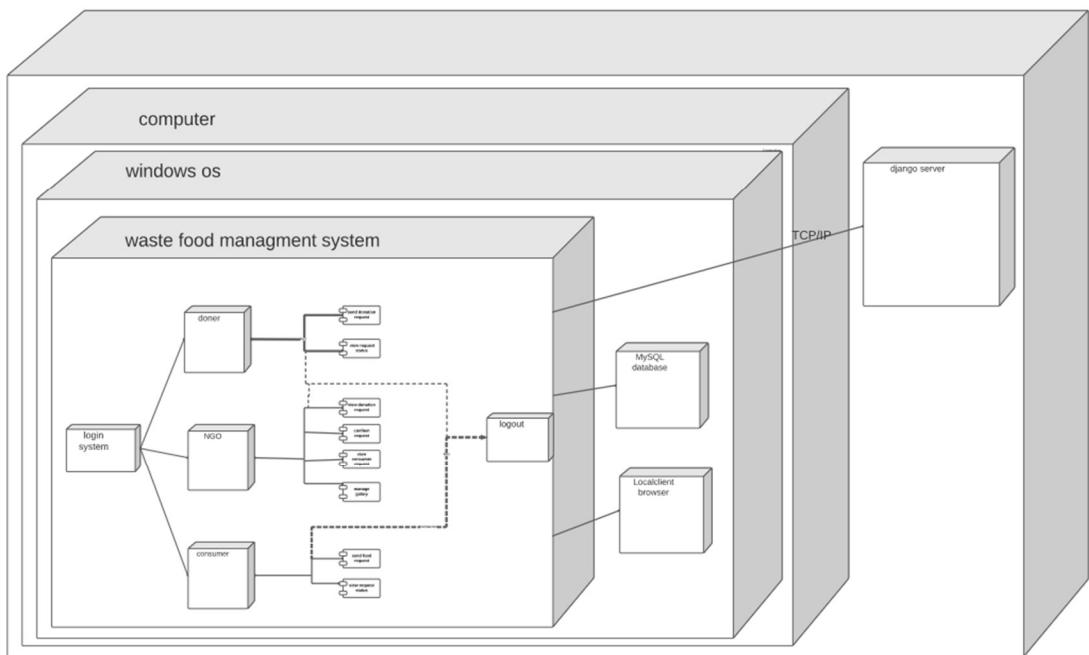
Select food and make order:



1.6.Component Diagram



1.7.Deployment Diagram



6. IMPLEMENTATION PLANNING

IMPLEMENTATION ENVIRONMENT:

During the complete implementation we have worked on Visual Studio Code (VSCode). Visual Studio Code is a free source-code GUI (Graphical User Interface) editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality. It also provides extension to work in a team with a multi user experience. Even compiling and deploying of the application was done easily with the help of VSCode.

For the database implementation, we have used MySQL. MySQL is a powerful, open-source object-relational database system Specially designed for Django Related Web Projects.

CODING SNIPPETS

While writing our code we took the utmost care to follow the basic coding standards while writing a JavaScript code like,

- Following proper naming conventions of local variables, global variables, constants and functions.
- Doing proper indentation.
- Proper error handling.
- Adding comments for better understanding.
- Proper Syntax

Here are some of the code snippets written during implementation,

Front End Snippets

//Home.html

```
<!DOCTYPE html>
<html style="font-size: 16px;">

<head>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta charset="utf-8">
    <meta name="keywords" content="DiningOn The Lake, breakfast, 01, 02, 03, 04, 05, 06, Delivery services, Fast Food, Burger Menu, Delivery Services, Subscribe, location, follow us">
    <meta name="description" content="">
    <meta name="page_type" content="np-template-header-footer-from-plugin">
    <title>FoodForAll</title>
    <link rel="stylesheet"
        href="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/dist/css/bootstrap.min.css"
        integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm"
        crossorigin="anonymous">
        <link rel="stylesheet" href="/static/css/userBase.css">
        <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
        <link rel="stylesheet" href="/static/css/nicepage.css">
        <link rel="stylesheet" href="/static/css/Page-2.css">

        <meta name="generator" content="Nicepage 4.2.6, nicepage.com">
        <link id="u-theme-google-font" rel="stylesheet"
            href="https://fonts.googleapis.com/css?family=Roboto:100,100i,300,300i,400,400i,500,500i,700,700i,900,900i|Open+Sans:300,300i,400,400i,600,600i,700,700i,800,800i">
            <link id="u-page-google-font" rel="stylesheet"
            href="https://fonts.googleapis.com/css?family=Ubuntu:300,300i,400,400i,500,500i,700,700i|Kalam:300,400,700|Alata:400">
            <script type="application/ld+json">
            {
```

```

        "@context": "http://schema.org",
        "@type": "Organization",
        "name": ""
    }
</script>
<meta name="theme-color" content="#478ac9">
<meta property="og:title" content="Page 2">
<meta property="og:type" content="website">
</head>

<body class="u-body">
    <div class="pic">
        <div class="container">
            <nav class="navbar abx navbar-expand-lg navbar-light" style="background-color: #e3f2fd;">
                <a class="navbar-brand" href="/">FoodForAll</a>
                <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
                    <span class="navbar-toggler-icon"></span>
                </button>

                <div class="collapse navbar-collapse" id="navbarSupportedContent">
                    <ul class="navbar-nav ml-auto">

                        <!-- <li class="nav-item active">
                            <a class="nav-link" href="gallery">Gallery</a>
                        </li> -->
                        <li class="nav-item active">
                            <a class="nav-link" href="login">Login <span class="sr-only">(current)</span></a>
                        </li>
                        <li class="nav-item active">
                            <a class="nav-link" href="register">SignUp <span class="sr-only">(current)</span></a>
                        </li>

                    </ul>
                </div>

```

```

        </nav>
    </div>
</header>
<section class="u-align-left u-clearfix u-section-1" id="carousel_560d">
    <div class="container">
        <div class="u-clearfix u-sheet u-sheet-1">
            <!-- <div class="u-clearfix u-gutter-32 u-layout-wrap u-layout-
wrap-1"> -->
            <div class="u-gutter-0 u-layout">
                <div class="u-layout-row">
                    <div class="u-container-style u-layout-cell u-size-60 u-
layout-cell-1">
                        <div class="u-container-layout u-container-layout-
1">
                            <br><br>
                            <h1 class="u-custom-font u-font-ubuntu u-text u-
text-1" data-animation-name="fadeIn" data-animation-duration="1000" data-animation-
direction="Left" data-animation-delay="500">
                                <span class="u-text-palette-2-base">
                                    <font class="u-text-palette-2-base"><b> make food available
for all</b>
                                </font>
                                </span>
                            </h1>
                            <br>
                            <h2 class="pl-3">
                                <p class="pl-3">when you are wasting food
someone is starving&nbsp; think.....!</p>
                            </h2>
                            <br>
                            <a href="gallery" class="u-active-grey-90 u-
border-none u-btn u-btn-round u-button-style u-color-scheme-summer-time u-color-
style-multicolor-1 u-hover-grey-90 u-palette-2-base u-radius-50 u-text-active-white
u-text-body-alt-color u-text-hover-white u-btn-1">Gallary</a>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>

```

```

        <div class="u-image u-image-square u-image-1" data-animation-
name="fadeIn" data-animation-duration="1000" data-animation-direction="Down" data-
animation-delay="500" data-image-width="2560" data-image-height="1920"></div>
        <div class="u-expanded-width u-gallery u-layout-grid u-lightbox u-
no-transition u-show-text-on-hover u-gallery-1">
            <div class="u-gallery-inner u-gallery-inner-1">
                <div class="u-effect-fade u-effect-hover-zoom u-gallery-item
u-shape-rectangle">
                    <div class="u-back-slide" data-image-width="626" data-
image-height="417">
                        
                    </div>
                    <div class="u-over-slide u-over-slide-1">
                        <h3 class="u-gallery-heading"></h3>
                        <p class="u-gallery-text"></p>
                    </div>
                </div>
                <div class="u-effect-fade u-effect-hover-zoom u-gallery-item
u-shape-rectangle">
                    <div class="u-back-slide" data-image-width="626" data-
image-height="646">
                        
                    </div>
                    <div class="u-over-slide u-over-slide-2">
                        <h3 class="u-gallery-heading"></h3>
                        <p class="u-gallery-text"></p>
                    </div>
                </div>
                <div class="u-effect-fade u-effect-hover-zoom u-gallery-item
u-shape-rectangle">
                    <div class="u-back-slide" data-image-width="626" data-
image-height="417">
                        
                    </div>
                    <div class="u-over-slide u-over-slide-3">
                        <h3 class="u-gallery-heading"></h3>
                        <p class="u-gallery-text"></p>

```

```

                </div>
            </div>
            <div class="u-effect-fade u-effect-hover-zoom u-gallery-item
u-shape-rectangle">
                <div class="u-back-slide">
                    
                </div>
                <div class="u-over-slide u-over-slide-4">
                    <h3 class="u-gallery-heading"></h3>
                    <p class="u-gallery-text"></p>
                </div>
            </div>
        </div>
    </div>
</section>
</div>
<section class="u-align-center u-clearfix u-grey-10 u-section-2"
id="carousel_987a">
    <div class="u-clearfix u-sheet u-sheet-1">
        <div class="u-palette-2-base u-shape u-shape-circle u-shape-1" data-
animation-name="fadeIn" data-animation-duration="1000" data-animation-
direction="Right" data-animation-delay="0"></div>
        
        <div class="u-align-left u-container-style u-group u-white u-group-1"
data-animation-name="fadeIn" data-animation-duration="1000" data-animation-
direction="Left" data-animation-delay="250">
            <div class="u-container-layout u-container-layout-1">
                <h2 class="u-text u-text-default u-text-1"> Our Aim</h2>
                <p class="u-text u-text-2">Wasting food is a common problem in
our society. Food waste management is crucial since it can improve our environmental
and economic sustainability.</p>
                <a href="about" class="u-active-grey-90 u-btn u-btn-round u-
button-style u-color-scheme-summer-time u-color-style-multicolor-1 u-hover-grey-90
u-palette-2-base u-radius-50 u-text-active-white u-text-body-alt-color u-text-hover-
white u-btn-1">Know more</a>

```

```

        </div>
    </div>
</div>
</section>
<section class="u-align-center u-clearfix u-image u-section-3"
id="carousel_979f" data-image-width="1920" data-image-height="1052">
    
        <div class="u-list u-list-1">
            <div class="u-repeater u-repeater-1">
                <div class="u-align-left u-container-style u-grey-80 u-list-item u-
opacity u-opacity-55 u-repeater-item u-shape-rectangle u-list-item-1">
                    <div class="u-container-layout u-similar-container u-valign-top
u-container-layout-1">
                        <div class="u-align-center u-container-style u-group u-
palette-2-base u-radius-50 u-shape-round u-group-1" data-animation-name="fadeIn"
data-animation-duration="1000" data-animation-direction="Left">
                            <div class="u-container-layout u-valign-middle u-
container-layout-2">
                                <h1 class="u-custom-font u-text u-text-1">01</h1>
                            </div>
                        </div>
                        <h5 class="u-text u-text-body-alt-color u-text-2" data-
animation-name="fadeIn" data-animation-duration="1000" data-animation-
direction="Right">Donate Food</h5>
                        <p class="u-text u-text-blue-30 u-text-3">If you have food
you can always donate your food here. Kindly register as donor.</p>
                        <a href="donerregister" class="u-active-none u-border-2 u-
border-active-white u-border-hover-white u-border-white u-btn u-button-style u-
hover-none u-none u-text-body-alt-color u-text-hover-grey-30 u-btn-1">Be Doner</a>
                    </div>
                </div>
                <div class="u-align-left u-container-style u-grey-80 u-list-item u-
opacity u-opacity-55 u-repeater-item u-shape-rectangle u-list-item-2">
                    <div class="u-container-layout u-similar-container u-valign-top
u-container-layout-3">
                        <div class="u-align-center u-container-style u-group u-
palette-2-base u-radius-50 u-shape-round u-video-cover u-group-2" data-animation-
name="fadeIn" data-animation-duration="1000" data-animation-direction="Left">

```

```


# 02



##### Food Collector



Collector plays major
role in this chain. They collect food from doner and store and give it to the
consumer.



# 03



##### Consumer



Consumer can order
food from collector If they needed food for needy people.

Be Consumer


```

```

    </section>
<section class="u-clearfix u-grey-10 u-section-4" id="carousel_a2bd">
    <div class="u-palette-2-base u-preserve-proportions u-shape u-shape-circle
u-shape-1" data-animation-name="fadeIn" data-animation-duration="1000" data-
animation-direction="Down" data-animation-delay="500"></div>
    <div class="u-palette-2-base u-shape u-shape-rectangle u-shape-2" data-
animation-name="fadeIn" data-animation-duration="1000" data-animation-
direction="Right"></div>
    
    <div class="u-align-left u-container-style u-group u-white u-group-1" data-
animation-name="fadeIn" data-animation-duration="1000" data-animation-direction="Up"
data-animation-delay="750">
        <div class="u-container-layout u-container-layout-1">
            <h2 class="u-text u-text-1">Why this is needed?</h2>
            <p class="u-text u-text-2">A study by Food Ministry, Government of
India found that about 20% of food at social events such as weddings goes waste.
Food waste also occurs at parties, functions and restaurants. This is a sad state of
affairs when one considers that
                two lakh street children in Mumbai go to sleep hungry every
day</p>
            <a href="https://thecsrjournal.in/food-wastage-india/" class="u-
active-grey-90 u-btn u-btn-round u-button-style u-color-scheme-summer-time u-color-
style-multicolor-1 u-hover-grey-90 u-palette-2-base u-radius-50 u-text-active-white
u-text-body-alt-color u-text-hover-white u-btn-1">learn more</a>
        </div>
    </div>
</section>

<section class="mainbox my-3">
    <div class="">
        <div class="row ">
            <div class="col-sm my-3 text-center">
                <a class="navbar-brand" href="/">
                    <p class="text-dark">FoodForAll</p>
                </a>
            </div>
        </div>
    </div>
</section>

```

```

        </div>
        <div class="col-sm my-3">
            <p class="who text-dark">Who we are</p>

            <li class="ab ">
                <a href="about">About us</a>
            </li>
            <li class="ab "><a href="contact">Contact us</a></li>

        </div>
        <div class="col-sm my-3">
            <p class="who text-dark">Get Invoved</p>
            <li class="ab ">
                <a href="login">Donate Food</a>
            </li>
        </div>
        <div class="col-sm my-3">
            <div class=" text-dark">Social</div>
            <br>
            <div class="">
                <a href="#" class="fa fa-facebook"> </a>&ampnbsp
                <a href="#" class="fa fa-twitter"> </a>&ampnbsp
                <a href="#" class="fa fa-instagram"> </a>&ampnbsp
            </div>
            </div>
        </div>
    </div>
</section>
<footer class="footer bg-dark">
    <div class="container text-center text-white">
        <p class="ph">Created For FoodForAll &nbsp;&nbsp;&nbsp;|<i
class="fa fa-envelope"></i>foodforall@gmail.com</p>
    </div>
</footer>
<!-- jQuery first, then Popper.js, then Bootstrap JS -->
<script src="/static/javascript/userBase.js"></script>
<script src="/static/javascript/home.js"></script>
<script src="/static/javascript/jquery.js"></script>
<script src="/static/javascript/nicepage.js"></script>

```

```

<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"
integrity="sha384-KJ3o2DKtIkYIK3UENzmM7KCKRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN"
crossorigin="anonymous"></script>
<script
src="https://cdn.jsdelivr.net/npm/popper.js@1.12.9/dist/umd/popper.min.js"
integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q"
crossorigin="anonymous"></script>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/dist/js/bootstrap.min.js"
integrity="sha384-JZR6Spejh4U02d8j0t6vLEHfe/JQGiRRSQxSfFWpi1MquVdAyjUar5+76PVCmYl"
crossorigin="anonymous"></script>
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
</body>

</html>
//NgoHome.html

```

```

<!doctype html>

<html lang="en">

<head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

    <!-- Bootstrap CSS -->

    <title>{% block title %}{{name}} | Food For All{% endblock title %}</title>

    <link rel="stylesheet" href="/static/css/ngohome.css">
    <link rel="stylesheet"
        href="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/dist/css/bootstrap.min.css"
        integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJ1SAwiGgFAW/dAiS6JXm"
        crossorigin="anonymous">

```

```

<link rel="stylesheet" href="/static/css/userBase.css">
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
<link rel="stylesheet" href="/static/css/nicepage.css">
<link rel="stylesheet" href="/static/css/Page-2.css">

<meta name="generator" content="Nicepage 4.2.6, nicepage.com">
<link id="u-theme-google-font" rel="stylesheet"
href="https://fonts.googleapis.com/css?family=Roboto:100,100i,300,300i,400,400i,500,
500i,700,700i,900,900i|Open+Sans:300,300i,400,400i,600,600i,700,700i,800,800i">
<link id="u-page-google-font" rel="stylesheet"
href="https://fonts.googleapis.com/css?family=Ubuntu:300,300i,400,400i,500,500i,700,
700i|Kalam:300,400,700|Alata:400">
<script type="application/ld+json">
{
    "@context": "http://schema.org",
    "@type": "Organization",
    "name": ""
}
</script>
<meta name="theme-color" content="#478ac9">
<meta property="og:title" content="Page 2">
<meta property="og:type" content="website">

</head>

<body>

<!-- header section -->
<header class="ngoPage">

    <!-- navbar -->
    <div class="">
        <section class="hadder">
            <div class="container">
                <nav class="navbar abx navbar-expand-lg navbar-light"
style="background-color: #e3f2fd;">
                    <a class="navbar-brand" href="ngopage">FoodForAll</a>
                    <button class="navbar-toggler" type="button" data-
toggle="collapse" data-target="#navbarSupportedContent" aria-

```

```

controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle
navigation">
    <span class="navbar-toggler-icon"></span>
</button>

    <div class="collapse navbar-collapse"
id="navbarSupportedContent">
        <ul class="navbar-nav ml-auto">
            <li class="nav-item active">
                <a class="nav-link" href="contactngo">Contact Us
<span
                    class="sr-only">(current)</span></a>
            </li>

            <li class="nav-item active">
                <a class="nav-link"
href="galleryNGO">Gallery</a>
            </li>
            <li class="nav-item dropdown">
                <a class="nav-link dropdown-toggle" href="#"
id="navbarDropdown" role="button" data-toggle="dropdown" aria-haspopup="true" aria-
expanded="false">
                    My Account
                </a>
                <div class="dropdown-menu" aria-
labelledby="navbarDropdown">
                    <a class="dropdown-item"
href="availableCookedMeal">Available Cooked Meal</a>
                    <a class="dropdown-item"
href="afood">Available Raw Grains</a>
                    <a class="dropdown-item"
href="viewfeedback">View Feedback</a>
                    <a class="dropdown-item"
href="listdoner">List of all Donations</a>
                    <a class="dropdown-item"
href="aboutngo">About Us</a>
                    <div class="dropdown-divider"></div>
                    <a class="dropdown-item"
href="#about">Update Profile</a>

```

```

                <a class="dropdown-item"
href="changepassword">Change Password</a>
                <a class="dropdown-item"
href="logout">Logout</a>
            </div>
        </li>

    </ul>

</div>
</nav>
</div>
<!-- endof navbar --&gt;
<!-- message --&gt;
&lt;div class="message"&gt;
    {% for message in messages %}
        &lt;li {% if message.tags %}&lt;div class="alert alert-success my-0
alert-dismissible tag"&gt;
            &lt;a href="#" class="close" data-dismiss="alert" aria-
label="close"&gt;&amp;times;&lt;/a&gt; {% endif %}{{ message }}
        &lt;/li&gt;
    &lt;/div&gt;
    {% endfor %}
&lt;/div&gt;
&lt;section class="u-align-left u-clearfix u-section-1" id="carousel_560d"&gt;
    &lt;div class="u-clearfix u-sheet u-sheet-1"&gt;
        &lt!-- &lt;div class="u-gutter-0 u-layout"&gt; --&gt;
        &lt!-- &lt;div class="u-layout-row"&gt; --&gt;
        &lt!-- &lt;div class="u-container-style u-layout-cell u-size-60 u-layout-
cell-1"&gt; --&gt;
            &lt!-- &lt;div class="u-container-layout u-container-layout-1"&gt; --&gt;
            &lt;h1 class="u-custom-font u-font-ubuntu u-text u-text-1" data-
animation-name="fadeIn" data-animation-duration="1000" data-animation-
direction="Left" data-animation-delay="500"&gt;
                &lt;span class="u-text-palette-2-base"&gt;
                    &lt;font class="u-text-palette-2-base"&gt;&lt;b&gt;
Welcome Back {{name}}&lt;/b&gt;
                    &lt;/font&gt;
                &lt;/span&gt;
            &lt;/h1&gt;
    &lt;/div&gt;
&lt;/section&gt;
</pre>

```

```

        </div>
    </section>

    <!-- ----- -->
--> <section class="u-section-3">
    <div class="u-list u-list-1 mainClass">
        <div class="u-repeater u-repeater-1">
            <div class="u-align-left u-container-style u-grey-80 u-list-item
u-opacity u-opacity-55 u-repeater-item u-shape-rectangle u-list-item-1">
                <div class="u-container-layout u-similar-container u-valign-
top u-container-layout-1">
                    <div class="u-align-center u-container-style u-group u-
palette-2-base u-radius-50 u-shape-round u-group-1" data-animation-name="fadeIn"
data-animation-duration="1000" data-animation-direction="Left">
                        <div class="u-container-layout u-valign-middle u-
container-layout-2">
                            <h1 class="u-custom-font u-text u-text-
1">01</h1>
                            </div>
                        </div>
                    <h5 class="u-text u-text-body-alt-color u-text-2" data-
animation-name="fadeIn" data-animation-duration="1000" data-animation-
direction="Right">See food donation requests from doners</h5>
                    <!-- <p class="u-text u-text-blue-30 u-text-3">If you
have food you can always donate your food here.Kindly register as doner.</p> -->
                    <a href="viewdonationrequest" class="u-active-none u-
border-2 u-border-active-white u-border-hover-white u-border-white u-btn u-button-
style u-hover-none u-none u-text-body-alt-color u-text-hover-grey-30 u-btn-1">Raw
                        Grains</a>
                    <a href="cookedMealRequest" class="u-active-none u-
border-2 u-border-active-white u-border-hover-white u-border-white u-btn u-button-
style u-hover-none u-none u-text-body-alt-color u-text-hover-grey-30 u-btn-1">Cooked
                        Meal</a>
                </div>
            </div>
            <div class="u-align-left u-container-style u-grey-80 u-list-item
u-opacity u-opacity-55 u-repeater-item u-shape-rectangle u-list-item-2">

```

```


<div class="u-align-center u-container-style u-group u-palette-2-base u-radius-50 u-shape-round u-video-cover u-group-2" data-animation-name="fadeIn" data-animation-duration="1000" data-animation-direction="Left">
        <div class="u-container-layout u-valign-middle u-container-layout-4">
            <h1 class="u-custom-font u-text u-text-4">02</h1>
            </div>
        </div>
        <h5 class="u-text u-text-body-alt-color u-text-5" data-animation-name="fadeIn" data-animation-duration="1000" data-animation-direction="Right">See food requests from consumers</h5>
        <!-- <p class="u-text u-text-blue-30 u-text-6">Coller plays major role in this chain. They collect food from doner and store and give it to the consumer.</p> -->
        <a href="foodrequest" class="u-active-none u-border-2 u-border-active-white u-border-hover-white u-border-white u-btn u-button-style u-hover-none u-text-body-alt-color u-text-hover-grey-30 u-btn-2">Food Requests</a>
    </div>
</div>
<div class="u-align-left u-container-style u-grey-80 u-list-item u-opacity u-opacity-55 u-repeater-item u-shape-rectangle u-list-item-3">
    <div class="u-container-layout u-similar-container u-valign-top u-container-layout-5">
        <div class="u-align-center u-container-style u-group u-palette-2-base u-radius-50 u-shape-round u-video-cover u-group-3" data-animation-name="fadeIn" data-animation-duration="1000" data-animation-direction="Left">
            <div class="u-container-layout u-valign-middle u-container-layout-6">
                <h1 class="u-custom-font u-text u-text-7">03</h1>
                </div>
            </div>
            <h5 class="u-text u-text-body-alt-color u-text-8" data-animation-name="fadeIn" data-animation-duration="1000" data-animation-direction="Right">Create food packages</h5>
        </div>
    </div>


```

```

        <!-- <p class="u-text u-text-blue-30 u-text-9">Consumer  
can order food from collector If they needed food for needy people.</p> -->
            <a href="createPakage" class="u-active-none u-border-2  
u-border-active-white u-border-hover-white u-border-white u-btn u-button-style u-  
hover-none u-text-body-alt-color u-text-hover-grey-30 u-btn-3">Create  
Food Pakage</a>
        </div>
    </div>
</div>
</section>
<!-- ----->
-->

</header>
<!-- header ending -->
<!-- footer -->
<section class="mainbox my-3">
<div class="">
    <div class="row ">
        <div class="col-sm my-3 text-center">
            <a class="navbar-brand" href="ngopage">
                <p class="text-dark">FoodForAll</p>
            </a>
        </div>
        <div class="col-sm my-3">
            <p class="who">Who we are</p>

            <li class="ab ">
                <a href="aboutngo">About us</a>
            </li>
            <li class="ab "><a href="contactngo">Contact us</a></li>

        </div>
        <div class="col-sm my-3">
            <p class="who ">Get Invoved</p>
            <li class="ab ">
                <a href="register">Donate Food</a>
            </li>
        </div>
    </div>
</section>

```

```

        </div>
        <div class="col-sm my-3">
            <div class="text-center">Social</div>
            <div class="">
                <a href="#" class="fa fa-facebook"> </a>&ampnbsp
                <a href="#" class="fa fa-twitter"> </a>&ampnbsp
                <a href="#" class="fa fa-instagram"> </a>&ampnbsp
            </div>
        </div>
    </div>
</section>
<footer class="footer bg-dark">
    <div class="container text-center text-white">
        <p class="ph">Created For FoodForAll &nbsp;&nbsp;&nbsp;|<i
class="fa fa-envelope"></i>foodforall@gmail.com</p>
    </div>
</footer>

<!-- footer end --&gt;

{% block content %}{% endblock content %}
<!-- Optional JavaScript --&gt;
<!-- jQuery first, then Popper.js, then Bootstrap JS --&gt;
&lt;script src="/static/javascript/userBase.js"&gt;&lt;/script&gt;
&lt;script src="/static/javascript/home.js"&gt;&lt;/script&gt;
&lt;script src="/static/javascript/jquery.js"&gt;&lt;/script&gt;
&lt;script src="/static/javascript/nicepage.js"&gt;&lt;/script&gt;
&lt;script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"
integrity="sha384-KJ3o2DKtIkYI1K3UENmM7KCkRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN"
crossorigin="anonymous"&gt;&lt;/script&gt;
&lt;script
src="https://cdn.jsdelivr.net/npm/popper.js@1.12.9/dist/umd/popper.min.js"
integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q"
crossorigin="anonymous"&gt;&lt;/script&gt;
&lt;script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/dist/js/bootstrap.min.js"
integrity="sha384-JZR6Spejh4U02d8j0t6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76PVCmYl"
crossorigin="anonymous"&gt;&lt;/script&gt;
</pre>

```

```
<script  
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>  
  <script  
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>  
</body>  
  
</html>
```

Back End Snippets

//settings.py

```
# Database
# https://docs.djangoproject.com/en/4.0/ref/settings/#databases

DATABASES = {
    'default': {
        # 'ENGINE': 'django.db.backends.sqlite3',
        # 'NAME': BASE_DIR / 'db.sqlite3',
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'food',
        'USER': 'root',
        'PASSWORD': '',
        'HOST': 'localhost',
        'PORT': '3306',
    }
}
```

//foodforall_urls.py

```
from unicodedata import name
from django.contrib import admin
from django.urls import path
from FoodForAll import views

urlpatterns = [
    path('',views.HomePage,name='HomePage'),

    path('login',views.login,name='login'),
    path('register',views.register,name='register'),
    path('registerview',views.registerview,name='registerview'),
    path('logout',views.logout,name='logout'),
    path('gallery',views.gallery,name='gallery'),
```

```

path('updateProfile',views.updateProfile,name='updateProfile'),
path('updateProfilec',views.updateProfilec,name='updateProfilec'),


path('uplodeImage',views.uplodeImage,name='uplodeImage'),
path('galleryNGO',views.galleryNGO,name='galleryNGO'),
path('galleryDoner',views.galleryDoner,name='galleryDoner'),
path('galleryConsumer',views.galleryConsumer,name='galleryConsumer'),


path('users',views.users,name='users'),
path('donatefood',views.donatefood,name='donatefood'),
    path('donate',views.donate,name='donate'),
path('requeststatus',views.requeststatus,name='requeststatus'),
path('predonation',views.predonation,name='predonation'),
path('meal',views.meal,name='meal'),
path('confirmc',views.confirmc,name='confirmc'),
path('cancelc',views.cancelc,name='cancelc'),


path('cookedMealRequest',views.cookedMealRequest,name='cookedMealRequest'),
path('availableCookedMeal',views.availableCookedMeal,name='availableCookedMeal')
,'

path('ngopage',views.ngopage,name='ngopage'),
path('viewdonationrequest',views.viewdonationrequest,name='viewdonationrequest')
,'

path('afood',views.afood,name='afood'),
path('foodrequest',views.foodrequest,name='foodrequest'),
path('listdoner',views.listdoner,name='listdoner'),
path('viewfeedback',views.viewfeedback,name='viewfeedback'),
path('feedbackinfo',views.feedbackinfo,name='feedbackinfo'),
path('createPakage',views.createPakage,name='createPakage'),


path('about',views.about,name='about'),
path('contact',views.contact,name='contact'),
path('aboutuser',views.aboutuser,name='aboutuser'),
path('cabout',views.cabout,name='cabout'),
path('ccontact',views.ccontact,name='ccontact'),
path('contactuser',views.contactuser,name='contactuser'),
path('aboutngo',views.aboutngo,name='aboutngo'),
path('contactngo',views.contactngo,name='contactngo'),
path('changepassword',views.changepassword,name='changepassword'),
```

```

path('registration',views.registration,name='registration') ,
path('ngoregister',views.ngoregister,name='ngoregister'),
path('donerregister',views.donerregister,name='donerregister'),
path('consumerregister',views.consumerregister,name='consumerregister'),
path('confirm',views.confirm,name='confirm'),
path('cancel',views.cancel,name='cancel'),


path('consumer',views.consumer,name='consumer'),
path('selectfood',views.selectfood,name='selectfood') ,
path('select',views.select,name='select'),
path('selectpack',views.selectpack,name='selectpake'),
path('mycart',views.mycart,name='mycart'),
path('order',views.order,name='order'),
path('remove',views.remove,name='remove'),
path('remove1',views.remove1,name='remove1'),
path('removeall',views.removeall,name='removeall'),
path('confirmOrder',views.confirmOrder,name='confirmOrder'),


path('foodConfirm',views.foodConfirm,name='foodConfirm'),
path('foodcancel',views.foodcancel,name='foodcancel'),
path('packconfirm',views.packconfirm,name='packconfirm'),
path('packcancel ',views.packcancel ,name='packcancel '),
path('previousOrder',views.previousOrder,name='previousOrder'),
]

```

//foodforall_models.py

```

from datetime import datetime
from email.mime import message
from email.policy import default
from operator import mod
from time import time
from django.db import models

# Create your models here.

```

```

# model for all users
class myUser(models.Model):
    usert=(
        ("ngo","NGO"),("doner","Doner"),("consumer","consumer")
    )
    usertype=models.CharField(max_length=10,default=' ',choices=usert)
    username=models.CharField(max_length=50)
    email=models.CharField(max_length=50,default=' ')
    password=models.CharField(max_length=50)

    def __str__(Self):
        return Self.username


class userdetail(models.Model):
    fullname=models.CharField(max_length=50,default=' ')
    phoneno=models.CharField(max_length=50,default = ' ')
    aadhar=models.CharField(max_length=50,default = ' ')
    address=models.CharField(max_length=500,default=' ')
    city=models.CharField(max_length=500,default=' ')
    state=models.CharField(max_length=500,default=' ')
    zip=models.CharField(max_length=50,default=' ')
    user=models.ForeignKey(myUser,on_delete=models.CASCADE)
    def __str__(Self):
        return Self.fullname


class donation(models.Model):
    foodt=(
        ("Rice","Rice"),("Wheat","Wheat"),("Corn","Corn"),
        ("Jowar","Jowar"),("Bajri","Bajri"),("Moong Dal","Moong Dal"),
        ("Masoor Dal","Masoor Dal"),("Urad Dal","Urad Dal"),("Chana Dal","Chana
        Dal"),
        ("Toor Dal","Toor Dal"),("Rajma","Rajma"),("Soy Bean","Soy Bean")
    )
    donate_date=models.DateTimeField(default=datetime.now)
    foodtype=models.CharField(max_length=50,default=' ',choices=foodt)
    quantity=models.IntegerField(default=0)

```

```

dateofc=models.DateField(default=' ')
timeofc=models.TimeField(default=' ')
# image = models.ImageField(upload_to='images',default=' ')
address=models.CharField(max_length=500,default=' ')
status=models.CharField(max_length=10,default='Pending')
user=models.ForeignKey(userdetail,on_delete=models.CASCADE)
def __str__(Self):
    return Self.foodtype

class foodata(models.Model):
    foodtype=models.CharField(max_length=50,default=' ')
    quantity=models.IntegerField(default=0)
    available=models.BooleanField(default=False)
    message=models.BooleanField(default=False)
    # image = models.ImageField(upload_to='images',default=' ')
    def __str__(Self):
        return Self.foodtype

class foodpack(models.Model):
    foodt=(
        ("Rice","Rice"),("Wheat","Wheat"),("Corn","Corn"),
        ("Jowar","Jowar"),("Bajri","Bajri"),("Moong Dal","Moong Dal"),
        ("Masoor Dal","Masoor Dal"),("Urad Dal","Urad Dal"),("Chana Dal","Chana
Dal"),
        ("Toor Dal","Toor Dal"),("Rajma","Rajma"),("Soy Bean","Soy Bean")
    )
    packName=models.CharField(max_length=50,default=' ')
    food1=models.CharField(max_length=50,default='',choices=foodt)
    food2=models.CharField(max_length=50,default='',choices=foodt)
    food3=models.CharField(max_length=50,default='',choices=foodt)
    food4=models.CharField(max_length=50,default='',choices=foodt)
    quantity=models.CharField(max_length=10,default=' ')
    available=models.BooleanField(default=False)
    message=models.BooleanField(default=False)
    def __str__(Self):
        return Self.packName

```

```

# cart model

class cart(models.Model):
    user=models.ForeignKey(userdetail,on_delete=models.CASCADE)
    created_date=models.DateTimeField(default=datetime.now)
    def __str__(Self):
        return Self.user.fullname

class cartItem(models.Model):
    foodtype=models.ForeignKey(fooddata,on_delete=models.CASCADE)
    quantity=models.CharField(max_length=10,default=0)
    cart=models.ForeignKey(cart,on_delete=models.CASCADE)
    status=models.CharField(max_length=10,default='Pending')
    order_date=models.DateTimeField(default=datetime.now)
    def __str__(Self):
        return Self.foodtype.foodtype

class packCart(models.Model):
    foodpack=models.ForeignKey(foodpack,on_delete=models.CASCADE)
    quantity=models.CharField(max_length=10,default=0)
    cart=models.ForeignKey(cart,on_delete=models.CASCADE)
    status=models.CharField(max_length=10,default='Pending')
    order_date=models.DateTimeField(default=datetime.now)
    def __str__(Self):
        return Self.foodpack.packName

class feedback(models.Model):

    desc=models.CharField(max_length=500)
    pic=models.ImageField(upload_to="images",null=True)
    user=models.ForeignKey(userdetail,on_delete=models.CASCADE)
    def __str__(self):
        return self.user.fullname

class gallarypics(models.Model):
    pics=models.ImageField(upload_to="img")
    # def __str__(self):
    #     return self.user.fullname

```

```
class cookedmeald(models.Model):
    name=models.CharField(max_length=50,default=' ')
    datedonation=models.DateTimeField(default=datetime.now)
    quantity=models.IntegerField(default=0)
    dateofc=models.DateField(default=' ')
    timeofc=models.TimeField(default=' ')
    address=models.CharField(max_length=500,default=' ')
    status=models.CharField(max_length=10,default='Pending')
    user=models.ForeignKey(userdetail,on_delete=models.CASCADE)
    def __str__(Self):
        return Self.name
```

7. TESTING

TESTING PLAN

The method of testing used in this project is White Box testing as in White Box testing, the tester has the complete knowledge and understanding of the code.

TESTING STRATEGY

The development process repeats this testing sub-process a number of times for the following phases.

- a) Unit Testing.
- b) Integration Testing

Unit Testing tests a unit of code (module or program) after coding of that unit is completed. Integration Testing tests whether the various programs that make up a system, interface with each other as desired, fit together and whether the interfaces between the programs are correct.

TESTING METHOD

There are two kinds of testing mainly Blackbox and Whitebox testing:

- In black-box testing a software item is viewed as a blackbox, without knowledge of its internal structure or behavior. Possible input conditions based on the specifications (and possible sequences of input conditions), are presented as test cases.
- In white-box testing, knowledge of internal structure and logic is explored. Test cases are presented such that possible paths of control flow through the software item are traced. Hence more defects than black-box testing are likely to be found.

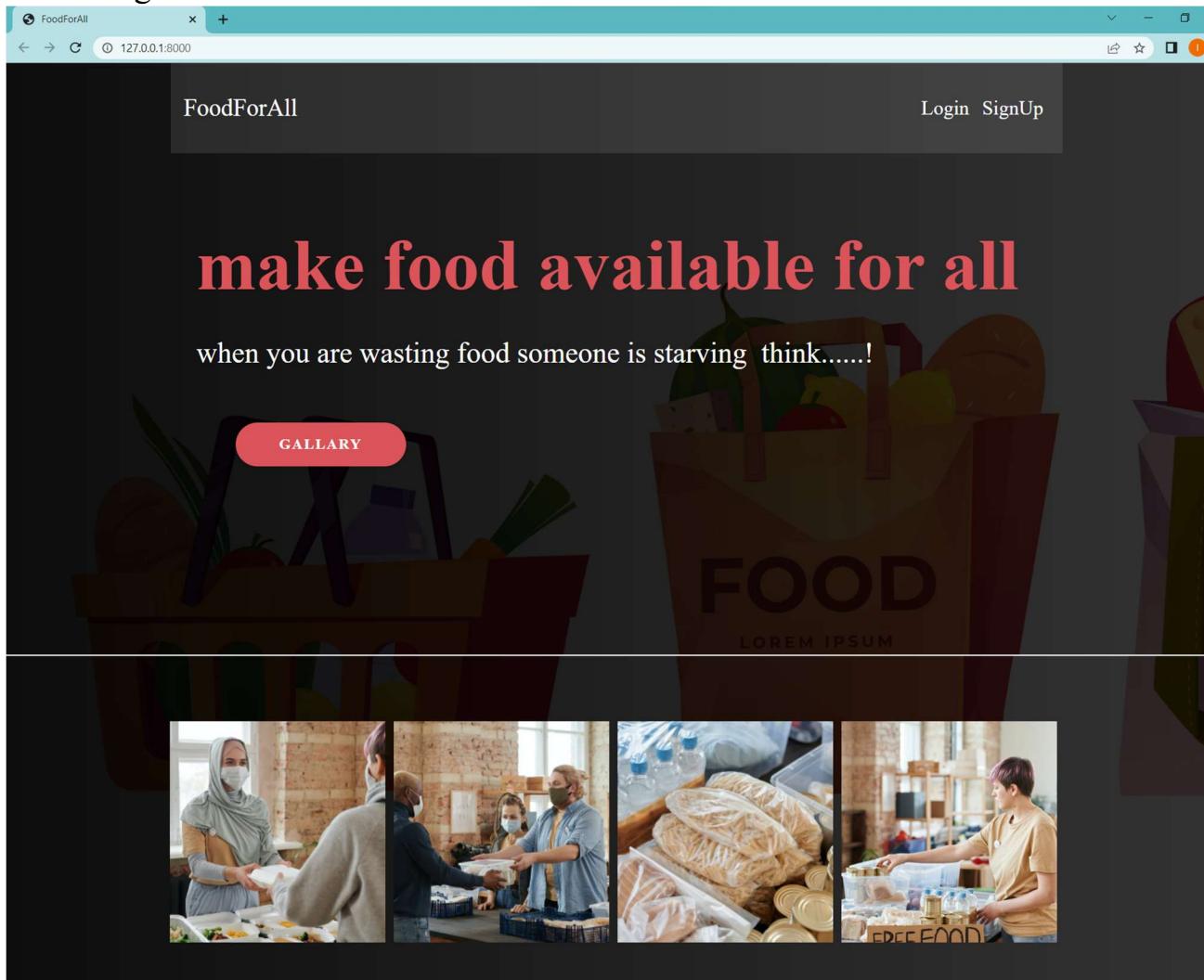
TEST CASES

ENTITY	TEST CASE	EXPECTED OUTPUT	ACTUAL OUTPUT	STATUS
Sign-up	Validation	Successfully register and redirect to log in page	Successfully register and redirect to log in page	pass
Login	Validation	Redirect to Home page	Redirect to Home page	pass
Sign-up as NGO	Validation	Redirect to Home page of donor portal	Redirect to Home page of donor portal	pass
Sign-up as consumer	Validation	Redirect to Home page of donor portal	Redirect to Home page of donor portal	pass
Sign-up as doner	Validation	Redirect to Home page of donor portal	Redirect to Home page of donor portal	Pass
Make food donation request	Enter food details	Redirect to donate food page after submitting form redirect to doner home page	Redirect to donate food page after submitting form redirect to doner home page	pass
Make food request	Enter food details	Redirect to consumer home page after selecting all required food and click on order	Redirect to consumer home	pass
View donation request	Post details	Display all the pending donation request from doner	Display all the pending donation request from doner	pass
View food request	Post details	Display all the pending food request from consumer	Display all the pending food request from consumer	pass

View previous donations	Post details	Display all previous donations of logged in donor	Display all previous donations of logged in donor	pass
View previous food orders	Post details	Display all previous food request of logged in consumer	Display all previous food request of logged in consumer	pass
View list of all doners	Post details	Display list of doners to NGO	Display list of doners to NGO	pass
View request status	Post details	Display status all requests for doner and consumer	Display status all requests for doner and consumer	Pass
Contact	Send details	The details will get submitted to database and redirect to index page Pass	The details will get submitted to database and redirect to index page	Pass
Change password	Validation	After changing password redirect to login page	Redirect to login page	pass

8. USER MANUAL

Home Page:



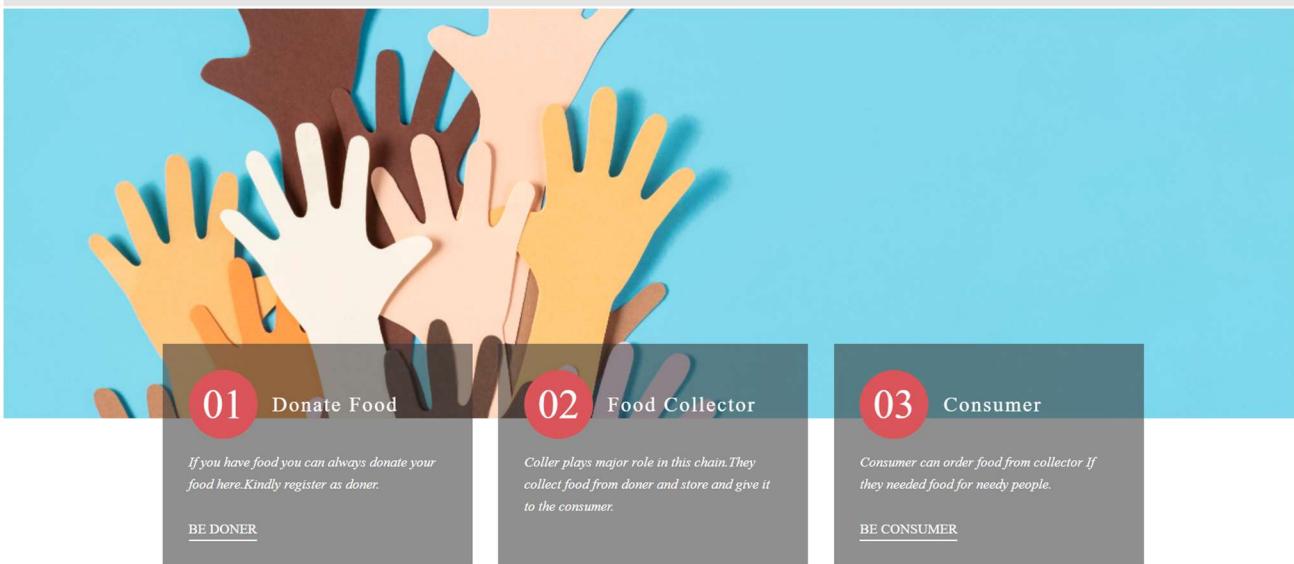


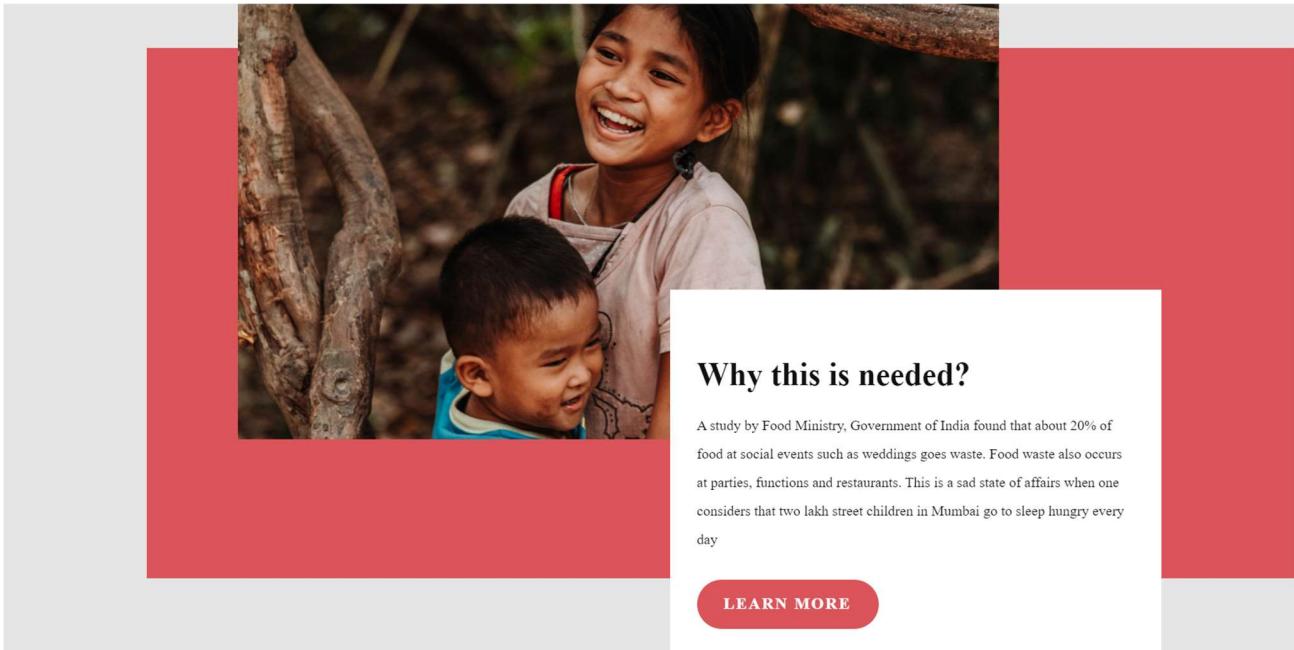
Our Aim

Wasting food is a common problem in our society. Food waste management is crucial since it can improve our environmental and economic sustainability.

[KNOW MORE](#)







Why this is needed?

A study by Food Ministry, Government of India found that about 20% of food at social events such as weddings goes waste. Food waste also occurs at parties, functions and restaurants. This is a sad state of affairs when one considers that two lakh street children in Mumbai go to sleep hungry every day

[LEARN MORE](#)

FoodForAll

Who we are

[About us](#)
[Contact us](#)

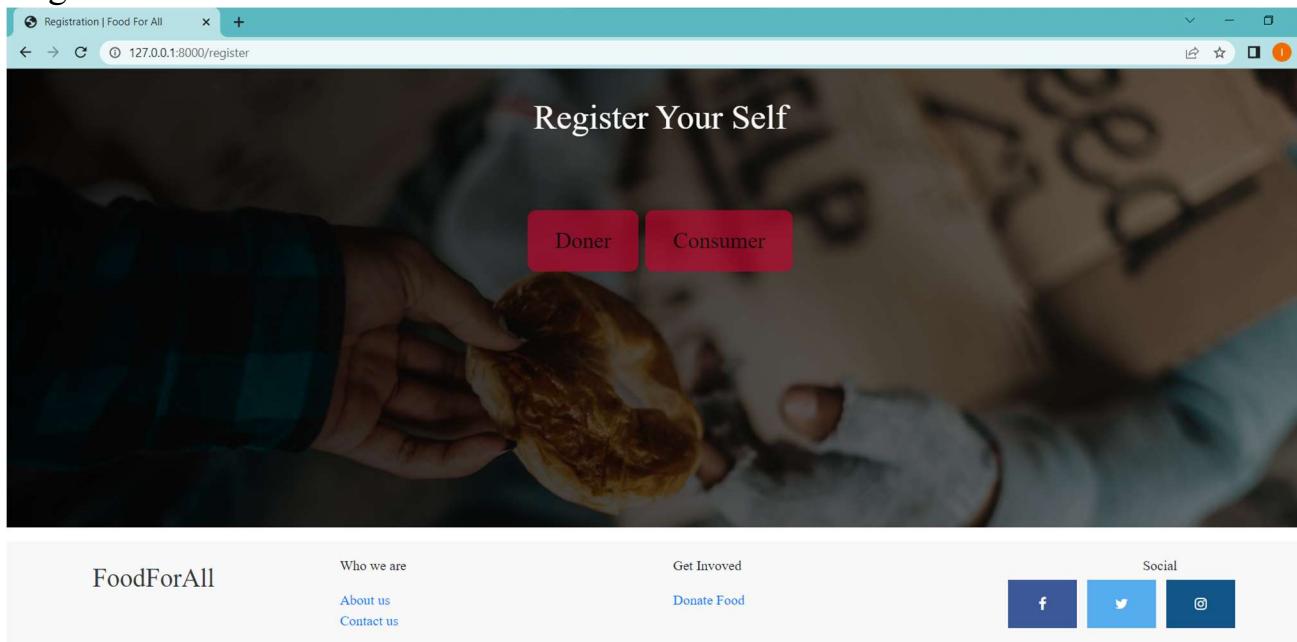
Get Involved

[Donate Food](#)

Social

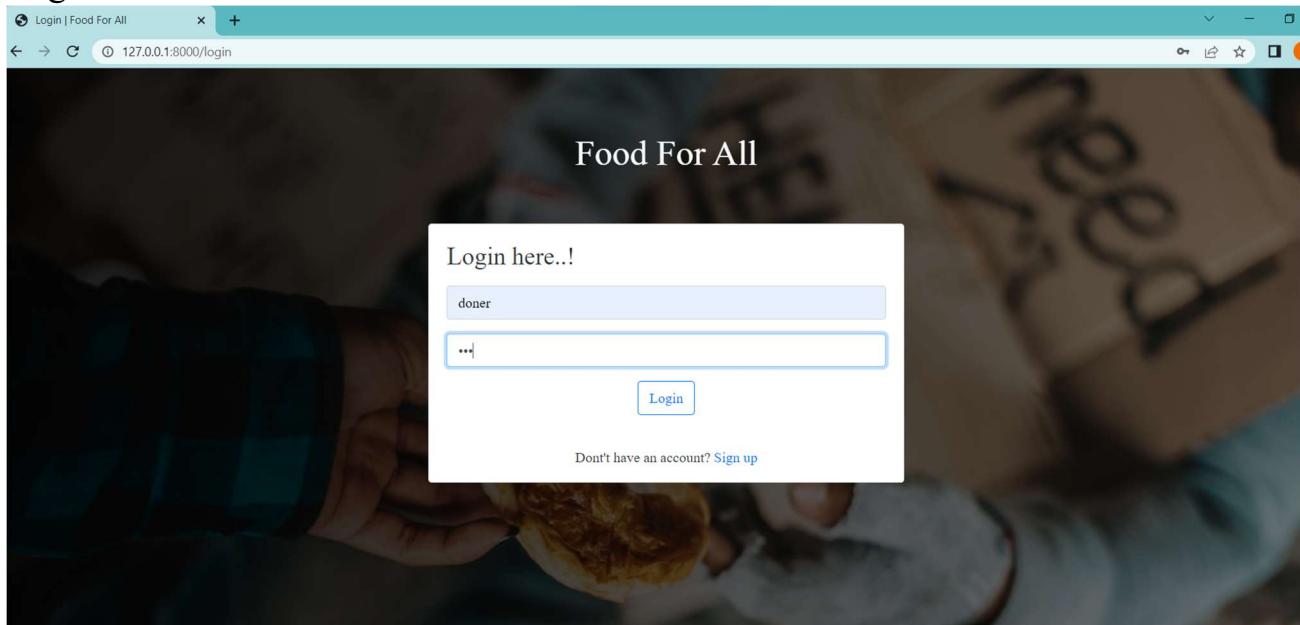
Created For FoodForAll | foodforall@gmail.com

Register

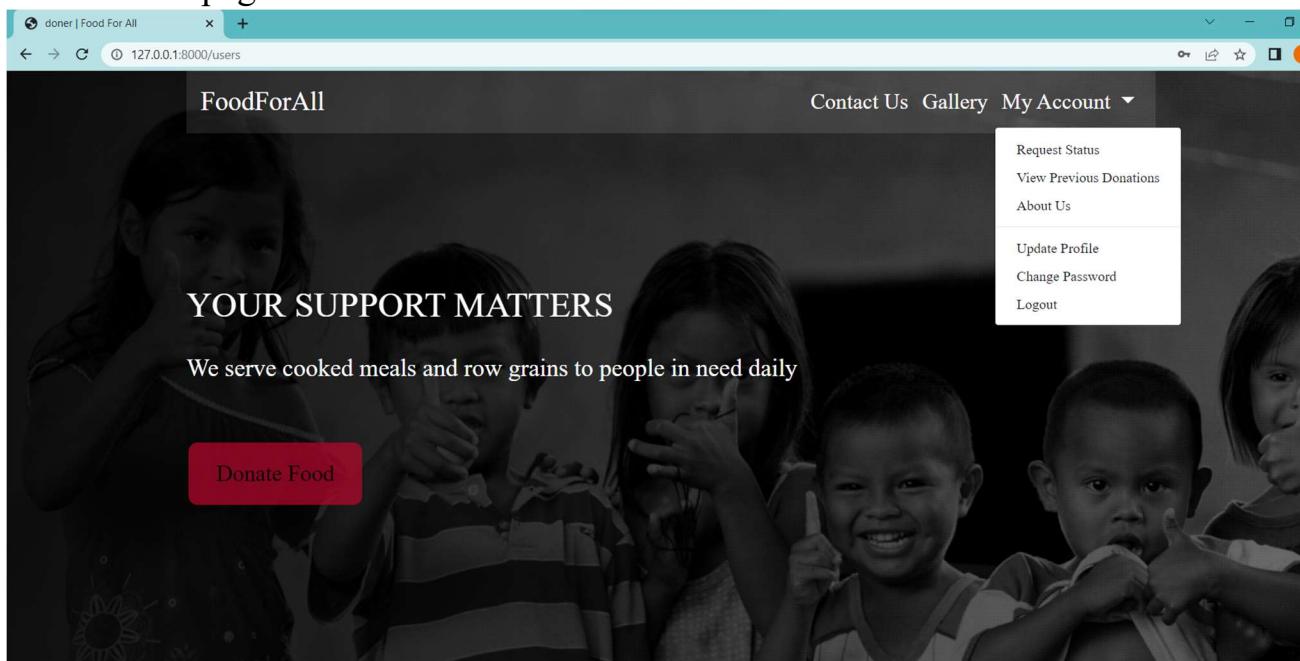


The screenshot shows a "Doner Registration" form titled "Food For All". The form is titled "Register here..!" and contains the following fields: "Enter your username", "Enter your Fullname", "Enter your Email", "Enter your Password", "Confirm your Password", "Enter your Phone Number", "Enter your Aadhar Number", "Apartment, studio, or floor", "City" (with dropdown options), "State" (with dropdown options), and "Zip". A "Register" button is located at the bottom of the form.

Login

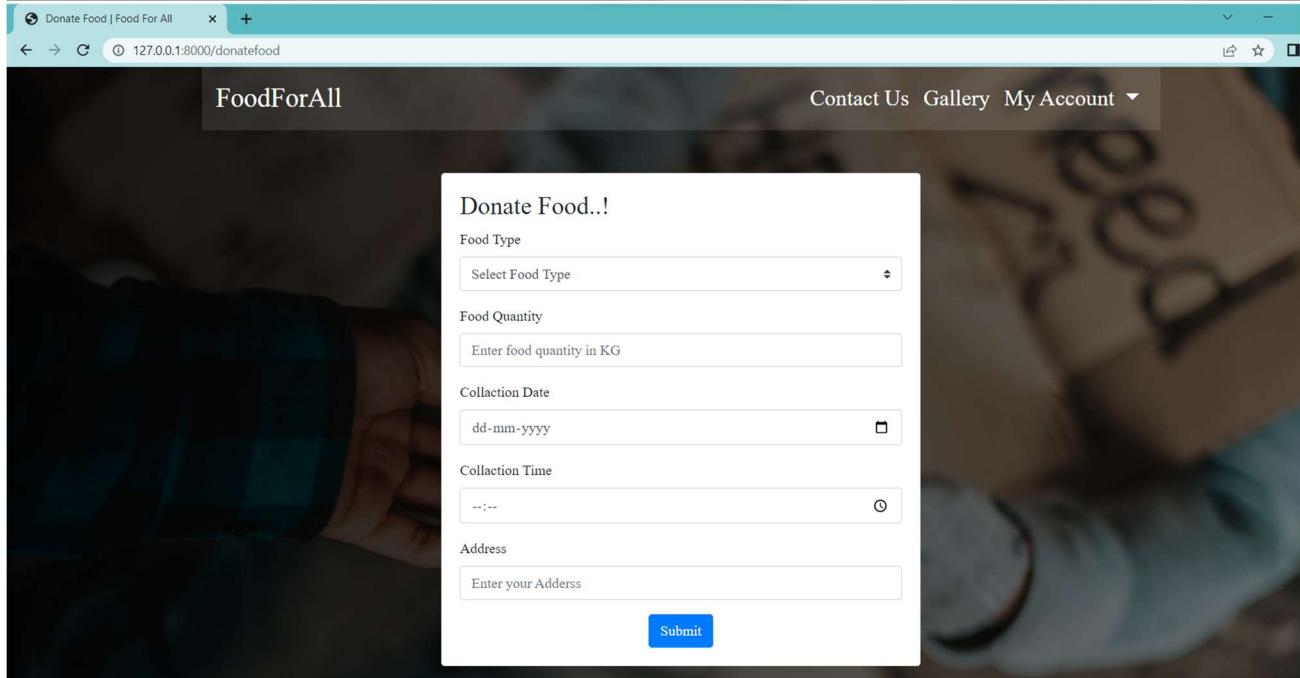
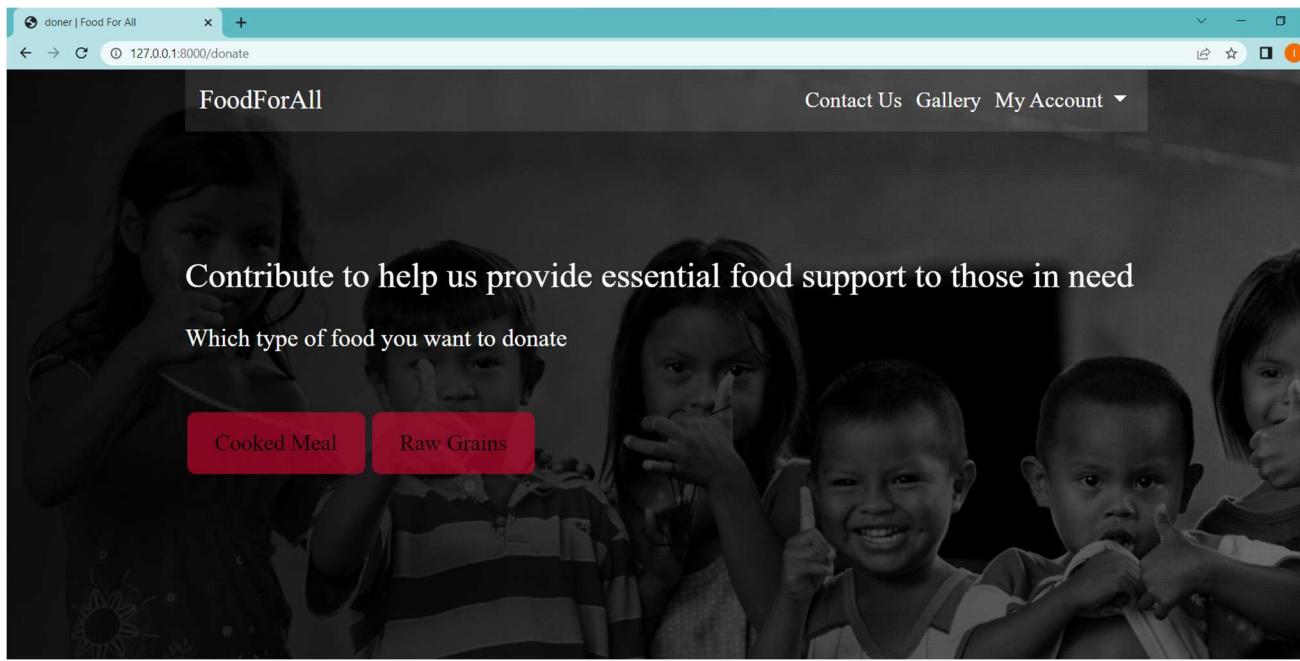


Doner home page



Donate Food

DDU(Faculty of Tech., Dept. of IT)



View request status

FoodForAll

Contact Us Gallery My Account ▾

Your Request Status

Food Type	Quantity	Status	Date of Donation
Wheat	10	confirm	March 13, 2022
Wheat	10	confirm	March 15, 2022
Rice	12	cancel	March 13, 2022

View previous donations

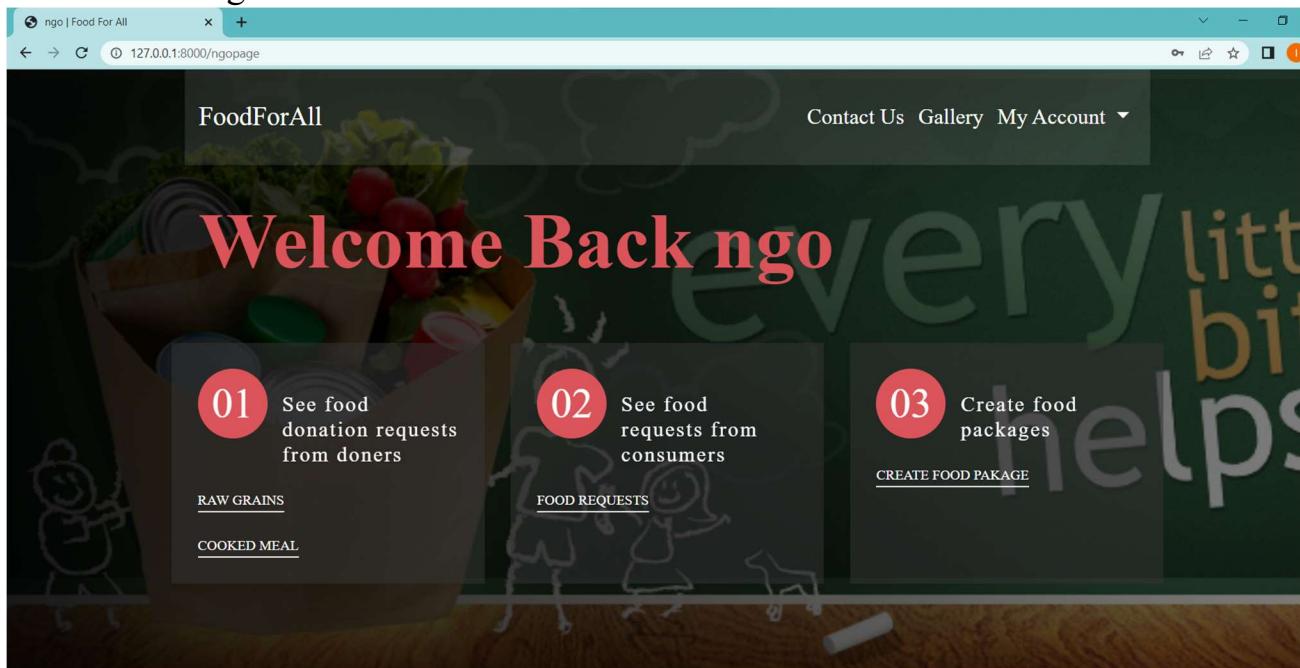
FoodForAll

Contact Us Gallery My Account ▾

Your Previous Donations

Food Type	Quantity	Status	Date of Donation	Time of Collection	Date of Collection
Wheat	10	confirm	March 6, 2022, 1:42 p.m.	5 p.m.	March 13, 2022
Wheat	10	confirm	March 15, 2022, 11:14 a.m.	5:20 p.m.	March 15, 2022

NGO home Page



View pending donation request

Doner Name	Food Type	Quantity	Status	Date of Donation	Time of Collection	Date of Collection
Doner	Bajri	10 KG	Pending	April 9, 2022, 1:02 p.m.	1 p.m.	April 10, 2022

Below the table, there are 'Confirm' and 'Cancel' buttons.

View pending donation request

The screenshot shows a web browser window titled "Food Requests" with the URL "127.0.0.1:8000/foodrequest#". The page header includes the "FoodForAll" logo, "Contact Us", "Gallery", and "My Account" dropdown. The main content area is titled "Pending Requests" and displays a table with the following data:

Food Type	Quantity	Date of Order
Rice	1 KG	March 11, 2022, 11:13 p.m.

Below the table, there are three buttons: "Confirm", "Cancel", and "User Details". At the bottom of the table row, there is a note: "Name:Consumer Address:antalia, bilmora, Gujrat Phone no:1234567890".

Available food

The screenshot shows a web browser window titled "127.0.0.1:8000/afood" with various navigation icons at the top. The page header includes the "FoodForAll" logo, "Contact Us", "Gallery", and "My Account" dropdown. The main content area is titled "Availabe Food" and displays three food items in cards:

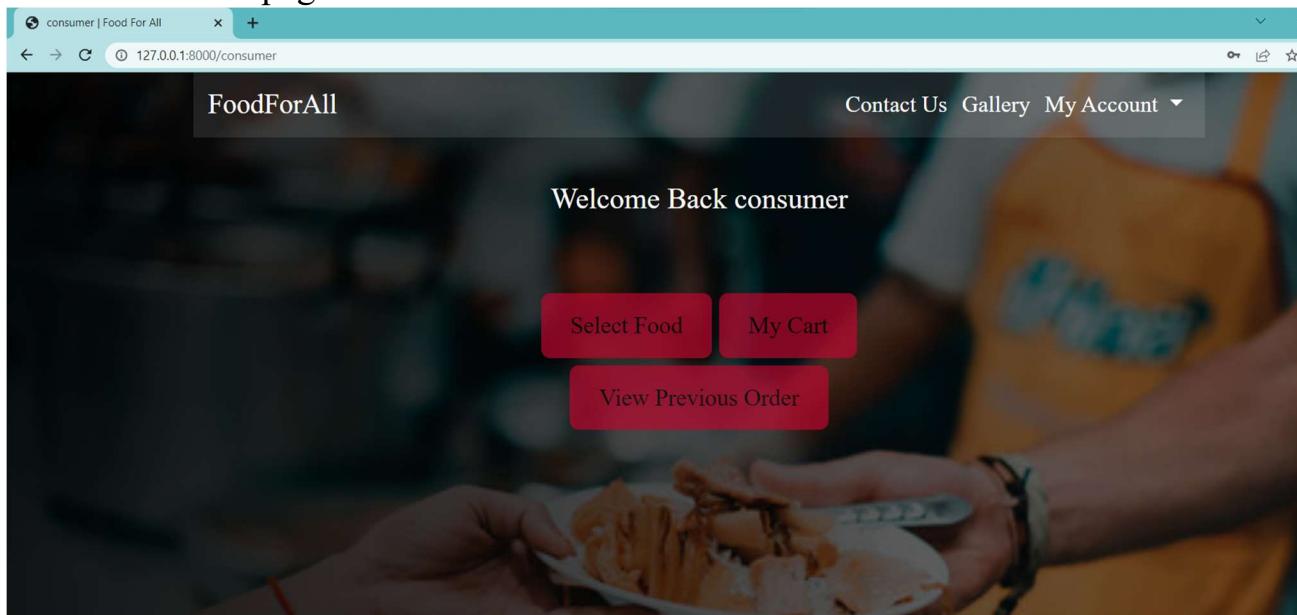
- 

Rice
39 KG
- 

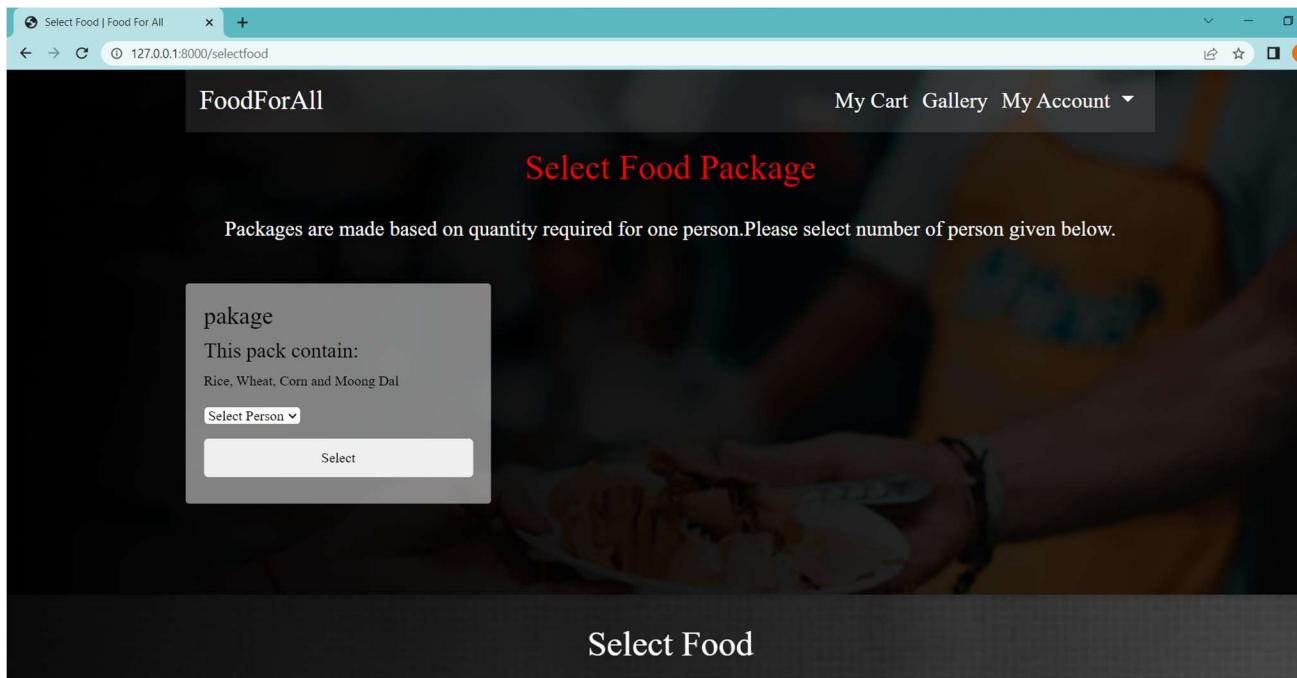
Wheat
20 KG
- 

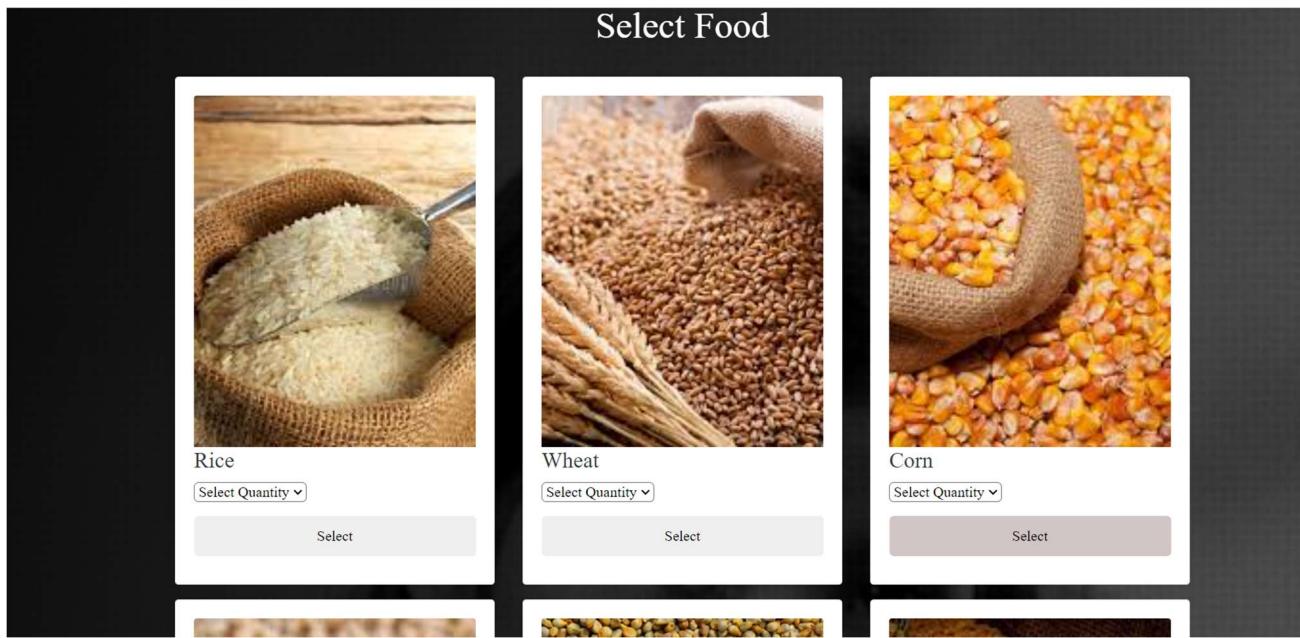
Corn
40 KG

Consumer home page



Select food





My cart

The screenshot shows a web browser window with the following details:

- Address Bar:** consumer | My Cart
- URL:** 127.0.0.1:8000/mycart
- Header:** FoodForAll (with a logo), Add Food, Gallery, My Account ▾
- Main Content:** A large box titled "MY CART" contains the text "pakage" and "3 Package".
- Item Details:** A sub-section shows "Rice" and "5 KG" with a "Confirm" button below it.

Update profile

A screenshot of a web browser showing the 'FoodForAll' consumer profile update page. The URL in the address bar is 127.0.0.1:8000/updateProfilec. The page title is 'consumer | Update Profile | FoodForAll'. The main content area is titled 'Food For All' and contains a form titled 'Change Profile'. The form fields are as follows:

Username	consumer
Email	consumer@gmail.com
Fullname	Consumer
PhoneNumber	1234567890
Aadhar Number	5432 2345 5443
Address	[Empty]

Change password

A screenshot of a web browser showing the 'FoodForAll' consumer password change page. The URL in the address bar is 127.0.0.1:8000/changepassword. The page title is 'consumer | Change Password | FoodForAll'. The main content area is titled 'Food For All' and contains a form titled 'Change Password'. The form fields are as follows:

Enter your Current Password
Enter your New Password
Confirm your new Password

. 9.LIMITATIONS AND FUTURE ENHANCEMENTS

LIMITATIONS:

- The Internet must be on the whole time while using the web application.
- It is for only one NGO. Only local service.

FUTURE ENHANCEMENTS:

- Implementing advanced facilities like more than one NGO can use.
- Improving the efficiency of the website.
- Keep on changing the UI-Design to make the website look more attractive.
- A mobile application for better portability, functioning and reliability.
- Making a map feature for users to find nearest NGO.

10. CONCLUSION AND DISCUSSION

CONCLUSION

- We provide 1 step solution for all the requirements related to a leftover food
- We have tried to keep the UI alluring and easy enough for our users to feel comfortable
- There would be obvious limitation in users as our application would require internet connection
- Small step to reduce food wastage.

DISCUSSION

Self-Analysis of Project Viabilities:

In our opinion, this project has served the goal that we set when we started. It provides a platform for people to reduce food wastage. Needy people can get food using this.

Problems encountered

There were so many problems encountered during this project: -

1. Problem to maintain databases and change them often.
2. Need to change some functionality fully which leads to doing the whole work again.
3. Many instances where the errors weren't understandable.
4. The entire database had to be rebuild thrice because of system errors

Summary of Project Work:

We have completed our project work using software engineering and system analysis and design approach. We have done our work with planned scheduling pertaining to time constraints and result oriented progress in project development. We have been given an industrial exposure which is very beneficial for our field.

11. BIBLIOGRAPHY

1. Youtube Channels - To learn all the related technologies

<https://www.youtube.com/user/javaboynavin>

2. Django Documentation - For all the knowledge related to

Django <https://docs.djangoproject.com>

3. Stack Overflow - For Debugging

<https://stackoverflow.com>