# 18CSC206J- Software Engineering & Project Management

# LAB REPORT

***Submitted by***

***Under the Guidance of***

Mrs. S Nagadevi

(Associate Professor, Department of Computer Science & Engineering)

***In partial satisfaction of the requirements for the degree of***

**BACHELOR OF TECHNOLOGY**

**in**

**COMPUTER SCIENCE ENGINEERING**

**SCHOOL OF COMPUTING**

**COLLEGE OF ENGINEERING AND TECHNOLOGY**

**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**

**KATTANKULATHUR - 603203**

**JUNE 2022**

# BONAFIDE CERTIFICATE

Certified that this lab report titled **“GoFood”** is the bonafide work done by who carried out the lab exercises under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other work.

**SIGNATURE**
Mrs. S Nagadevi
SEPM – Course Faculty
Guide
Professor
Dept. of Computer Science & Engineering

# ABSTRACT

This project is used to manage wastage foods in a useful way. Every day the people are wasting lots of foods. So we have to reduce that food wastage problem through online. If anyone has wastage foods they are entering their food quantity details and their address in the site and then the admin maintain the details of food donator. The donator can create the account and whenever they are having wastage food they can login and give request to the admin. And the admin also maintain the receiver (orphanage, NGOs,…) details too. After the admin view the donator request and give the alert message like time to come and collect the food. And the admin collect foods from donator and display it on the website, the receiver register themselves and selects the item they want to buy and places the order. Then provide to nearest orphanages or poor people the delivery will be done. If the donator needs any detail about the orphanage with helping thought they can give request to the admin and collect the orphanage details. This project is food redistribution is an enormously successful social innovation that tackles food waste and food poverty.

## TABLE OF CONTENTS

# PROBLEM STATEMENT

Developing a mini project to create a responsive website on a food donating system that makes a link between the donor and the receiver meeting the following requirements

- Admin acts as an intermediate
- The whole system works free of cost
- The delivery system is managed both at the donor and receiver’s side
- The site has separate login portals
- Main web page
- Dashboard

# LIST OF FIGURES

# LIST OF ABBREVIATIONS

ER - Diagram Entity Relationship

CAPEX - Capital expenditures

WBS - Work Breakdown Structure

SWOT - Strength, Weakness, Opportunities, Threats

RMMM - Risk Mitigation, Monitoring and Management

DFD - Data flow diagram

## CHAPTER-1
## PROBLEM STATEMENT-LOGO

To Frame a project team, analyze and identify a Software project. To create a business case and Arrive at a Problem Statement for the student performance.

| | |
|---|---|
| DATE | 15-03-2021 |
| SUBMITTED BY | Sakshi Shruti, Harsh, Anurag |
| TITLE / ROLE | Student performance |

fig.1.0 logo

### 1.1 PROBLEM STATEMENT

Developing a mini project to create a responsive website on a food donating system that makes a link between the donor and the receiver meeting the following requirements

- Admin acts as an intermediate
- The whole system works free of cost
- The delivery system is managed both at the donor and receiver's side
- The site has separate login portals
- Main web page
- Dashboard

### 1.2 Project Description:-

### 1.3 THE PROJECT

In bullet points, describe the problem this project aims to solve or the opportunity it aims to develop.

- Our aim is to provide a service platform that will connect the world in order to decrease food wastage and at the same time help the needy.
- We will create a website that will connect the donors and receivers which may include donors such as Restaurants, Big Function Venues etc. and receivers as NFOs or mini workplaces that are willing to buy leftover foods.

### 1.4 THE HISTORY

- Every year, 70% of the nation's population sleep empty stomach despite the fact that plenty of food is available.
- Weddings and other ceremonies are being magnificently conducted and even more is that food that goes to waste.

### 1.5 LIMITATIONS

List what could prevent the success of the project, such as the need for expensive equipment, bad weather, lack ofspecial training, etc.

- Natural causes such as warmer temperatures etc.
- Lack of support from donors.
- Bad packing by the restaurants.

### 1.6 APPROACH

List what is needed to complete the project.

- Tech stack needed:

  – frontend – Bootstrap, Material UI, React JS
  - Backend – Nodejs, Express JS

### 1.7 BENEFITS

list the benefits that this project will bring to the organization.

- Minimize food wastage.
- Assist non-profit organizations in their step towards conquering Global Hunger by starting at a regional and national level.
- Reduce the waste produced(direct consequence)

# CHAPTER-2

# STAKEHOLDERS & PROCESS MODELS

To identify the appropriate Process Model for the project and prepare Stakeholder and User Description.

**2.1**

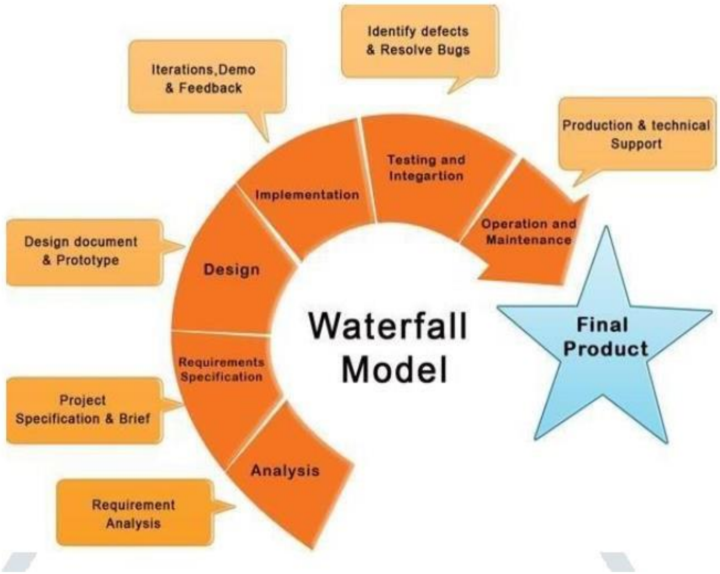


**Fig.2.1 process model**

2.2 SUMMARIZATION OF THE METHODOLOGY:-

□ Requirements: During this initial phase, the potential requirements of the application are methodically analyzed and written down in a specification document that serves as the basis for all future development. The result is typically a requirements document that defines what the application should do, but not how it should do it.

□ Analysis: During this second stage, the system is analyzed in order to properly generate the models and business logic that will be used in the application.

□ Design: This stage largely covers technical design requirements, such as programming language, data layers, services, etc. A design specification will typically be created that outlines how exactly the business logic covered in analysis will be technically implemented.

□ Coding: The actual source code is finally written in this fourth stage, implementing all models, business logic, and service integrations that were specified in the prior stages.

 Testing: During this stage, QA, beta testers, and all other testers systematically discover and report issues within the application that need to be resolved. It is not uncommon for this phase to cause a “necessary repeat” of the previous coding phase, in order for revealed bugs to be properly squashed.

 Deployment: Finally, the application is ready for deployment to a live environment. This stage entails not just the deployment of the application, but also subsequent support and maintenance that may be required to keep it functional and up-to-date.

| sl.no | Stakeholder Name | Activity/ Area /Phase | Interest | Influence | Priority (High/ Medium/ Low) |
|---|---|---|---|---|---|
| 1. | Owner | Managing resources | High | High | 1 |
| 2. | Customers | Website with receiver login | High | High | 1 |
| 3. | Supplier | Mobile app with donor login | High | High | 1 |
| 4. | Designer | Provide design to website | High | Low | 2 |
| 5. | Investor | Provie necessary financial resources | Low | High | 2 |
| 6. | Distributor | Distribute the supplies | Low | High | 3 |

# CHAPTER-3

## IDENTIFYING REQUIREMENTS

To identify the system, functional and non-functional requirements for the project

### 3.1 System Requirements

3.1.1. Hardware Requirements:-

- 8 GB RAM
- i5 10th gen processor

3.1.2. Software Requirements:-

- IDE
- Chrome Browser
- MongoDB

### 3.2 User Requirements

- ordering left-over food
- searching for registered

restaurants and vice-versa.

- large-scale service provider.

### 3.2 Functional Requirements:-

React JS, Node JS, MongoDB, Cloudinary, Express JS

### 3.3 Non-Functional Requirements:-

We are focusing on the people who are willing to order leftover food, cannot afford daily food and for the unit on a large scale. For example: NGOs and Social Group

# CHAPTER-4
# PROJECT PLAN & EFFORT

To Prepare Project Plan based on scope, Calculate Project effort based on resources, Find Job roles and responsibilities.

## 4.1 Project Management Plan

Describe the key issues driving the project.

| Focus Area | Details |
|---|---|
| schedule management | Front-end Developed in 1 months<br>Back-End integration in 0.5 month<br>Reach Out to Suppliers and Receivers in the<br>500 Orders in 3 months |
| Procurement Management | Communication with Suppliers and Receiving organizations<br>Quality Checks<br>Transportation Charges |
| Risk Management | Natural Causes<br>Lack of supply from users<br>Package and Quality Problems<br>Fraud Registrations<br>Breakdown of Communication |

## 4.2 Estimation

### 4.5.1 Effort and Cost Estimation

| Activity Description | Sub-Task | Sub-Task Description | Effort (in hours) | Cost in INR |
|---|---|---|---|---|
| Design the user screen | E1R1A1T1 (Effort-Requirement-Activity-Task) | Confirm the user requirements (acceptance criteria) | 3 | 1500 |
| | E1R1A1T2 | linking the data with octave or matlab | 1 | 500 |
| | E1R1A1T3 | Datafilling | 8 | 4000 |
| Identify Data Source for | | Go through Interface | 5 | 2500 |
| displaying units of Energy | | contract (Application | | |
| Consumption | | Data Exchange) | | |
| | | documents | 2 | 1000 |

| Effort (hr) | Cost (INR) |
|---|---|
| 1 | 500 |

## 4.3 Infrastructure/Resource Cost

| Infrastructure Requirement | Qty | Cost per qty | Cost per item |
|---|---|---|---|
| Map Integration | 1 | 2000 | 2000 |

### 4.5.2 Maintenance and Support Cost

| Category | Details | Qty | Cost per qty per annum | Cost per item |
|---|---|---|---|---|
| People | Network, System, Middleware and DB admin<br>Developer , Support Consultant | 4 | 2,000,000 | 6,000,000 |
| License | Operating System<br>Database<br>Middleware<br>IDE | 10 | 10000 | 100,000 |
| Infrastructures | Server, Storage and Network | 20 | 20000 | 400,000 |

## 4.6 Project Team Formation

### 4.6.1 Identification Team members

| Name | Role | Responsibilities |
|---|---|---|
| Harsh | Key Business User (Product Owner) | Provide clear business and user requirements |
| Sakshi | Project Manager | Manage the project |
| Anurag | Business Analyst | Discuss and Document Requirements |
| Harsh | Technical Lead | Design the end-to-end architecture |
| Anurag | Frontend Developer | Develop User interface |
| Harsh | Backend Developer | Design, Develop and Unit Test Services/API/DB |
| Sakshi | UX Designer | Design the user experience |
| Sakshi | Cloud Architect | Design the cost effective, highly available and scalable architecture |
| Sakshi | Cloud Operations | Provision required Services |
| Anurag | tester | Define Test Cases and Perform Testing |

## 4.6.2 Responsibility Assignment Matrix

| RACI Matrix | Team Members | | | |
|---|---|---|---|---|
| **Activity** | **Name (BA)** | **Name (Developer)** | **Name (Project Manager)** | **Key Business User** |
| User Requirement Documentation | Sakshi | harsh | Anurag | CRS institutions |

| | |
|---|---|
| A | Accountable |
| R | Responsible |
| C | Consult |
| I | Inform |

# CHAPTER-5

# WORK BREAKDOWN STRUCTURE & RISK ANALYSIS

To Prepare Work breakdown structure, Timeline chart and Risk identification table.

Project work breakdown structure with timeline chart and risk table is incorporated.

## 5.1 Work break down structure

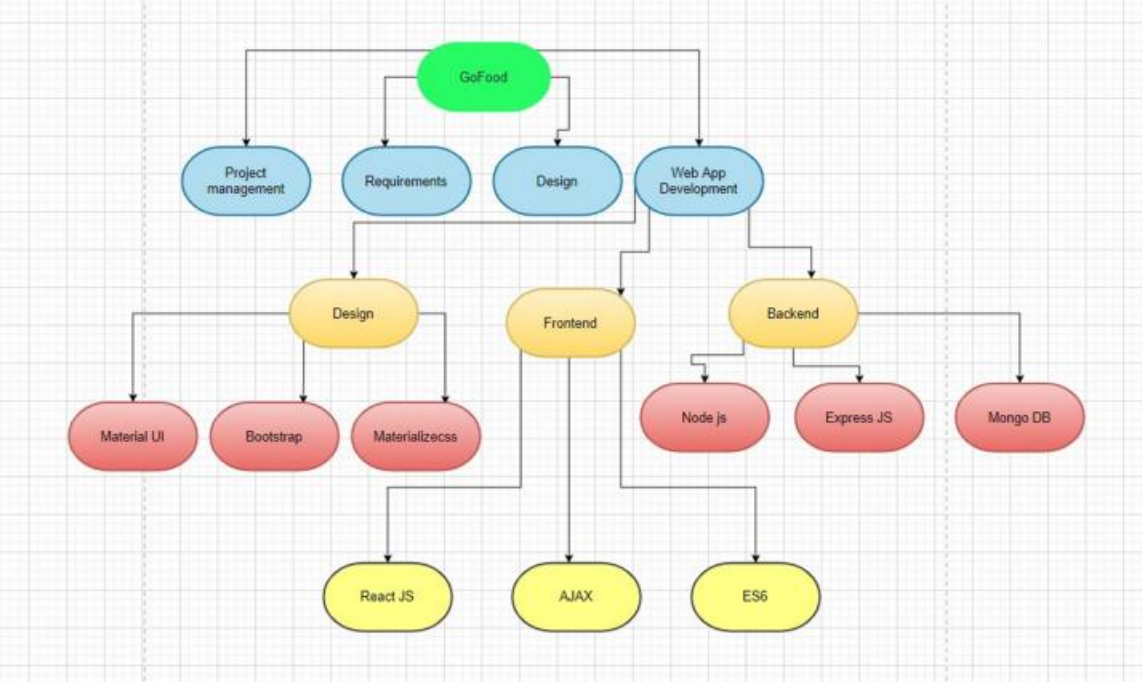
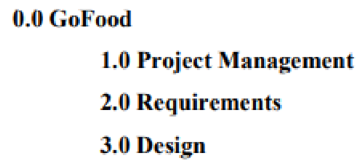


**Fig.5.1.WBS**



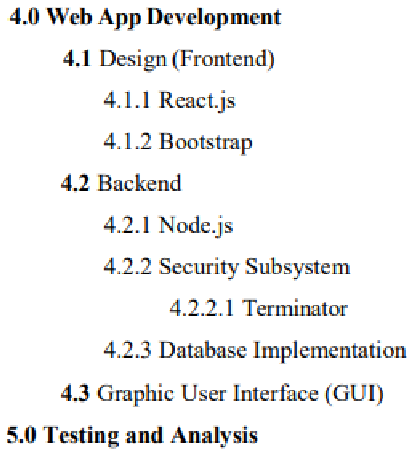
**0.0 GoFood**

**1.0 Project Management**

**2.0 Requirements**

**3.0 Design**

□ S



**4.0 Web App Development**

**4.1** Design (Frontend)

4.1.1 React.js

4.1.2 Bootstrap

**4.2** Backend

4.2.1 Node.js

4.2.2 Security Subsystem

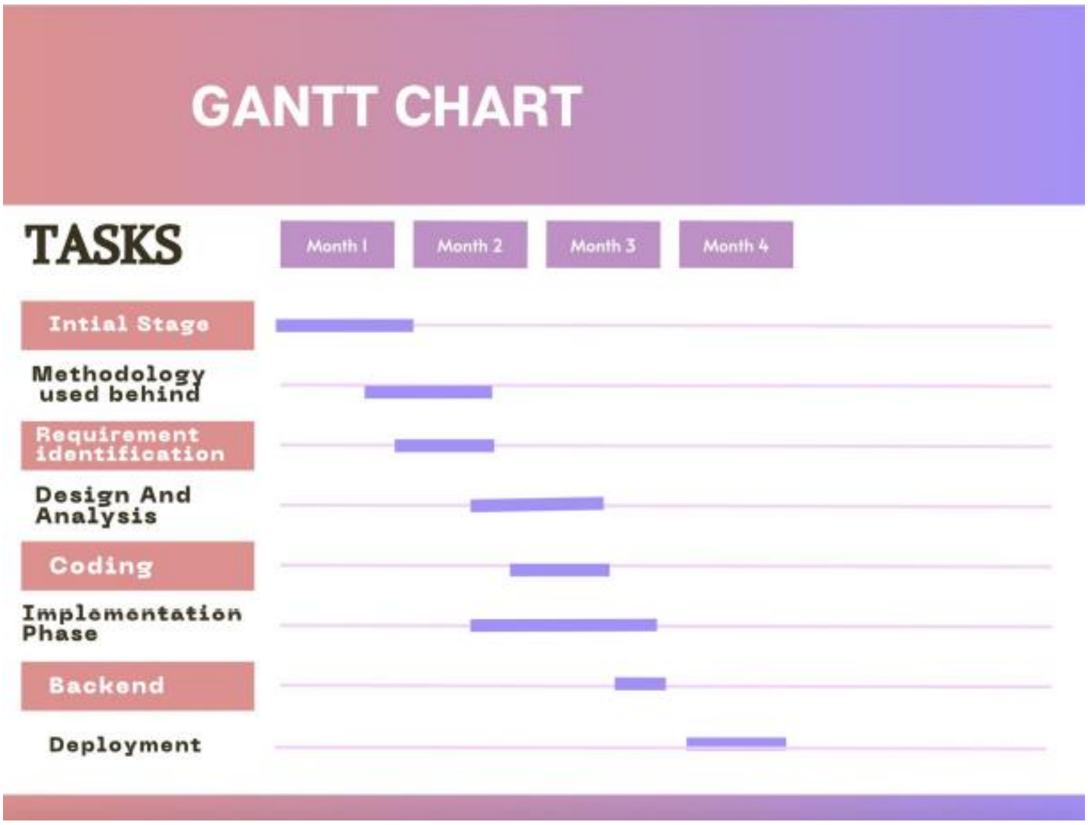
4.2.2.1 Terminator

4.2.3 Database Implementation

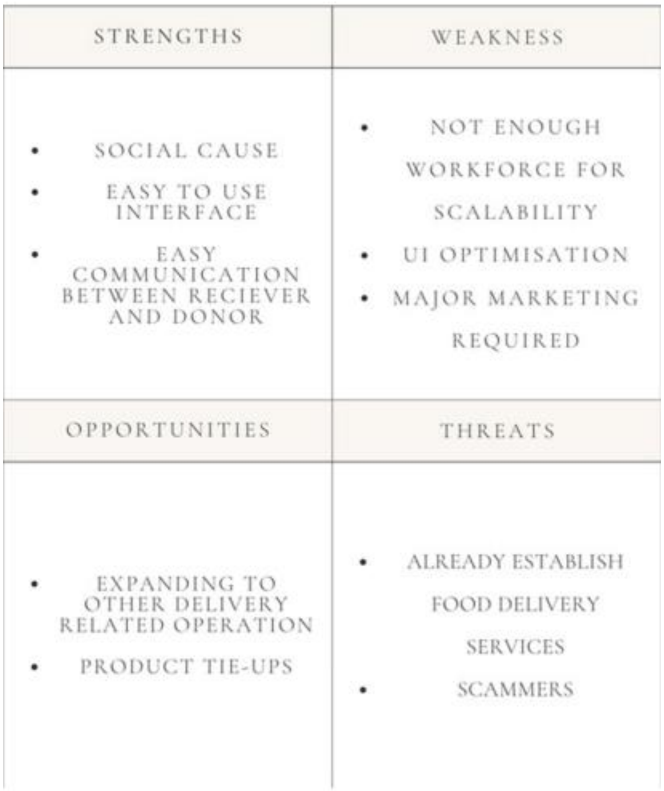
**4.3** Graphic User Interface (GUI)

**5.0 Testing and Analysis**

**Gantt Chart:**

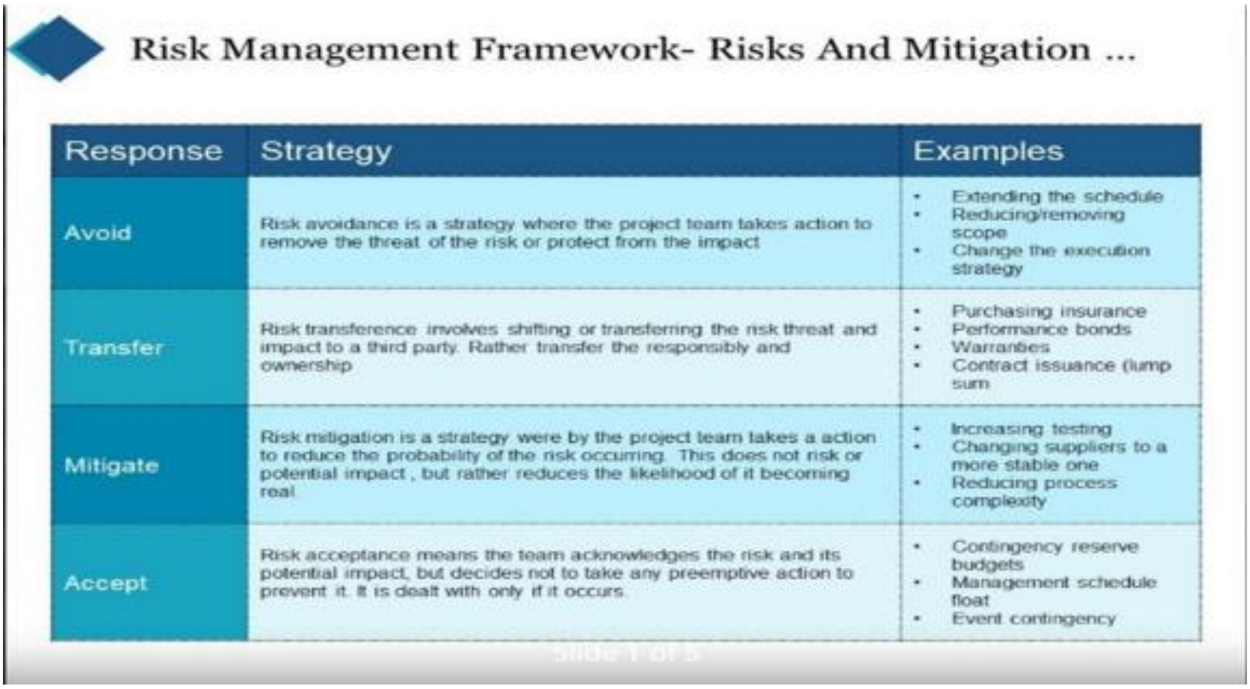


**SWOT ANALYSIS:**



| STRENGTHS | WEAKNESS |
|---|---|
| • SOCIAL CAUSE<br>• EASY TO USE INTERFACE<br>• EASY COMMUNICATION BETWEEN RECIEVER AND DONOR | • NOT ENOUGH WORKFORCE FOR SCALABILITY<br>• UI OPTIMISATION<br>• MAJOR MARKETING REQUIRED |
| OPPORTUNITIES | THREATS |
| • EXPANDING TO OTHER DELIVERY RELATED OPERATION<br>• PRODUCT TIE-UPS | • ALREADY ESTABLISH FOOD DELIVERY SERVICES<br>• SCAMMERS |

## 5.4 RMMM



Risk Management Framework- Risks And Mitigation ...

| Response | Strategy | Examples |
|---|---|---|
| Avoid | Risk avoidance is a strategy where the project team takes action to remove the threat of the risk or protect from the impact | • Extending the schedule<br>• Reducing/removing scope<br>• Change the execution strategy |
| Transfer | Risk transference involves shifting or transferring the risk threat and impact to a third party. Rather transfer the responsibly and ownership | • Purchasing insurance<br>• Performance bonds<br>• Warranties<br>• Contract issuance (lump sum |
| Mitigate | Risk mitigation is a strategy were by the project team takes a action to reduce the probability of the risk occurring. This does not risk or potential impact , but rather reduces the likelihood of it becoming real | • Increasing testing<br>• Changing suppliers to a more stable one<br>• Reducing process complexity |
| Accept | Risk acceptance means the team acknowledges the risk and its potential impact, but decides not to take any preemptive action to prevent it. It is dealt with only if it occurs | • Contingency reserve budgets<br>• Management schedule float<br>• Event contingency |

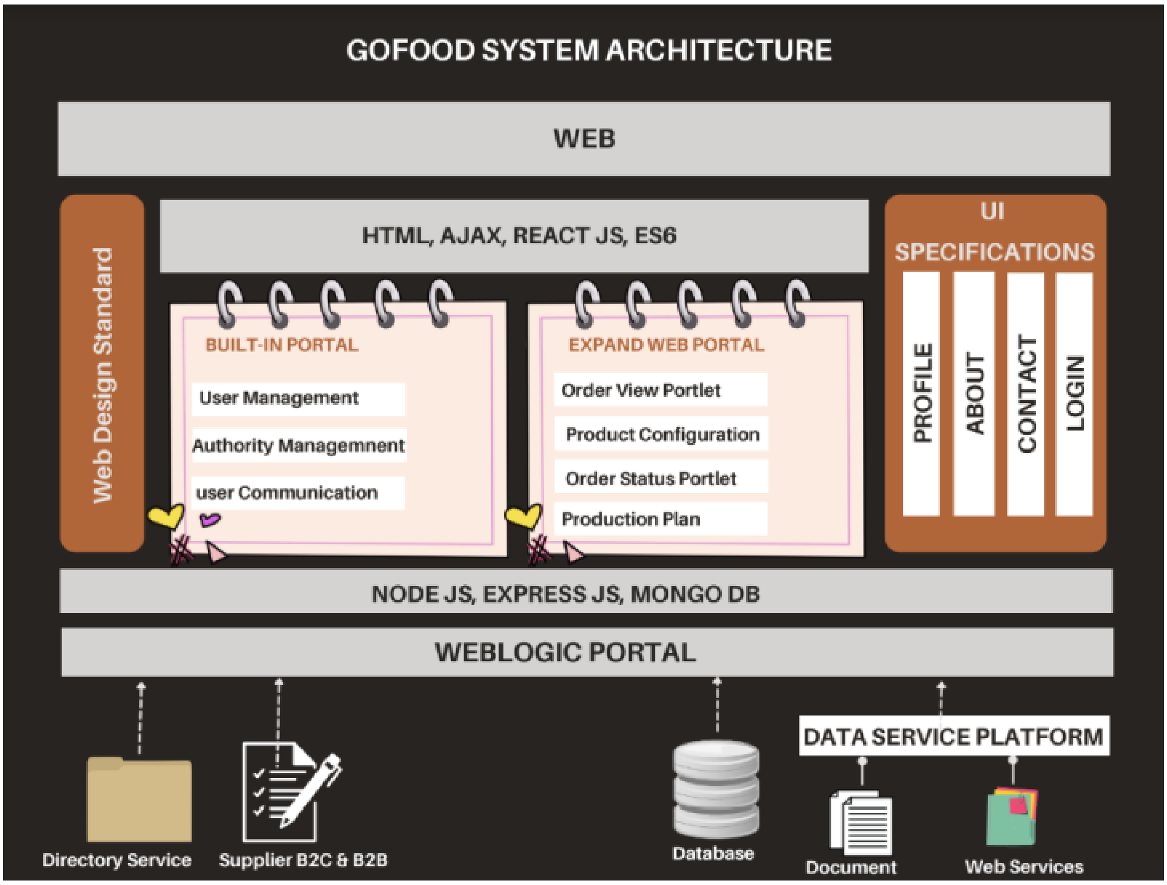
Slide 1 of 5

# CHAPTER-6

# SYSTEM ARCHITECTURE, USE CASE & CLASS DIAGRAM

To Design a System Architecture, use case and Class Diagram for student performace details.

## 6.1 SYSTEM ARCHITECTURE

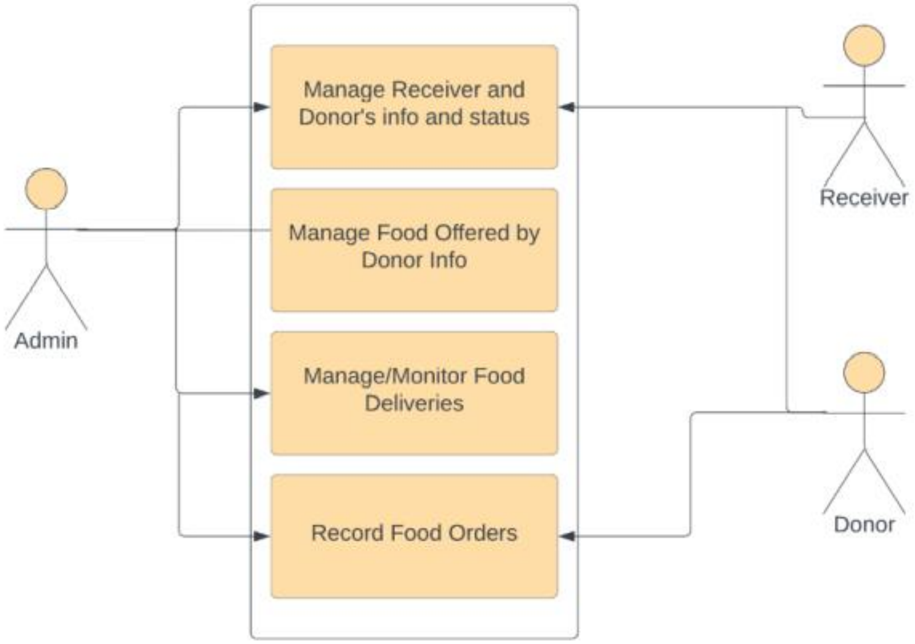


### 6.1.1 SYSTEM ARCHITECTURE

A system architecture diagram is the distribution of the functional correspondences.

These are formal elements, the embodiment of concepts and information. Architecture defines the relations between elements, among features, and the surrounding elements. Creating an Architecture diagram is not easy.

## 6.2 USE CASE DIAGRAM


Manage Receiver and Donor's info and status
Manage Food Offered by Donor Info
Manage/Monitor Food Deliveries
Record Food Orders
Admin
Receiver
Donor

**6.2.1 use case diagram**

The use case diagram are usually referred to as behavior diagram used to describe the actions of all user in a system. All user describe in use case are actors and the functionality as action of system.

The Use case diagram is a collection of diagram and text together that make action on goal of a process.

Use case diagram for student information system project. In Student information system there is a only one actor Admin can do all the activities to run the system.

## CLASS DIAGRAM

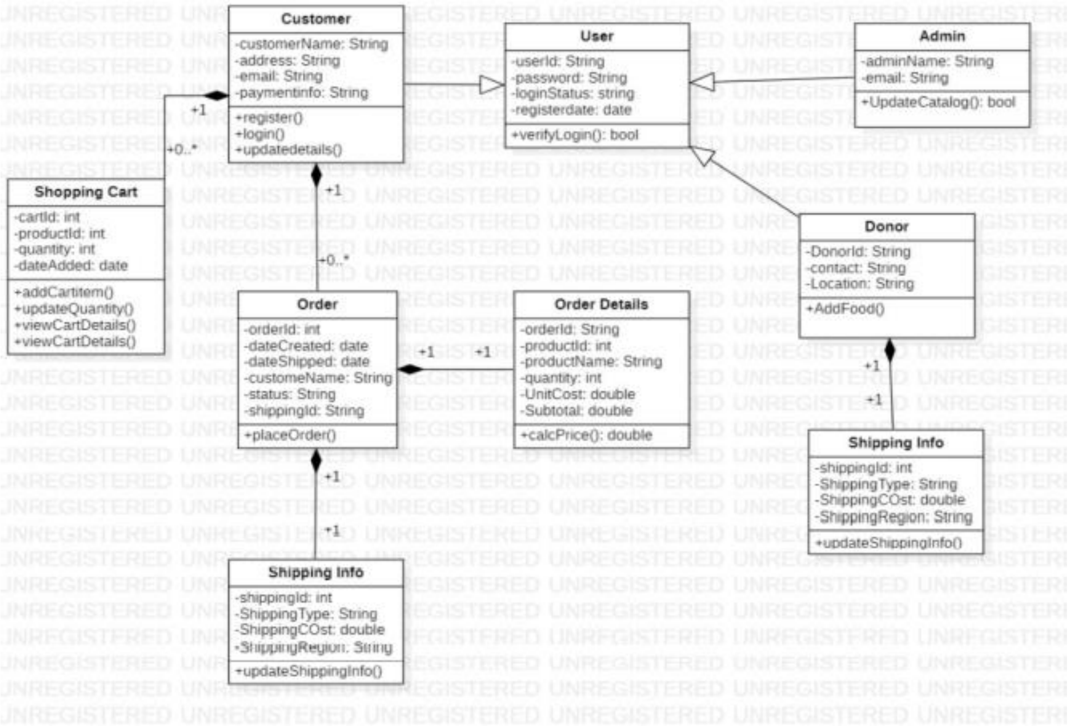


**Fig. 6.3 class diagram**

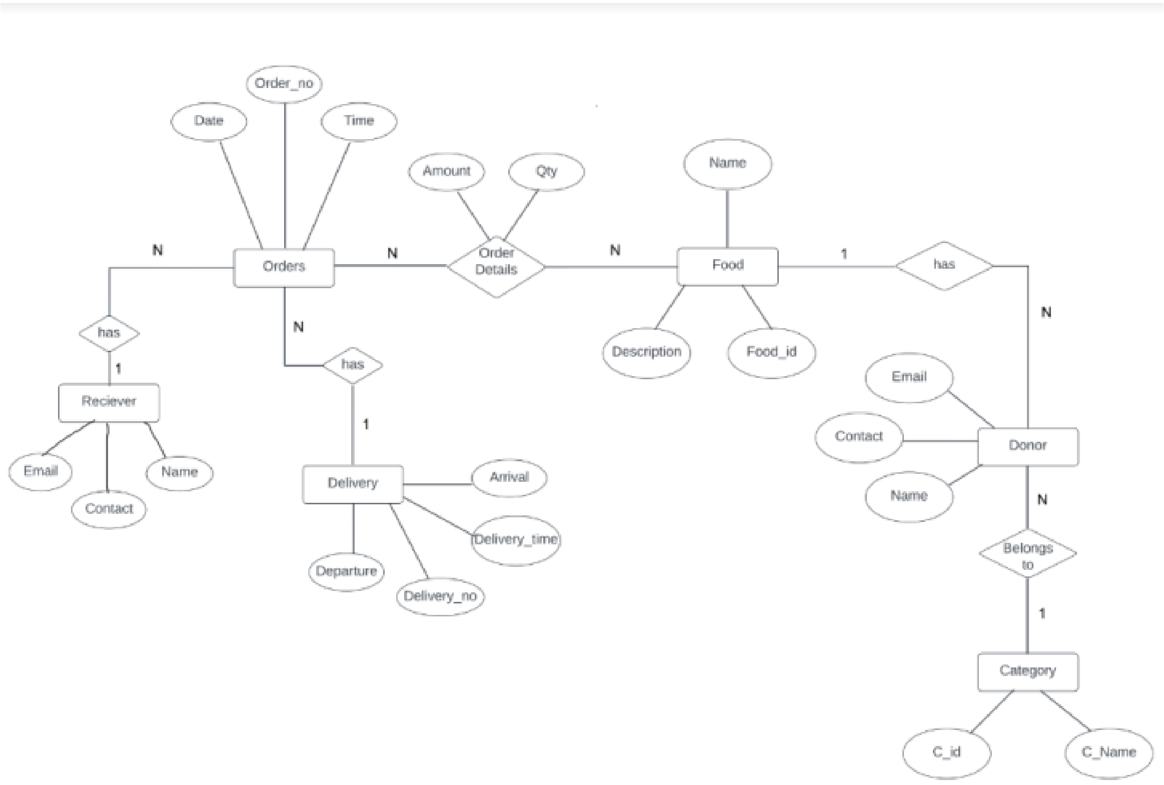
Class diagram is a static diagram. It represents the static view of an application.

Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.

# CHAPTER-7
# ENTITY RELATIONSHIP DIAGRAM

To create the Entity Relationship Diagram for Secure Digital Authentication System

**Entity Relationship Diagram:-**



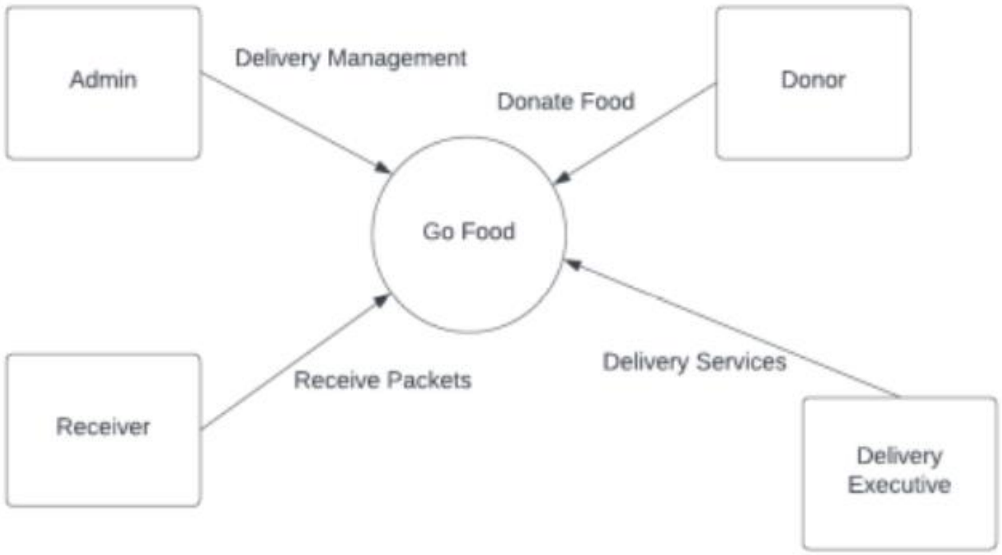
**7.1**

# CHAPTER-8

# DATA FLOW DIAGRAM

To develop the data flow diagram up to level 1 for the student performance details

## 8.1 Data Flow Diagram-level 0



**8.1.1**

The 0 level dfd known as **context level** data flow diagram. The context level data flow diagram (dfd) is describe the whole system. The (o) level dfd describe the all user modules who run the system.

## 8.2 Data Flow Diagram-level 1

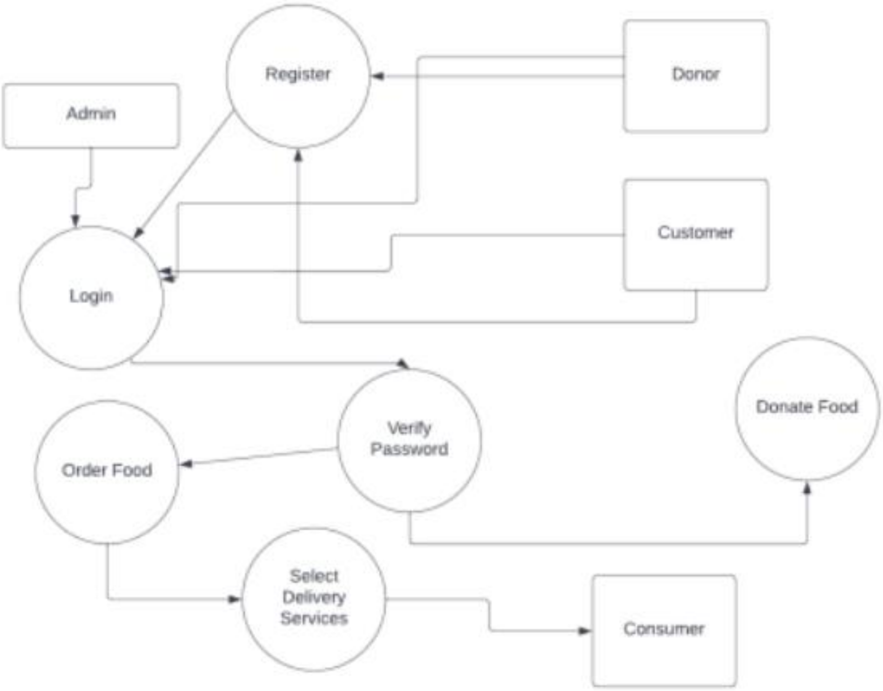


Fig.8.2.1 DFD L-1

# CHAPTER-9

## SEQUENCE & COLLABORATION DIAGRAM

To create the sequence and collaboration diagram for the student performace details.

### 9.1 Sequence Diagram:-

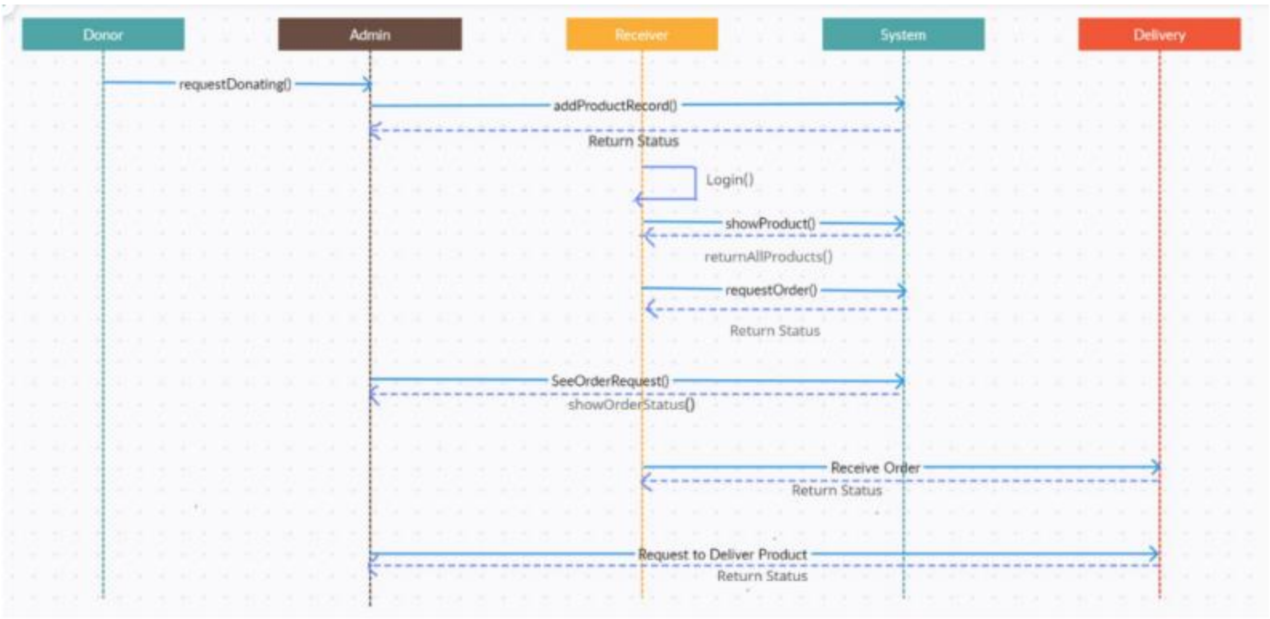


**Fig.9.1.1 sequence diagram**

The messages are shown and annotated to assist you in creating a Food Donation System. If you have further ideas, you can change the design. You can also expand on this design and use it as a blueprint for your project.

By drawing a sequence diagram, you will be able to comprehend and educate yourself on how the Food Donation System operates. Because it determines the objects, actors, and messages thatare required, as well as their interactions.

By drawing a sequence diagram, you will be able to comprehend and comprehend how the System operates. It aids in the identification of required objects, actors, and messages, as well as their interactions.

**9.2 collaboration diagram:-**

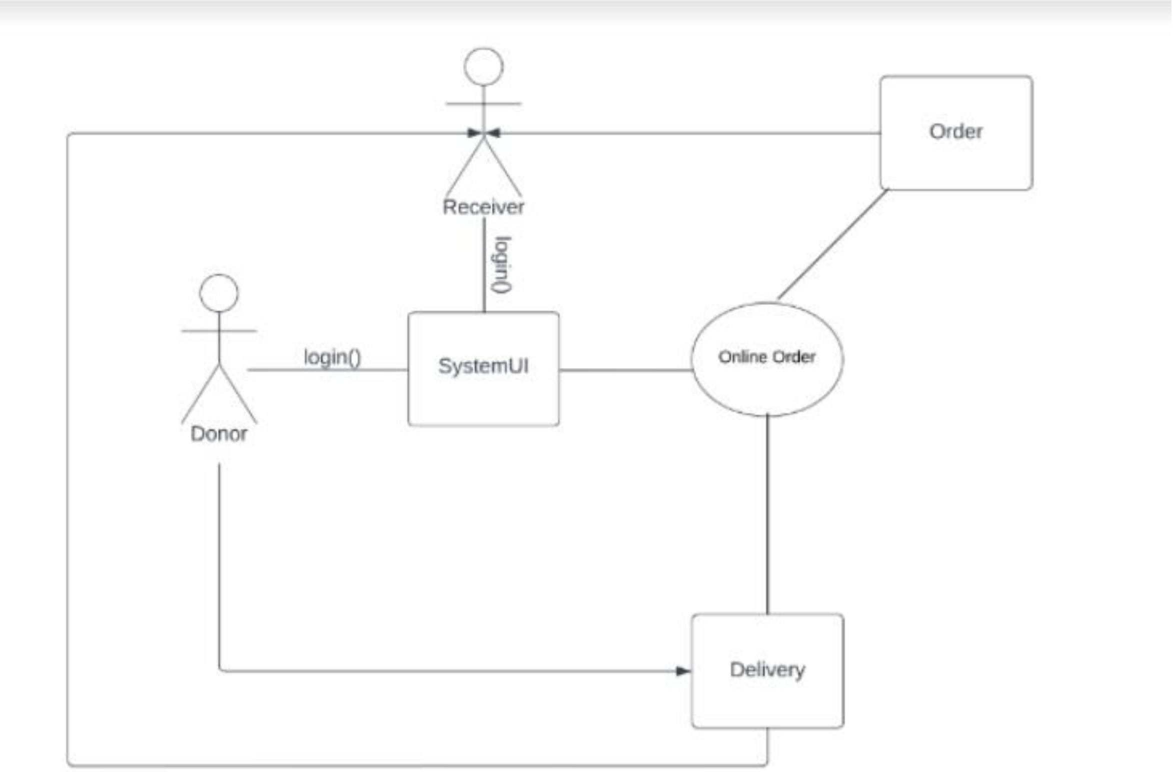


**Fig.9.2.1 collaboration diagram**

# CHAPTER-10

## DEVELOPMENT OF TESTING FRAMEWORK/USER INTERFACE

To develop the testing framework and/or user interface framework for the student performace details.

### Executive Summary

Testing the software application using linear automation framework

### Test Plan

### Scope of Testing

Product features to be tested to build a reliable assessment of the product's quality:

1. Website Server Connectivity
2. Layout Testing
3. Dashboard
4. Authentication

**Functional:** Backend and Frontend modules covered and tested for navigation, user input and checkpoints

**Non-Functional:** All non functional requirements including performance, configuration and security were tested.

### Types of Testing, Methodology, Tools

| Category | Methodology | Tools Required |
|---|---|---|
| Functional Requirements | Manual | Linear Automation |

## CHAPTER-11

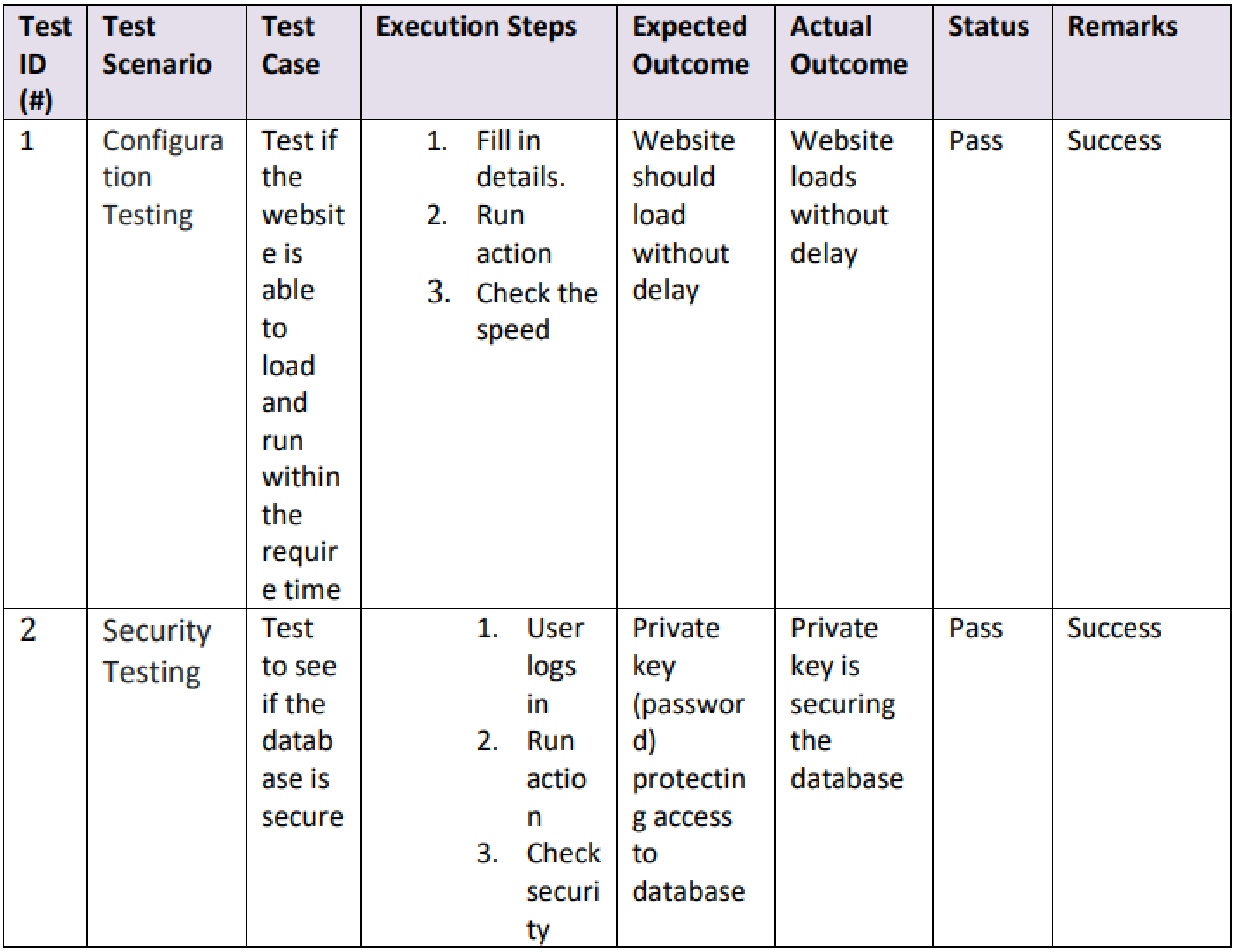
## TEST CASES & REPORTING

To develop the test cases manual for the student performace details..

### 11.1 Functional Test Cases

| est<br>D<br>#) | Test Scenario | Test Case | Execution Steps | Expected Outcome | Actual Outcome | Status | Remarks |
|---|---|---|---|---|---|---|---|
| | User Authentication | Accept a valid email address and location for creating account and logging in | 1. User clicks on User Registration link<br>2. Enter the email id<br>3. Click Register button | User should be taken to homepage of the website | User is taken to the homepage of the website | Pass | success |
| | dashboard | Provides information regarding donor's order details | User can add/edit items available for delivery | Added item should be displayed | Item is added and user is redirected to dashboard. | Pass | success |
| | Menu-page | The list of food cards are displayed with food name, quantity, donor's contact number. | User clicks on food and view details. | The receiver can contact the donor and place order. | The food photo and its details. | Pass | Success |

# Non-Functional Test Cases



| Test ID (#) | Test Scenario | Test Case | Execution Steps | Expected Outcome | Actual Outcome | Status | Remarks |
|---|---|---|---|---|---|---|---|
| 1 | Configuration Testing | Test if the website is able to load and run within the require time | 1. Fill in details.<br>2. Run action<br>3. Check the speed | Website should load without delay | Website loads without delay | Pass | Success |
| 2 | Security Testing | Test to see if the database is secure | 1. User logs in<br>2. Run action<br>3. Check security | Private key (password) protecting access to database | Private key is securing the database | Pass | Success |

# CHAPTER-12

## ARCHITECTURE/DESIGN/FRAMEWORK/IMPLE-MENTATION

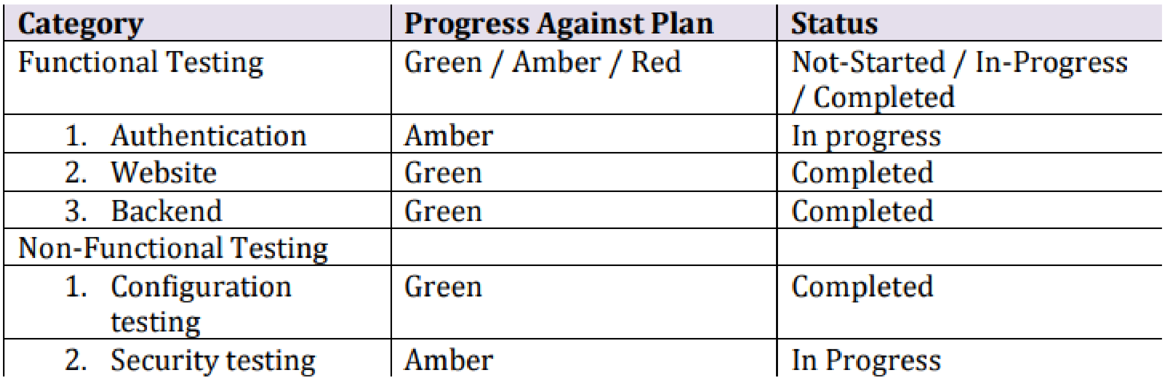
To provide the details of architectural design/framework/implementation. Thus, the details of architectural design/framework/implementation along with the screen shots were provided.

### CURRENT STATUS OF TESTING:

1. Nonfunctional tests for configuration and security were conducted.
2. Functional tests included user authentication and testing out the order placed by receiver.

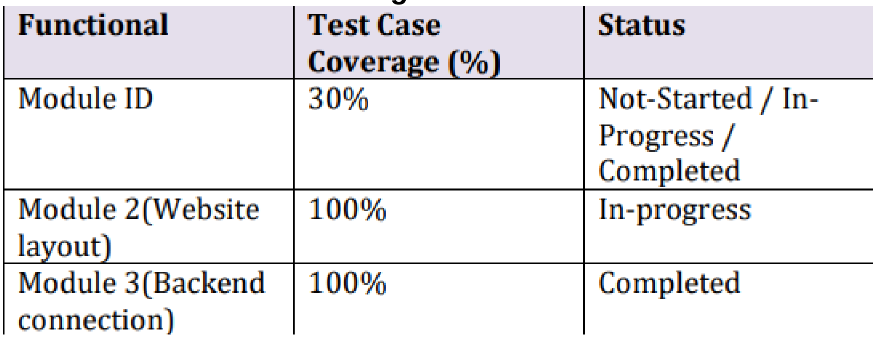
### PRESENT OBSTACLES:

1. The database is not connected to the project.
2. Increasing the speed at which the code is formatted.
3. Single page design pattern needs to be implemented.



| Category | Progress Against Plan | Status |
|---|---|---|
| Functional Testing | Green / Amber / Red | Not-Started / In-Progress / Completed |
| 1. Authentication | Amber | In progress |
| 2. Website | Green | Completed |
| 3. Backend | Green | Completed |
| Non-Functional Testing | | |
| 1. Configuration testing | Green | Completed |
| 2. Security testing | Amber | In Progress |

**Fig.12.1**



| Functional | Test Case Coverage (%) | Status |
|---|---|---|
| Module ID | 30% | Not-Started / In-Progress / Completed |
| Module 2(Website layout) | 100% | In-progress |
| Module 3(Backend connection) | 100% | Completed |

**Fig.12.2**

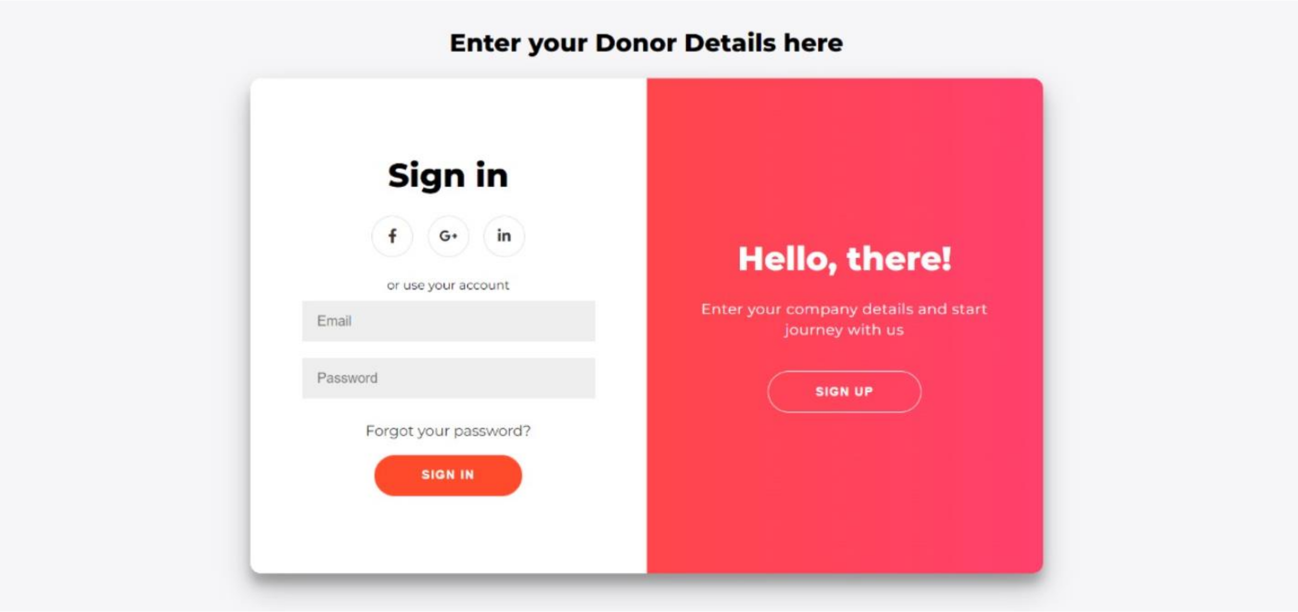
- Poor planning can only lead to project failure or delays and cost overruns. It will also infuriate all stakeholders.
- For the project manager, this presents a host of problems that need to be addressed, ranging from planning for the new situation and strategizing for a solution that entails minimal inconvenience for stakeholders.
- Assessing your stakeholder's interests also provides a strong basis for legal compliance, enabling you to put protocols in place that protect the privacy of stakeholders.
- By developing a deep understanding of your project stakeholders, project managers can create sound plans to eliminate delays, mitigate risks, and better align projects with wider business goals.

| Category | Progress Against Plan | Status |
|---|---|---|
| Functional Testing | amber | Still in progress |
| Non-Functional Testing | green | completed |

| Functional | Test Case Coverage (%) | Status |
|---|---|---|
| 73196 | 60% | In progress |
| 3197 | 100% | completed |
| 3198 | 100% | completed |
| 3199 | 100% | completed |

## Exp-13
## Architectural design/framework

### 13.1



Enter your Donor Details here

Sign in

or use your account

Email

Password

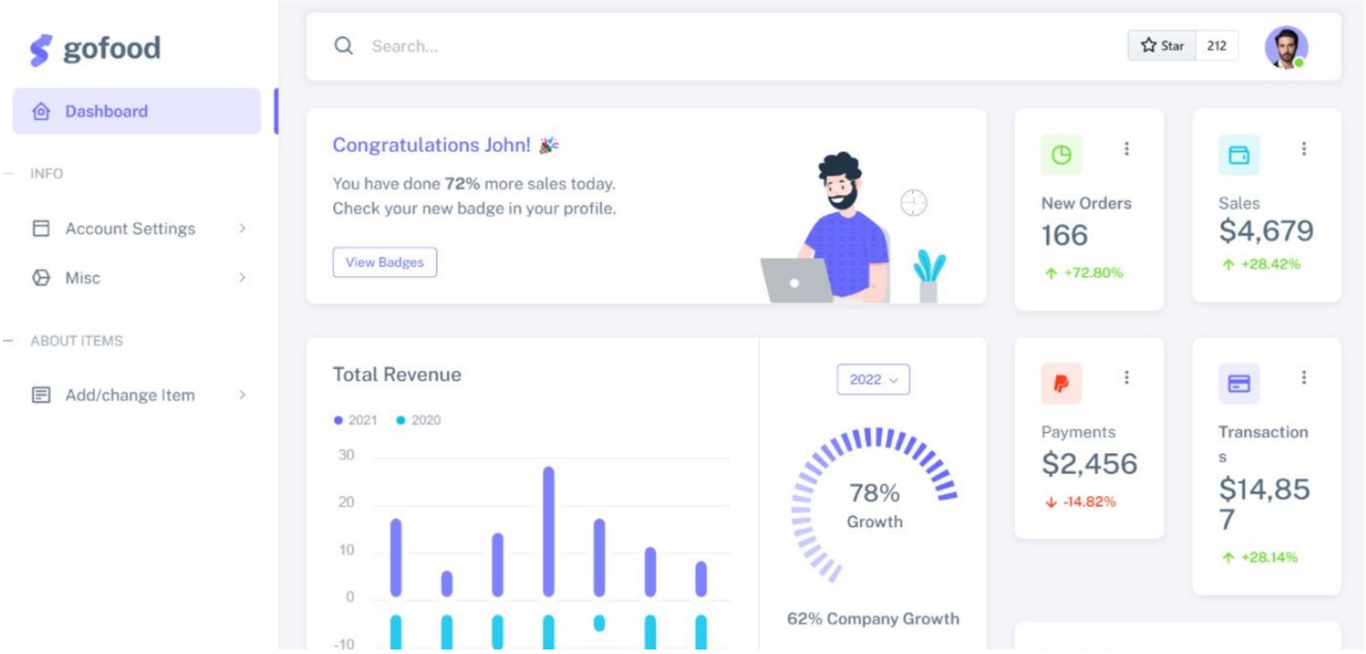
Forgot your password?

Hello, there!

Enter your company details and start journey with us

Fig.13.2.1 student login page

### 13.2



Congratulations John!

You have done **72%** more sales today.
Check your new badge in your profile.

New Orders 166 +72.80%

Sales $4,679 +28.42%

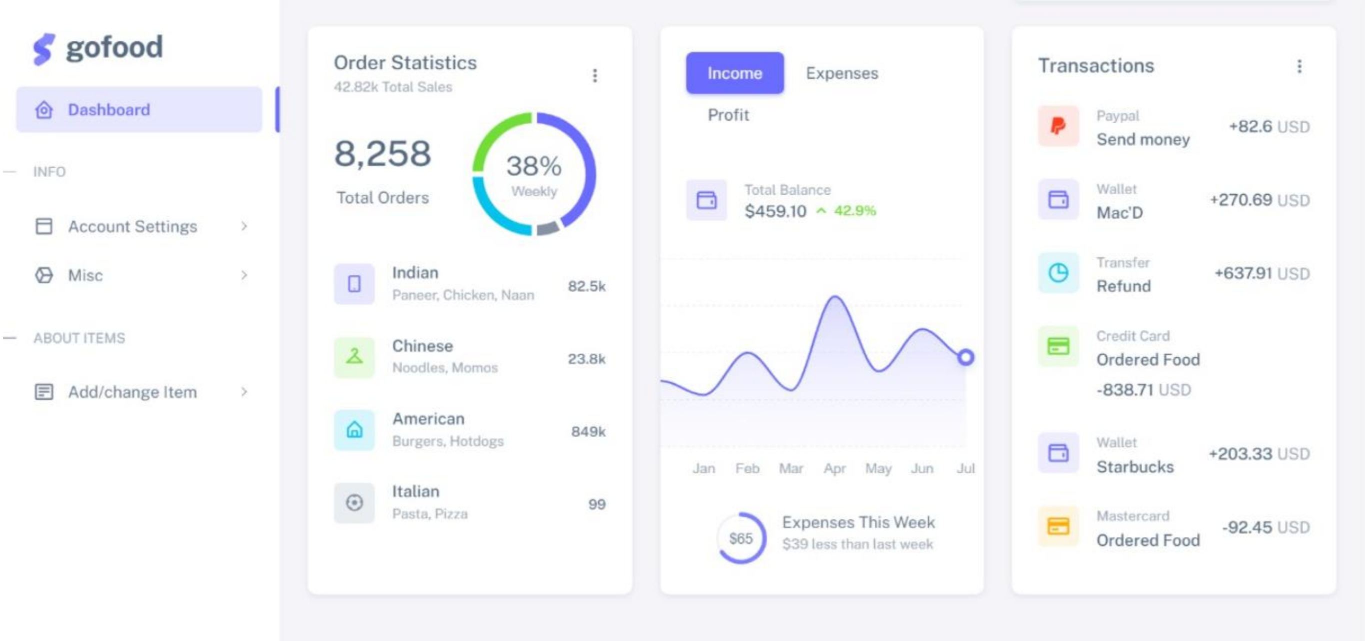
Total Revenue

78% Growth

62% Company Growth

Payments $2,456 -14.82%

Transaction
s
$14,85
7
+28.14%



gofood

Dashboard

INFO

Account Settings

Misc

ABOUT ITEMS

Add/change Item

Order Statistics

42.82k Total Sales

8,258

Total Orders

38%

Weekly

| | |
|---|---|
| Indian<br>Paneer, Chicken, Naan | 82.5k |
| Chinese<br>Noodles, Momos | 23.8k |
| American<br>Burgers, Hotdogs | 849k |
| Italian<br>Pasta, Pizza | 99 |

Income Expenses Profit

Total Balance

$459.10 ^ 42.9%

$65

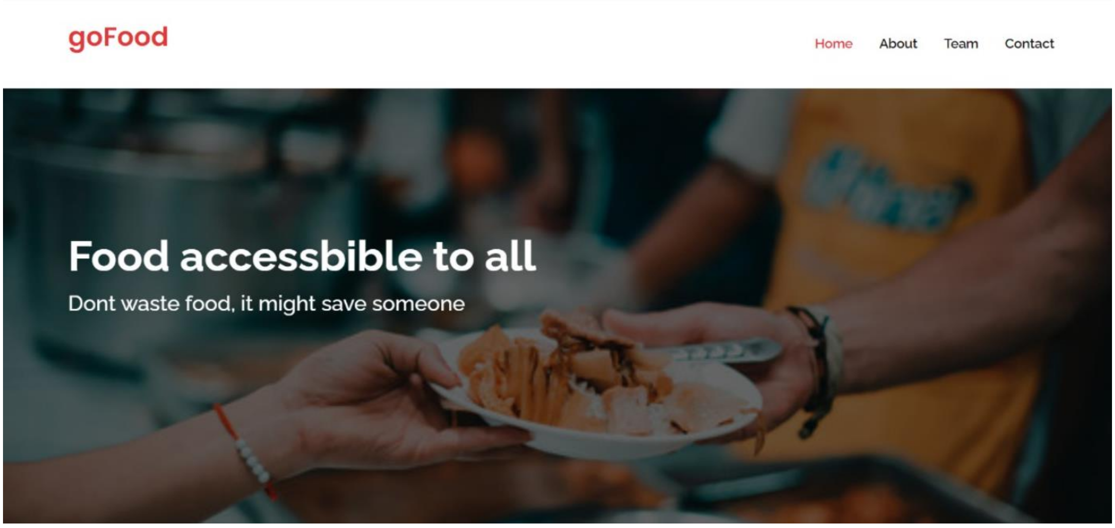
Expenses This Week

$39 less than last week

Transactions

| | |
|---|---|
| Paypal<br>Send money | +82.6 USD |
| Wallet<br>Mac'D | +270.69 USD |
| Transfer<br>Refund | +637.91 USD |
| Credit Card<br>Ordered Food | -838.71 USD |
| Wallet<br>Starbucks | +203.33 USD |
| Mastercard<br>Ordered Food | -92.45 USD |

Fig.13.3.1 dashboard



goFood

Home About Team Contact

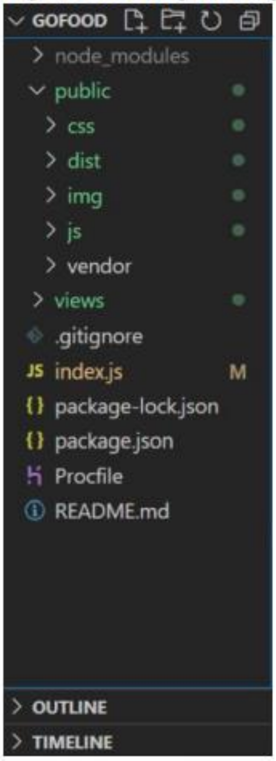
Food accessbible to all

Dont waste food, it might save someone

Fig. 13.3.2 Home Page

## IMPLEMENTATION:

## FOLDER ARCHITECTURE:

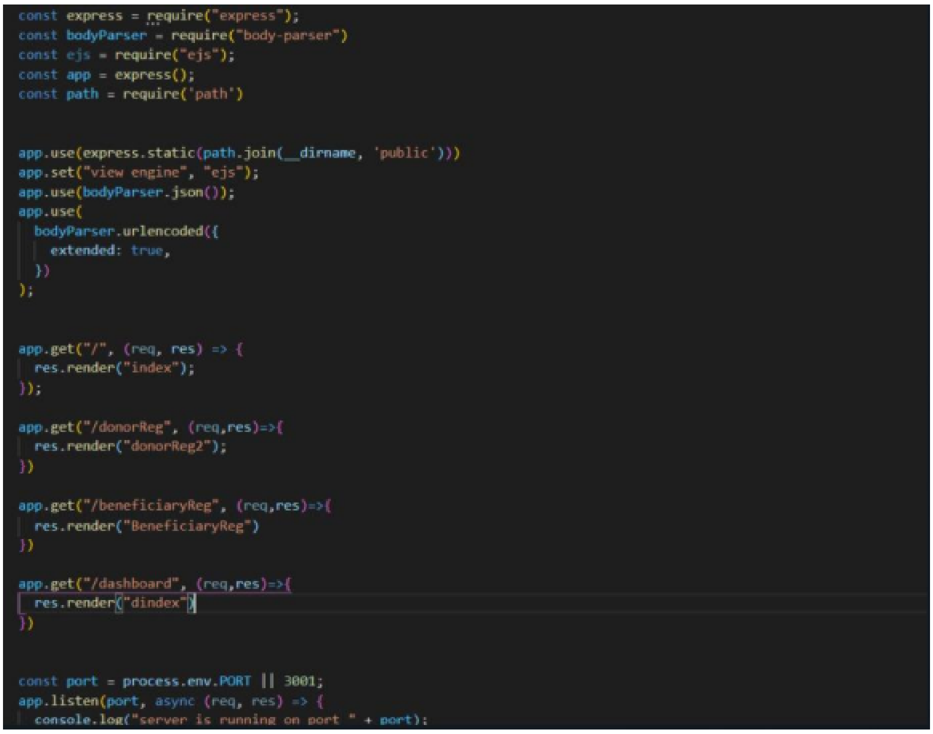


```
GOFOOD
  node_modules
  public
    css
    dist
    img
    js
    vendor
  views
  .gitignore
  index.js      M
  package-lock.json
  package.json
  Procfile
  README.md
OUTLINE
TIMELINE
```

## SOURCE CODE:

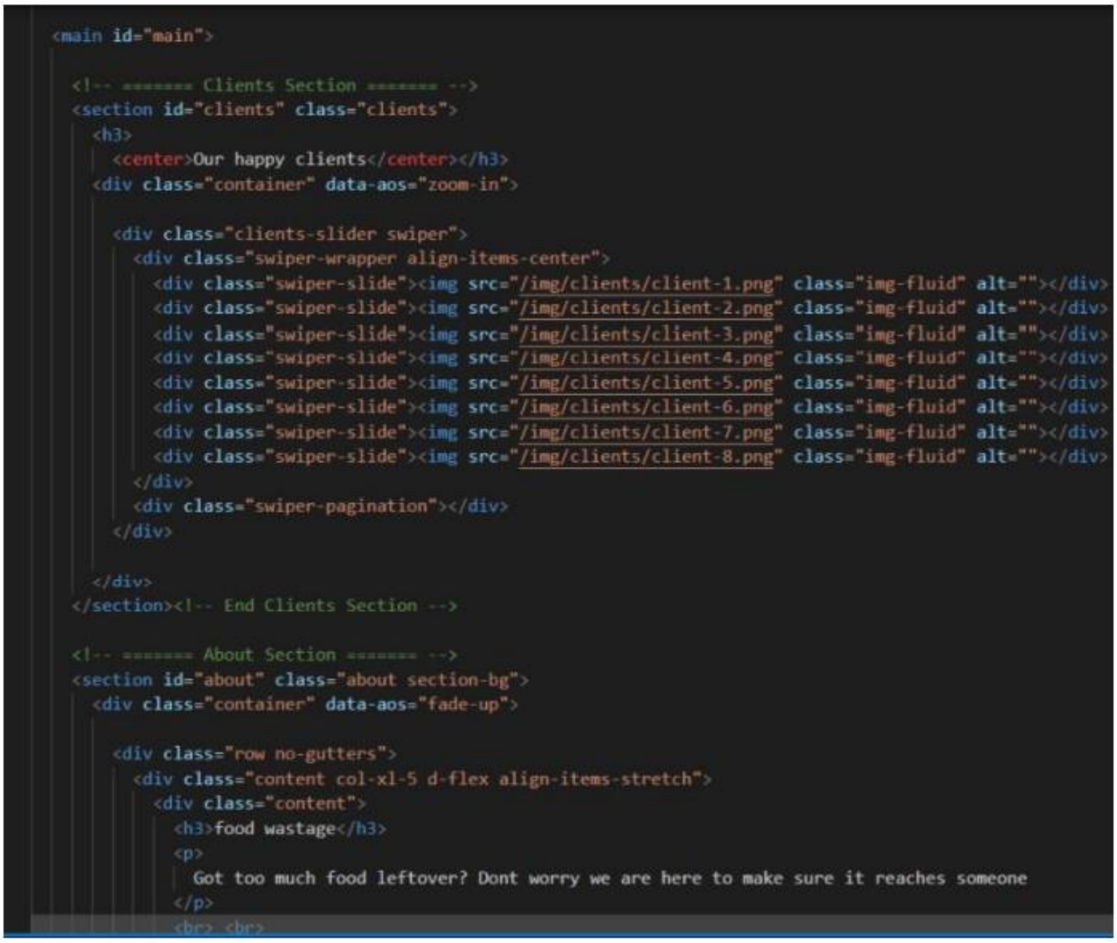
## MODULES:

### 1. BACKEND:



```
const express = require("express");
const bodyParser = require("body-parser")
const ejs = require("ejs");
const app = express();
const path = require('path')


app.use(express.static(path.join(__dirname, 'public')))
app.set("view engine", "ejs");
app.use(bodyParser.json());
app.use(
  bodyParser.urlencoded({
    extended: true,
  })
);


app.get("/", (req, res) => {
  res.render("index");
});

app.get("/donorReg", (req,res)=>{
  res.render("donorReg2");
})

app.get("/beneficiaryReg", (req,res)=>{
  res.render("BeneficiaryReg")
})

app.get("/dashboard", (req,res)=>{
  res.render("dindex")
})


const port = process.env.PORT || 3001;
app.listen(port, async (req, res) => {
  console.log("server is running on port " + port);
```

## 2. FRONT-END:



```html
<main id="main">

  <!-- ======= Clients Section ======= -->
  <section id="clients" class="clients">
    <h3>
      <center>Our happy clients</center></h3>
    <div class="container" data-aos="zoom-in">

      <div class="clients-slider swiper">
        <div class="swiper-wrapper align-items-center">
          <div class="swiper-slide"><img src="/img/clients/client-1.png" class="img-fluid" alt=""></div>
          <div class="swiper-slide"><img src="/img/clients/client-2.png" class="img-fluid" alt=""></div>
          <div class="swiper-slide"><img src="/img/clients/client-3.png" class="img-fluid" alt=""></div>
          <div class="swiper-slide"><img src="/img/clients/client-4.png" class="img-fluid" alt=""></div>
          <div class="swiper-slide"><img src="/img/clients/client-5.png" class="img-fluid" alt=""></div>
          <div class="swiper-slide"><img src="/img/clients/client-6.png" class="img-fluid" alt=""></div>
          <div class="swiper-slide"><img src="/img/clients/client-7.png" class="img-fluid" alt=""></div>
          <div class="swiper-slide"><img src="/img/clients/client-8.png" class="img-fluid" alt=""></div>
        </div>
        <div class="swiper-pagination"></div>
      </div>

    </div>
  </section><!-- End Clients Section -->

  <!-- ======= About Section ======= -->
  <section id="about" class="about section-bg">
    <div class="container" data-aos="fade-up">

      <div class="row no-gutters">
        <div class="content col-xl-5 d-flex align-items-stretch">
          <div class="content">
            <h3>food wastage</h3>
            <p>
              Got too much food leftover? Dont worry we are here to make sure it reaches someone
            </p>
            <br> <br>
```

# CONCLUSION

Hence the documentation for GoFood was created and completed successfully. This documentation will help the stakeholders and creators of the application better understand the project and give them an in-depth analysis for future prospects.

# REFERENCES

Wikipedia, Medium: For Several references on various topics.
Geeks For Geeks, Udemy: For learning technical Concepts.
W3Schools and Lucidchart for Diagram and related things.
Canva and Whimsical: For designing the user interface and flow charts.