FOOD DONATION SYSTEM FOR THE NEEDY

A MAJOR PROJECT REPORT

Submitted by

CH.EN.U4AIE20001	ANBAZHAGAN E
CH.EN.U4AIE20053	RAMYA POLAKI
CH.EN.U4AIE20069	TRINAYA KODAVATI
CH.EN.U4AIE20031	SMITHIN REDDY K
CH.EN.U4AIE20035	ABHIRAM KUNCHAPU

in partial fulfillment of the award of the degree

Of

BACHELOR OF TECHNOLOGY

IN

DATABASE MANAGEMENT SYSTEM

AMRITA SCHOOL OF ENGINEERING, CHENNAI

AMRITA VISHWA VIDYAPEETHAM

CHENNAI – 601103, TAMIL NADU

AMRITA VISHWA VIDYAPEETHAM

AMRITA SCHOOL OF ENGINEERING, CHENNAI, 601103

BONAFIDE CERTIFICATE

This is to certify that the major project report entitled “**FOOD DONATION SYSTEM FOR THE NEEDY”** submitted by

CH.EN.U4AIE20001	ANBAZHAGAN E
CH.EN.U4AIE20053	RAMYA POLAKI
CH.EN.U4AIE20069	TRINAYA KODAVATI
CH.EN.U4AIE20031	SMITHIN REDDY K
CH.EN.U4AIE20035	ABHIRAM KUNCHAPU

in partial fulfillment of the requirements for the award of the **bachelor of Master of Technology** in **COMPUTER SCIENCE AND ENGINEERING** is a bonafide record of the work carried out under my guidance and supervision at Amrita School of Engineering, Chennai.

Signature

Mr. R Annamalai

CSE Dept Assistant Professor

This project report was evaluated by us on ……………...

INTERNAL EXAMINER EXTERNAL EXAMINER

DECLARATION

We the undersigned solemnly declare that the thesis “Food Donation System for the needy” is based on our own work carried out during the course of our study under the supervision of Mr. R Annamalai, Assistant professor of Computer Science & Engineering, and has not for the basis for the award of any other degree or diploma,in this or any other Institution or University. In keeping with the ethical practice of reporting scientific information, due acknowledgment has been made wherever the findings of others have been cited.

Anbazhagan, Smithin, Ramya, Trinaya, Abhiram

ABSTRACT

The food donation system provides an interface between people wanting to donate food and people who are in need of food. It is useful for NGOs to recognize the donators and work with them to provide food for the poor. It helps reduce manual work for volunteers and encourages people to join the community due to the application’s ease of use. The donor simply uploads the type of food, the number of people it is sufficient for and the location of the Food Corner branch they will drop the food the ones in need of food can see a list of all the options available, and they can like the post and go to a respective branch to collect the food. Avoids the process of needing to search for an organization for donation. This way it can help build trust and a safe and reliable ground for donation and distribution in the community.

ACKNOWLEDGEMENT

I wish to record my deep sense of gratitude and profound thanks to my research supervisor Mr.R Annamalai, Assistant Professor, Department of Computer Science Engineering, Amrita Vishwa Vidyapeetham, Chennai, for his / her keen interest, guidance, and constant encouragement during all stages of my work, to bring this thesis into fruition.

I am extremely indebted to Dr. Prasanna Kumar, Department of CSE-AI, Amrita Vishwa Vidyapeetham, Chennai, and panel members for their valuable suggestions and support during the course of my research work.

I also thank the faculty members of the Department of CSE, Amrita Vishwa Vidyapeetham, Chennai, for their valuable support throughout the course of my research work.

Anbazhagan, Smithin, Ramya, Trinaya, Abhiram

LIST OF FIGURES

S.No Topic **Page. No**

INDEX

CHAPTER 1

INTRODUCTION

1.1 PROBLEM STATEMENT

All food produced is lost or squandered to the tune of up to one-third. Food is wasted and decomposes instead of feeding the estimated 811 million hungry people worldwide, which contributes 8% to 10% of the world's greenhouse gas emissions. Loss and waste occur along the entire food supply chain: upstream due to inadequate storage facilities, consumer confusion or inefficient behavior, and downstream due to surplus or faulty products on farms.

Manufacturers, retailers, and eateries frequently point to storage and transportation expenses as justifications for not donating food. In Colombia, safe food donations to registered charity organizations or food banks are rewarded with a national income tax credit, balancing costs and making the act of giving more inexpensive. Strong laws and regulations have made it possible for ABACO, or the Network of Colombian Food Banks, to recover and distribute an estimated 25,089 tonnes of food to more than 650,000 individuals across the nation.

1.2 AIM AND OBJECTIVE

This project works to provide:

i. A common reliable ground for donation and distribution
ii. Faster approach through a web application
iii. Bridge communication between NGOs, restaurants, and individual donations

Our project allows for a more effective, socially responsible distribution of surpluses, which improved the food system and reduced the amount of food that ended up in landfills. In summary, such projects benefit both the environment and humans.

1.3 EXISTING SYSTEMS AND DRAWBACKS

Currently, anyone who wants to donate food or other supplies must go in person to the charities. If not, they must look for websites where they can contribute extra food. Large producers, wholesalers, and organized groups typically donate food to food banks or throw away tons of food every day. They must look for a charity that requires meals. Contacting the organization to confirm the requirement takes a long time in this process. The person must get in touch with another organization if they do not require the meals. The donor feels worn out and drained as a result.

Another system available provides an interface between the donator and receiver, but the receiver has to go and pick up the food at the location that the donator is willing. This creates a traveling issue for the poor due to inconvenience or lack of transportation facilities on their budget.

1.4 PROPOSED SYSTEM

Our system creates a web interface between the donator and receiver and the donator has to drop off the food at nearby branches the ones in need of food can come and collect food at the branches that are near to them, or if they are willing to travel they can go to other branches based on the menu of items available at different branches. This overcomes the existing systems as it helps to create a common and constant place of donation and distribution. If these branches are installed at locations where there is demand among the poor, then it can be helpful for them in terms of ease to access. We shall look into the system in depth further in chapter 3.

CHAPTER 2

LITERATURE SURVEY

The paper on Helping Hands is a web-based program that offers a platform for giving used items and food scraps to all charitable causes. If someone has leftover food, they can input the quantity and their address in the application, and the administrator will keep track of the information for the food donor. The donor can set up an account, and whenever there is food waste, they can log in and submit a request to the administrator. And the admin also keeps track of the buyer's information. Following the admin's review of the donor request, the admin issues an alert message, such as the time to arrive and collect the food. And the administrator receives food from the donor through their local agent before giving it to the closest orphans or underprivileged individuals. Giving the donor a warning message after receiving the food from the agent on behalf of the admin. The administrator can be asked to gather information on the orphanage if the donor requires any information about it in order to aid it. [1]

By donating leftover food to individuals or organizations that are in need, they have attempted to eliminate food waste via a smartphone app.

In the event that a donor has any extra food, the needy will increase their request. The donors on the list are informed of this request. The request is then approved by the available donor. By hiring delivery managers, the delivery system is controlled. The second delivery system option is for volunteers to accompany us as we distribute meals in nearby communities. The Food Bank is the third choice. If we get food at unusual hours, we can donate it to a food bank, where it will be cared for. [2]

A user can send a message within the program if they want to give something. Other users will see a notification of this message in the donations tab. This message will be kept in the database on the backend. The orphanages that want to claim the donations can get in touch with the donor after receiving a notification and reply. The system's user interface will be easy and approachable, and Android is the intended OS. [3]

A food recipient is an NGO in need of food that will check for any food requests in the system and can accept those requests if necessary. A donor can be any institution or individual who wishes to donate food and will submit their request to the system. This message will be displayed as a notification in the system to other NGOs (admin). The volunteer will then be assigned by the NGO, get the notification, and be directed to the donor location using GPS API to pick up food. The volunteer will next use a food moisture sensor from the Internet of Things to assess quality assurance. and a volunteer will also inspect the food's shelf life to determine its quality. If the food is in good condition after being checked for shelf life, volunteers will collect it, and a notification will be issued to the system. The system will determine the closest slum region to distribute food to using the k-NN algorithm. [4]

CHAPTER 3

PROPOSED SYSTEM

3.1 SYSTEM ARCHITECTURE

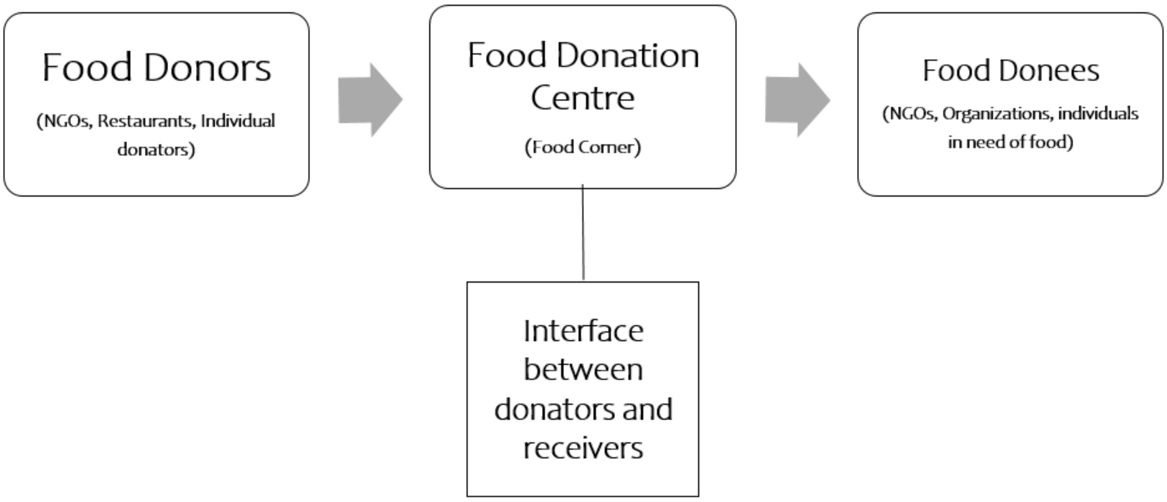

Food Donors
(NGOs, Restaurants, Individual donators)
Food Donation Centre
(Food Corner)
Food Donees
(NGOs, Organizations, individuals in need of food)
Interface between donators and receivers

Fig 3.1 Architecture of the proposed system

3.2 USE CASE ACTIVITY DIAGRAM

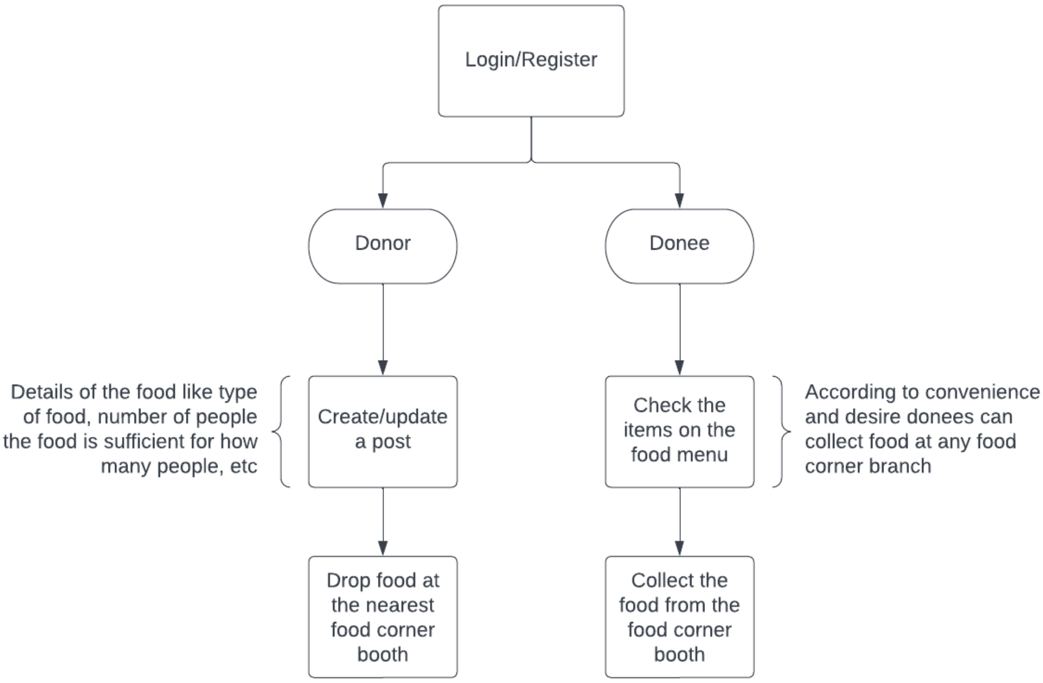

Login/Register
Donor
Donee
Details of the food like type of food, number of people the food is sufficient for how many people, etc
Create/update a post
Check the items on the food menu
According to convenience and desire donees can collect food at any food corner branch
Drop food at the nearest food corner booth
Collect the food from the food corner booth

Fig 3.2 Activity Diagram of the model

3.3 ENTITY RELATIONSHIP DIAGRAM

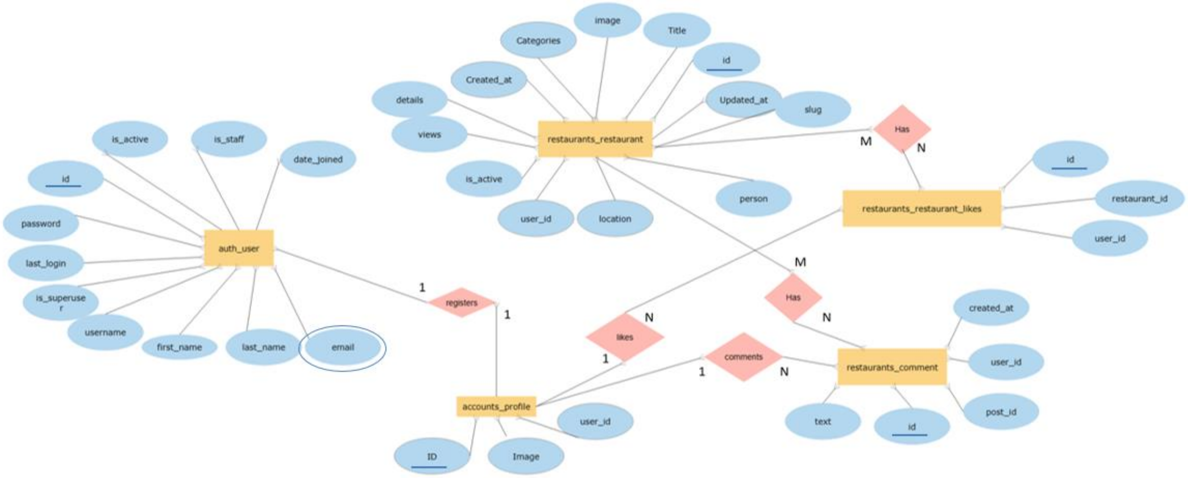


Fig 3.3 ER Diagram of the model

CHAPTER 4

IMPLEMENTATION AND CODES

4.1 WEB INTERFACE

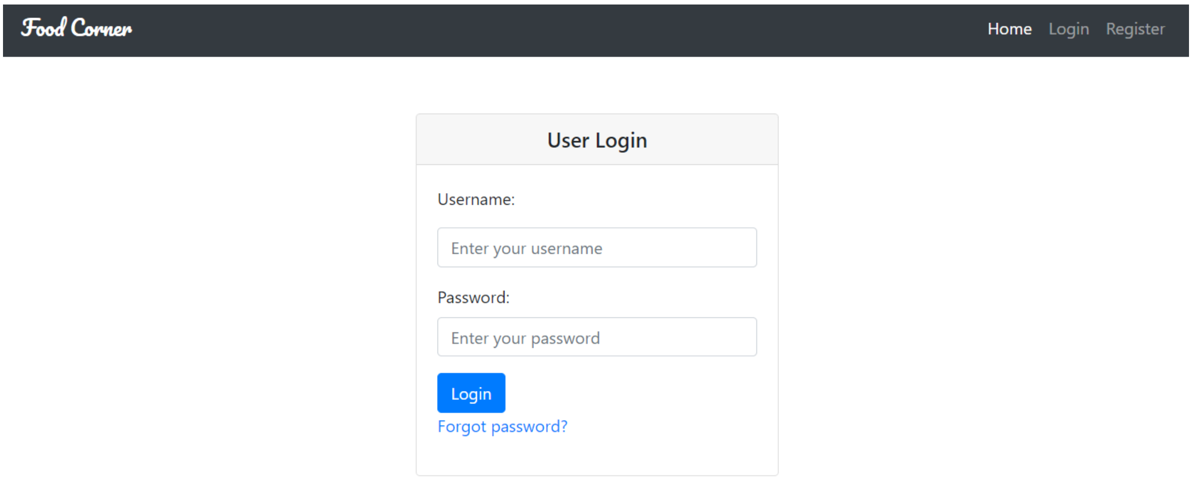
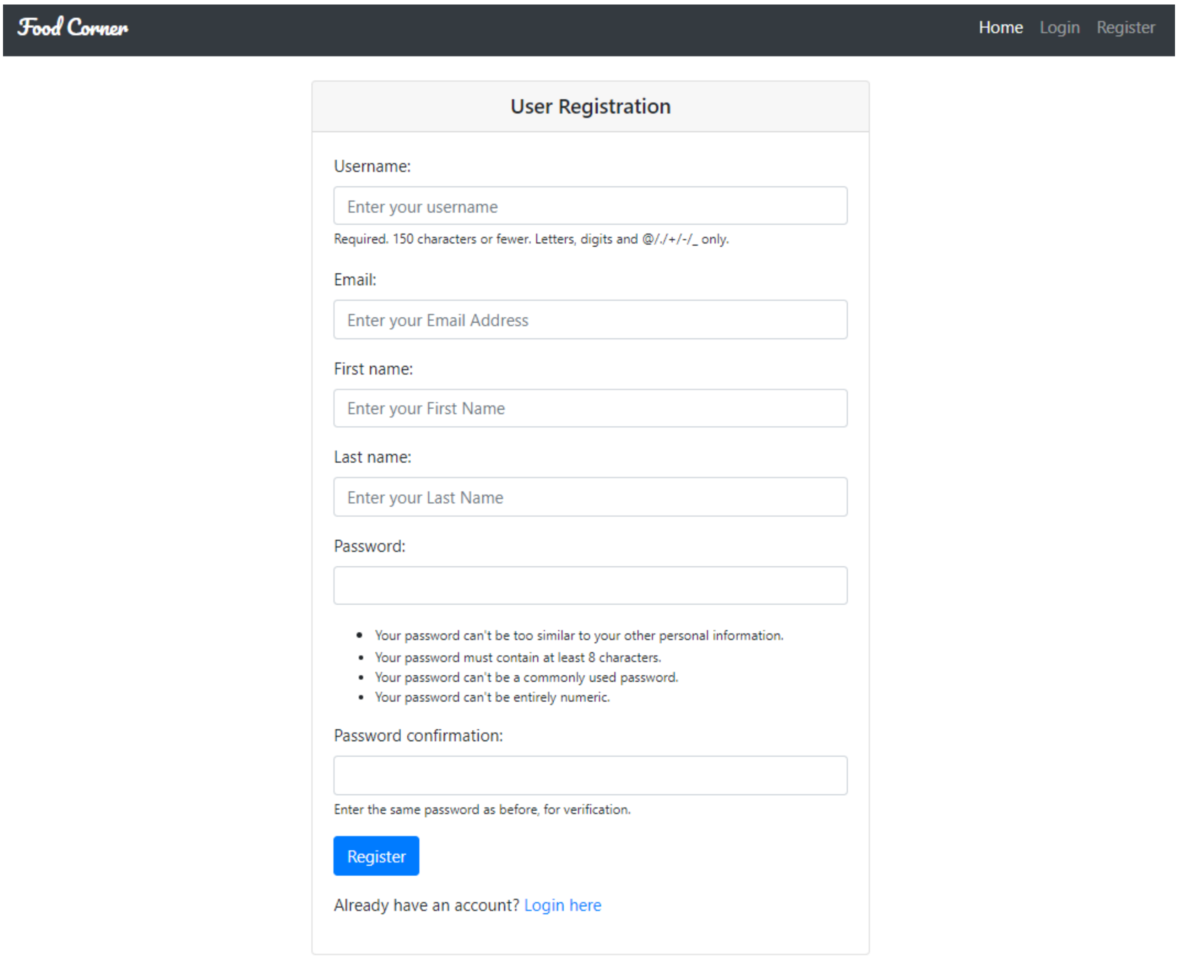


Fig 4.1 Login page



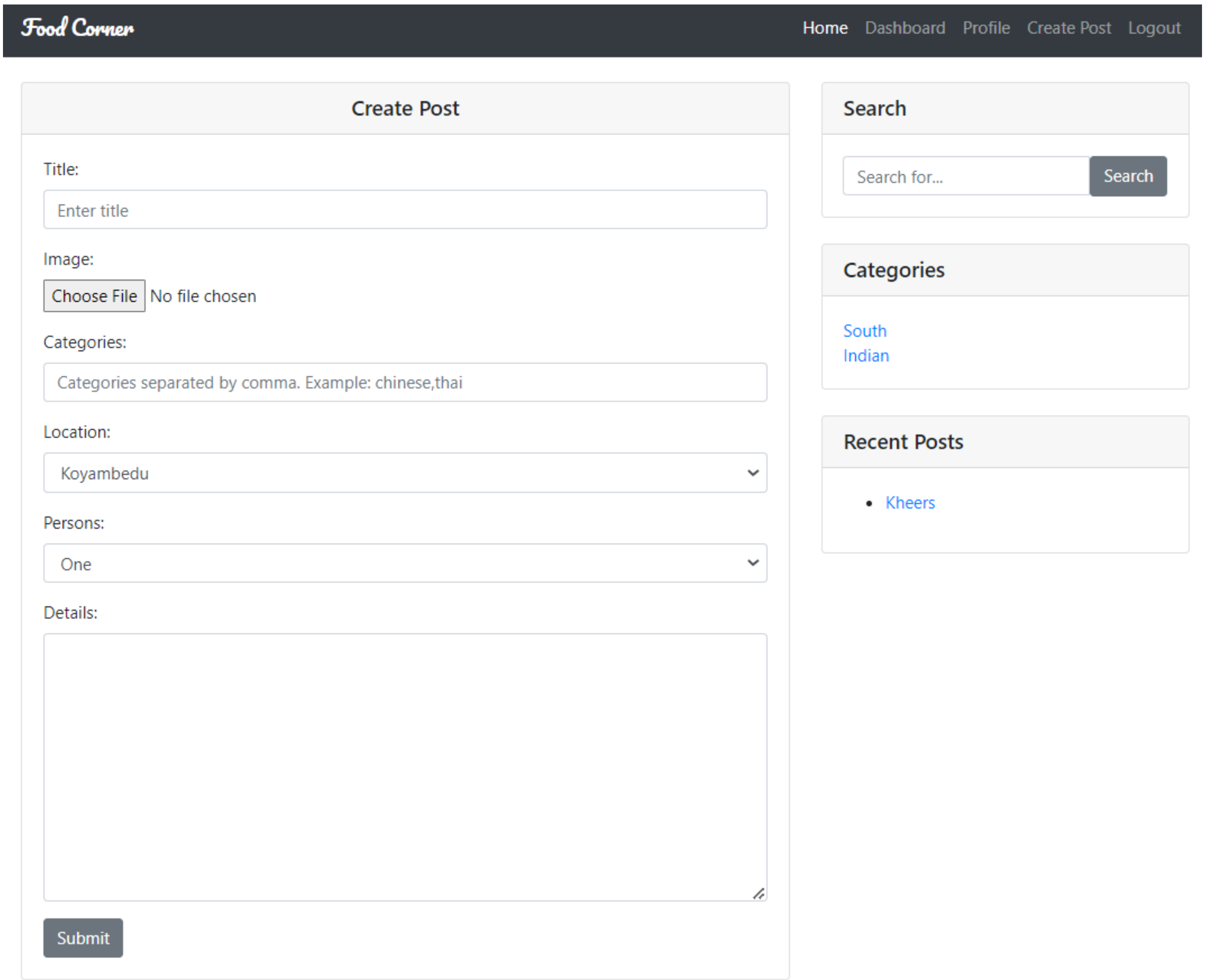
Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

Enter the same password as before, for verification.

Already have an account? Login here

Fig 4.2 Registration page



Create Post

Title:

Enter title

Image:

No file chosen

Categories:

Categories separated by comma. Example: chinese,thai

Location:

Koyambedu

Persons:

One

Details:

Search

Search for...

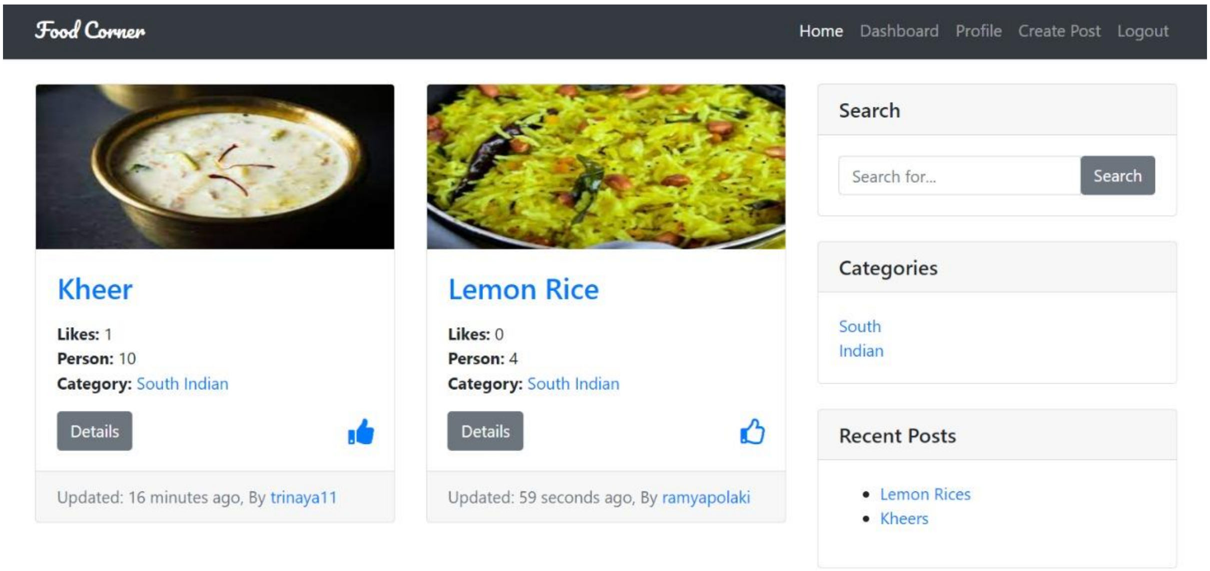
Categories

South Indian

Recent Posts

- Kheers

Fig 4.3 Creating a post



Kheer

Likes: 1
Person: 10
Category: South Indian

Updated: 16 minutes ago, By trinaya11

Lemon Rice

Likes: 0
Person: 4
Category: South Indian

Updated: 59 seconds ago, By ramyapolaki

Search

Search for...

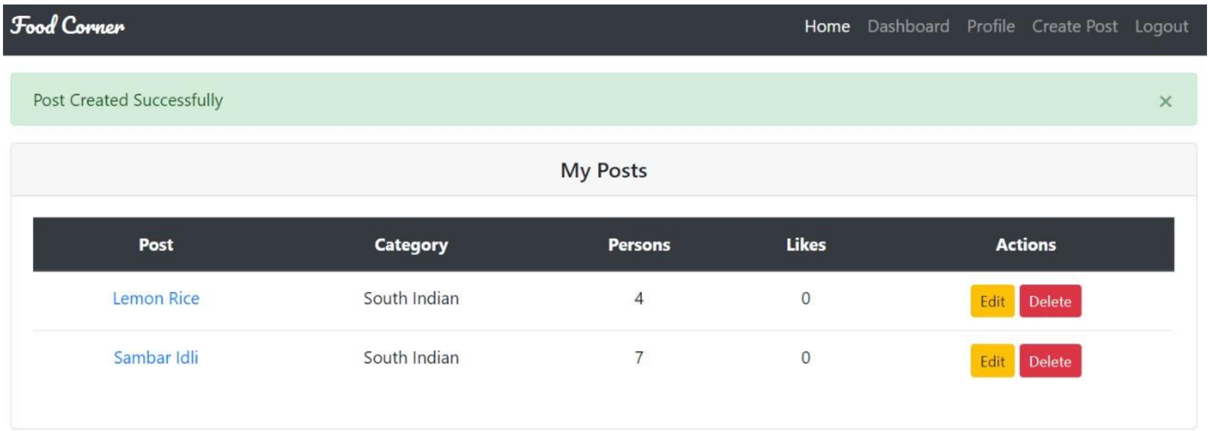
Categories

South Indian

Recent Posts

- Lemon Rices
- Kheers

Fig 4.4 Home page: views the menu list and widgets

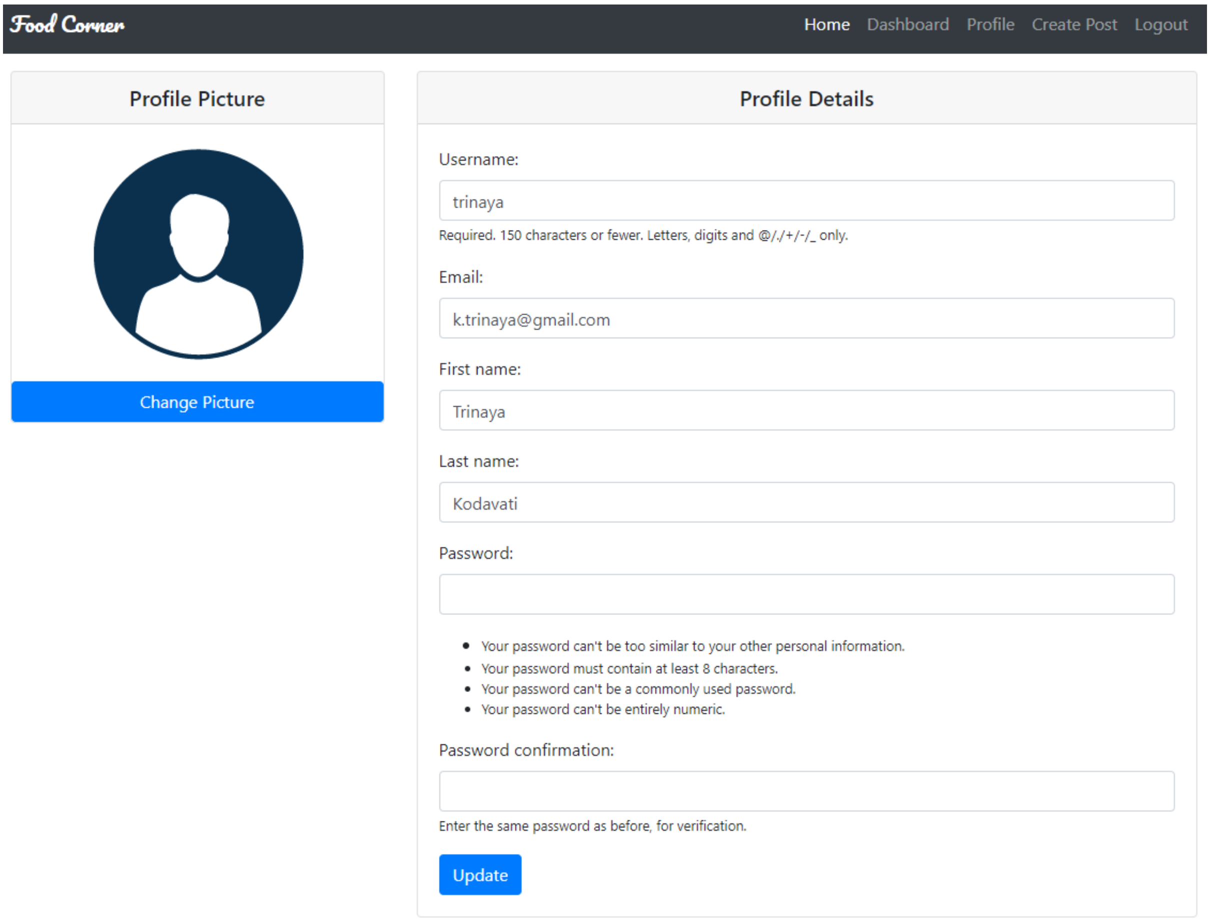


Post Created Successfully

My Posts

Post	Category	Persons	Likes	Actions
Lemon Rice	South Indian	4	0	Edit Delete
Sambar Idli	South Indian	7	0	Edit Delete

Fig 4.5 Dashboard: views posts uploaded



Profile Details

Username: trinaya

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Email: k.trinaya@gmail.com

First name: Trinaya

Last name: Kodavati

Password:

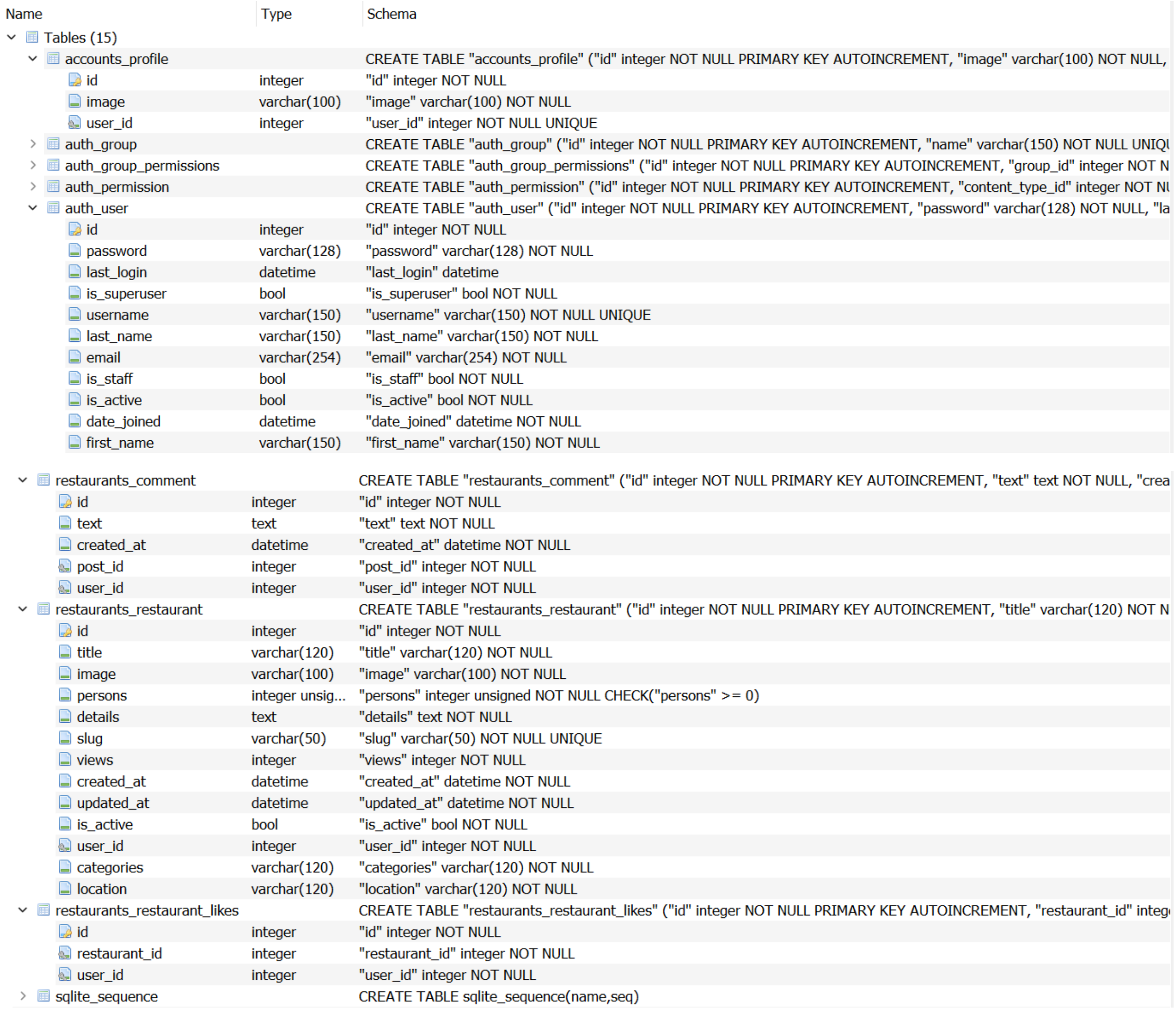
- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

Password confirmation:

Enter the same password as before, for verification.

Fig 4.6 Profile Details

4.2 SQLITE3 DATABASE



Name	Type	Schema
Tables (15)		
accounts_profile		CREATE TABLE "accounts_profile" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "image" varchar(100) NOT NULL,
id	integer	"id" integer NOT NULL
image	varchar(100)	"image" varchar(100) NOT NULL
user_id	integer	"user_id" integer NOT NULL UNIQUE
auth_group		CREATE TABLE "auth_group" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "name" varchar(150) NOT NULL UNIQU
auth_group_permissions		CREATE TABLE "auth_group_permissions" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "group_id" integer NOT N
auth_permission		CREATE TABLE "auth_permission" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "content_type_id" integer NOT NU
auth_user		CREATE TABLE "auth_user" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "password" varchar(128) NOT NULL, "la
id	integer	"id" integer NOT NULL
password	varchar(128)	"password" varchar(128) NOT NULL
last_login	datetime	"last_login" datetime
is_superuser	bool	"is_superuser" bool NOT NULL
username	varchar(150)	"username" varchar(150) NOT NULL UNIQUE
last_name	varchar(150)	"last_name" varchar(150) NOT NULL
email	varchar(254)	"email" varchar(254) NOT NULL
is_staff	bool	"is_staff" bool NOT NULL
is_active	bool	"is_active" bool NOT NULL
date_joined	datetime	"date_joined" datetime NOT NULL
first_name	varchar(150)	"first_name" varchar(150) NOT NULL
restaurants_comment		CREATE TABLE "restaurants_comment" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "text" text NOT NULL, "crea
id	integer	"id" integer NOT NULL
text	text	"text" text NOT NULL
created_at	datetime	"created_at" datetime NOT NULL
post_id	integer	"post_id" integer NOT NULL
user_id	integer	"user_id" integer NOT NULL
restaurants_restaurant		CREATE TABLE "restaurants_restaurant" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "title" varchar(120) NOT N
id	integer	"id" integer NOT NULL
title	varchar(120)	"title" varchar(120) NOT NULL
image	varchar(100)	"image" varchar(100) NOT NULL
persons	integer unsig...	"persons" integer unsigned NOT NULL CHECK("persons" >= 0)
details	text	"details" text NOT NULL
slug	varchar(50)	"slug" varchar(50) NOT NULL UNIQUE
views	integer	"views" integer NOT NULL
created_at	datetime	"created_at" datetime NOT NULL
updated_at	datetime	"updated_at" datetime NOT NULL
is_active	bool	"is_active" bool NOT NULL
user_id	integer	"user_id" integer NOT NULL
categories	varchar(120)	"categories" varchar(120) NOT NULL
location	varchar(120)	"location" varchar(120) NOT NULL
restaurants_restaurant_likes		CREATE TABLE "restaurants_restaurant_likes" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "restaurant_id" integ
id	integer	"id" integer NOT NULL
restaurant_id	integer	"restaurant_id" integer NOT NULL
user_id	integer	"user_id" integer NOT NULL
sqlite_sequence		CREATE TABLE sqlite_sequence(name,seq)

Fig 4.7 Some of the tables of the database displayed along with their respective attributes (including primary keys)

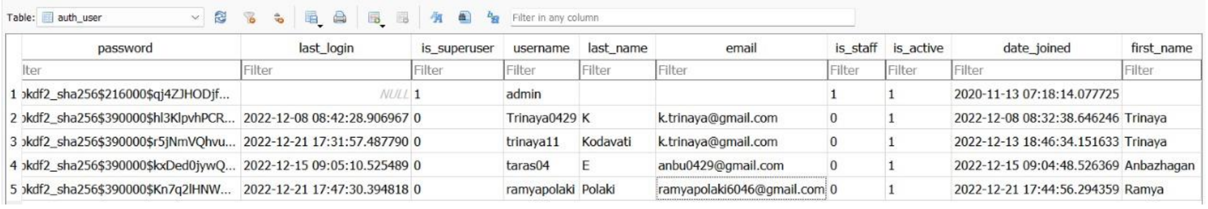


Table: auth_user

Filter in any column

	password	last_login	is_superuser	username	last_name	email	is_staff	is_active	date_joined	first_name
	lter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	›kdf2_sha256$216000$qj4ZJHODjf...	NULL	1	admin			1	1	2020-11-13 07:18:14.077725	
2	›kdf2_sha256$390000$hl3KlpvhPCR...	2022-12-08 08:42:28.906967	0	Trinaya0429	K	k.trinaya@gmail.com	0	1	2022-12-08 08:32:38.646246	Trinaya
3	›kdf2_sha256$390000$r5jNmVQhvu...	2022-12-21 17:31:57.487790	0	trinaya11	Kodavati	k.trinaya@gmail.com	0	1	2022-12-13 18:46:34.151633	Trinaya
4	›kdf2_sha256$390000$kxDed0jywQ...	2022-12-15 09:05:10.525489	0	taras04	E	anbu0429@gmail.com	0	1	2022-12-15 09:04:48.526369	Anbazhagan
5	›kdf2_sha256$390000$Kn7q2lHNW...	2022-12-21 17:47:30.394818	0	ramyapolaki	Polaki	ramyapolaki6046@gmail.com	0	1	2022-12-21 17:44:56.294359	Ramya

Fig 4.8 auth_user table

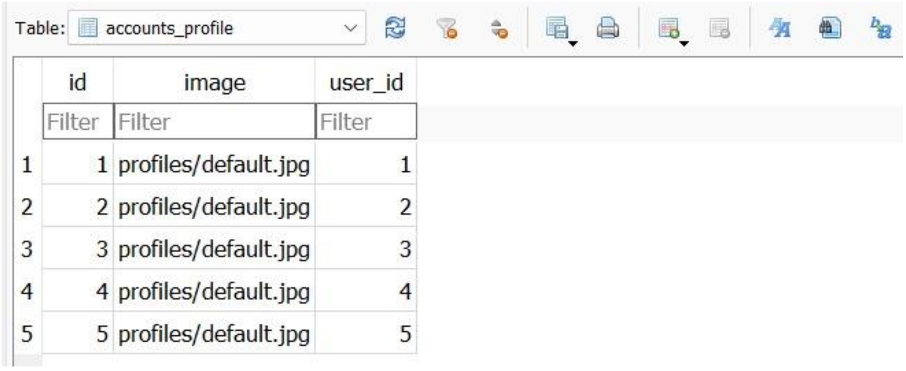


Table: accounts_profile

	id	image	user_id
1	1	profiles/default.jpg	1
2	2	profiles/default.jpg	2
3	3	profiles/default.jpg	3
4	4	profiles/default.jpg	4
5	5	profiles/default.jpg	5

Fig 4.9 accounts_profile table

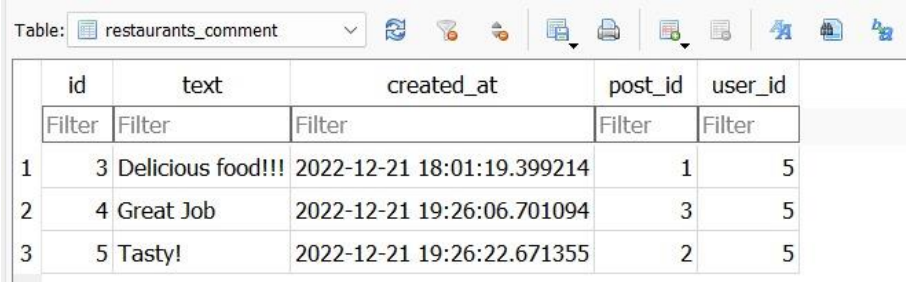


Table: restaurants_comment

	id	text	created_at	post_id	user_id
1	3	Delicious food!!!	2022-12-21 18:01:19.399214	1	5
2	4	Great Job	2022-12-21 19:26:06.701094	3	5
3	5	Tasty!	2022-12-21 19:26:22.671355	2	5

Fig 4.10 restaurants_comment table

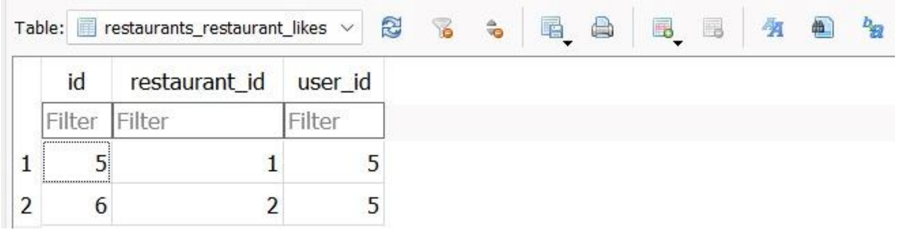


Table: restaurants_restaurant_likes

	id	restaurant_id	user_id
1	5	1	5
2	6	2	5

Fig 4.11 restaurants_restaurant_likes

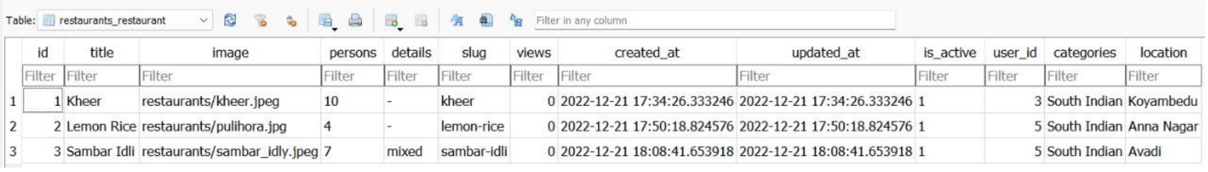


Table: restaurants_restaurant

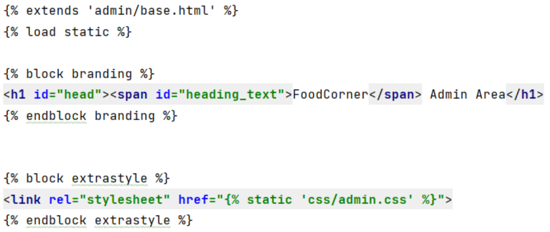
Filter in any column

	id	title	image	persons	details	slug	views	created_at	updated_at	is_active	user_id	categories	location
1	1	Kheer	restaurants/kheer.jpeg	10	-	kheer	0	2022-12-21 17:34:26.333246	2022-12-21 17:34:26.333246	1	3	South Indian	Koyambedu
2	2	Lemon Rice	restaurants/pulihora.jpg	4	-	lemon-rice	0	2022-12-21 17:50:18.824576	2022-12-21 17:50:18.824576	1	5	South Indian	Anna Nagar
3	3	Sambar Idli	restaurants/sambar_idly.jpeg	7	mixed	sambar-idli	0	2022-12-21 18:08:41.653918	2022-12-21 18:08:41.653918	1	5	South Indian	Avadi

Fig 4.12 restaurants_restaurant table

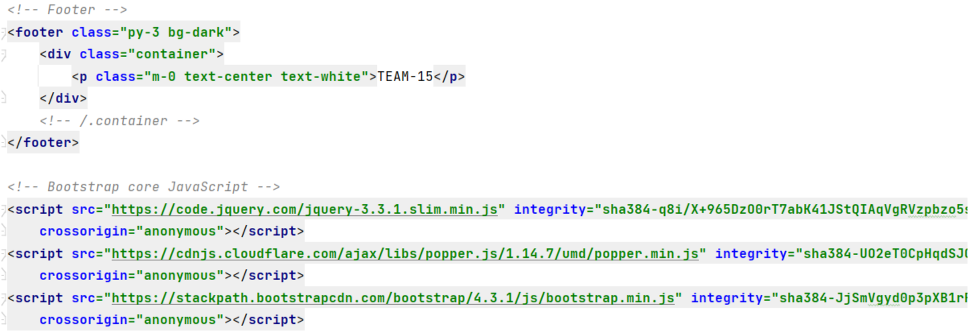
4.3 CODES

base_site.html



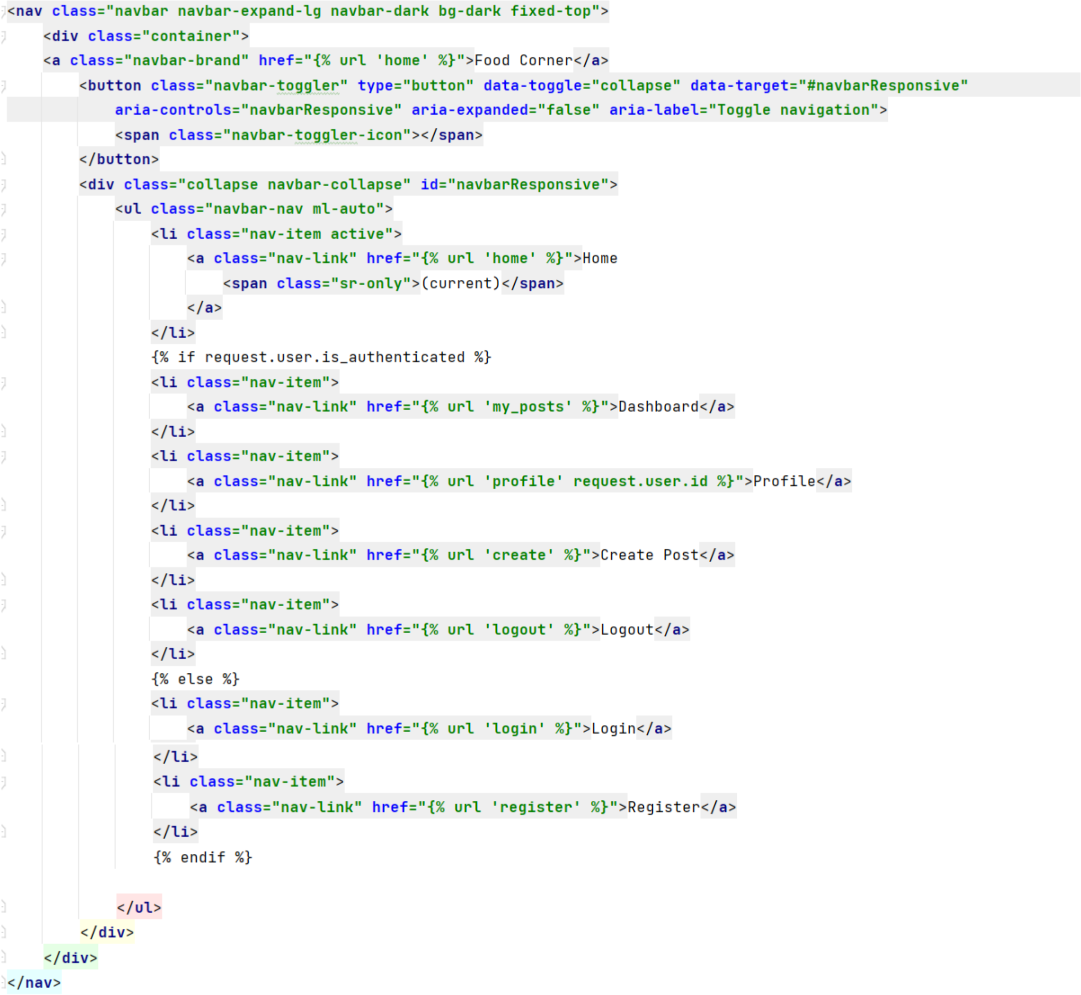
```
{% extends 'admin/base.html' %}
{% load static %}

{% block branding %}
<h1 id="head"><span id="heading_text">FoodCorner</span> Admin Area</h1>
{% endblock branding %}


{% block extrastyle %}
<link rel="stylesheet" href="{% static 'css/admin.css' %}">
{% endblock extrastyle %}
```

_footer.html



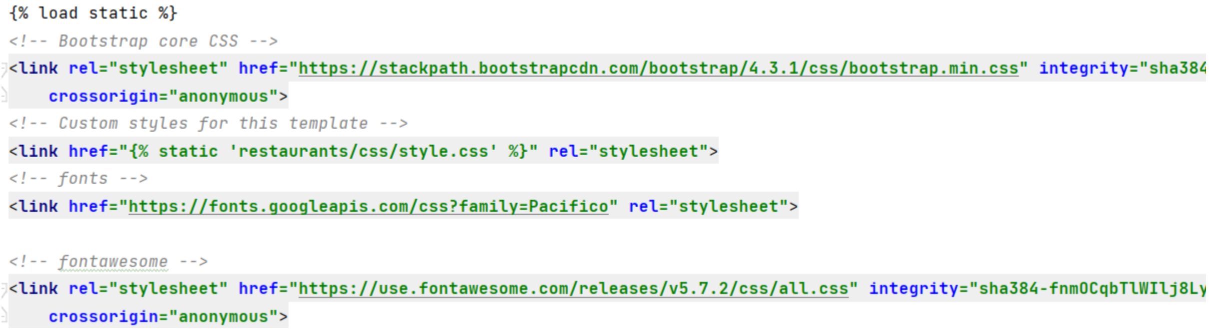
```
<!-- Footer -->
<footer class="py-3 bg-dark">
    <div class="container">
        <p class="m-0 text-center text-white">TEAM-15</p>
    </div>
    <!-- /.container -->
</footer>

<!-- Bootstrap core JavaScript -->
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5
    crossorigin="anonymous"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-UO2eT0CpHqdSJ
    crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1r
    crossorigin="anonymous"></script>
```

_navbar.html



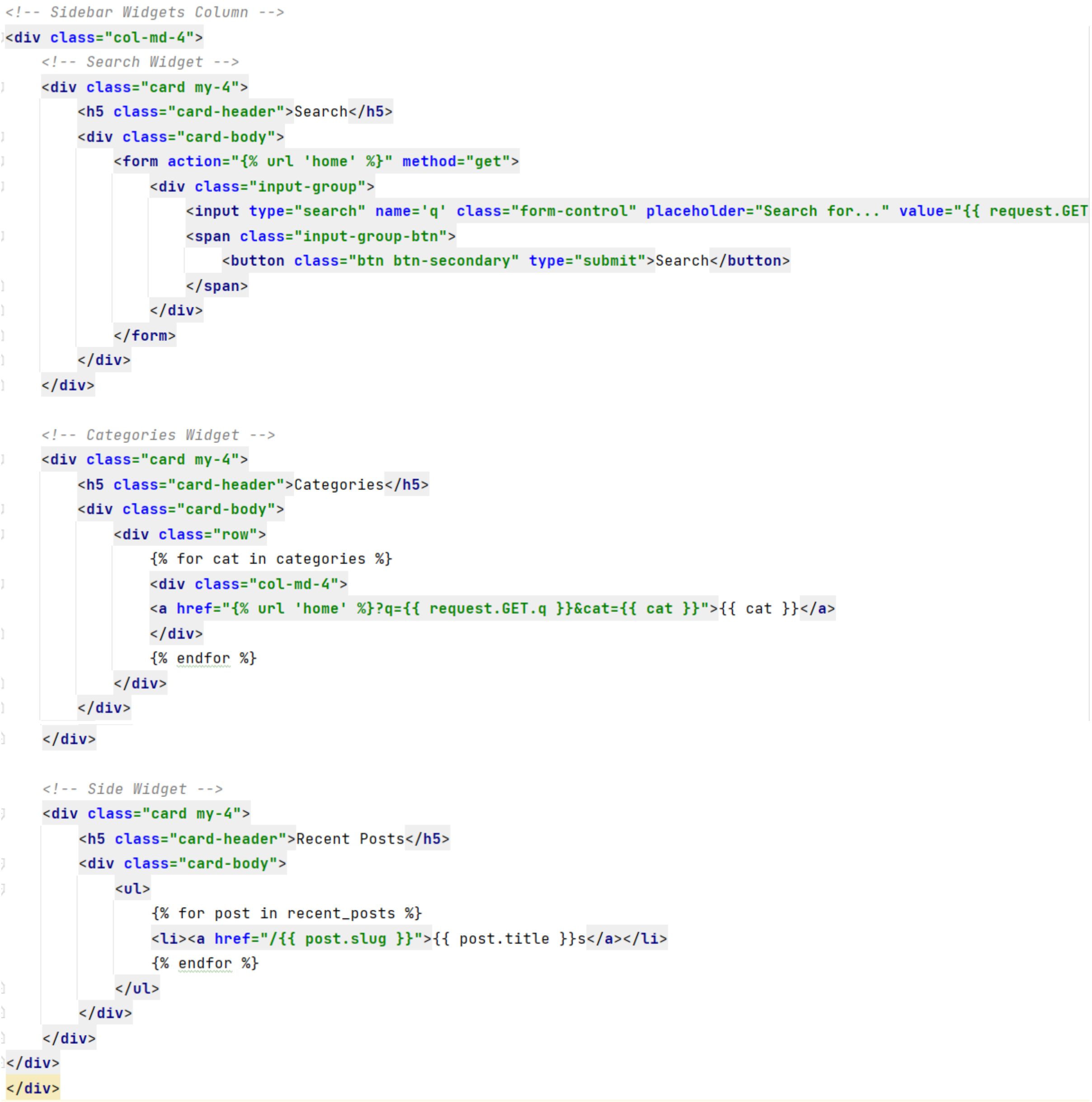
```
<nav class="navbar navbar-expand-lg navbar-dark bg-dark fixed-top">
    <div class="container">
    <a class="navbar-brand" href="{% url 'home' %}">Food Corner</a>
        <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarResponsive"
            aria-controls="navbarResponsive" aria-expanded="false" aria-label="Toggle navigation">
            <span class="navbar-toggler-icon"></span>
        </button>
        <div class="collapse navbar-collapse" id="navbarResponsive">
            <ul class="navbar-nav ml-auto">
                <li class="nav-item active">
                    <a class="nav-link" href="{% url 'home' %}">Home
                        <span class="sr-only">(current)</span>
                    </a>
                </li>
                {% if request.user.is_authenticated %}
                <li class="nav-item">
                    <a class="nav-link" href="{% url 'my_posts' %}">Dashboard</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{% url 'profile' request.user.id %}">Profile</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{% url 'create' %}">Create Post</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{% url 'logout' %}">Logout</a>
                </li>
                {% else %}
                <li class="nav-item">
                    <a class="nav-link" href="{% url 'login' %}">Login</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{% url 'register' %}">Register</a>
                </li>
                {% endif %}

            </ul>
        </div>
    </div>
</nav>
```

_styles.html



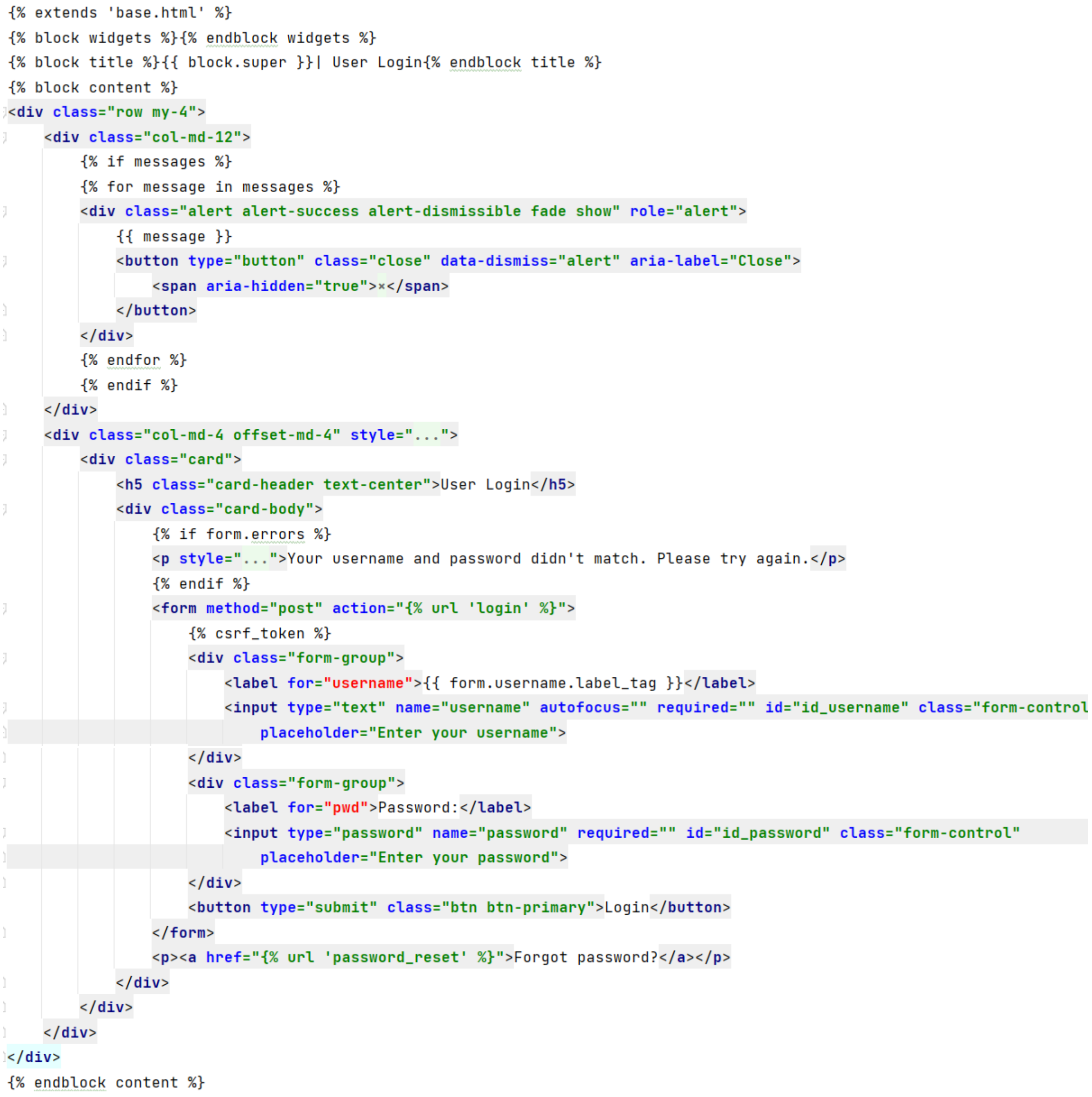
```
{% load static %}
<!-- Bootstrap core CSS -->
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384
    crossorigin="anonymous">
<!-- Custom styles for this template -->
<link href="{% static 'restaurants/css/style.css' %}" rel="stylesheet">
<!-- fonts -->
<link href="https://fonts.googleapis.com/css?family=Pacifico" rel="stylesheet">

<!-- fontawesome -->
<link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.7.2/css/all.css" integrity="sha384-fnmOCqbTlWIlj8Ly
    crossorigin="anonymous">
```

_widgets.html



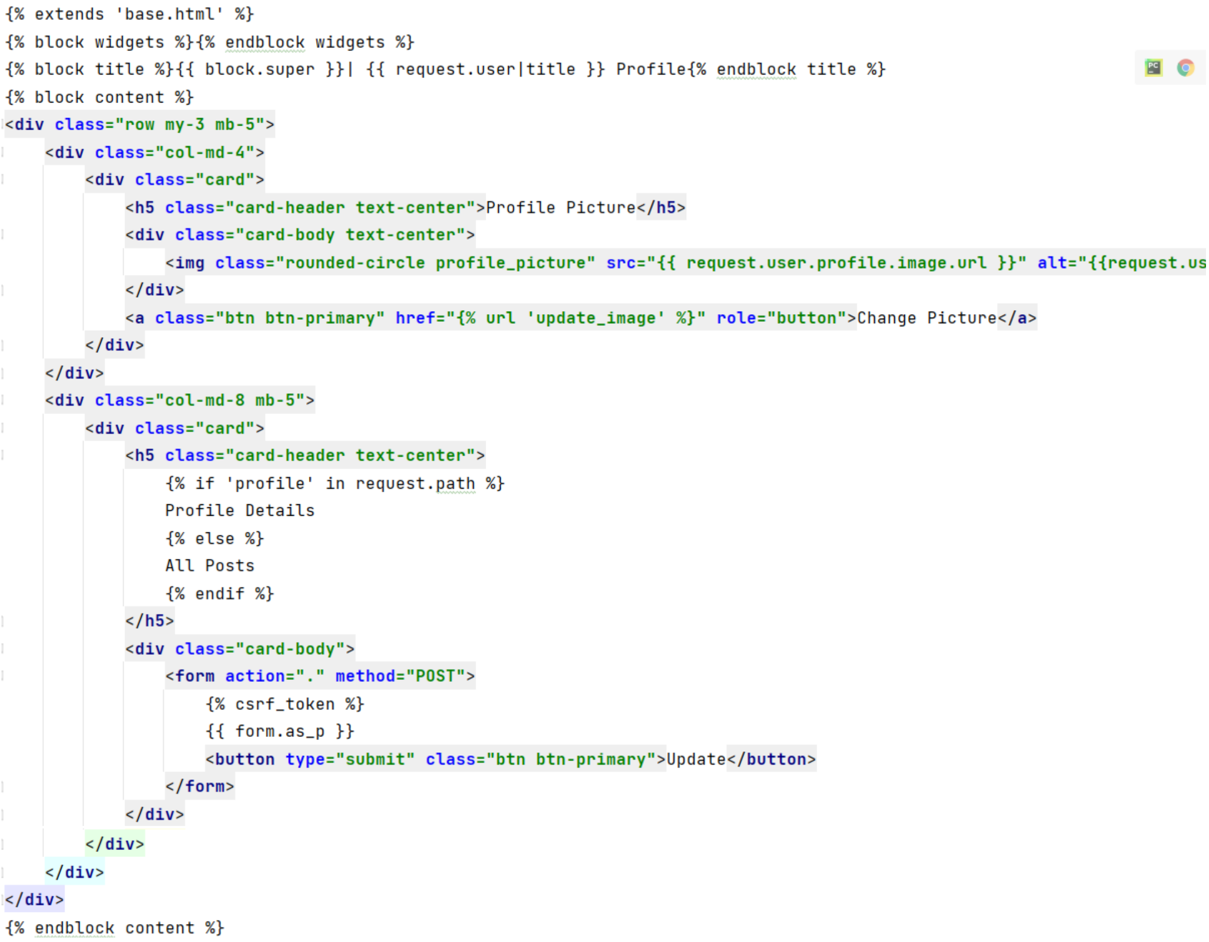
```
<!-- Sidebar Widgets Column -->
<div class="col-md-4">
    <!-- Search Widget -->
    <div class="card my-4">
        <h5 class="card-header">Search</h5>
        <div class="card-body">
            <form action="{% url 'home' %}" method="get">
                <div class="input-group">
                    <input type="search" name='q' class="form-control" placeholder="Search for..." value="{{ request.GET
                    <span class="input-group-btn">
                        <button class="btn btn-secondary" type="submit">Search</button>
                    </span>
                </div>
            </form>
        </div>
    </div>

    <!-- Categories Widget -->
    <div class="card my-4">
        <h5 class="card-header">Categories</h5>
        <div class="card-body">
            <div class="row">
                {% for cat in categories %}
                <div class="col-md-4">
                <a href="{% url 'home' %}?q={{ request.GET.q }}&cat={{ cat }}">{{ cat }}</a>
                </div>
                {% endfor %}
            </div>
        </div>
    </div>

    <!-- Side Widget -->
    <div class="card my-4">
        <h5 class="card-header">Recent Posts</h5>
        <div class="card-body">
            <ul>
                {% for post in recent_posts %}
                <li><a href="/{{ post.slug }}">{{ post.title }}s</a></li>
                {% endfor %}
            </ul>
        </div>
    </div>
</div>
</div>
```

login.html



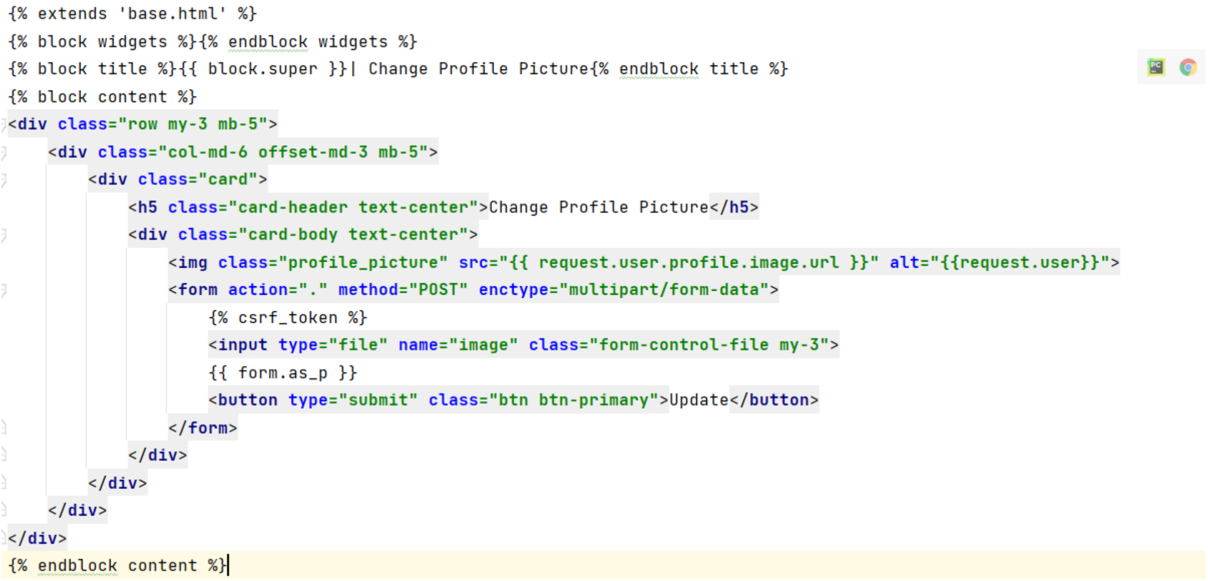
```
{% extends 'base.html' %}
{% block widgets %}{% endblock widgets %}
{% block title %}{{ block.super }}| User Login{% endblock title %}
{% block content %}
<div class="row my-4">
    <div class="col-md-12">
        {% if messages %}
        {% for message in messages %}
        <div class="alert alert-success alert-dismissible fade show" role="alert">
            {{ message }}
            <button type="button" class="close" data-dismiss="alert" aria-label="Close">
                <span aria-hidden="true">×</span>
            </button>
        </div>
        {% endfor %}
        {% endif %}
    </div>
    <div class="col-md-4 offset-md-4" style="...">
        <div class="card">
            <h5 class="card-header text-center">User Login</h5>
            <div class="card-body">
                {% if form.errors %}
                <p style="...">Your username and password didn't match. Please try again.</p>
                {% endif %}
                <form method="post" action="{% url 'login' %}">
                    {% csrf_token %}
                    <div class="form-group">
                        <label for="username">{{ form.username.label_tag }}</label>
                        <input type="text" name="username" autofocus="" required="" id="id_username" class="form-control
                            placeholder="Enter your username">
                    </div>
                    <div class="form-group">
                        <label for="pwd">Password:</label>
                        <input type="password" name="password" required="" id="id_password" class="form-control"
                            placeholder="Enter your password">
                    </div>
                    <button type="submit" class="btn btn-primary">Login</button>
                </form>
                <p><a href="{% url 'password_reset' %}">Forgot password?</a></p>
            </div>
        </div>
    </div>
</div>
{% endblock content %}
```

profile.html



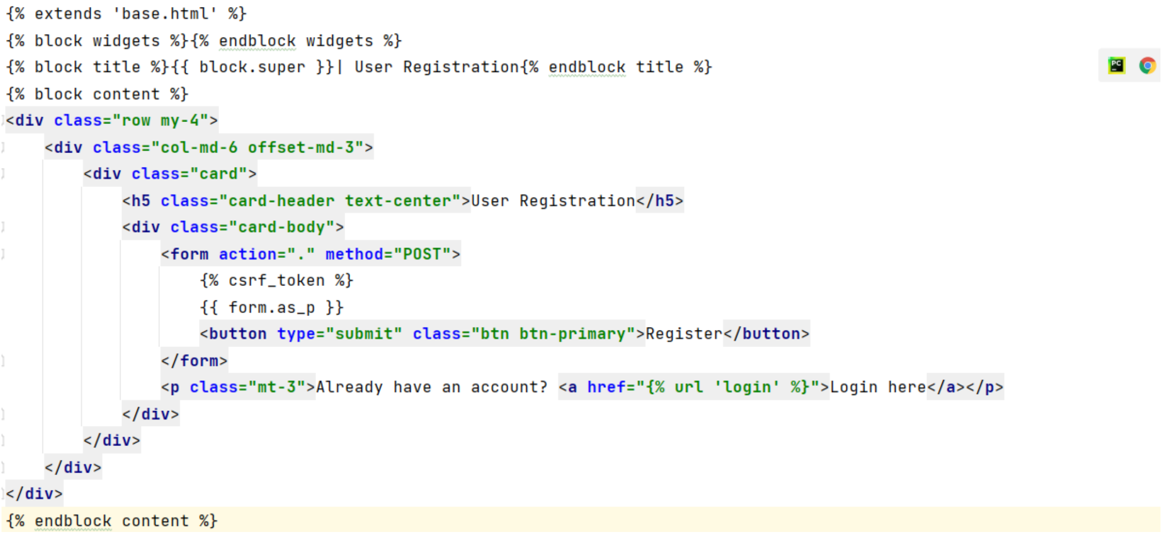
```
{% extends 'base.html' %}
{% block widgets %}{% endblock widgets %}
{% block title %}{{ block.super }}| {{ request.user|title }} Profile{% endblock title %}
{% block content %}
<div class="row my-3 mb-5">
    <div class="col-md-4">
        <div class="card">
            <h5 class="card-header text-center">Profile Picture</h5>
            <div class="card-body text-center">
                <img class="rounded-circle profile_picture" src="{{ request.user.profile.image.url }}" alt="{{request.us
            </div>
            <a class="btn btn-primary" href="{% url 'update_image' %}" role="button">Change Picture</a>
        </div>
    </div>
    <div class="col-md-8 mb-5">
        <div class="card">
            <h5 class="card-header text-center">
                {% if 'profile' in request.path %}
                Profile Details
                {% else %}
                All Posts
                {% endif %}
            </h5>
            <div class="card-body">
                <form action="." method="POST">
                    {% csrf_token %}
                    {{ form.as_p }}
                    <button type="submit" class="btn btn-primary">Update</button>
                </form>
            </div>
        </div>
    </div>
</div>
{% endblock content %}
```

profile_picture.html



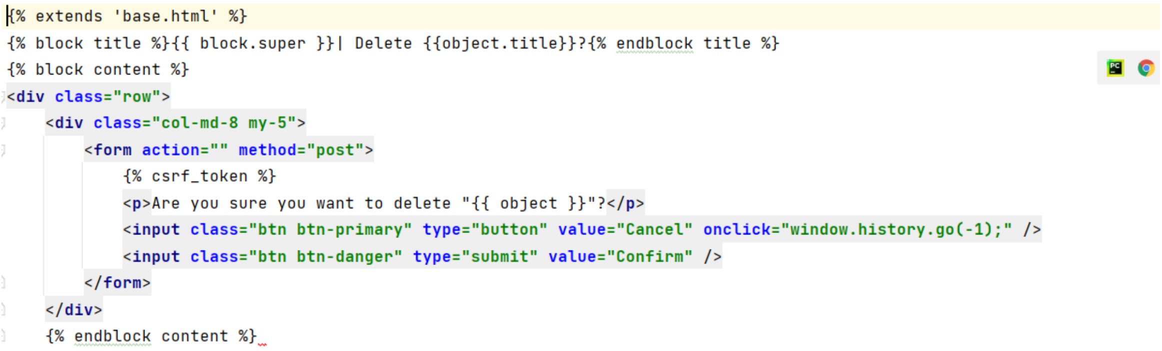
```
{% extends 'base.html' %}
{% block widgets %}{% endblock widgets %}
{% block title %}{{ block.super }}| Change Profile Picture{% endblock title %}
{% block content %}
<div class="row my-3 mb-5">
    <div class="col-md-6 offset-md-3 mb-5">
        <div class="card">
            <h5 class="card-header text-center">Change Profile Picture</h5>
            <div class="card-body text-center">
                <img class="profile_picture" src="{{ request.user.profile.image.url }}" alt="{{request.user}}">
                <form action="." method="POST" enctype="multipart/form-data">
                    {% csrf_token %}
                    <input type="file" name="image" class="form-control-file my-3">
                    {{ form.as_p }}
                    <button type="submit" class="btn btn-primary">Update</button>
                </form>
            </div>
        </div>
    </div>
</div>
{% endblock content %}
```

register.html



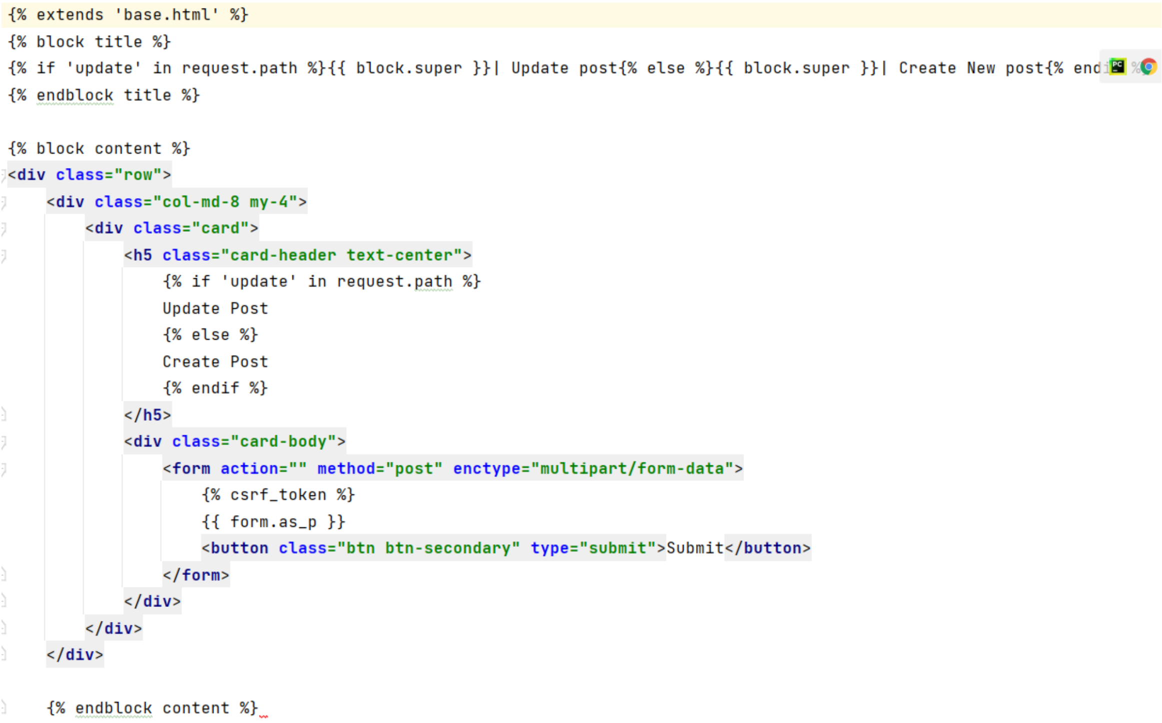
```
{% extends 'base.html' %}
{% block widgets %}{% endblock widgets %}
{% block title %}{{ block.super }}| User Registration{% endblock title %}
{% block content %}
<div class="row my-4">
    <div class="col-md-6 offset-md-3">
        <div class="card">
            <h5 class="card-header text-center">User Registration</h5>
            <div class="card-body">
                <form action="." method="POST">
                    {% csrf_token %}
                    {{ form.as_p }}
                    <button type="submit" class="btn btn-primary">Register</button>
                </form>
                <p class="mt-3">Already have an account? <a href="{% url 'login' %}">Login here</a></p>
            </div>
        </div>
    </div>
</div>
{% endblock content %}
```

restaurant_confirm_delete.html



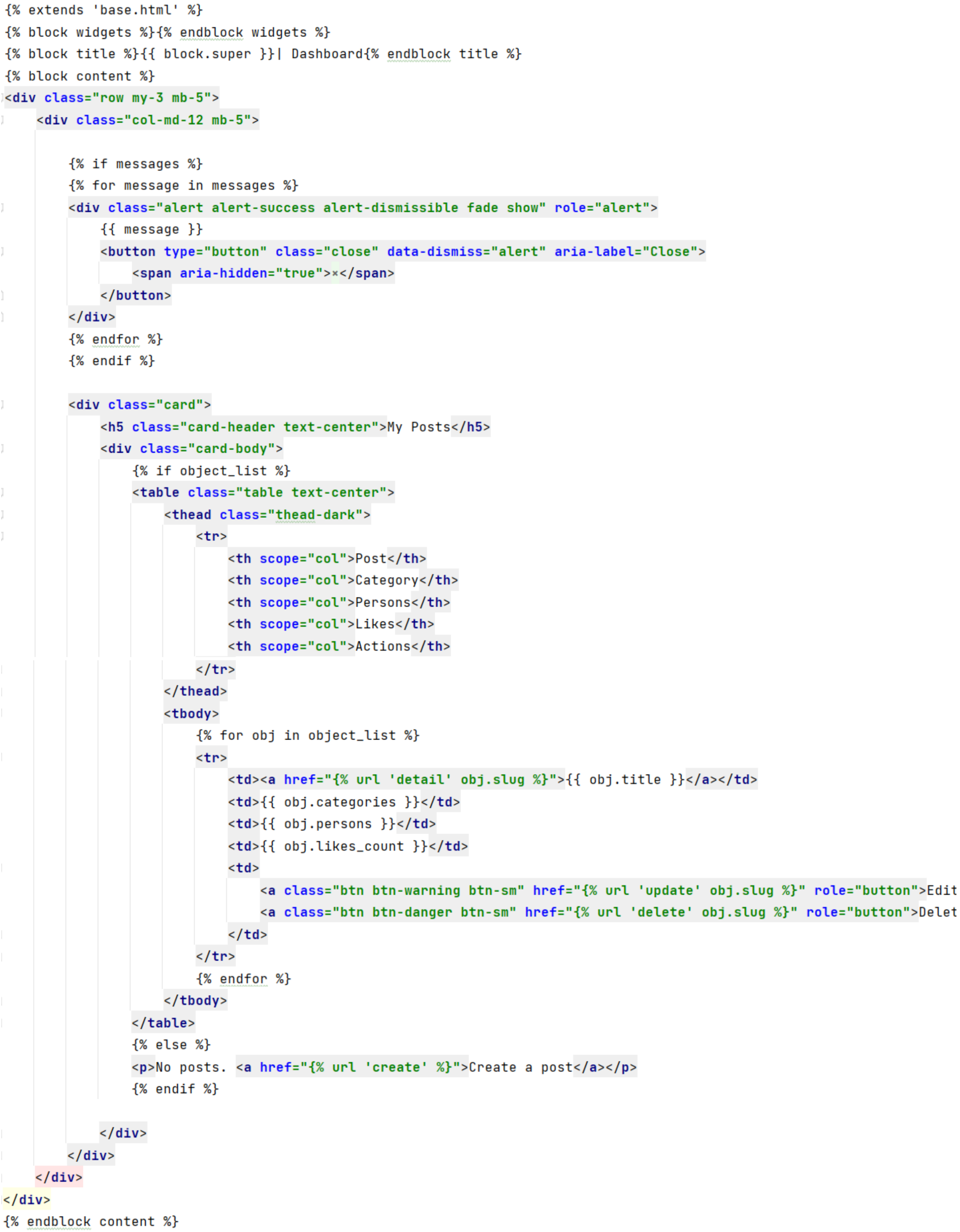
```
{% extends 'base.html' %}
{% block title %}{{ block.super }}| Delete {{object.title}}?{% endblock title %}
{% block content %}
<div class="row">
    <div class="col-md-8 my-5">
        <form action="" method="post">
            {% csrf_token %}
            <p>Are you sure you want to delete "{{ object }}"?</p>
            <input class="btn btn-primary" type="button" value="Cancel" onclick="window.history.go(-1);" />
            <input class="btn btn-danger" type="submit" value="Confirm" />
        </form>
    </div>
    {% endblock content %}
```

restaurant_form.html



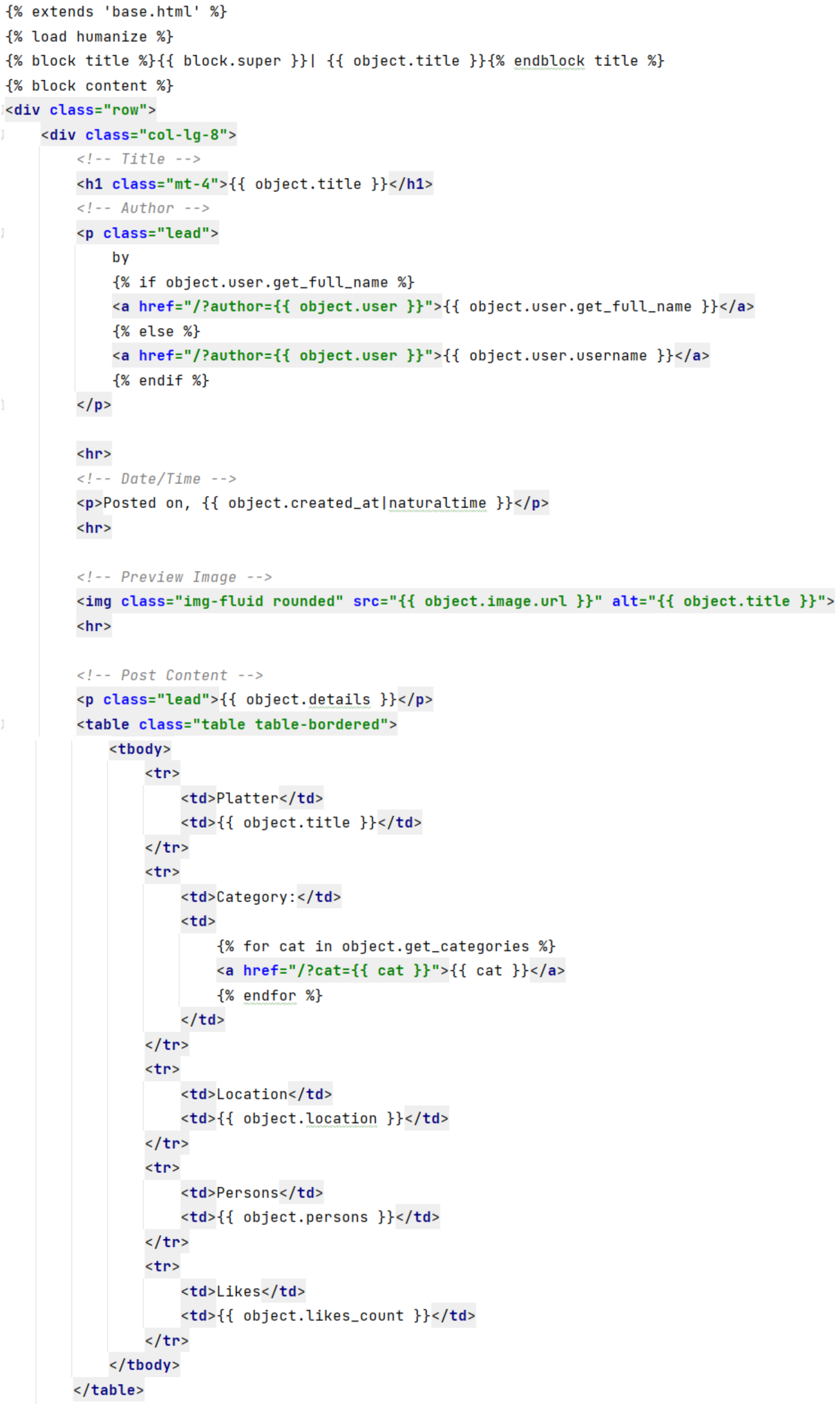
```
{% extends 'base.html' %}
{% block title %}
{% if 'update' in request.path %}{{ block.super }}| Update post{% else %}{{ block.super }}| Create New post{% end
{% endblock title %}

{% block content %}
<div class="row">
    <div class="col-md-8 my-4">
        <div class="card">
            <h5 class="card-header text-center">
                {% if 'update' in request.path %}
                Update Post
                {% else %}
                Create Post
                {% endif %}
            </h5>
            <div class="card-body">
                <form action="" method="post" enctype="multipart/form-data">
                    {% csrf_token %}
                    {{ form.as_p }}
                    <button class="btn btn-secondary" type="submit">Submit</button>
                </form>
            </div>
        </div>
    </div>

    {% endblock content %}
```

my_posts.html

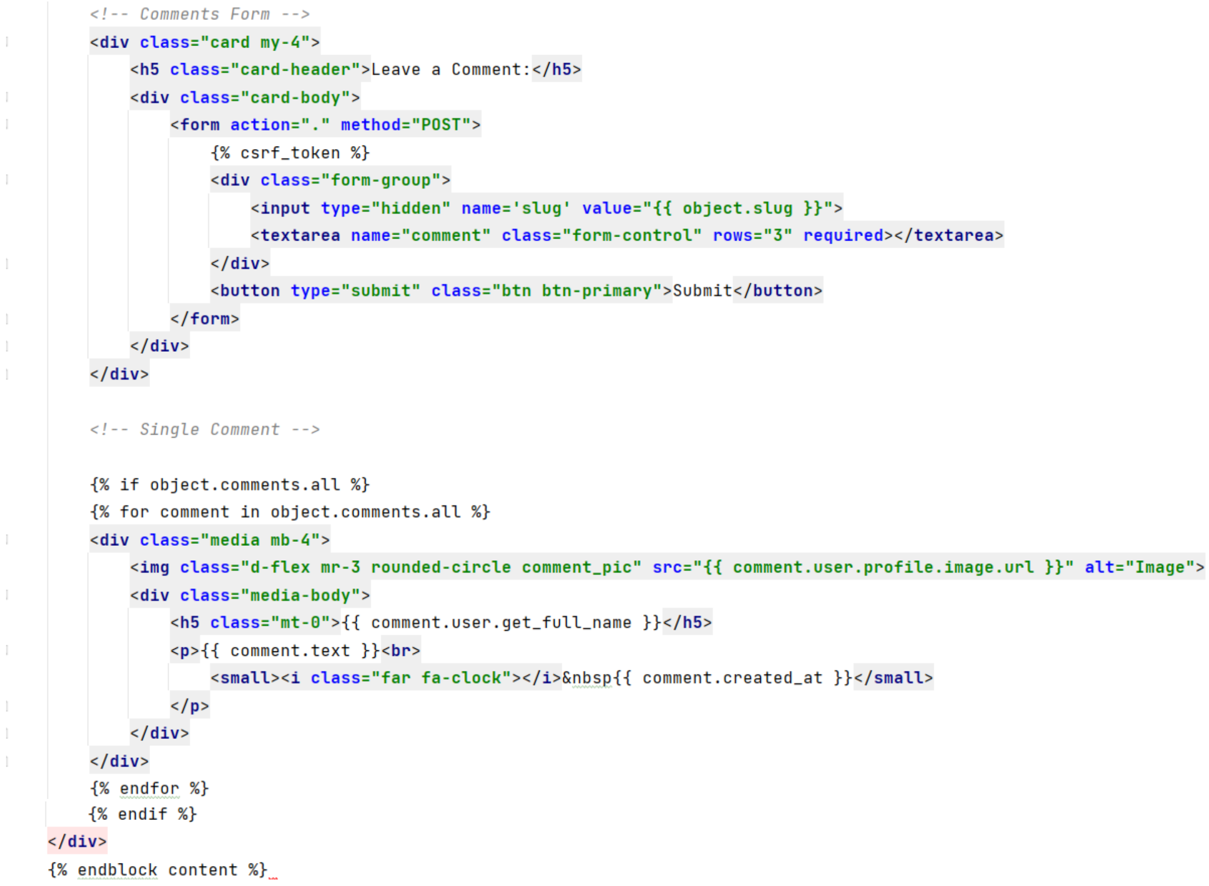


```
{% extends 'base.html' %}
{% block widgets %}{% endblock widgets %}
{% block title %}{{ block.super }}| Dashboard{% endblock title %}
{% block content %}
<div class="row my-3 mb-5">
    <div class="col-md-12 mb-5">

        {% if messages %}
        {% for message in messages %}
        <div class="alert alert-success alert-dismissible fade show" role="alert">
            {{ message }}
            <button type="button" class="close" data-dismiss="alert" aria-label="Close">
                <span aria-hidden="true">×</span>
            </button>
        </div>
        {% endfor %}
        {% endif %}

        <div class="card">
            <h5 class="card-header text-center">My Posts</h5>
            <div class="card-body">
                {% if object_list %}
                <table class="table text-center">
                    <thead class="thead-dark">
                        <tr>
                            <th scope="col">Post</th>
                            <th scope="col">Category</th>
                            <th scope="col">Persons</th>
                            <th scope="col">Likes</th>
                            <th scope="col">Actions</th>
                        </tr>
                    </thead>
                    <tbody>
                        {% for obj in object_list %}
                        <tr>
                            <td><a href="{% url 'detail' obj.slug %}">{{ obj.title }}</a></td>
                            <td>{{ obj.categories }}</td>
                            <td>{{ obj.persons }}</td>
                            <td>{{ obj.likes_count }}</td>
                            <td>
                                <a class="btn btn-warning btn-sm" href="{% url 'update' obj.slug %}" role="button">Edit
                                <a class="btn btn-danger btn-sm" href="{% url 'delete' obj.slug %}" role="button">Delet
                            </td>
                        </tr>
                        {% endfor %}
                    </tbody>
                </table>
                {% else %}
                <p>No posts. <a href="{% url 'create' %}">Create a post</a></p>
                {% endif %}

            </div>
        </div>
    </div>
</div>
{% endblock content %}
```

restaurant_detail.html

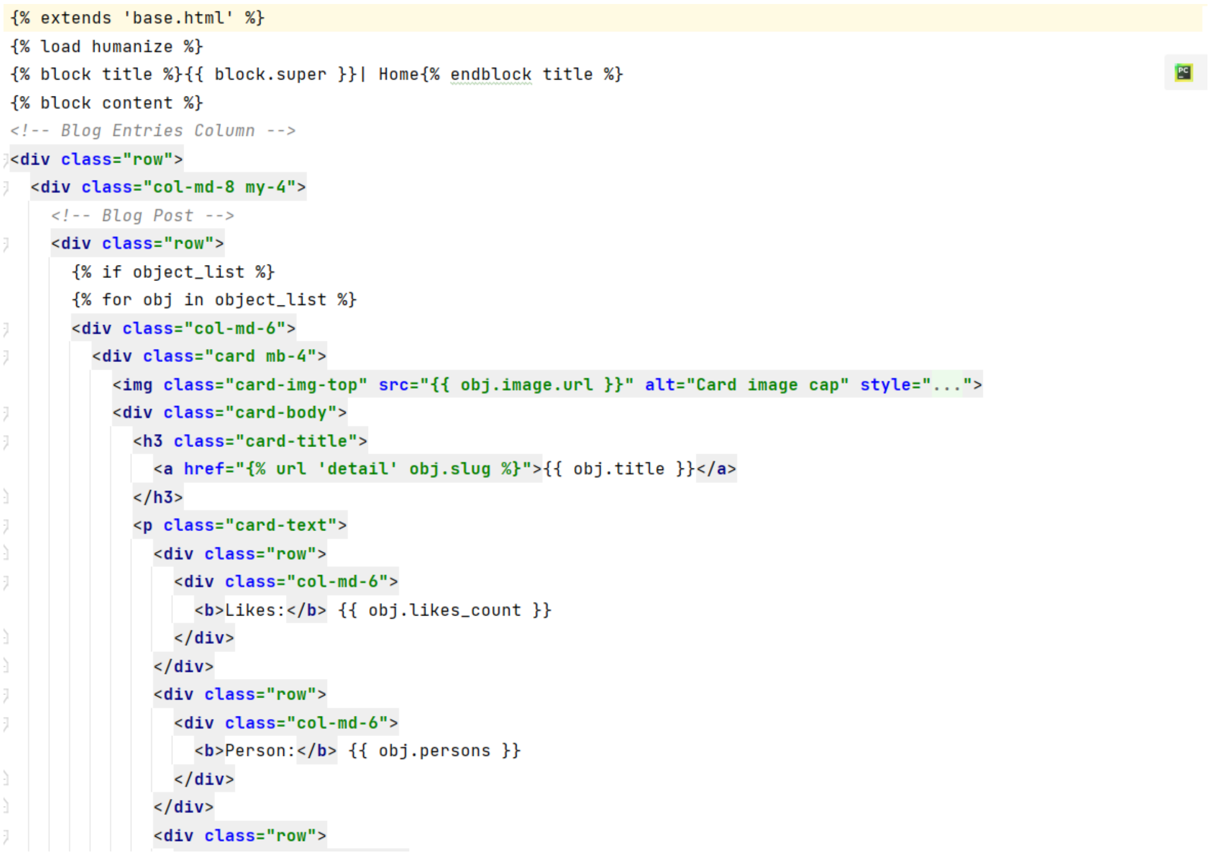


```
{% extends 'base.html' %}
{% load humanize %}
{% block title %}{{ block.super }}| {{ object.title }}{% endblock title %}
{% block content %}
<div class="row">
    <div class="col-lg-8">
        <!-- Title -->
        <h1 class="mt-4">{{ object.title }}</h1>
        <!-- Author -->
        <p class="lead">
            by
            {% if object.user.get_full_name %}
            <a href="/?author={{ object.user }}">{{ object.user.get_full_name }}</a>
            {% else %}
            <a href="/?author={{ object.user }}">{{ object.user.username }}</a>
            {% endif %}
        </p>

        <hr>
        <!-- Date/Time -->
        <p>Posted on, {{ object.created_at|naturaltime }}</p>
        <hr>

        <!-- Preview Image -->
        <img class="img-fluid rounded" src="{{ object.image.url }}" alt="{{ object.title }}">
        <hr>

        <!-- Post Content -->
        <p class="lead">{{ object.details }}</p>
        <table class="table table-bordered">
            <tbody>
                <tr>
                    <td>Platter</td>
                    <td>{{ object.title }}</td>
                </tr>
                <tr>
                    <td>Category:</td>
                    <td>
                        {% for cat in object.get_categories %}
                        <a href="/?cat={{ cat }}">{{ cat }}</a>
                        {% endfor %}
                    </td>
                </tr>
                <tr>
                    <td>Location</td>
                    <td>{{ object.location }}</td>
                </tr>
                <tr>
                    <td>Persons</td>
                    <td>{{ object.persons }}</td>
                </tr>
                <tr>
                    <td>Likes</td>
                    <td>{{ object.likes_count }}</td>
                </tr>
            </tbody>
        </table>
```

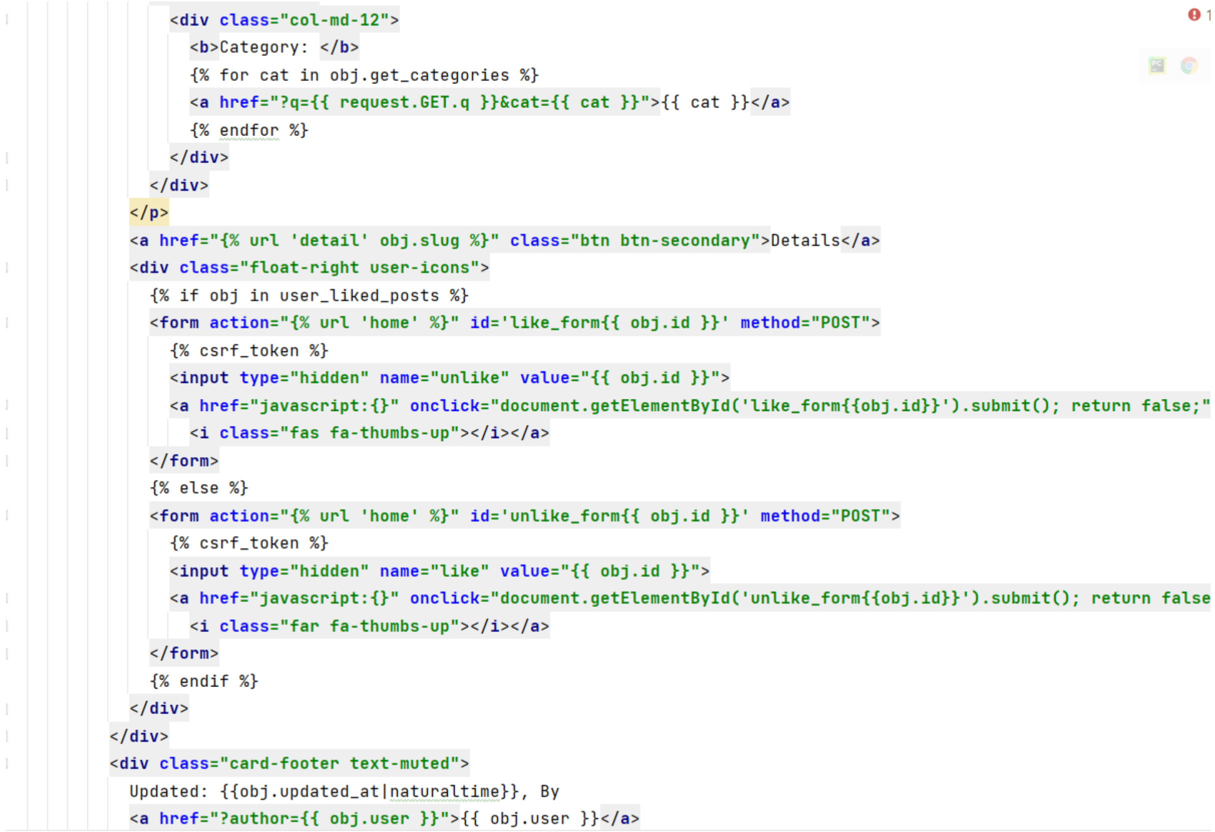


```
    <!-- Comments Form -->
    <div class="card my-4">
        <h5 class="card-header">Leave a Comment:</h5>
        <div class="card-body">
            <form action="." method="POST">
                {% csrf_token %}
                <div class="form-group">
                    <input type="hidden" name='slug' value="{{ object.slug }}">
                    <textarea name="comment" class="form-control" rows="3" required></textarea>
                </div>
                <button type="submit" class="btn btn-primary">Submit</button>
            </form>
        </div>
    </div>

    <!-- Single Comment -->

    {% if object.comments.all %}
    {% for comment in object.comments.all %}
    <div class="media mb-4">
        <img class="d-flex mr-3 rounded-circle comment_pic" src="{{ comment.user.profile.image.url }}" alt="Image">
        <div class="media-body">
            <h5 class="mt-0">{{ comment.user.get_full_name }}</h5>
            <p>{{ comment.text }}<br>
                <small><i class="far fa-clock"></i>&nbsp;{{ comment.created_at }}</small>
            </p>
        </div>
    </div>
    {% endfor %}
    {% endif %}
</div>
{% endblock content %}
```

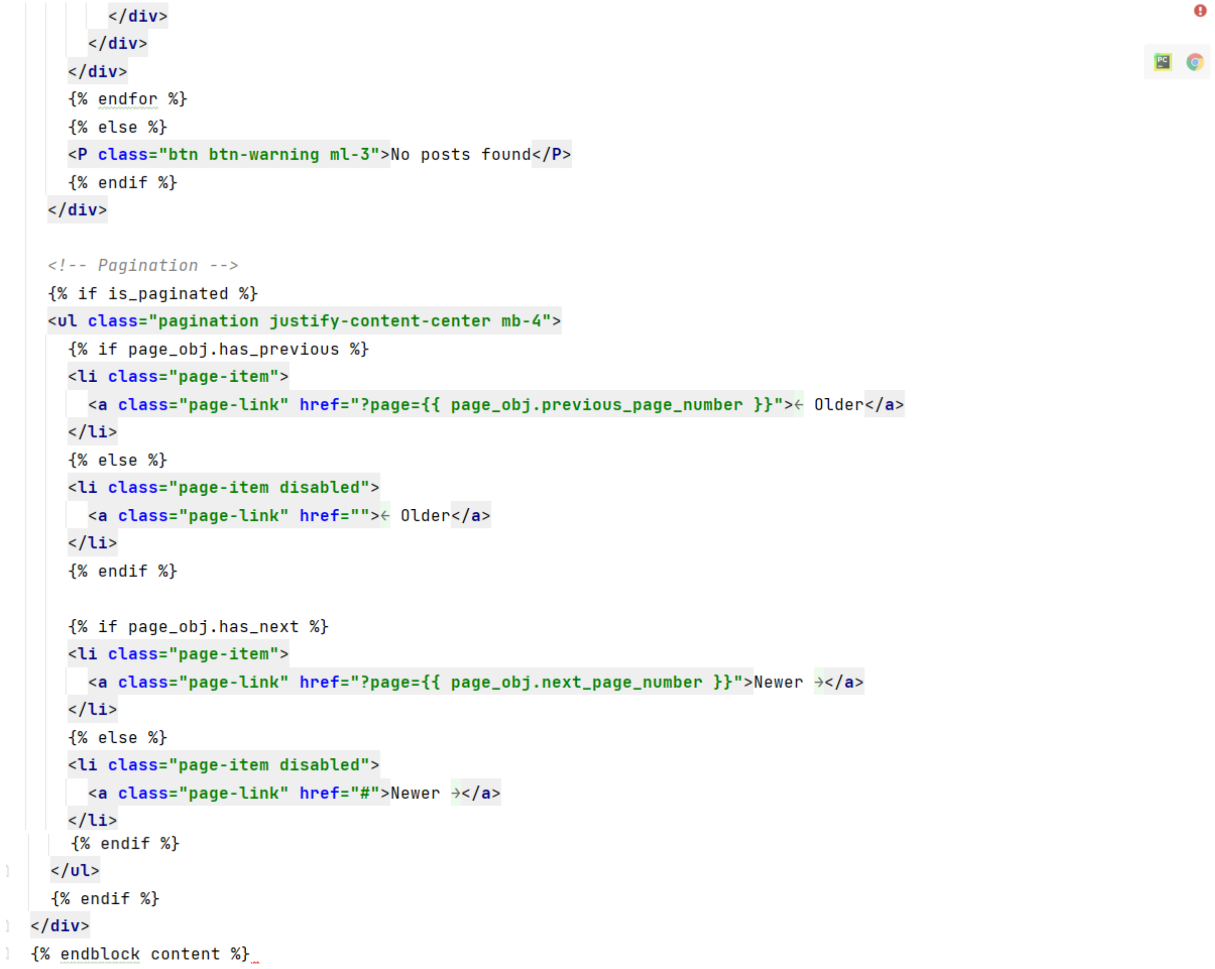
restaurant_list.html



```
{% extends 'base.html' %}
{% load humanize %}
{% block title %}{{ block.super }}| Home{% endblock title %}
{% block content %}
<!-- Blog Entries Column -->
<div class="row">
  <div class="col-md-8 my-4">
    <!-- Blog Post -->
    <div class="row">
      {% if object_list %}
      {% for obj in object_list %}
      <div class="col-md-6">
        <div class="card mb-4">
          <img class="card-img-top" src="{{ obj.image.url }}" alt="Card image cap" style="...">
          <div class="card-body">
            <h3 class="card-title">
              <a href="{% url 'detail' obj.slug %}">{{ obj.title }}</a>
            </h3>
            <p class="card-text">
              <div class="row">
                <div class="col-md-6">
                  <b>Likes:</b> {{ obj.likes_count }}
                </div>
              </div>
              <div class="row">
                <div class="col-md-6">
                  <b>Person:</b> {{ obj.persons }}
                </div>
              </div>
              <div class="row">
```

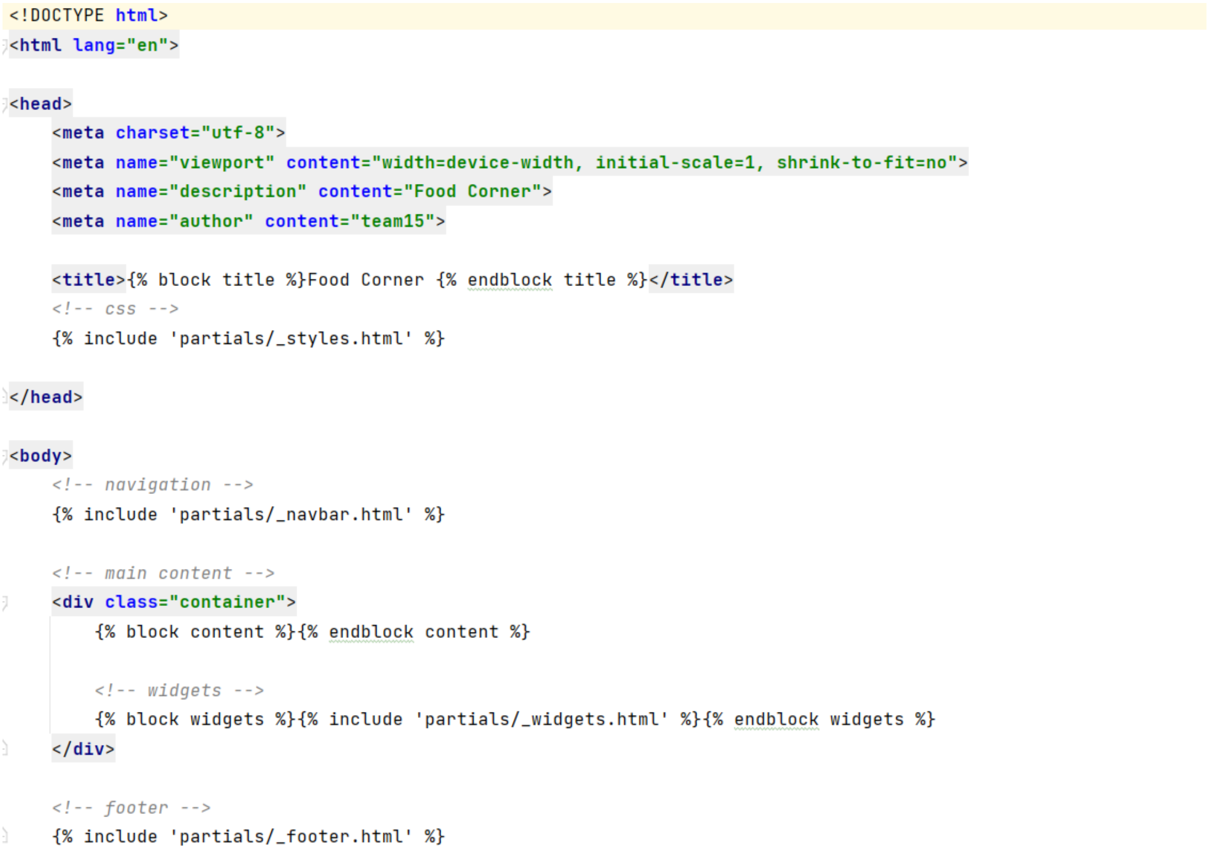


```
      <div class="col-md-12">
        <b>Category: </b>
        {% for cat in obj.get_categories %}
        <a href="?q={{ request.GET.q }}&cat={{ cat }}">{{ cat }}</a>
        {% endfor %}
      </div>
    </div>
  </p>
  <a href="{% url 'detail' obj.slug %}" class="btn btn-secondary">Details</a>
  <div class="float-right user-icons">
    {% if obj in user_liked_posts %}
    <form action="{% url 'home' %}" id='like_form{{ obj.id }}' method="POST">
      {% csrf_token %}
      <input type="hidden" name="unlike" value="{{ obj.id }}">
      <a href="javascript:{}" onclick="document.getElementById('like_form{{obj.id}}').submit(); return false;"
        <i class="fas fa-thumbs-up"></i></a>
    </form>
    {% else %}
    <form action="{% url 'home' %}" id='unlike_form{{ obj.id }}' method="POST">
      {% csrf_token %}
      <input type="hidden" name="like" value="{{ obj.id }}">
      <a href="javascript:{}" onclick="document.getElementById('unlike_form{{obj.id}}').submit(); return false
        <i class="far fa-thumbs-up"></i></a>
    </form>
    {% endif %}
  </div>
</div>
<div class="card-footer text-muted">
  Updated: {{obj.updated_at|naturaltime}}, By
  <a href="?author={{ obj.user }}">{{ obj.user }}</a>
```



```
        </div>
      </div>
    </div>
    {% endfor %}
    {% else %}
    <P class="btn btn-warning ml-3">No posts found</P>
    {% endif %}
  </div>

  <!-- Pagination -->
  {% if is_paginated %}
  <ul class="pagination justify-content-center mb-4">
    {% if page_obj.has_previous %}
    <li class="page-item">
      <a class="page-link" href="?page={{ page_obj.previous_page_number }}">← Older</a>
    </li>
    {% else %}
    <li class="page-item disabled">
      <a class="page-link" href="">← Older</a>
    </li>
    {% endif %}

    {% if page_obj.has_next %}
    <li class="page-item">
      <a class="page-link" href="?page={{ page_obj.next_page_number }}">Newer →</a>
    </li>
    {% else %}
    <li class="page-item disabled">
      <a class="page-link" href="#">Newer →</a>
    </li>
    {% endif %}
  </ul>
  {% endif %}
</div>
{% endblock content %}
```

base.html



```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="description" content="Food Corner">
    <meta name="author" content="team15">

    <title>{% block title %}Food Corner {% endblock title %}</title>
    <!-- css -->
    {% include 'partials/_styles.html' %}

</head>

<body>
    <!-- navigation -->
    {% include 'partials/_navbar.html' %}

    <!-- main content -->
    <div class="container">
        {% block content %}{% endblock content %}

        <!-- widgets -->
        {% block widgets %}{% include 'partials/_widgets.html' %}{% endblock widgets %}
    </div>

    <!-- footer -->
    {% include 'partials/_footer.html' %}
```

CHAPTER 5

CONCLUSION AND FUTURE ENHANCEMENTS

This project model was successful in creating a web interface which is to deployed in number of booths around the city and shall help in reducing hunger among the poor as well as food wastage. This project shall also help in communication and act as mediating interface between donors and donees. The future scope of this project could be for checking the quality of the food before accepting and distributing food from donors. A volunteering choice option could also be implemented for people to register as employees who can help in deliveries. A differentiation between perishable and non-perishable items can be made and the number of hours or days the food can sustain.

CHAPTER 6

REFERENCES

[1] M. S. Elavarasan, Mr. C. Daniel Nesakumar "Food Wastage Reduction Mobile Application" International Journal of Computer Science and Mobile Computing, ISSN 2320–088X, Vol. 8, pg.103 – Issue. 10, October 2019.

[2] Harshada Mhaske, Siddhi Kengar, Arti Singh "Waste Food Management And Donation App" International Journal of Advance Research and Innovative Ideas in Education, ISSN(Online) 3395-4396, Vol. 8, pg.3491- Issue. 3, 2022.

[3] Vt, Sankar. (2020). Review in Food Wastage Reduction Through Donation Application. 8. 2321-5526. 10.17148/IJIREEICE.2020.8611.

[4] Juhi Patil, Gayatri More, Pooja Mahale, Nikita Harale and Vijaylaxmi Bittal, "Zero Hunger: Smart Food Donation System using IoT", International Journal of Advanced Research in Science, Communication, and Technology, ISSN (Online) 2581-9429, Vol. 5, pg.292- Issue 1, May 2021.