INDEX

ACKNOWLEGDEMENT

We are proud to present our research report of our final project for our B.Sc. Statistics course. We would like to express our sincere gratitude to everyone that contributed to the successful execution of this our project. This endeavour would not have been possible without the support, guidance, and encouragement of several individuals.

Firstly, we would like to express our sincere gratitude to Dr. George Lekha, the Principal of Mount Carmel College for giving us this opportunity to work on this project.

We would also like to acknowledge and thank the assistance of Mrs. Renju K, Head of the Department of Computer Science and our project guide Ms. Uma K, who generously shared her knowledge and provided constructive feedback throughout the project.

Finally, we are deeply grateful to our family and friends for their support and understanding during the demanding phases of this project. This project would not be possible without them.

INTRODUCTION

In the past, the challenges faced by students regarding limited break time and continuous classes have been a persistent issue in our college community. Many students struggled to find quick dining options that could satisfy their hunger without sacrificing precious study or relaxation time. Traditional methods of finding food on campus often led to long lines, delays, and ultimately, frustration. Recognizing these difficulties, Campus Cravings embarked on a mission to address this problem by proposing the development of an innovative website.

Presently, as technology continues to advance, the need for efficient and user-centric solutions has become increasingly apparent. The proposed website by Campus Cravings aims to bridge the gap between technology and student needs by offering a seamless online platform that prioritizes quick dining experiences. By leveraging the power of the internet and mobile devices, students can now easily access a variety of dining options available on campus, allowing them to make informed decisions and minimize time spent waiting in lines. With a focus on user-centric design, the website promises to enhance the overall dining experience for students, making mealtime more convenient and enjoyable.

Looking towards the future, Campus Cravings envisions further enhancements to the website to better serve the evolving needs of the college community. These enhancements may include features such as personalized recommendations based on dietary preferences, real-time updates on menu availability and wait times, and integration with mobile payment systems for added convenience. By continually innovating and adapting to changing circumstances, Campus Cravings seeks to solidify its position as a valuable resource for students seeking quick and satisfying dining experiences on campus. Through the ongoing development and refinement of the website, Campus Cravings remains committed to promoting student well-being and academic success in our college community.

The Campus Cravings website not only addresses the immediate challenge of long waiting times but also aligns with broader trends in digital transformation and consumer expectations. As students increasingly rely on digital platforms for various aspects of their lives, from communication to entertainment and now dining, the website represents a forward-looking approach to meeting their evolving needs. Moreover, by promoting quick and convenient dining options, Campus Cravings contributes to a more balanced and productive campus environment, where students can efficiently manage their time and prioritize their academic and personal pursuits.

PROBLEM OVERVIEW

1. Long Waiting Times: Students often face long queues during peak hours, leading to frustration and delays in getting their meals. During peak hours, such as lunchtime or breaks between classes, the number of students visiting the canteen tends to surge, resulting in overcrowded lines and delays in service. Students experience frustration and impatience due to the extended periods spent waiting in line, especially when they have limited time between classes or other commitments. Extended wait times diminish the overall satisfaction levels of students with the campus experience. They may perceive the canteen as inefficient or poorly managed. For students with tight schedules, long queues can disrupt their daily routines and impact their ability to manage time effectively. This can lead to rushed meals or missed opportunities to study or engage in extracurricular activities. Prolonged waiting times may deter students from using the campus canteen regularly, leading to a decline in customer retention and loyalty. Dissatisfied students may share their experiences with peers, faculty, or online platforms, potentially creating a negative reputation for the canteen and discouraging others from patronizing it.

2. Network Issues: The canteen may have network issues due to which payment may not go through and the student might have to wait for a long time for payment confirmation or skip food instead. It can disrupt the payment process, causing delays in payment confirmation. Students may experience frustration and inconvenience as they wait for the payment to go through. In some cases, the delay may be significant enough that students choose to skip purchasing food altogether. This not only affects their dining experience but also impacts the canteen's revenue and customer satisfaction. Addressing network issues is crucial to ensuring a smooth and efficient payment process, minimizing wait times, and providing a positive experience for students.

3. Cash Handling: Handling cash transactions can be cumbersome and time-consuming, especially when dealing with small denominations. Counting, verifying, and storing cash requires additional resources and increases the risk of errors. Additionally, reconciling cash transactions and maintaining accurate records can be labour-intensive and prone to discrepancies, impacting overall operational efficiency. Implementing cashless payment options can streamline transactions, reduce administrative burdens, and enhance security, offering a more convenient and efficient payment experience.

4. Order Errors: Human errors in taking and processing orders can lead to incorrect deliveries or missing items, causing dissatisfaction among students. Order errors significantly impacting students' satisfaction levels. These errors not only cause frustration but also disrupt the seamless dining experience expected by customers. Addressing these issues requires implementing robust order management systems and training staff to minimize errors by using an online system with order names, ensures accurate and timely deliveries for enhanced customer satisfaction.

5. Lack of Personalization: The system may not offer personalized recommendations or customization options based on user preferences, leading to a generic dining experience. The lack of personalization manifests as a standardized menu without options for customization or catering to specific dietary needs. This leads to a one-size-fits-all dining experience, where users may struggle to find suitable food choices. Consequently, users may experience dissatisfaction or boredom with the limited variety and lack of options tailored to their preferences. Integrating personalized features, such as dietary filters or customizable meal options, can significantly enhance the traditional canteen experience by offering users more control and satisfaction with their dining choices which is possible in online systems.

6. Inefficient Feedback Mechanism: Gathering feedback from students about their dining experiences and suggestions for improvement may be lacking or ineffective. The inefficient feedback mechanism in the traditional canteen system refers to a lack of structured processes for collecting and utilizing feedback from students regarding their dining experiences. This deficiency can result in missed opportunities to gather valuable insights, suggestions, and areas for improvement. Without an effective feedback system, the canteen management may struggle to address issues promptly, leading to persistent dissatisfaction among users. Implementing a robust feedback mechanism from online systems that encourages active participation, collects meaningful data, and facilitates actionable changes can greatly enhance the canteen's responsiveness to user needs and overall satisfaction levels.

7. Poor Integration with Technology: The canteen system may not leverage modern technology such as online ordering, mobile payments, or digital menus, resulting in a less convenient experience for tech-savvy students who prefer digital solutions for convenience and efficiency. Without these technological features, students may face challenges such as long wait times, manual payment processes, and limited access to menu information, ultimately impacting their overall dining experience.

OBJECTIVES

1. Revolutionize Campus Dining: Campus Cravings aims to leverage technology to create an efficient online system that prioritizes user convenience and promotes healthy living, addressing the time constraints faced by students. By harnessing digital solutions, Campus Cravings prioritizes user convenience and promotes healthy living, directly addressing the time constraints often faced by busy students. This approach not only streamlines the ordering process but also enhances the overall dining experience, making it more accessible, personalized, and conducive to a balanced lifestyle.

2. Streamlined Menus and Diverse Options: Campus Cravings revolutionizes the campus dining experience by introducing streamlined menus and a diverse range of food options. This strategic approach empowers students with choices that cater to their individual preferences and dietary needs, even within the limited time available between classes. The platform's user-friendly interface allows students to browse menus effortlessly, select meals with ease, and place orders quickly, enhancing convenience and reducing wait times. By incorporating healthy options alongside indulgent favourites, Campus Cravings promotes a balanced approach to dining, encouraging students to make nutritious choices without sacrificing taste or satisfaction. This comprehensive menu strategy not only addresses time constraints but also contributes to a more inclusive and enjoyable dining environment for the entire college community. With an emphasis on variety, quality, and efficiency, Campus Cravings sets a new standard for campus dining, ensuring that students can fuel their academic pursuits with delicious and nourishing meals.

3. Cashless Transactions: Implementing cashless transactions through Campus Cravings eliminates the inconvenience of carrying cash, providing a seamless and efficient ordering process. This innovation significantly speeds up transactions, particularly during peak dining hours, where time constraints are critical. By promoting a cashless environment, the platform enhances overall efficiency, reduces wait times, and improves the overall dining experience for students and staff alike. This modern approach not only aligns with current digital payment trends but also ensures a more secure and convenient payment experience, further contributing to a streamlined and user-centric dining environment on campus.

4. Reduced Wait Times: Ensuring reduced wait times for orders allows students to quickly grab a meal between classes, optimizing their time on campus and minimizing disruptions to

their academic schedules. Reducing wait times for orders is paramount for Campus Cravings as it enables students to efficiently grab meals between classes, maximizing their time on campus and minimizing disruptions to their academic schedules. This streamlined approach to food service ensures that students can enjoy timely meals without long queues or delays, promoting productivity and convenience throughout their busy day. By implementing efficient order processing systems, optimizing kitchen workflows, and leveraging technology for faster service, Campus Cravings prioritizes student needs and enhances their overall campus experience.

5. Tackles Network Issues: Tackling network issues is paramount for the smooth functioning of Campus Cravings, especially since food orders can be placed from anywhere within the campus. Network disruptions or slowdowns can significantly impact the ordering process, causing delays and frustration for users. Implementing robust network infrastructure, including reliable internet connectivity, load balancing mechanisms, and redundancy protocols, is essential to ensure uninterrupted service. Regular monitoring and proactive troubleshooting of network issues can help identify and resolve potential issues before they escalate. Additionally, employing cybersecurity measures to protect against cyber threats and ensuring data integrity further enhances the reliability and security of the platform, ultimately providing a seamless and efficient ordering experience for users across the campus**.**

6. Convenience: Campus Craving aims to provide the convenience of ordering food from anywhere inside the campus so that the students never have to skip food due to network issue or unable to confirm the payment status. Convenience is at the core of Campus Craving's mission, offering students the flexibility to order food from any location on campus. This ensures that students never have to skip meals due to network issues or uncertainty about payment status. By providing a seamless and reliable ordering experience, Campus Craving aims to enhance student satisfaction and convenience throughout their academic journey.

7. Personalization and User Feedback: The platform emphasizes personalization and user feedback to continuously improve the dining experience, ensuring it remains tailored to students' evolving preferences even within the constraints of their tight schedules. This commitment to customization allows Campus Cravings to adapt quickly to changing needs and preferences, fostering a more engaging and satisfying dining experience for students.

METHODOLOGY

The methodology of Campus Cravings involves a user-centric approach to ordering and managing food transactions on the platform along with the technology stack used:

User Arrival and Registration/Login: Frontend: HTML, CSS, JavaScript for user interface design and user interaction. Backend: Django framework for handling user authentication, registration, and profile creation. Database: SQLite3 for storing user credentials and profile information securely.

Menu Exploration and Selection: Frontend: HTML, CSS, JavaScript for displaying the dynamic menu interface with filtering and sorting options. Backend: Django for fetching and rendering menu data from the database based on user preferences and selections.

Order Placement and Cart Management: Frontend: HTML, CSS, JavaScript for managing the shopping cart functionality, allowing users to add, remove, and modify items. Backend: Django for handling cart operations, updating item quantities, and calculating the total cost of the order.

Order Processing and Inventory Check: Backend: Django for real-time inventory management, checking item availability, and updating stock levels after order placement.

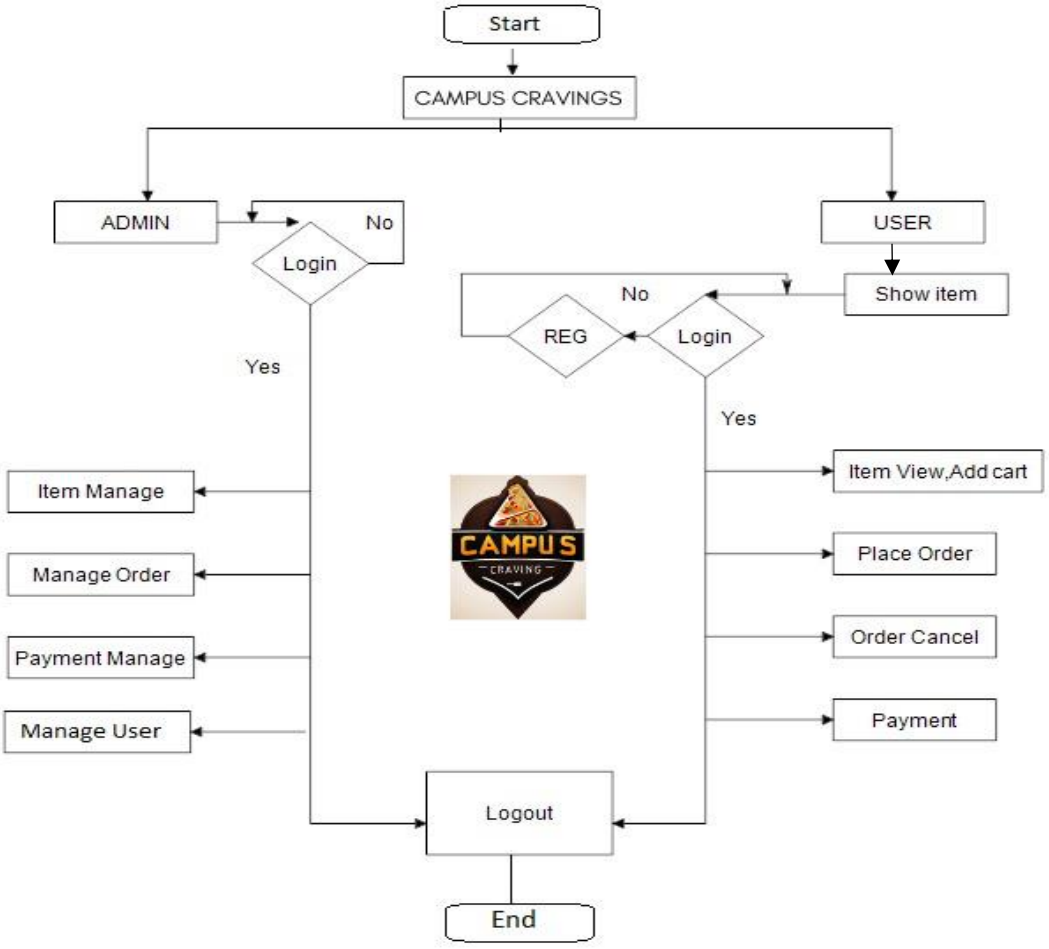
Order Confirmation and Payment Processing: Frontend: HTML, CSS, JavaScript for displaying order confirmation details and initiating payment processing. Backend: Django with integration of payment gateway APIs (e.g., Stripe, PayPal) for secure payment processing and transaction management.

Order Completion and Feedback: Backend: Django for marking orders as completed, sending confirmation emails, and storing order history. Frontend: HTML, CSS, JavaScript for displaying feedback forms, collecting user ratings, reviews, and comments.

Account Management: Backend: Django for fetching and displaying order and payment history respectively, managing account settings, and handling user preferences.

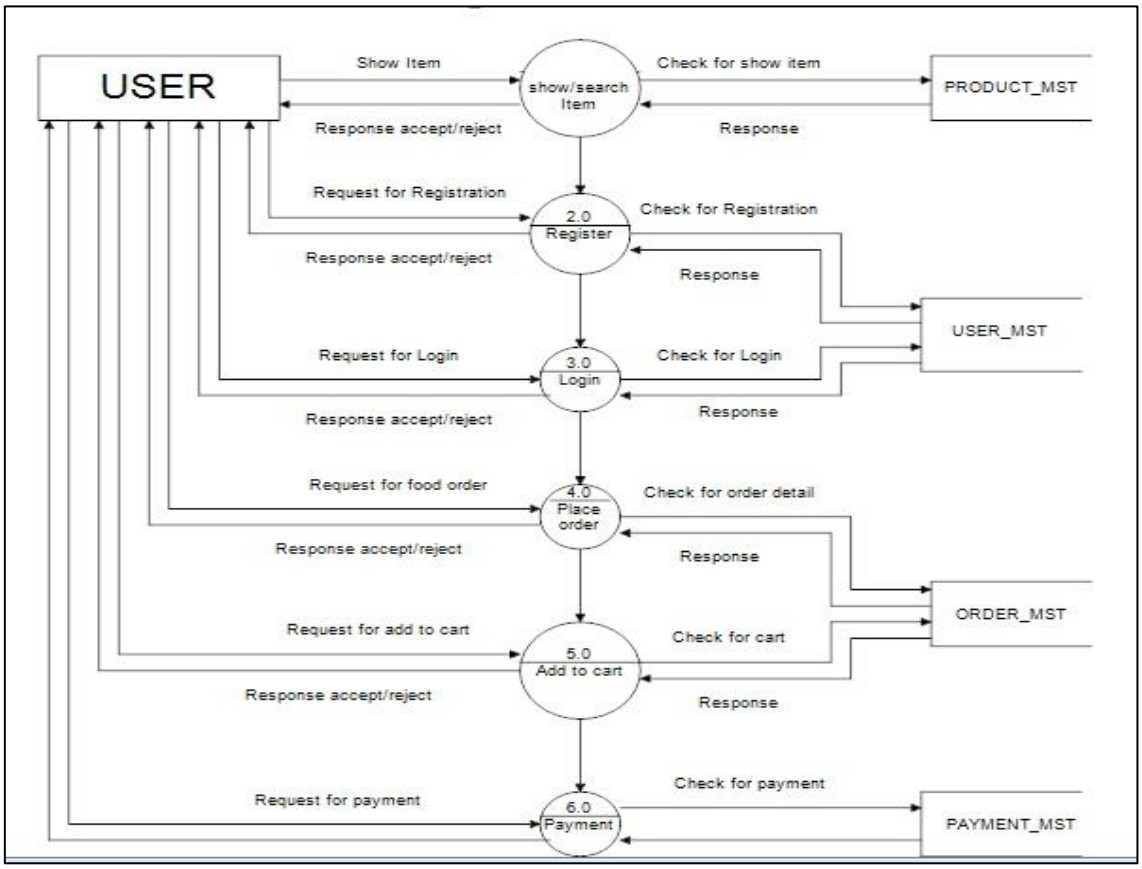
This technology stack ensures a robust and scalable architecture for Campus Cravings, combining frontend technologies (HTML, CSS, JavaScript) for a responsive and interactive user interface, Django framework for backend logic, database management, and seamless integration of payment gateways, and SQLite3 as the relational database for efficient data storage and retrieval.

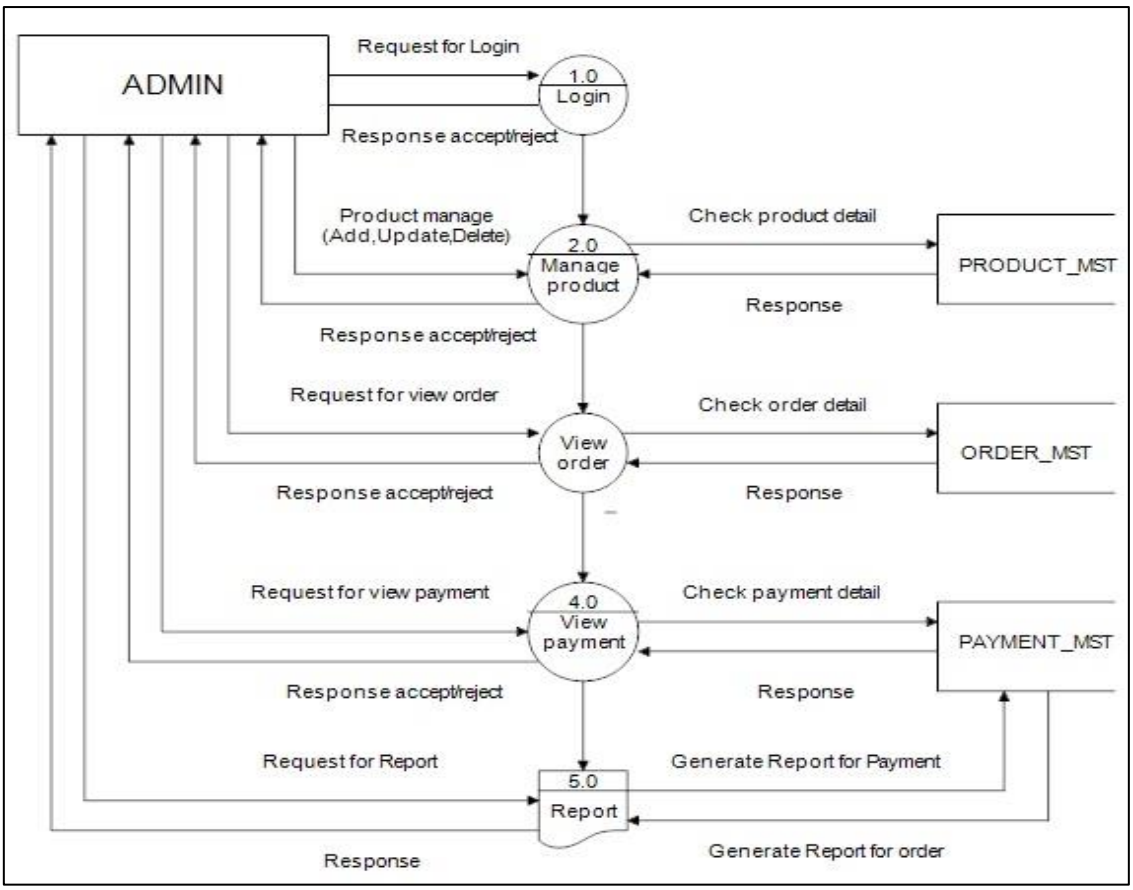
FLOW DIAGRAM



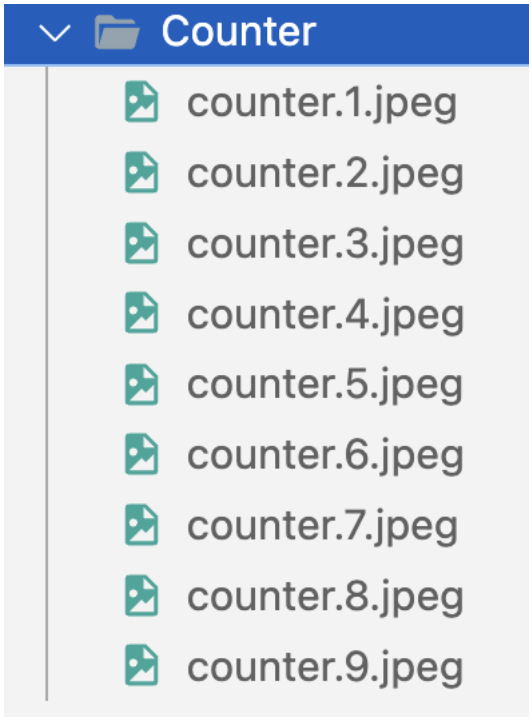
- User arrives at the website.
- They can either log in with existing credentials or register as a new user.
- After logging in, the user can browse the menu items available in the canteen.
- User selects items from the menu.
- Adds items to the cart.
- Proceeds to place an order.
- The system processes the order, checking item availability.
- If items are available, the order is confirmed.
- User proceeds to the payment step.
- Enters payment details and confirms the transaction.
- The system marks the order as completed.
- Sends confirmation to the user.
- User can view their order history and details of previous transactions.

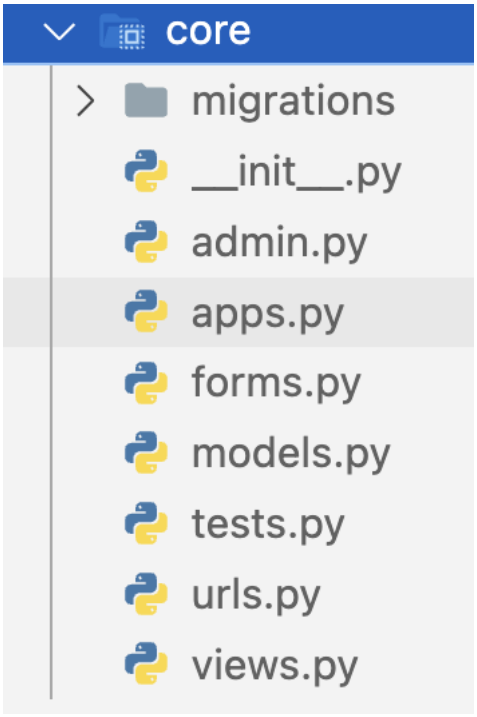
DATA FLOW DIAGRAM

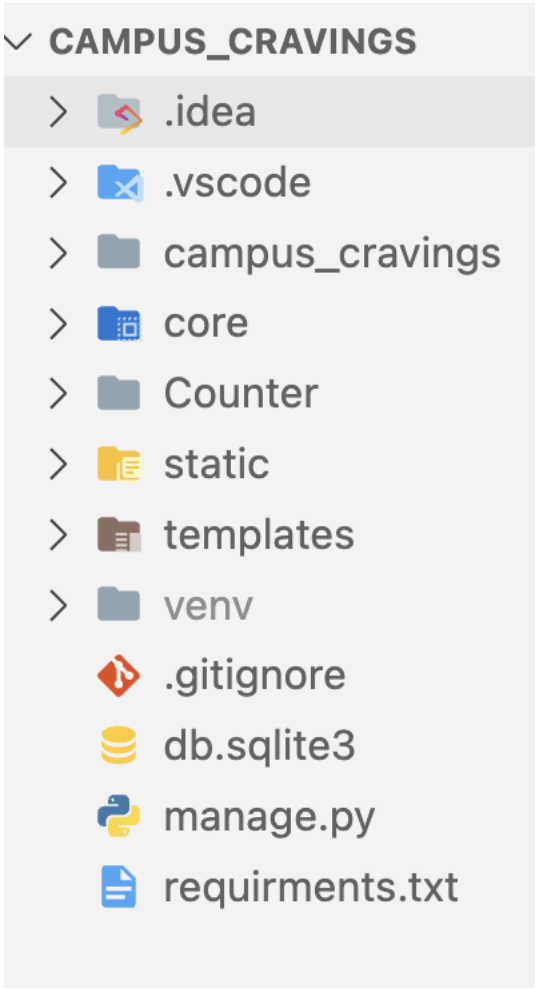


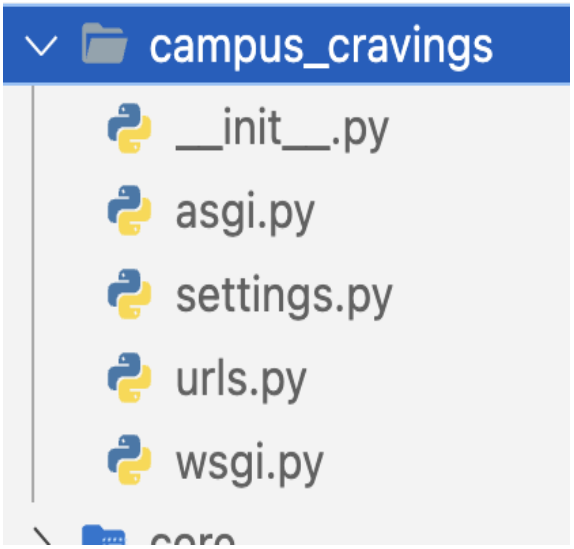


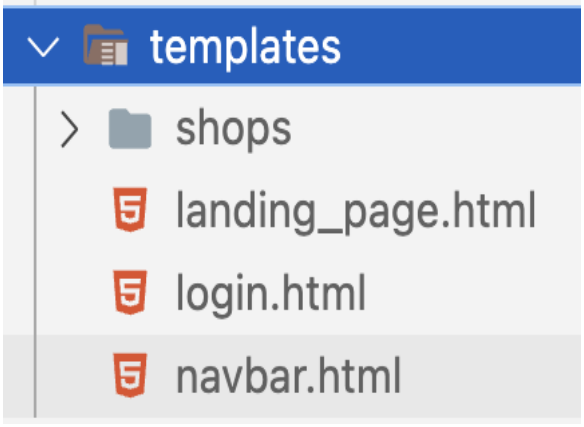
MODULARISATION











SOFTWARE TOOLS

1. FRONTEND DEVELOPMENT:

HTML: Provides the structure and content of the Campus Cravings website, defining elements like headings, paragraphs, buttons, and forms.

CSS: Styles the HTML elements, ensuring a visually appealing and consistent layout across different devices. It handles colours, fonts, spacing, and responsiveness for improved user experience.

JavaScript: Adds interactive features to the website, such as dropdown menus, sliders, form validation, and dynamic content updates. It enhances user engagement and functionality, making the site more user-friendly.

2. BACKEND DEVELOPMENT:

Django: Powers the backend of Campus Cravings, managing server-side logic, URL routing, database interactions (using SQLite), user authentication, and session management. It simplifies development with its built-in features and follows the Model-View-Template (MVT) architecture.

SQLite: Serves as the relational database management system for storing user data, menu items, orders, and feedback securely. It integrates seamlessly with Django, allowing efficient data retrieval and manipulation.

3. PAYMENT GATEWAY INTEGRATION:

Razorpay: Integrated with Campus Cravings to handle secure and seamless online transactions. Razorpay's APIs facilitate payment processing, including credit/debit cards, net banking, and digital wallets, ensuring a smooth checkout experience for users.

4.DESIGN:

Figma: Utilized for designing the website's visual elements, wireframes, and prototypes collaboratively. Figma's features enable seamless design iteration, version control, and feedback integration, ensuring a user-friendly and aesthetically pleasing interface.

Canva: Used for creating graphics such as logos, icons, banners, and interface elements. Canva's intuitive tools and templates help in designing visually appealing assets for the website, enhancing its overall branding and user experience.

HARDWARE REQUIREMENTS

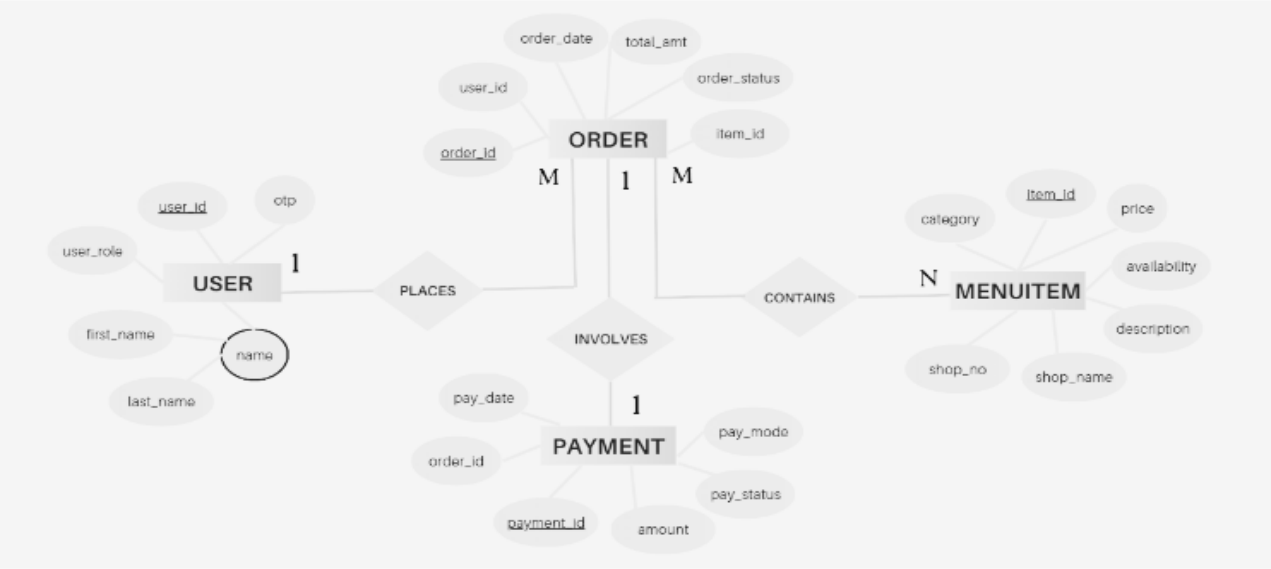
1. Frontend Development: HTML, CSS, and JavaScript are essential components of frontend development. HTML provides the structure and content of web pages, defining elements like headings, paragraphs, buttons, and forms. CSS is used for styling and design, ensuring an attractive and user-friendly interface by handling colours, fonts, spacing, and responsiveness for different devices. JavaScript adds interactivity and dynamic functionality to the website, such as real-time updates, animations, form validation, and interactive elements like dropdown menus and sliders. These technologies collectively require a standard computer or laptop with a modern web browser for coding, styling, and adding interactive features to the website.

2. Backend Development: Django is a high-level Python web framework used for backend development. It facilitates rapid development and clean, pragmatic design by handling server-side logic, URL routing, database interactions (using SQLite), user authentication, and session management. Django requires a computer or server capable of running Python and the Django framework, typically with at least 2GB RAM and a multi-core processor for development. SQLite, a lightweight and efficient relational database management system, is often used with Django for storing user data, menu items, orders, and feedback securely.

3. Payment Gateway Integration: Razorpay integration is essential for enabling secure and seamless online transactions on the Campus Cravings website. Razorpay provides payment gateway APIs that facilitate payment processing, including credit/debit cards, net banking, and digital wallets, ensuring a smooth checkout experience for users. Integrating Razorpay requires a stable internet connection and a computer or laptop capable of running web browsers for testing payment functionalities securely.

4.Design: Figma and Canva are design tools used for creating visually appealing and user-friendly websites. Figma is a versatile design tool that enables seamless collaboration and prototyping for designing the website's visual elements, wireframes, and interfaces collaboratively. Figma requires a computer or laptop with internet connectivity and modern web browsers for designing and iterating on website designs. Canva, on the other hand, is used for creating graphics such as logos, icons, banners, and interface elements. Canva is accessible on computers or laptops with web browsers, making it convenient for designing visual assets for the website interface. Both tools contribute to enhancing the overall branding, aesthetics, and user experience of the Campus Cravings website.

ER DIAGRAM



The ER diagram depicts the entities and relationships involved in Campus Cravings:-

User: This entity represents the system's users who can place orders. It has attributes such as user ID, first name, last name, and user role.

Order: This entity represents an order placed by a user. It has attributes such as order ID, order date, total amount, and order status.

Payment: This entity represents the payment made for an order. It has attributes such as payment ID, order ID, pay date, amount, and payment status.

Menu Item: This entity represents the items available for order. It has attributes such as item ID, description, price, and category.

The relationships between the entities are:

A User can place many Orders (one-to-many).

An Order involves one User (one-to-many).

An Order contains many Menu Items (one-to-many).

A Menu Item can be included in many Orders (many-to-many).

An Order is associated with one Payment (one-to-one).

A Payment is made for one Order (one-to-one).

CODING

#backend operations

```
# forms.py
from django import forms
from django.contrib.auth.models import User
from django.contrib.auth.forms import UserCreationForm
class UserRegisterForm(UserCreationForm):
    email = forms.EmailField()
    class Meta:
        model = User
        fields = ['username', 'email', 'password1', 'password2']
# core/urls.py
from django.urls import path
from . import views
from django.contrib.auth import views as auth_views
app_name = 'core'  # Namespace for the 'core' app
urlpatterns = [
    path('', views.landing_page, name='landing_page'),
    path('login/', views.signup_login, name='login'),
    path('logout/', auth_views.LogoutView.as_view(), name="logout"),
    path('shops/', views.ShopListView.as_view(), name='shop_list'),
    path('shops/<int:pk>/', views.ShopDetailView.as_view(),
name='shop_detail'),
    path('shops/<int:shop_id>/food_items/', views.food_item_list,
name='food_item_list'),
    path('my_orders/', views.user_orders, name='user_orders'),
]
 #Models.py
from django.db import models
```

```
from django.contrib.auth.models import User

import uuid

class Shop(models.Model):

    name = models.CharField(max_length=100)

    owner = models.ForeignKey(User, related_name='owned_shops',
on_delete=models.CASCADE)

    location = models.CharField(max_length=100)

    description = models.TextField()

    image = models.URLField(null=True, blank=True)

    def _str_(self):

        return self.name

class FoodItem(models.Model):

    shop = models.ForeignKey(Shop, related_name='food_items',
on_delete=models.CASCADE)

    name = models.CharField(max_length=100)

    description = models.TextField()

    price = models.DecimalField(max_digits=6, decimal_places=2)

    def _str_(self):

        return self.name

class Order(models.Model):

    user = models.ForeignKey(User, related_name='orders',
on_delete=models.CASCADE)

    shop = models.ForeignKey(Shop, related_name='orders',
on_delete=models.CASCADE)

    items = models.ManyToManyField(FoodItem, through='OrderItem')

    created_at = models.DateTimeField(auto_now_add=True)

    status = models.CharField(max_length=20, choices=[('PENDING',
'Pending'), ('CONFIRMED', 'Confirmed'), ('READY', 'Ready')])

    waiting_time = models.IntegerField(help_text="Estimated waiting
time in minutes", null=True, blank=True)

    confirmation_time = models.DateTimeField(null=True, blank=True)
```

```
    def _str_(self):
        return f"Order {self.id}"
class OrderItem(models.Model):
    order = models.ForeignKey(Order, on_delete=models.CASCADE)
    item = models.ForeignKey(FoodItem, on_delete=models.CASCADE)
    quantity = models.PositiveIntegerField()
    def _str_(self):
        return f"{self.quantity} of {self.item.name}"
def generate_uuid():
    return uuid.uuid4().hex
class OTP(models.Model):
    order = models.OneToOneField(Order, on_delete=models.CASCADE)
    code = models.CharField(max_length=6, default=generate_uuid,
unique=True)
    is_active = models.BooleanField(default=True)
    created_at = models.DateTimeField(auto_now_add=True)
    def _str_(self):
        return f"OTP {self.code} for Order {self.order.id}"
class UserProfile(models.Model):
    user = models.OneToOneField(User, on_delete=models.CASCADE)
    phone_number = models.CharField(max_length=15, blank=True,
null=True)
    address = models.TextField(blank=True, null=True)
    def _str_(self):
        return self.user.username
[01:33, 19/03/2024] Jalak Sharma: #Admin.py
from django.contrib import admi
from core.models import Shop, FoodItem, Order, UserProfile
# Register your models here.
```

```
admin.site.register(Shop)

admin.site.register(FoodItem)

admin.site.register(Order)

admin.site.register(UserProfile)

[01:34, 19/03/2024] Jalak Sharma: #Apps.py

from django.apps import AppConfig

class CoreConfig(AppConfig):

    default_auto_field = "django.db.models.BigAutoField"

    name = "core"

# core/views.py

from django.shortcuts import render, redirect, get_object_or_404

from django.views.generic import ListView, DetailView

from django.contrib.auth import login as auth_login, authenticate,
login

from django.contrib.auth.forms import UserCreationForm

from django.contrib.auth.decorators import login_required

from django.contrib import messages

from .models import Shop, FoodItem, Order

def landing_page(request):

    return render(request, 'landing_page.html')

# User Registration View

from django.shortcuts import render, redirect

from django.contrib.auth import authenticate, login

from django.contrib.auth.forms import AuthenticationForm,
UserCreationForm

def signup_login(request):

    action = request.GET.get('action', 'login')

    # Initialize form variable based on the action parameter

    if action == 'register':
```

```
        form = UserCreationForm()
    else:
        form = AuthenticationForm()
    if request.method == 'POST':
        if 'login' in request.POST:
            # Create a new instance of the login form with the POST
data
            form = AuthenticationForm(request, data=request.POST)
            if form.is_valid():
                user = form.get_user()
                login(request, user)
                return redirect('core:landing_page')  # Redirect to
a home page or other target
        elif 'register' in request.POST:
            # Create a new instance of the registration form with
the POST data
            form = UserCreationForm(request.POST)
            if form.is_valid():
                form.save()
                username = form.cleaned_data.get('username')
                password = form.cleaned_data.get('password1')
                user = authenticate(username=username,
password=password)
                login(request, user)
                return redirect('core:landing_page')  # Redirect to
a home page or other target
    return render(request, 'login.html', {'form': form, 'action':
action})
# Shop Listing View
class ShopListView(ListView):
    model = Shop
```

```
    template_name = 'shops/list.html'
    context_object_name = 'shops'
    queryset = Shop.objects.all().order_by("-id")
# Shop Detail View
class ShopDetailView(DetailView):
    model = Shop
    template_name = 'shops/detail.html'
    context_object_name = 'shop'
# Food Item Listing View
def food_item_list(request, shop_id):
    shop = get_object_or_404(Shop, pk=shop_id)
    food_items = FoodItem.objects.filter(shop=shop)
    return render(request, 'food_items/list.html', {'shop': shop,
'food_items': food_items})
# User Order Listing View
@login_required
def user_orders(request):
    orders = Order.objects.filter(user=request.user).order_by('-id')
    return render(request, 'orders/list.html', {'orders': orders})
#frontend operations
#landing page html
{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <!-- Google fonts -->
```

```
    <link rel="preconnect" href="https://fonts.googleapis.com">
    <link rel="preconnect" href="https://fonts.gstatic.com"
crossorigin>
    <link
href="https://fonts.googleapis.com/css2?family=Open+Sans:ital,wght@0
,300..800;1,300..800&family=Poppins:ital,wght@0,100;0,200;0,300;0,40
0;0,500;0,600;0,700;0,800;0,900;1,100;1,200;1,300;1,400;1,500;1,600;
1,700;1,800;1,900&display=swap"
        rel="stylesheet">
    <!-- font awesome -->
    <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">
    <!-- connect css -->
    <link rel="stylesheet" href="{% static 'css/landing_page.css'
%}">
    <title>Main Page</title>
</head>
<body>
    <div id="page1">
        <div id="page1_content">
            <h1>CampusCravings</h1>
            <h2>Fuel Your Study Sessions with Campus Craving
Delights!</h2>
            <a href=" ">Shops</a>
        </div>
    </div>
    <!-- header section -->
    <div id="header">
        <nav id="navbar">
            <h1>CrampusCravings</h1>
            <ul>
```

```
                <li><a href="{% url 'core:landing_page'
%}">Home</a></li>
                <li><a href="{% url 'core:shop_list'
%}">Shops</a></li>
                <li><a href="">Cart</a></li>
                <li><a href="">Contact</a></li>
                <li><a href="">Profile</a></li>
               {% if not user.is_authenticated %}
                    <li><a href="{% url 'core:login'
%}">Login</a></li>
                {% else %}
                    <li><a>Hello, {{ user.username }}</a></li>
                {% endif %}
            </ul>
            <div id="phone">
                <a href="tel:+91-7814064545">Call Customer Care</a>
            </div>
        </nav>
        <div id="mobile_menu">
            <ul>
                <li><a href="">Home</a></li>
                <li><a href="">Shop</a></li>
                <li><a href="">Cart</a></li>
                <li><a href="">Contact</a></li>
                <li><a href="">Profile</a></li>
                {% if not user.is_authenticated %}
                    <li><a href="{% url 'core:login'
%}">Login</a></li>
                {% else %}
                    <li><a>Hello, {{ user.username }}</a></li>
```

```
                {% endif %}
            </ul>
        </div>
    </div>
    </div>
    </div>
</body>
</html>
#landing page css
* {
    margin: 0;
    padding: 0;
}
html {
    font-size: 100%;
    scroll-behavior: smooth;
}
body {
    font-family: 'Poppins', sans-serif;
    overflow-x: hidden;
    /*in body of image will stop as soon as it comes down the x */
}
main {
    padding-top: 100px; /* Adjust this value if your header height
changes */
}
#page1 {
    position: relative;
    width: 100vw;
```

```
    height: 100vh;
    display: flex;
    justify-content: center;
    align-items: center;
    text-align: center;
}
#page1:
before {
    content: "";
    position: absolute;
    top: 0;
    left: 0;
    width: 100%;
    height: 100%;
    background-image: url('https://mccblr.edu.in/wp-
content/uploads/2022/06/cafeteria.png');
    background-size: cover;
    background-position: center;
    background-repeat: no-repeat;
    background-attachment: fixed;
    filter: brightness(40%);
}
#page1_content
{
    position: absolute;
}
#page1_content h1 {
    color: #FFFFFF;
    font-size: 2.8rem;
```

```
    text-shadow: 2px 2px 2px #000;
}
#page1_content h2 {
    color: #FFFFFF;
    font-size: 2rem;
    font-family: cursive;
    margin-top: 1rem;
    margin-bottom: 4rem;
    text-shadow: 2px 2px 2px #000;
}
#page1_content a {
    color: #FFFFFF;
    padding: 3px 20px 3px 20px;
    border: 1px solid orange;
    background: rgb(5, 5, 5);
    font-size: 1.5rem;
    text-decoration: none;
    /*it will remove the anchor line */
    border-radius: 25px;
}
#page1_content a:hover {
    background: #F9E6099C;
}
#header {
    position: fixed;
    top: 0;
    width: 100vw;
    height: 70px;
    line-height: 70px;
```

```
    z-index: 1000;
    /*the content will be in middle */
}
#navbar {
    display: flex;
    justify-content: space-around;
    background: rgb(0, 0, 0, 0.5);
}
#navbar h1 {
    color: orange;
    font-size: 1.8rem;
    /* it will give that hd effect */
    text-shadow: 2px 2px 2px #000;
    font-family: cursive;
}
#navbar ul {
    display: flex;
}
#navbar ul li {
    list-style: none;
    padding: 3px 15px 3px 15px;
}
#navbar ul li:hover,
#phone a:hover,
#mobile_menu ul li:hover,
#phone a:hover {
    background: #F9E6099C;
    cursor: pointer;
}
```

```
#navbar ul li a {
    text-decoration: none;
    color: #FFFCFC;
    font-size: 1.1rem;
}
#phone a {
    text-decoration: none;
    color: #FFFFFF;
    font-size: 1.1rem;
    padding: 3px 15px 3px 15px;
    border: 1px solid orange;
}
#mobile_menu {
    display: none;
    height: 40px;
    line-height: 40px;
    background-color: magenta;
}
#mobile_menu ul {
    display: flex;
    justify-content: center;
}
#mobile_menu ul li {
    list-style: none;
    padding: 0 5px 0 5px;
}
#mobile_menu ul li a {
    text-decoration: none;
    color: #FFFFFF;
```

```
}
@media screen and (max-width : 768px) {
    #mobile_menu {
        display: block;
    }
    #navbar ul {
        display: none;
    }
    html {
        font-size: 95%;
    }
}
@media screen and (max-width : 360px) {
    html {
        font-size: 70%;
    }
}
#login.html
{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <link rel="stylesheet"
href='https://unpkg.com/boxicons@2.1.4/css/boxicons.min.css'>
    <link rel="stylesheet" href="{% static 'css/LoginPage.css' %}">
    <title>{% if action == 'register' %}Register{% else %}Login{%
endif %} Page</title>
```

```
</head>
<body>
    <div class="wrapper">
        <div class="login_box">
            <form method="post" action="?action={{ action }}">
                {% csrf_token %}
                <div class="login-header">
                    <span>{% if action == 'register' %}Register{%
else %}Login{% endif %}</span>
                </div>
                {{ form.as_p }}
                <div class="input_box">
                    <input type="submit" name="{{ action }}"
class="input-submit" value="{% if action == 'register' %}Register{%
else %}Login{% endif %}">
                </div>
                {% if action == 'register' %}
                    <div class="register">
                        <span>Already have an account? <a
href="?action=login">Login</a></span>
                    </div>
                {% else %}
                    <div class="register">
                        <span>Don't have an account? <a
href="?action=register">Register</a></span>
                    </div>
                {% endif %}
            </form>
        </div>
    </div>
```

```
</body>

</html>

#login.css

@import
url('https://fonts.googleapis.com/css2?family=Poppins:wght@400;500;6
00;700;800&display=swap');

* {

    margin: 0;

    padding: 0;

    box-sizing: border-box;

    font-family: 'Poppins', sans-serif;

}

body {

    background-image:
url('https://media.istockphoto.com/id/1191930970/vector/spoon-and-
forks-cute-cutlery-abstract-background-vector-
illustration.jpg?s=612x612&w=0&k=20&c=VHQ-
Ix3j_4FdCWtCcragbZHnDk6tOKrV3SASGP5GCf8=');

    background-position: center;

    background-size: cover;

    background-repeat: no-repeat;

    background-attachment: fixed;

    backdrop-filter: blur(5px);

}

a {

    text-decoration: none;

    color: #ffa928;

}

a:hover {

    text-decoration: underline;

}
```

```
.wrapper {
    width: 100%;
    display: flex;
    justify-content: center;
    align-items: center;
    min-height: 100vh;
}
.login_box {
    position: relative;
    width: 450px;
    backdrop-filter: blur(4px);
    border: 2px solid rgba(35, 244, 247, 0.853);
    border-radius: 15px;
    padding: 7.5em 2.5em 4em 2.5em;
    color: rgb(0, 0, 0);
    box-shadow: 0px 0px 10px 2px #f6f935;
}
.login-header {
    position: absolute;
    top: 0;
    left: 50%;
    transform: translate(-50%);
    display: flex;
    align-items: center;
    justify-content: center;
    background-color: #f7e150fc;
    width: 140px;
    height: 70px;
    border-radius: 0 0 20px 20px;
```

```
}
.login-header span {
    font-size: 25px;
    color: black
}
.login-header::before {
    content: "";
    position: absolute;
    top: 0;
    left: -30px;
    width: 30px;
    height: 30px;
    border-top-right-radius: 50%;
    background: transparent;
    box-shadow: 15px 0 0 0 #f7e150fc
}
.login-header::after {
    content: "";
    position: absolute;
    top: 0;
    right: -30px;
    width: 30px;
    height: 30px;
    border-top-left-radius: 50%;
    background: transparent;
    box-shadow: -15px 0 0 0 #f7e150fc
}
.input_box {
    position: relative;
```

```
    display: flex;
    flex-direction: column;
    margin: 20px 0;
}
.input-field {
    width: 100%;
    height: 55px;
    font-size: 16px;
    background: transparent;
    color: #000000;
    padding-inline: 20px 50px;
    border: 2px solid rgb(0, 0, 0);
    border-radius: 30px;
    outline: none;
}
#user {
    margin-bottom: 10px;
}
.label {
    position: absolute;
    top: 15px;
    left: 20px;
    transition: 2s;
}
.input-field:focus~.label,
.input-field:valid~.label {
    position: absolute;
    top: -10px;
    left: 20px;
```

```
    font-size: 14px;
    background-color: hsla(58, 100%, 63%, 0.824);
    border-radius: 30px;
    color: #713ABE;
    padding: 0 10px;
}
.icon {
    position: absolute;
    top: 18px;
    right: 25px;
    font-size: 20px;
}
.remember-forgot {
    display: flex;
    justify-content: space-between;
    font-size: 15px;
}
.input-submit {
    width: 100%;
    height: 50px;
    background: whitesmoke;
    font-size: 16px;
    font-weight: 500;
    border: none;
    cursor: pointer;
    transition: 3s;
}
.input-submit:hover {
    background: #FFA41B;
```

```
}
.register {
    text-align: center;
}
.register a {
    font-weight: 500;
}
@media only screen and (max-width: 564px) {
    .wrapper {
        padding: 20px;
    }
    .login_box {
        padding: 7.5em 1.5em 4em 1.5em;
    }
}
#navbar.html
<div id="header">
        <nav id="navbar">
            <h1>CrampusCravings</h1>
            <ul>
                <li><a href="{% url 'core:landing_page'
%}">Home</a></li>
                <li><a href="{% url 'core:shop_list'
%}">Shops</a></li>
                <li><a href="">Cart</a></li>
                <li><a href="">Contact</a></li>
                <li><a href="">Profile</a></li>
                {% if not user.is_authenticated %}
                    <li><a href="{% url 'core:login'
%}">Login</a></li>
```

```
                {% else %}
                    <li><a>Hello, {{ user.username }}</a></li>
                {% endif %}
            </ul>
            <div id="phone">
                <a href="tel:+91-7814064545">Call Customer Care</a>
                <a href="{% url 'core:logout' %}">Logout</a>
            </div>
        </nav>
        <div id="mobile_menu">
            <ul>
                <li><a href="{% url 'core:landing_page'
%}">Home</a></li>
                <li><a href="{% url 'core:shop_list'
%}">Shops</a></li>
                <li><a href="">Cart</a></li>
                <li><a href="">Contact</a></li>
                <li><a href="">Profile</a></li>
                {% if not user.is_authenticated %}
                    <li><a href="{% url 'core:login'
%}">Login</a></li>
                {% else %}
                    <li><a>Hello, {{ user.username }}</a></li>
                {% endif %}
            </ul>
        </div>
</div>
```

#shop.css

```
#menu {
    padding: 25px 0 25px 0;
```

```
}
#section {
    padding: 25px 0 25px 0;
    text-align: center;
    font-size: 2rem;
    font-family: verdana;
}
#menu_row {
    display: flex;
    flex-wrap: wrap; /* Add this line to wrap items to the next line
*/
    justify-content: center; /* Center items horizontally */
    padding: 0 100px; /* Adjust padding as needed */
}
#menu_col {
    box-shadow: 2px 2px 2px #bbb;
    border: 1px solid #bbb;
    background-color: #FFFFFF;
    margin: 5px;
    padding: 5px;

}
#menu_col h2 {
    background-color: #07255F;
    color: #FFFFFF;
    text-align: center;
    padding: 5px;
    font-family: cursive;
}
```

```
#image {
    width: 150px;
    height: 150px;
    border-radius: 50%;
    padding: 5px;
    border: 2px solid orange;
}
#image img {
    width: 50%;
    height: 50%;
    border-radius: 50%;
    object-fit: cover;
}
.box {
    display: flex;
    flex-direction: column;
    align-items: center;
    text-align: center;
    margin: 5px;
    width: 200px; /* Adjust width as needed */
    height: 250px; /* Adjust height as needed */
}
* {
    margin: 0;
    padding: 0;
}
html {
    font-size: 100%;
    scroll-behavior: smooth;
```

```
}
body {
    font-family: 'Poppins', sans-serif;
    overflow-x: hidden;
    /*in body of image will stop as soon as it comes down the x */
}
main {
    padding-top: 100px; /* Adjust this value if your header height
changes */
}
#page1 {
    position: relative;
    width: 100vw;
    height: 100vh;
    display: flex;
    justify-content: center;
    align-items: center;
    text-align: center;
}
#page1::before {
    content: "";
    position: absolute;
    top: 0;
    left: 0;
    width: 100%;
    height: 100%;
    background-image: url('https://mccblr.edu.in/wp-
content/uploads/2022/06/cafeteria.png');
    background-size: cover;
```

```
    background-position: center;
    background-repeat: no-repeat;
    background-attachment: fixed;
    filter: brightness(40%);
}
#page1_content {
    position: absolute;
}
#page1_content h1 {
    color: #FFFFFF;
    font-size: 2.8rem;
    text-shadow: 2px 2px 2px #000;
}
#page1_content h2 {
    color: #FFFFFF;
    font-size: 2rem;
    font-family: cursive;
    margin-top: 1rem;
    margin-bottom: 4rem;
    text-shadow: 2px 2px 2px #000;
}
#page1_content a {
    color: #FFFFFF;
    padding: 3px 20px 3px 20px;
    border: 1px solid orange;
    background: rgb(5, 5, 5);
    font-size: 1.5rem;
    text-decoration: none;
    /*it will remove the anchor line */
```

```
    border-radius: 25px;
}
#page1_content a:hover {
    background: #F9E6099C;
}
#header {
    position: fixed;
    top: 0;
    width: 100vw;
    height: 70px;
    line-height: 70px;
    z-index: 1000;
    /*the content will be in middle */
}
#navbar {
    display: flex;
    justify-content: space-around;
    background: rgb(0, 0, 0, 0.5);
}
#navbar h1 {
    color: orange;
    font-size: 1.8rem;
    /* it will give that hd effect */
    text-shadow: 2px 2px 2px #000;
    font-family: cursive;
}
#navbar ul {
    display: flex;
}
```

```
#navbar ul li {
    list-style: none;
    padding: 3px 15px 3px 15px;
}
#navbar ul li:hover,
#phone a:hover,
#mobile_menu ul li:hover,
#phone a:hover {
    background: #F9E6099C;
    cursor: pointer;
}
#navbar ul li a {
    text-decoration: none;
    color: #FFFCFC;
    font-size: 1.1rem;
}
#phone a {
    text-decoration: none;
    color: #FFFFFF;
    font-size: 1.1rem;
    padding: 3px 15px 3px 15px;
    border: 1px solid orange;
}
#mobile_menu {
    display: none;
    height: 40px;
    line-height: 40px;
    background-color: magenta;
}
```

```
#mobile_menu ul {
    display: flex;
    justify-content: center;
}
#mobile_menu ul li {
    list-style: none;
    padding: 0 5px 0 5px;
}

#mobile_menu ul li a {
    text-decoration: none;
    color: #FFFFFF;
}
@media screen and (max-width : 768px) {
    #mobile_menu {
        display: block;
    }
    #navbar ul {
        display: none;
    }
    html {
        font-size: 95%;
    }
}
@media screen and (max-width : 360px) {
    html {
        font-size: 70%;
    }
}
```

UNIT TESTING

Various components of the Campus Cravings online canteen system were tested to ensure functionality, accuracy, and reliability.

1. User Authentication and Registration:

Successfully tested the registration process for new users, ensuring accounts are created without errors.

Tested login functionality with correct credentials, verifying that existing users can log in securely.

Validated handling of invalid login attempts, such as incorrect passwords or non-existent users, with appropriate error messages displayed.

2. Menu Management:

Added new menu items and confirmed they appeared correctly in the menu interface.

Updated existing menu items to ensure changes were reflected accurately.

Verified that deleted menu items were removed from the menu interface as expected.

3. Cart and Order Processing:

Added items to the cart and confirmed they displayed accurate quantities and prices. Tested cart functionality, including updating item quantities and removing items, without errors.

Successfully completed the checkout process, ensuring orders were processed correctly and payment details were captured securely.

4. Inventory Management:

Tested inventory tracking, verifying that items were deducted from stock upon order placement.

Validated inventory replenishment, confirming that stock levels were updated accurately.

5. User Preferences and Profile Management:

Updated user profiles with new information and verified changes were saved correctly.

Successfully reset passwords through the password reset functionality.

6. Error Handling and Data Validation:

Conducted input validation tests for various forms, preventing users from submitting invalid data.

Verified error handling for network errors, database connectivity issues, and server timeouts.

Tested error messages and notifications for clarity and helpfulness in guiding users to resolve issues.

7. Performance and Scalability:

Tested system performance under different loads, ensuring responsiveness and stability even during peak ordering times.

Verified that the system could handle a large volume of transactions without significant slowdowns or errors.

The unit testing process for Campus Cravings was comprehensive and successful in validating the functionality, accuracy, and reliability of the online canteen system's key components. Identified issues were addressed and resolved, ensuring a robust and user-friendly experience for users.

SOFTWARE TESTING

The software testing process for Campus Cravings was comprehensive and successful in identifying and resolving issues across different testing categories. The system demonstrated functionality, performance, security, usability, compatibility, and accessibility compliance.

1. Functional Testing:

User Authentication and Registration: Verified that users can register, log in, and update their profiles without errors. Tested password reset functionality.

Menu Management: Tested adding, updating, and deleting menu items. Ensured accurate display of menus and availability of items.

Cart and Order Processing: Tested adding items to the cart, updating quantities, and completing orders. Verified payment processing and order confirmation.

Inventory Management: Tested inventory tracking, out-of-stock handling, and inventory replenishment processes.

2. Performance Testing:

Conducted load testing to assess system performance under various loads. Ensured responsiveness and stability during peak usage times.

Tested system scalability to handle a large number of concurrent users and transactions without performance degradation.

3. Security Testing:

Conducted penetration testing to identify vulnerabilities and ensure data security.

Verified secure handling of user data, including encryption of sensitive information like passwords and payment details.

Tested authentication mechanisms and access controls to prevent unauthorized access.

4. Usability Testing:

Evaluated the user interface for consistency, clarity, and ease of navigation.

Conducted user acceptance testing with actual users to gather feedback on usability and overall user experience.

Tested mobile responsiveness and compatibility with different devices and browsers.

5. Compatibility Testing:

Tested compatibility with various operating systems (Windows, macOS, Linux) and web browsers (Chrome, Firefox, Safari, Edge).

Ensured the system functions correctly across different devices, screen sizes, and resolutions.

6. Regression Testing:

Conducted regression testing after system updates or bug fixes to ensure new changes did not introduce new issues or affect existing functionality.

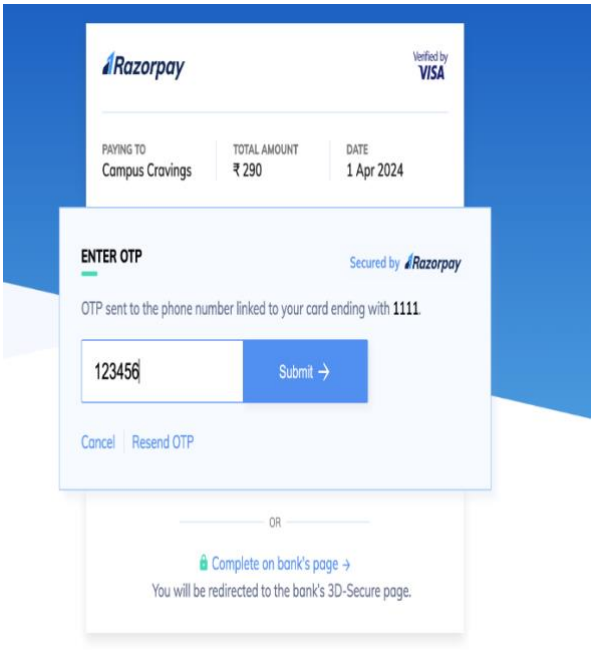
7. Accessibility Testing:

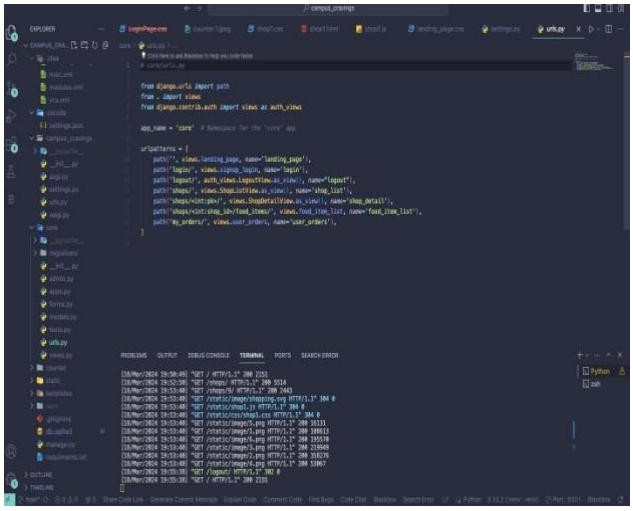
Evaluated the system for accessibility compliance, including screen reader compatibility, keyboard navigation, and colour contrast for visually impaired users.

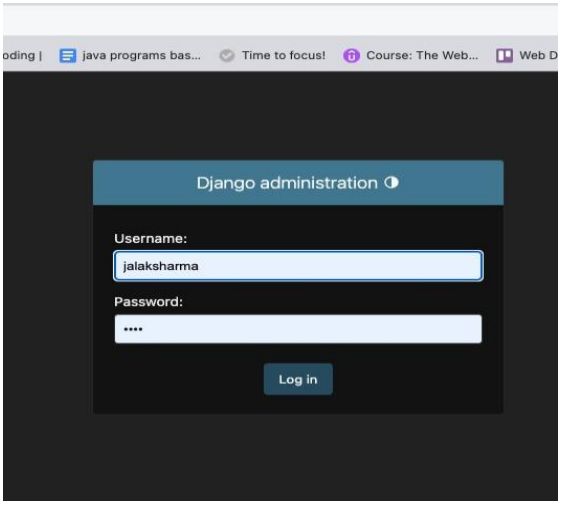
8. Error Handling and Recovery Testing:

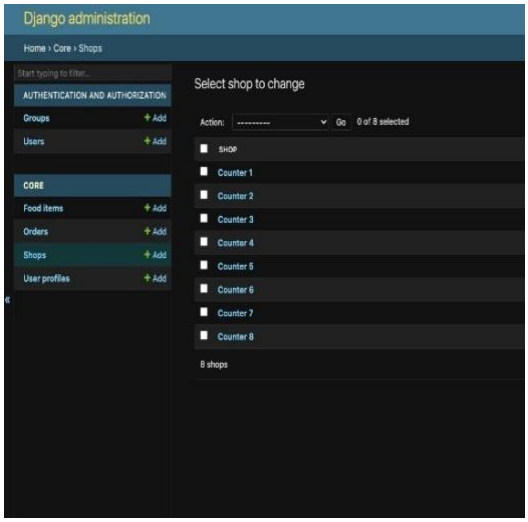
Tested error handling mechanisms for various scenarios, such as network errors, server downtime, and user input errors. Verified that error messages were clear, informative, and guided users to resolve issues effectively.

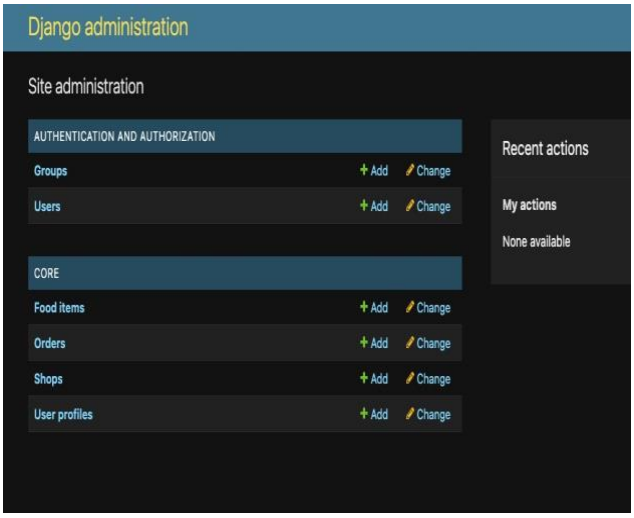
WORKING VALIDATIONS

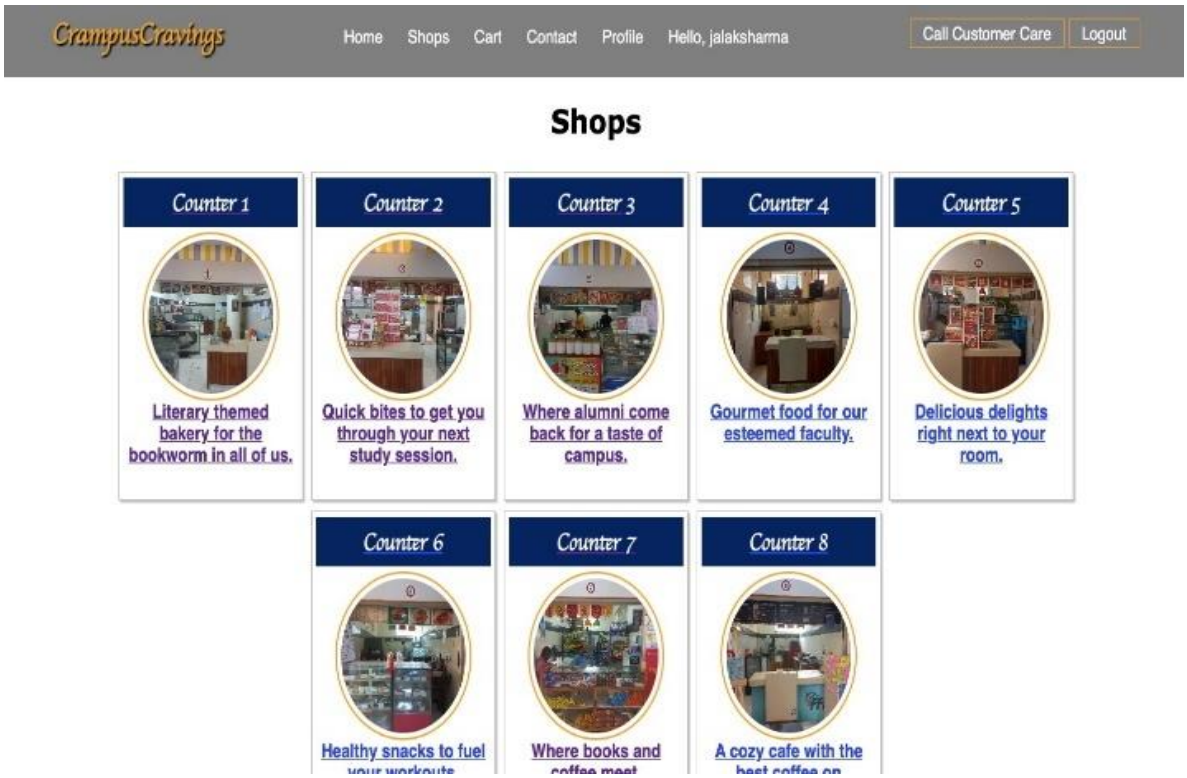


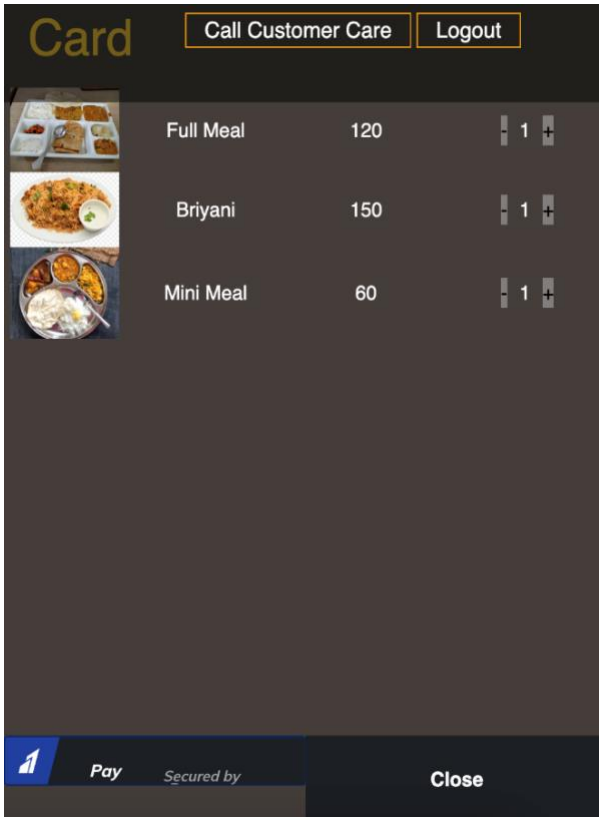


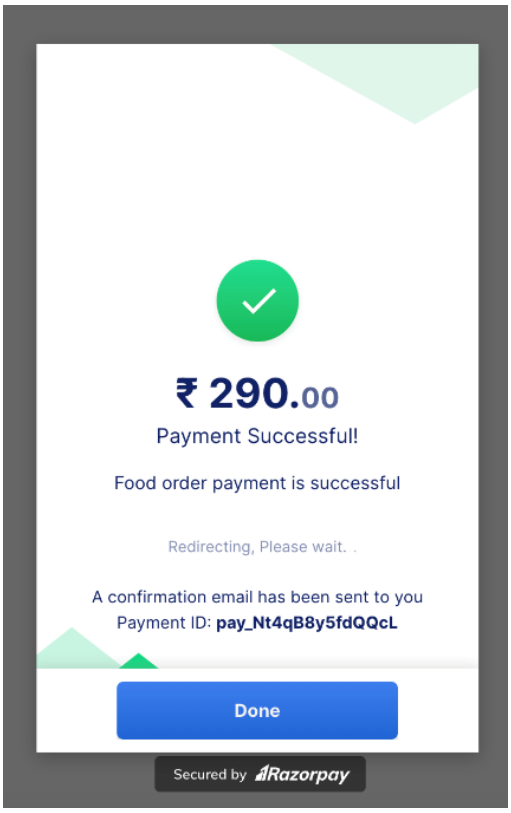

Django administration
Username:
jalaksharma
Password:
Log in

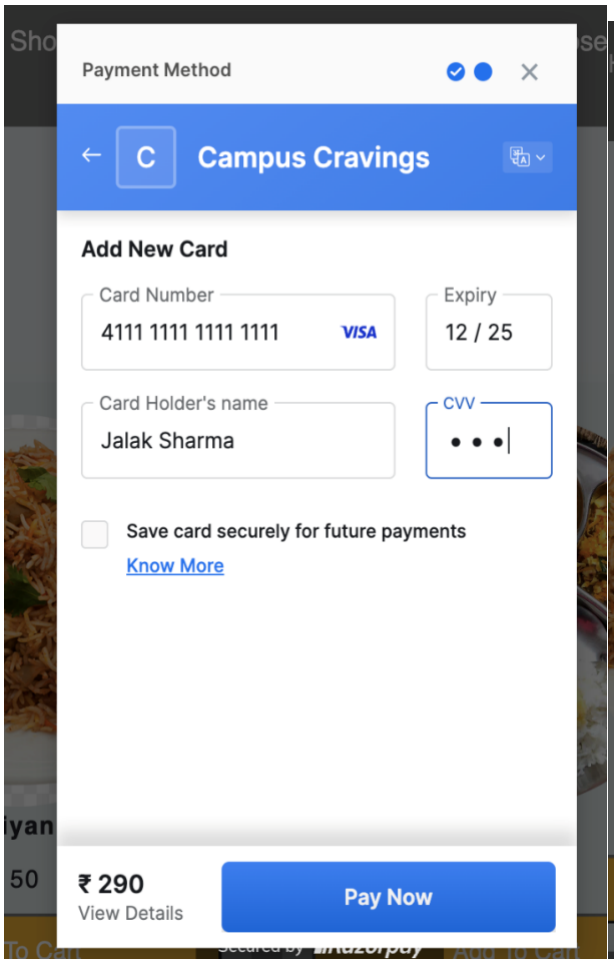

Django administration
Home › Core › Shops
Select shop to change
AUTHENTICATION AND AUTHORIZATION
Groups
Users
CORE
Food items
Orders
Shops
User profiles
Counter 1
Counter 2
Counter 3
Counter 4
Counter 5
Counter 6
Counter 7
Counter 8
8 shops

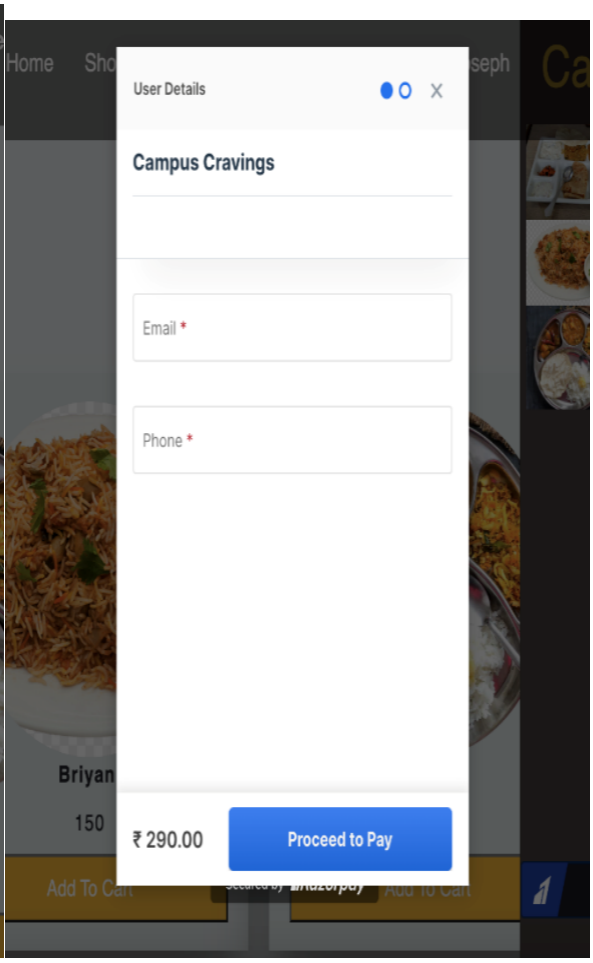

Django administration
Site administration
AUTHENTICATION AND AUTHORIZATION
Groups
Users
CORE
Food items
Orders
Shops
User profiles
Recent actions
My actions
None available


CampusCravings
Home
Shops
Cart
Contact
Profile
Hello, jalaksharma
Call Customer Care
Logout
Shops
Counter 1
Literary themed bakery for the bookworm in all of us.
Counter 2
Quick bites to get you through your next study session.
Counter 3
Where alumni come back for a taste of campus.
Counter 4
Gourmet food for our esteemed faculty.
Counter 5
Delicious delights right next to your room.
Counter 6
Healthy snacks to fuel
Counter 7
Where books and
Counter 8
A cozy cafe with the


Card
Call Customer Care
Logout
Full Meal
120
1
Briyani
150
1
Mini Meal
60
1
Pay
Secured by
Close


₹ 290.00
Payment Successful!
Food order payment is successful
Redirecting, Please wait. .
A confirmation email has been sent to you
Payment ID: pay_Nt4qB8y5fdQQcL
Done
Secured by Razorpay


Payment Method
Campus Cravings
Add New Card
Card Number
4111 1111 1111 1111
VISA
Expiry
12 / 25
Card Holder's name
Jalak Sharma
CVV
Save card securely for future payments
Know More
₹ 290
View Details
Pay Now


User Details
Campus Cravings
Email *
Phone *
₹ 290.00
Proceed to Pay

CONCLUSION

In conclusion, the proposed online canteen experience offers a comprehensive solution to the dining needs of college students, faculty, and staff within the campus environment. With its intuitive user interface, menu exploration capabilities, robust ordering system, seamless payment integration, user account functionality, feedback and ratings features, and compatible design, the platform prioritizes user convenience, customization, and satisfaction.

By providing users with easy access to a diverse range of food options, personalized ordering experiences, secure payment options, and opportunities for engagement and feedback, the online canteen experience aims to revolutionize the way individuals interact with campus dining services. Through continuous refinement based on user feedback and technological advancements, the platform seeks to enhance the overall dining experience, making it more convenient, enjoyable, and accessible for all members of the college community.

The platform offers an intuitive and user-friendly interface that allows users to navigate seamlessly. Clear menu categories, search functionality, and visually appealing layouts make it easy for users to explore food options and place orders efficiently. Users can browse through a diverse range of food options, including various cuisines, dietary preferences (e.g., vegetarian, vegan, gluten-free), and special promotions. The platform's ordering system is robust and efficient. Users can add items to their cart, customize orders (e.g., toppings, sides), and review their selections before finalizing their orders. The system ensures accuracy and completeness in order processing. Secure payment integration allows users to complete transactions seamlessly.

Multiple payment options, such as credit/debit cards, digital wallets, and campus meal plans, provide flexibility and convenience for users. Registered users have access to personalized features and benefits. The platform is designed to be compatible across different devices, including desktops, laptops, tablets, and smartphones. Responsive design ensures a seamless user experience regardless of the device used.

Overall, this comprehensive solution represents a significant step towards modernizing campus dining services and creating a dining environment that is efficient, user-centric, and conducive to a positive campus experience for all users.

SCOPE AND ENHANCEMENTS

In envisioning the future of the Student Canteen Website, several exciting enhancements are on the horizon to elevate the user experience and cater to evolving student needs. One of the key features in the pipeline is the integration of a loyalty program, aimed at rewarding frequent users and fostering long-term engagement. By offering incentives such as discounts or free items, the loyalty program will incentivize students to return to the website for their dining needs, thus fostering a sense of loyalty and appreciation among users.

Another significant enhancement planned for the website is the implementation of real-time order tracking. This feature will provide users with enhanced transparency and visibility into the status of their orders, from preparation to delivery. By leveraging real-time data and notifications, students can track their orders accurately and plan accordingly, reducing uncertainty and potential wait times. This added layer of convenience aligns with the website's goal of promoting quick dining experiences and further enhances user satisfaction.

Looking ahead, the Student Canteen Website aims to expand its offerings to cater to a broader range of student preferences and dietary requirements. The introduction of a dedicated mobile application will enable on-the-go ordering and notifications, providing students with greater flexibility and convenience in accessing the canteen's services. Furthermore, plans to diversify the menu to accommodate various dietary preferences, including vegetarian, vegan, and gluten-free options, underscore the website's commitment to inclusivity and catering to the diverse needs of the student community includes:

1. Integrating a Loyalty Program: Implementing a loyalty program will incentivize and reward frequent users, encouraging them to return to the platform for their dining needs. Points-based rewards, discounts, and exclusive offers for loyal customers can foster a sense of appreciation and loyalty among users.

2. Real-Time Order Tracking: By implementing real-time order tracking functionality, users can track the status of their orders from placement to delivery. This enhanced transparency provides users with peace of mind and allows them to plan their activities accordingly while awaiting their meals.

3. Customization Options for Nutritional Information Display: Providing customization options for displaying nutritional information allows users to view information relevant to their dietary preferences and health goals. This feature enables users to make informed decisions about their food choices based on their individual nutritional needs.

4. Building a Mobile Application with Real-Time Location Detection: Developing a dedicated mobile application equipped with real-time location detection capabilities enables users to access the application only within the college vicinity. This ensures that the platform remains exclusive to campus users, enhancing security and convenience.

5. Continuous Optimization Based on User Feedback: Continuously gathering and analysing user feedback allows for ongoing optimization of the platform. Regular updates and improvements based on user suggestions ensure that the platform remains responsive to user needs and preferences, fostering long-term user satisfaction and engagement.

6. Profile Details and Order History for Marketing Automation: Utilizing profile details and order history data enables the implementation of targeted marketing strategies. By leveraging this information, personalized promotions, recommendations, and marketing campaigns can be tailored to individual user preferences, enhancing user engagement and loyalty.

7. Incorporation of OTP-Based Login System: Implementing an OTP-based login system enhances security by adding an additional layer of authentication. This feature ensures that user accounts remain secure, protecting sensitive information such as personal details and payment credentials from unauthorized access.

By incorporating these future enhancements, the online canteen experience can be further optimized to meet the diverse needs and preferences of users. From enhancing convenience and transparency to fostering user loyalty and engagement, these enhancements contribute to creating a more seamless, personalized, and enjoyable dining experience for all users within the campus community.

Through continuous optimization based on user feedback and the integration of emerging technologies, the Student Canteen Website seeks to remain dynamic and responsive, ensuring that it stays aligned with the ever-changing demands and expectations of students. Potential partnerships with local food vendors also present exciting opportunities for collaboration and expansion, further enriching the dining experience offered through the platform.

BIBLIOGRAPHY

https://www.codecademy.com/learn/learn-html

https://css-tricks.com/

https://www.freecodecamp.org/

https://www.educative.io/courses/learn-html-css-javascript-from-scratch

https://www.youtube.com/watch?v=PtQiiknWUcI

https://codepen.io/pen/

https://www.javatpoint.com/software-engineering

https://skills.github.com/

https://www.figma.com/resource-library/design-basics/

https://techvidvan.com/tutorials/django-database-connectivity/

https://www.zomato.com/bangalore

https://www.swiggy.com/city/bangalore

https://justvend.in/web/

https://kiosk.com/