

Online Food Ordering System



GROUP MEMBERS

“Syed Muhammad Shahzaib Ali”

Student1297252

“Syed Shahrukh Ahmed”

Student1289573

“Muhammad Ali”

Student1296037

“Muhammad Muawia”

Student1296575

CURRICULUM

6740

BATCH

2101B1

FACULTY

“MS. SAMREEN RAFIQ”

CO-ORDINATOR

“MS. MAHA HAIDER”

View

```
from django.http import request
from django.shortcuts import render
from .models import Contact_msg, Restaurant, category, dish, WebsiteInfo

# from .models import WebsiteInfo
# var = WebsiteInfo.objects.all().values()
# request.session['add'] = var
# Create your views here.

def index(request):
    one_res = Restaurant.objects.all().order_by('-Res_id')[:3]
    cate = category.objects.all()
    dishlist = dish.objects.all()
    context = {
        'cate': cate,
        'dishlist': dishlist,
        'one_res': one_res
    }
    return render(request, "index.html", context)

def restaurant(request):
    cate = category.objects.all()
    res = Restaurant.objects.all()
    context = {
        'cate': cate,
        'res': res,
    }

    return render(request, "restaurant.html", context)

def dishes(request):
    cate = category.objects.all()
    dishlist = dish.objects.all()
    context = {
        'dishlist': dishlist,
        'cate': cate,
    }

    return render(request, "dishes.html", context)

def dishCategory(request, id):
    cate = category.objects.all()
    dishlist = dish.objects.all().filter(cate_id=id)
    context = {
```

```

        'cate': cate,
        'dishlist': dishlist,
    }

    return render(request, "dishes.html", context)

def resCategory(request, id):
    cate = category.objects.all()
    res = Restaurant.objects.all().filter(cate_id=id)
    context = {
        'cate': cate,
        'res': res,
    }

    return render(request, "restaurant.html", context)

def shop(request):
    return render(request, "shop-grid.html")

def contact(request):
    info = WebsiteInfo.objects.all()
    context = {
        'info': info,
    }

    if request.method == "POST":
        name = request.POST["con_name"]
        email = request.POST["con_email"]
        message = request.POST["con_msg"]

        Contact_msg.objects.create(Con_Name=name, Con_Email=email,
Con_msg=message)

        alert_msg = "Message sent successfully!"
        return render(request, "contact.html", {"m": alert_msg})

    return render(request, "contact.html", context)

def shop_details(request):
    return render(request, "shop-details.html")

def shopping_cart(request):
    return render(request, "shopping-cart.html")

def checkout(request):
    return render(request, "checkout.html")

def searchbar_dish(request):
    if request.method == 'GET':

```

```

        query = request.GET.get("search_query")
        if query:
            cate = category.objects.all()
            dishlist = dish.objects.all().filter(Dish_Name__icontains=query)

            return render(request, 'search_dish.html', {'dishlist': dishlist,
'cate': cate})
        else:

            return render(request, 'search_dish.html')

def searchbar_res(request):

    if request.method == 'GET':
        query = request.GET.get("search_query")
        if query:
            cate = category.objects.all()
            reslist = Restaurant.objects.all().filter(Res_Name__icontains=query)

            return render(request, 'search_res.html', {'reslist': reslist, 'cate':
cate})
        else:

            return render(request, 'search_res.html')

def restaurant_details(request, id):
    res = Restaurant.objects.get(Res_id=id)
    context = {
        'res': res,
    }
    return render(request, "restaurant-details.html", context)

def user_login(request):
    return render(request, "user-login.html")

```



Model

```
from django.db import models

# Create your models here.

class category(models.Model):
    Cate_id = models.AutoField(primary_key=True)
    Cate_Name = models.CharField(max_length=50, unique=True)

    def __str__(self):
        return self.Cate_Name

class dish(models.Model):
    Dish_id = models.AutoField(primary_key=True)
    Dish_Name = models.CharField(max_length=50, unique=True)
    Dish_img = models.ImageField(upload_to='dish_img/', blank=True, null=True,
default='')
    Dish_Des = models.CharField(max_length=255)
    cate_id = models.ForeignKey(category, on_delete=models.CASCADE, null=True)
    Dish_Price = models.IntegerField()

    def __str__(self):
        return self.Dish_Name

class Area(models.Model):
    Ar_id = models.AutoField(primary_key=True)
    Ar_Name = models.CharField(max_length=50, unique=True)

    def __str__(self):
        return self.Ar_Name

class Restaurant(models.Model):
    Res_id = models.AutoField(primary_key=True)
    cate_id = models.ForeignKey(category, on_delete=models.CASCADE, null=True)
    Res_Name = models.CharField(max_length=100, unique=True)
    Res_img = models.FileField(upload_to='Res_img/', blank=True, null=True,
default='')
    Res_Email = models.CharField(max_length=100)
    Res_Password = models.CharField(max_length=50)
    Res_Phone = models.CharField(max_length=50)
    Ar_id = models.ForeignKey(Area, on_delete=models.CASCADE, null=True)
    Res_Address = models.CharField(max_length=250)
    Dish_id = models.ForeignKey(dish, on_delete=models.CASCADE, null=True)
    Res_MenuFile = models.FileField(upload_to='menu/', blank=True, null=True,
default='')
    Res_OpenAT = models.DateTimeField()
    Res_CloseAT = models.DateTimeField()

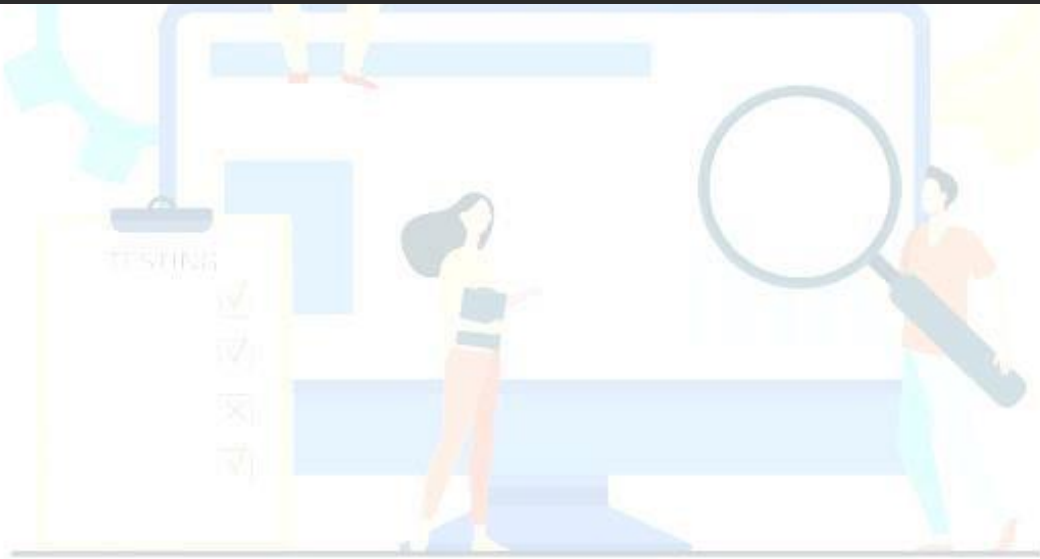
    def __str__(self):
```

```
        return self.Res_Name

class Contact_msg(models.Model):
    Con_id = models.AutoField(primary_key=True)
    Con_Name = models.CharField(max_length=50, blank=False, null=False)
    Con_Email = models.CharField(max_length=50, blank=False, null=False)
    Con_msg = models.CharField(max_length=255)

    def __str__(self):
        return self.Con_Name

class WebsiteInfo(models.Model):
    Web_id = models.AutoField(primary_key=True)
    Web_address = models.CharField(max_length=50)
    Web_email = models.CharField(max_length=50)
    Web_phone = models.CharField(max_length=50)
    Web_facebook_link = models.CharField(max_length=50)
    Web_instagram_link = models.CharField(max_length=50)
    Web_twitter_link = models.CharField(max_length=50)
    Web_whatsapp_link = models.CharField(max_length=50)
```



URL

```
"""
URL configuration for Ordering_System project.

The `urlpatterns` list routes URLs to views. For more information please see:
    https://docs.djangoproject.com/en/4.2/topics/http/urls/
Examples:
Function views
    1. Add an import:  from my_app import views
    2. Add a URL to urlpatterns:  path('', views.home, name='home')
Class-based views
    1. Add an import:  from other_app.views import Home
    2. Add a URL to urlpatterns:  path('', Home.as_view(), name='home')
Including another URLconf
    1. Import the include() function: from django.urls import include, path
    2. Add a URL to urlpatterns:  path('blog/', include('blog.urls'))
"""
from django.contrib import admin
from django.urls import path
from Ordering_System_app import views
from django.conf.urls.static import static
from django.conf import settings

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', views.index),
    path('shop', views.shop),
    path('contact', views.contact),
    path('shop-details', views.shop_details),
    path('shoping-cart', views.shoping_cart),
    path('checkout', views.checkout),
    path('restaurant', views.restaurant),
    path('dishes', views.dishes),
    path('dishesCategory/<int:id>', views.dishCategory),
    path('restaurantsCategory/<int:id>', views.resCategory),
    path('search_dish', views.searchbar_dish),
    path('search_restaurant', views.searchbar_res),
    path('restaurant-details/<int:id>', views.restaurant_details),
    path('user_login', views.user_login),

    # For CRUD Operations

    # path('contact', views.Create),

]+ static(settings.MEDIA_URL, document_root= settings.MEDIA_ROOT)
```


Admin

```
from django.contrib import admin
from .models import Restaurant, dish, category, Contact_msg, WebsiteInfo, Area
# Register your models here.
admin.site.register(Restaurant)
admin.site.register(dish)
admin.site.register(category)
admin.site.register(Contact_msg)
admin.site.register(WebsiteInfo)
admin.site.register(Area)
admin.site.site_header = "OFOSA Admin"
admin.site.site_title = "Online Food Ordering System"
admin.site.index_title = "Online Food Ordering System"
```

