



الجمهورية العربية السورية

جامعة اللاذقية

كلية الهندسة المعلوماتية

قسم البرمجيات و نظم المعلومات

Pizza Shop Management System

تقديم :

نغم حيدر

محمود حلواني

رياض حمدان

محمود الخنسا

محسن حبيب

بإشراف :

م.حمزة دواي

م.زينب محفوظ

د.غيث بلال

العام الدراسي 2024_2025

تحليل نظام متجر لبيع البيتزا :pizza shop Management System

نبدأ الان مشروعا بناشاء تطبيق api لنظام متجر البيتزا الذي تحدثنا عنه مع توثيق كامل له :

١- اعداد البيئة والمشروع:

إنشاء البيئة الافتراضية وتفعيلها:

```
python -m venv virtual
```

```
virtual \Scripts\activate
```

تثبيت الحزم المطلوبة:

```
pip install django
```

```
pip install djangorestframework
```

```
pip install django-filter
```

٢- انشاء المشروع والتطبيق :

```
django-admin startproject project
```

```
cd project
```

```
python manage.py migrate
```

```
python manage.py startapp Task
```

٣- تكوين الاعدادات : Task/settings.py

```
INSTALLED_APPS = [  
    . . .  
    'rest_framework',  
    'Task',  
    'django_filters',  
    . . .  
,  
    REST_FRAMEWORK = {  
        'DEFAULT_FILTER_BACKENDS':  
        ['django_filters.rest_framework.DjangoFilterBackend']  
    }
```

٤- نماذج البيانات Task/models.py :

```
from django.db import models

class Task(models.Model):
    name = models.CharField(max_length=200)
    size = models.CharField(max_length=100, choices=[
        ('S', 'Small'),
        ('M', 'Medium'),
        ('L', 'Large'),
        ('XL', 'Extra Large'),
    ])
    price = models.DecimalField(max_digits=5, decimal_places=2)
    def __str__(self) :
        return self.name
```

٥- التسلسل Task/serializers.py :

```
from rest_framework import serializers
from .models import Task
class TaskSerializer(serializers.ModelSerializer):
    class Meta:
        model = Task
        fields = '__all__'
```

٦- واجهات العرض Task/view.py :

```
from rest_framework import viewsets
from .models import Task
from .serializers import TaskSerializer
from django_filters.rest_framework import DjangoFilterBackend
class TaskViewSet(viewsets.ModelViewSet):
    queryset = Task.objects.all()
    serializer_class = TaskSerializer
    filter_backends = [DjangoFilterBackend]
    filterset_fields = ["size"]
    def perform_create(self, serializer):
        serializer.save(customer=self.request.user)
```

```
def get_queryset(self):  
    return Task.objects.filter(name=self.request.user)
```

٧-المسارات project/urls.py :

```
from django.contrib import admin  
from django.urls import path, include  
from Task.views import TaskViewSet  
from rest_framework.routers import DefaultRouter  
router = DefaultRouter()  
router.register('orders', TaskViewSet)  
urlpatterns = [  
    path('api/', include(router.urls)),  
    path('api-auth/', include('rest_framework.urls')),]
```

٨-نقوم بالترحيل وتشغيل الخادم :

python manage.py makemigrations

python manage.py migrate

python manage.py createsuperuser

python manage.py runserver