

Test 5.2 Implement Your Django Project

Name: Dhaneshwari Kolhe

Date : 27-06-24

- Model Creation

```
• from django.db import models
•
• # Create your models here.
•
• class Department(models.Model):
•     dept_name = models.CharField(max_length=100)
•
•     def __str__(self) -> str:
•         return self.dept_name
•
• class AppointmentStatus(models.Model):
•     status = models.CharField(max_length=50)
•     # active, canceled, expired, complete
•     def __str__(self) -> str:
•         return self.status
•
•
• class TimeSlot(models.Model):
•     timeslot = models.CharField(max_length=50)
•     #11am-12noon, 1pm-2pm etc
•
•     def __str__(self) -> str:
•         return self.timeslot
•
•
• class Doctor(models.Model):
•     first_name = models.CharField(max_length=50)
•     last_name = models.CharField(max_length=50)
•     Dept_name = models.ForeignKey(Department, on_delete=models.CASCADE)
•     email = models.EmailField(unique=True)
•     contact_number = models.BigIntegerField()
•
•     def __str__(self) -> str:
•         return f"Dr. {self.first_name} {self.last_name}"
•
•
• class Patient(models.Model):
```

```

• first_name = models.CharField(max_length=50)
• last_name = models.CharField(max_length=50)
• email = models.EmailField(unique=True,blank=True,null=True)
• contact_number = models.BigIntegerField()
•
• def __str__(self) -> str:
•     return f"{self.first_name} {self.last_name}"
•
•
• class DoctorAvailability(models.Model):
•     day_of_week = (
•         ("Mon", "Monday"),
•         ("Tue", "Tuesday"),
•         ("Wed", "Wednesday"),
•         ("Thr", "Thursday"),
•         ("Fri", "Friday"),
•         ("Sat", "Saturday"),
•         ("Sun", "Sunday")
•     )
•     #
•     Doctor = models.ForeignKey(Doctor,on_delete=models.CASCADE)
•     day_of_wk =
models.CharField('Type',max_length=10,choices=day_of_week,default='Mon'
)
•     timeslot = models.ManyToManyField(TimeSlot)
•
•     # def __str__(self) -> str:
•     #     return self.Doctor
•
• class Appointment(models.Model):
•     Doctor = models.ForeignKey(Doctor,on_delete=models.CASCADE)
•     patient = models.ForeignKey(Patient,on_delete=models.CASCADE)
•     appointment_date = models.DateField(null=False)
•     timeslot = models.ForeignKey(TimeSlot,on_delete=models.CASCADE)
•     appointment_status =
models.ForeignKey(AppointmentStatus,on_delete=models.CASCADE)
•
•     # validation required with doctor timeslot day of wk and status will be
updated automatically
•     # contact number validation also required

```

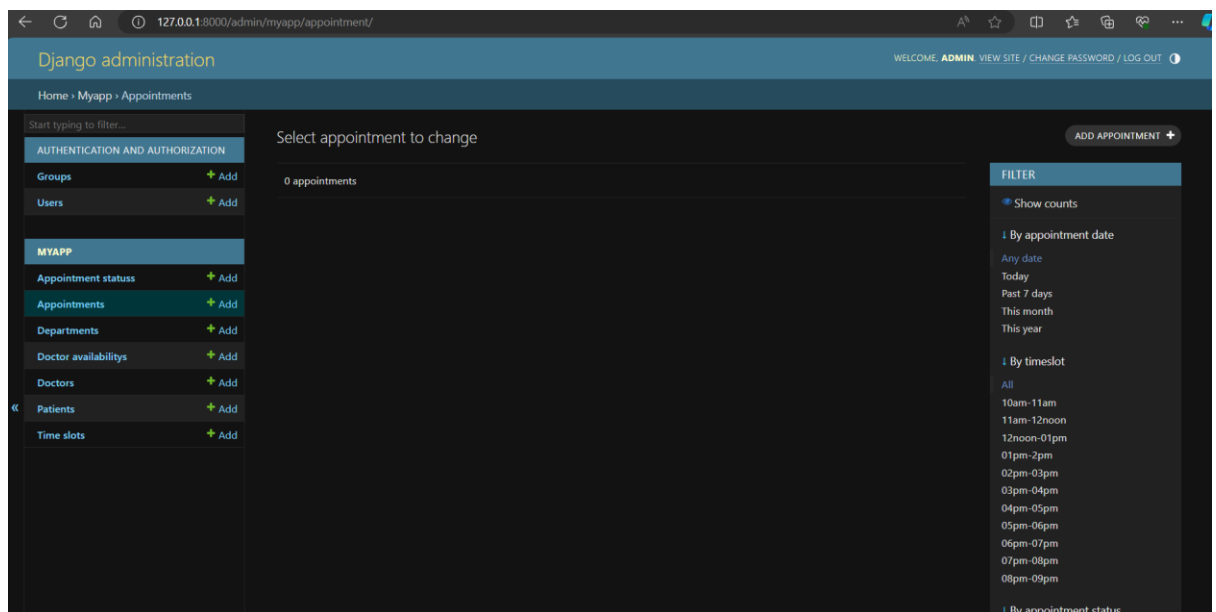
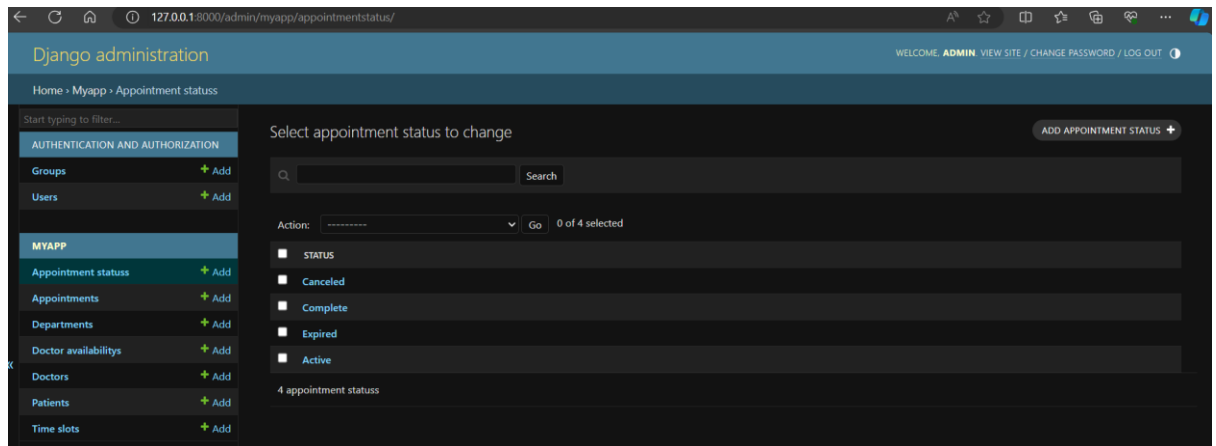
- DataBase Creation
SQLite through
python manage.py makemigrations
pythom manage.py migrate

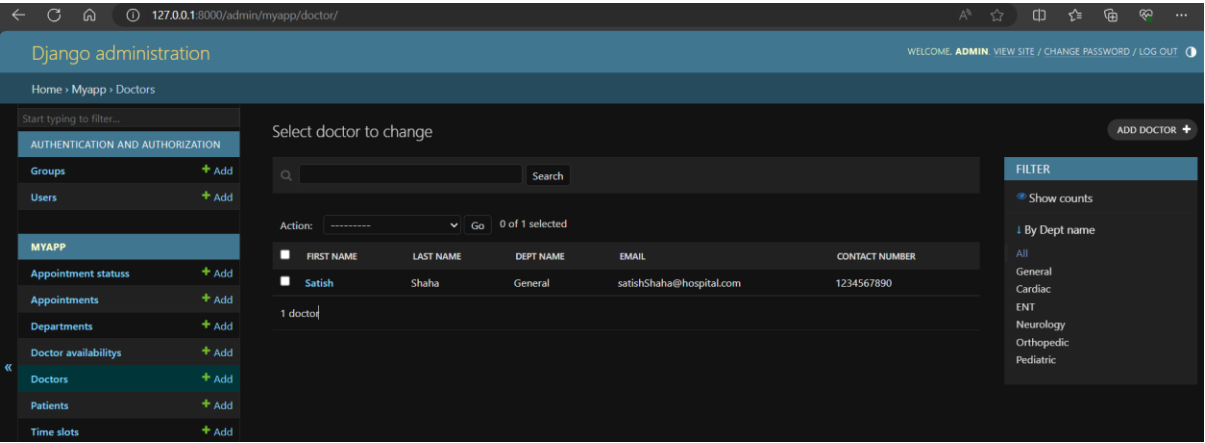
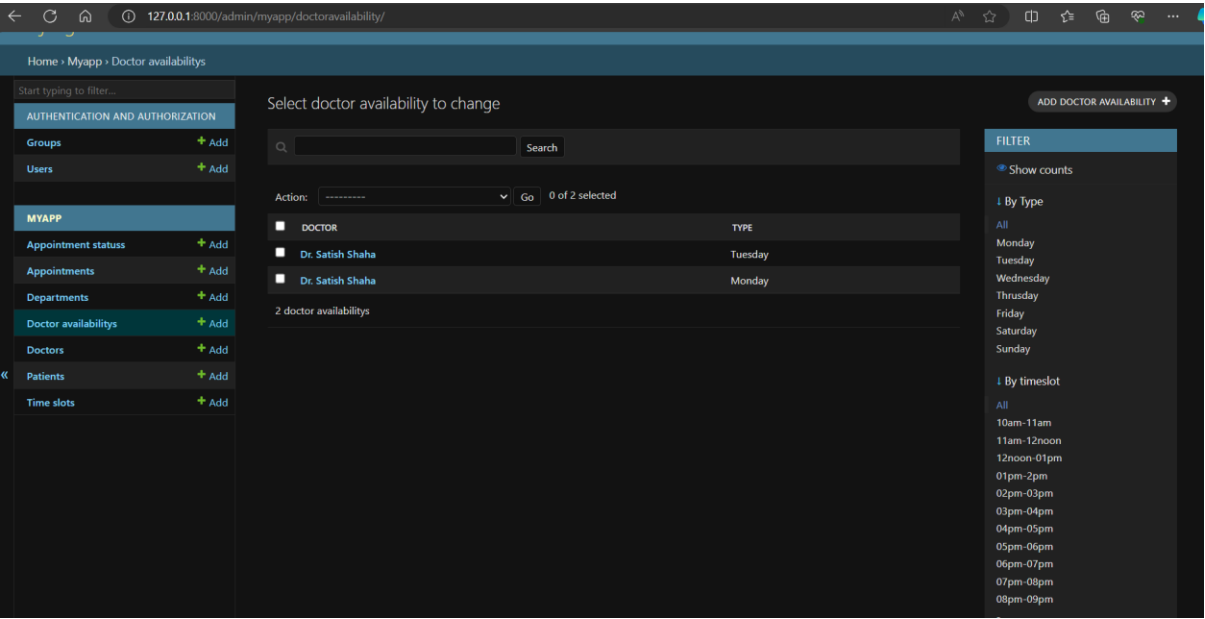
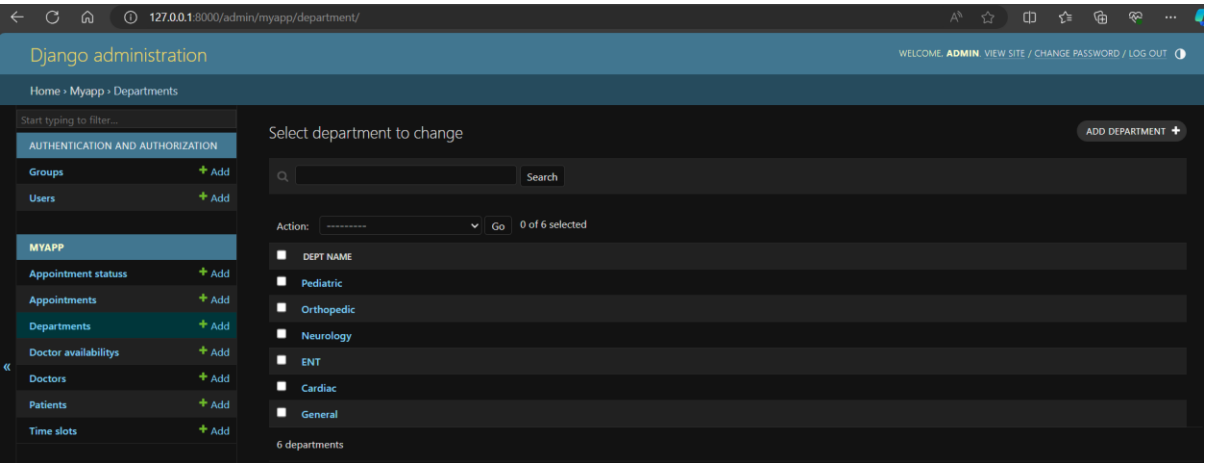
- Customize Admin panel

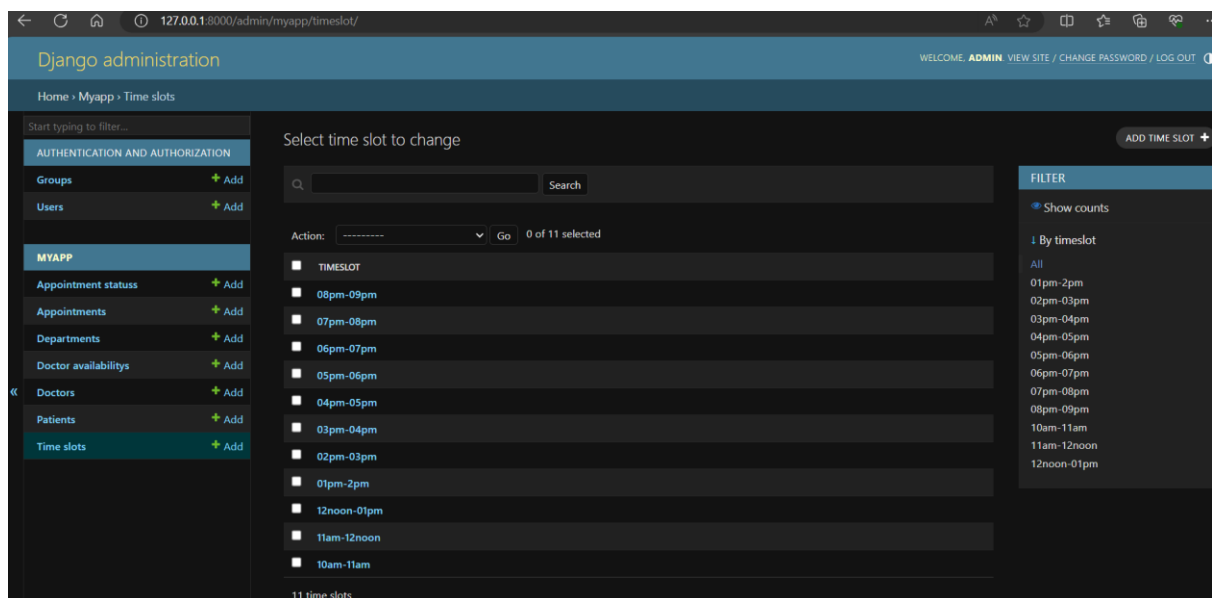
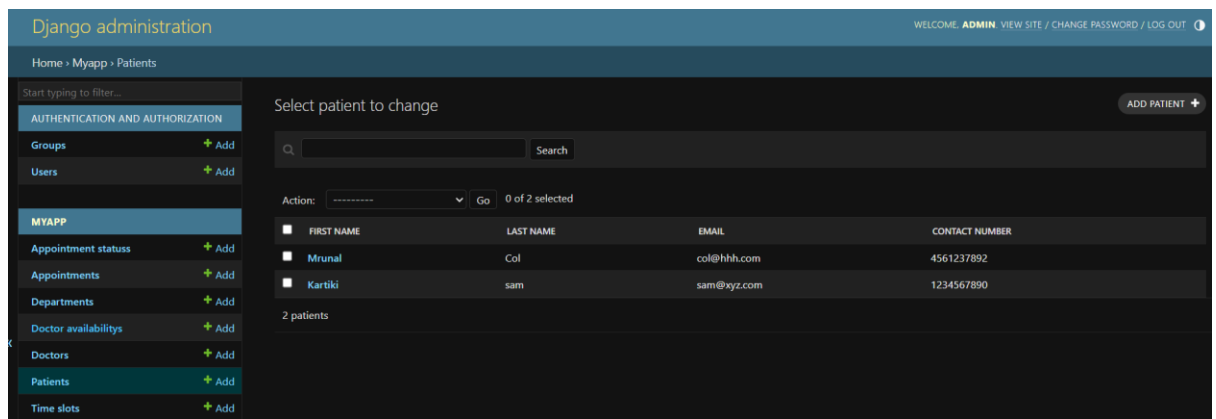
```
• from django.contrib import admin
• from .models import *
•
• @admin.register(Department)
• class DepartmentAdmin(admin.ModelAdmin):
•     list_display = ('dept_name',)
•     search_fields = ('dept_name',)
•
• @admin.register(AppointmentStatus)
• class AppointmentStatusAdmin(admin.ModelAdmin):
•     list_display = ('status',)
•     search_fields = ('status',)
•
• @admin.register(TimeSlot)
• class TimeSlotAdmin(admin.ModelAdmin):
•     list_display = ('timeslot',)
•     search_fields = ('timeslot',)
•     list_filter = ('timeslot',)
•
• @admin.register(Doctor)
• class DoctorAdmin(admin.ModelAdmin):
•     list_display = ('first_name',
• 'last_name', 'Dept_name', 'email', 'contact_number')
•     search_fields = ('first_name', 'last_name', 'contact_number')
•     list_filter = ('Dept_name',)
•
• @admin.register(Patient)
• class PatientAdmin(admin.ModelAdmin):
•     list_display = ('first_name', 'last_name', 'email', 'contact_number')
•     search_fields = ('first_name', 'last_name', 'contact_number')
•
• @admin.register(DoctorAvailability)
• class DoctorAvailabilityAdmin(admin.ModelAdmin):
•     list_display = ['Doctor', 'day_of_wk']
•     search_fields = ['Doctor', 'day_of_wk']
•     list_filter = ['Doctor', 'day_of_wk', 'timeslot']
•
• @admin.register(Appointment)
• class AppointmentAdmin(admin.ModelAdmin):
•     list_display =
• ['Doctor', 'patient', 'appointment_date', 'timeslot', 'appointment_status']
•     list_filter =
• ['Doctor', 'appointment_date', 'timeslot', 'appointment_status']
•
• # admin.site.register(Department)
• # admin.site.register(AppointmentStatus)
• # admin.site.register(TimeSlot)
```

- # admin.site.register(Doctor)
- # admin.site.register(Patient)
- # admin.site.register(DoctorAvailability)
- # admin.site.register(Appointment)
-

• Submission







Customize admin panel for all seven models