Backend (Django Rest Framework):

1. Install required packages:
   ```bash
   pip install django djangorestframework django-cors-headers
   ```

2. Set up Django project and app:
   ```bash
   django-admin startproject myproject
   cd myproject
   python manage.py startapp myapp
   ```

3. Configure settings (`myproject/settings.py`):
   ```python
   INSTALLED_APPS = [
       # ...
       'rest_framework',
       'myapp',
       'corsheaders',
   ]
   ```

```python
MIDDLEWARE = [
    # ...

'corsheaders.middleware.CorsMiddleware',
    ]


    # Allow all origins for simplicity (customize
for production)
    CORS_ALLOW_ALL_ORIGINS = True
```

4. Define models (`myapp/models.py`):
```python
from django.db import models
from django.contrib.auth.models import User


class Doctor(models.Model):
    user = models.OneToOneField(User,
on_delete=models.CASCADE)


class Patient(models.Model):
    user = models.OneToOneField(User,
on_delete=models.CASCADE)
    doctor = models.ForeignKey(Doctor,
on_delete=models.CASCADE, null=True,
blank=True)


class Illness(models.Model):
```

```python
class Illness(models.Model):
    patient = models.ForeignKey(Patient,
on_delete=models.CASCADE)
    doctor = models.ForeignKey(Doctor,
on_delete=models.CASCADE)
    description = models.TextField()
```

5. Create serializers (`myapp/serializers.py`):
```python
from rest_framework import serializers
from .models import Doctor, Patient, Illness

class
DoctorSerializer(serializers.ModelSerializer):
    class Meta:
        model = Doctor
        fields = '__all__'

class
PatientSerializer(serializers.ModelSerializer):
    class Meta:
        model = Patient
        fields = '__all__'

class
IllnessSerializer(serializers.ModelSerializer):
    class Meta:
        model = Illness
```

6. Implement views (`myapp/views.py`):

```python
from rest_framework import generics
from .models import Doctor, Patient, Illness
from .serializers import DoctorSerializer, PatientSerializer, IllnessSerializer

class DoctorList(generics.ListCreateAPIView):
    queryset = Doctor.objects.all()
    serializer_class = DoctorSerializer

class PatientList(generics.ListCreateAPIView):
    queryset = Patient.objects.all()
    serializer_class = PatientSerializer

class IllnessList(generics.ListCreateAPIView):
    queryset = Illness.objects.all()
    serializer_class = IllnessSerializer
```

7. Set up URLs (`myapp/urls.py`):

```python
from django.urls import path
from .views import DoctorList, PatientList, IllnessList
```

```python
   urlpatterns = [
      path('doctors/', DoctorList.as_view(),
name='doctor-list'),
      path('patients/', PatientList.as_view(),
name='patient-list'),
      path('illnesses/', IllnessList.as_view(),
name='illness-list'),
   ]
```

8. Configure main `urls.py`:
```python
   from django.contrib import admin
   from django.urls import path, include

   urlpatterns = [
      path('admin/', admin.site.urls),
      path('api/', include('myapp.urls')),
   ]
```

Run migrations and start the server:

```
   python manage.py makemigrations
   python manage.py migrate
   python manage.py runserver
```