

# Complete Postman Testing Guide - Step by Step

## 🔧 Initial Postman Setup

### Step 1: Create Environment Variables

1. Go to **Environments** → **Create Environment**

2. Name it: **Healthcare API**

3. Add these variables:

Variable	Initial Value	Current Value
base_url	http://127.0.0.1:8000/api	http://127.0.0.1:8000/api
access_token	(leave empty)	(will be filled after login)
refresh_token	(leave empty)	(will be filled after login)
patient_id	(leave empty)	(will be filled after creating patient)
doctor_id	(leave empty)	(will be filled after creating doctor)
mapping_id	(leave empty)	(will be filled after creating mapping)

### Step 2: Select Environment

- In top-right corner, select your **Healthcare API** environment

## ✍ 1. AUTHENTICATION APIs

### 1.1 Register New User

#### Postman Configuration:

- Method:** **POST**
- URL:** **{{base\_url}}/auth/register/**
- Headers:**

Content-Type: application/json

- Body → raw → JSON:**

```
json

{
    "username": "johndoe",
    "email": "john.doe@email.com",
    "first_name": "John",
    "last_name": "Doe",
    "password": "securepass123",
    "password_confirm": "securepass123"
}
```

### Expected Response (201 Created):

```
json

{
    "message": "User registered successfully",
    "user": {
        "id": 1,
        "username": "johndoe",
        "email": "john.doe@email.com",
        "first_name": "John",
        "last_name": "Doe"
    },
    "tokens": {
        "refresh": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9...",
        "access": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9..."
    }
}
```

### Post-Request Action:

Copy the **access** token and update your environment variable **access\_token**

---

## 1.2 User Login

### Postman Configuration:

- **Method:** `POST`
- **URL:** `{{base_url}}/auth/login/`
- **Headers:**

Content-Type: application/json

- **Body → raw → JSON:**

```
json

{
    "username": "johndoe",
    "password": "securepass123"
}
```

### Expected Response (200 OK):

```
json

{
    "message": "Login successful",
    "user": {
        "id": 1,
        "username": "johndoe",
        "email": "john.doe@email.com",
        "first_name": "John",
        "last_name": "Doe"
    },
    "tokens": {
        "refresh": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9...",
        "access": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9..."
    }
}
```

### Post-Request Action:

Update both `access_token` and `refresh_token` in your environment

---

## 1.3 Refresh Token

### Postman Configuration:

- **Method:** `POST`
- **URL:** `{{base_url}}/auth/token/refresh/`
- **Headers:**

Content-Type: application/json

- **Body → raw → JSON:**

```
json
{
  ...
  "refresh": "{{refresh_token}}"
}
```

### Expected Response (200 OK):

```
json
{
  ...
  "access": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9...",
  "refresh": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9..."
}
```

---

## 2. PATIENT MANAGEMENT APIs

### 2.1 Create Patient

#### Postman Configuration:

- **Method:** POST
- **URL:** {{base\_url}}/patients/
- **Headers:**

Content-Type: application/json  
Authorization: Bearer {{access\_token}}

- **Body → raw → JSON:**

json

```
{  
    "name": "Alice Johnson",  
    "age": 28,  
    "gender": "F",  
    "phone": "1234567890",  
    "email": "alice.johnson@email.com",  
    "address": "123 Main Street, New York, NY 10001",  
    "medical_history": "No known allergies. Underwent appendectomy in 2019. Regular checkups s  
}  

```

### Expected Response (201 Created):

json

```
{  
    "id": 1,  
    "name": "Alice Johnson",  
    "age": 28,  
    "gender": "F",  
    "phone": "1234567890",  
    "email": "alice.johnson@email.com",  
    "address": "123 Main Street, New York, NY 10001",  
    "medical_history": "No known allergies. Underwent appendectomy in 2019. Regular checkups s  
    "created_by": "johndoe",  
    "created_at": "2024-06-08T10:30:00Z",  
    "updated_at": "2024-06-08T10:30:00Z"  
}  

```

### Post-Request Action:

Save the **id** (1) in your environment as **patient\_id**

---

## 2.2 Get All Patients (User's Own)

### Postman Configuration:

- **Method:** **GET**
- **URL:** **{{{base\_url}}}/patients/**
- **Headers:**

Authorization: Bearer {{access\_token}}

- **Body:** None (GET request)

#### Expected Response (200 OK):

json

```
{  
    "count": 1,  
    "next": null,  
    "previous": null,  
    "results": [  
        {  
            "id": 1,  
            "name": "Alice Johnson",  
            "age": 28,  
            "gender": "F",  
            "phone": "1234567890",  
            "email": "alice.johnson@email.com",  
            "address": "123 Main Street, New York, NY 10001",  
            "medical_history": "No known allergies...",  
            "created_by": "johndoe",  
            "created_at": "2024-06-08T10:30:00Z",  
            "updated_at": "2024-06-08T10:30:00Z"  
        }  
    ]  
}
```

---

### 2.3 Get Specific Patient

#### Postman Configuration:

- **Method:** `GET`
- **URL:** `{{base_url}}/patients/{{patient_id}}/`
- **Headers:**

Authorization: Bearer {{access\_token}}

- **Body:** None

#### Expected Response (200 OK):

```
json

{
    "id": 1,
    "name": "Alice Johnson",
    "age": 28,
    "gender": "F",
    "phone": "1234567890",
    "email": "alice.johnson@email.com",
    "address": "123 Main Street, New York, NY 10001",
    "medical_history": "No known allergies...",
    "created_by": "johndoe",
    "created_at": "2024-06-08T10:30:00Z",
    "updated_at": "2024-06-08T10:30:00Z"
}
```

---

## 2.4 Update Patient

### Postman Configuration:

- **Method:** `PUT`
- **URL:** `{{base_url}}/patients/{{patient_id}}/`
- **Headers:**

`Content-Type: application/json`  
`Authorization: Bearer {{access_token}}`

- **Body → raw → JSON:**

```
json

{
    "name": "Alice Johnson Smith",
    "age": 29,
    "gender": "F",
    "phone": "1234567890",
    "email": "alice.smith@email.com",
    "address": "456 Oak Avenue, New York, NY 10002",
    "medical_history": "No known allergies. Underwent appendectomy in 2019. Regular checkups s"
}
```

### Expected Response (200 OK):

json

```
{  
    "id": 1,  
    "name": "Alice Johnson Smith",  
    "age": 29,  
    "gender": "F",  
    "phone": "1234567890",  
    "email": "alice.smith@email.com",  
    "address": "456 Oak Avenue, New York, NY 10002",  
    "medical_history": "No known allergies. Underwent appendectomy in 2019. Regular checkups scheduled.",  
    "created_by": "johndoe",  
    "created_at": "2024-06-08T10:30:00Z",  
    "updated_at": "2024-06-08T11:15:00Z"  
}
```

## 2.5 Delete Patient

### Postman Configuration:

- **Method:** `DELETE`
- **URL:** `{{base_url}}/patients/{{patient_id}}/`
- **Headers:**

`Authorization: Bearer {{access_token}}`

- **Body:** None

### Expected Response (204 No Content):

No response body (successful deletion)



## 3. DOCTOR MANAGEMENT APIs

### 3.1 Create Doctor

### Postman Configuration:

- **Method:** `POST`
- **URL:** `{{base_url}}/doctors/`

- **Headers:**

```
Content-Type: application/json
Authorization: Bearer {{access_token}}
```

- **Body → raw → JSON:**

json

```
{
  "name": "Sarah Wilson",
  "specialization": "Cardiology",
  "phone": "9876543210",
  "email": "dr.wilson@hospital.com",
  "experience_years": 12,
  "qualification": "MD, FACC, Fellowship in Interventional Cardiology",
  "consultation_fee": 300.00,
  "is_available": true
}
```

### Expected Response (201 Created):

json

```
{
  "id": 1,
  "name": "Sarah Wilson",
  "specialization": "Cardiology",
  "phone": "9876543210",
  "email": "dr.wilson@hospital.com",
  "experience_years": 12,
  "qualification": "MD, FACC, Fellowship in Interventional Cardiology",
  "consultation_fee": "300.00",
  "is_available": true,
  "patient_count": 0,
  "created_at": "2024-06-08T11:30:00Z",
  "updated_at": "2024-06-08T11:30:00Z"
}
```

### Post-Request Action:

Save the **[id]** (1) in your environment as **doctor\_id**

---

## 3.2 Create Another Doctor (for testing)

### Postman Configuration:

- **Method:** POST
- **URL:** {{base\_url}}/doctors/
- **Headers:**

Content-Type: application/json  
Authorization: Bearer {{access\_token}}

- **Body → raw → JSON:**

```
json

{
  "name": "Michael Chen",
  "specialization": "Neurology",
  "phone": "5555551234",
  "email": "dr.chen@hospital.com",
  "experience_years": 8,
  "qualification": "MD, Neurology Residency, Fellowship in Stroke Medicine",
  "consultation_fee": 250.00,
  "is_available": false
}
```

---

## 3.3 Get All Doctors

### Postman Configuration:

- **Method:** GET
- **URL:** {{base\_url}}/doctors/
- **Headers:**

Authorization: Bearer {{access\_token}}

- **Body:** None

### Expected Response (200 OK):

json

```
{  
    "count": 2,  
    "next": null,  
    "previous": null,  
    "results": [  
        {  
            "id": 2,  
            "name": "Michael Chen",  
            "specialization": "Neurology",  
            "phone": "5555551234",  
            "email": "dr.chen@hospital.com",  
            "experience_years": 8,  
            "qualification": "MD, Neurology Residency, Fellowship in Stroke Medicine",  
            "consultation_fee": "250.00",  
            "is_available": false,  
            "patient_count": 0,  
            "created_at": "2024-06-08T11:35:00Z",  
            "updated_at": "2024-06-08T11:35:00Z"  
        },  
        {  
            "id": 1,  
            "name": "Sarah Wilson",  
            "specialization": "Cardiology",  
            "phone": "9876543210",  
            "email": "dr.wilson@hospital.com",  
            "experience_years": 12,  
            "qualification": "MD, FACC, Fellowship in Interventional Cardiology",  
            "consultation_fee": "300.00",  
            "is_available": true,  
            "patient_count": 0,  
            "created_at": "2024-06-08T11:30:00Z",  
            "updated_at": "2024-06-08T11:30:00Z"  
        }  
    ]  
}
```

---

### 3.4 Filter Doctors by Specialization

**Postman Configuration:**

- **Method:**

- **URL:** {{base\_url}}/doctors/?specialization=cardiology

- **Headers:**

Authorization: Bearer {{access\_token}}

- **Body:** None

### Expected Response (200 OK):

json

```
{
  "count": 1,
  "next": null,
  "previous": null,
  "results": [
    {
      "id": 1,
      "name": "Sarah Wilson",
      "specialization": "Cardiology",
      "phone": "9876543210",
      "email": "dr.wilson@hospital.com",
      "experience_years": 12,
      "qualification": "MD, FACC, Fellowship in Interventional Cardiology",
      "consultation_fee": "300.00",
      "is_available": true,
      "patient_count": 0,
      "created_at": "2024-06-08T11:30:00Z",
      "updated_at": "2024-06-08T11:30:00Z"
    }
  ]
}
```

---

## 3.5 Filter Available Doctors

### Postman Configuration:

- **Method:** GET

- **URL:** {{base\_url}}/doctors/?available=true

- **Headers:**

Authorization: Bearer {{access\_token}}

- **Body:** None
- 

### 3.6 Get Specific Doctor

#### Postman Configuration:

- **Method:** `GET`
- **URL:** `{{base_url}}/doctors/{{doctor_id}}/`
- **Headers:**

`Authorization: Bearer {{access_token}}`

- **Body:** None
- 

### 3.7 Update Doctor

#### Postman Configuration:

- **Method:** `PUT`
- **URL:** `{{base_url}}/doctors/{{doctor_id}}/`
- **Headers:**

`Content-Type: application/json`  
`Authorization: Bearer {{access_token}}`

- **Body → raw → JSON:**

```
json
{
  "name": "Sarah Wilson",
  "specialization": "Cardiology",
  "phone": "9876543210",
  "email": "dr.wilson@hospital.com",
  "experience_years": 13,
  "qualification": "MD, FACC, Fellowship in Interventional Cardiology, Board Certified",
  "consultation_fee": 320.00,
  "is_available": true
}
```

---

### 3.8 Delete Doctor

## **Postman Configuration:**

- **Method:** `DELETE`
- **URL:** `{{base_url}}/doctors/{{doctor_id}}/`
- **Headers:**

`Authorization: Bearer {{access_token}}`

- **Body:** None

## **Expected Response (204 No Content):**

No response body (successful deletion)

---

## **4. PATIENT-DOCTOR MAPPING APIs**

### **4.1 Assign Doctor to Patient**

#### **Prerequisites:**

- Create a new patient (follow step 2.1)
- Ensure you have a doctor available

## **Postman Configuration:**

- **Method:** `POST`
- **URL:** `{{base_url}}/mappings/`
- **Headers:**

`Content-Type: application/json`  
`Authorization: Bearer {{access_token}}`

- **Body → raw → JSON:**

json

```
{  
    "patient": 1,  
    "doctor": 1,  
    "notes": "Initial consultation scheduled for next Tuesday. Patient reported chest pain during exercise."}
```

## Expected Response (201 Created):

```
json

{
    "id": 1,
    "patient": 1,
    "doctor": 1,
    "patient_name": "Alice Johnson Smith",
    "doctor_name": "Sarah Wilson",
    "doctor_specialization": "Cardiology",
    "assigned_date": "2024-06-08T12:00:00Z",
    "notes": "Initial consultation scheduled for next Tuesday. Patient reported chest pain during exercise.",
    "is_active": true
}
```

## Post-Request Action:

Save the `id` (1) in your environment as `mapping_id`

---

## 4.2 Get All Patient-Doctor Mappings

### Postman Configuration:

- **Method:** `GET`
- **URL:** `{{base_url}}/mappings/`
- **Headers:**

`Authorization: Bearer {{access_token}}`

- **Body:** None

## Expected Response (200 OK):

```
json
```

```
{  
    "count": 1,  
    "next": null,  
    "previous": null,  
    "results": [  
        {  
            "id": 1,  
            "patient": 1,  
            "doctor": 1,  
            "patient_name": "Alice Johnson Smith",  
            "doctor_name": "Sarah Wilson",  
            "doctor_specialization": "Cardiology",  
            "assigned_date": "2024-06-08T12:00:00Z",  
            "notes": "Initial consultation scheduled for next Tuesday. Patient reported chest pain.",  
            "is_active": true  
        }  
    ]  
}
```

## 4.3 Get Doctors for Specific Patient

### Postman Configuration:

- **Method:** `GET`
- **URL:** `{{{base_url}}}/mappings/{{patient_id}}/`
- **Headers:**

```
Authorization: Bearer {{access_token}}
```

- **Body:** None

### Expected Response (200 OK):

```
json
```

```
{
    "patient": {
        "id": 1,
        "name": "Alice Johnson Smith",
        "age": 29,
        "gender": "F",
        "phone": "1234567890",
        "email": "alice.smith@email.com",
        "address": "456 Oak Avenue, New York, NY 10002",
        "medical_history": "No known allergies. Underwent appendectomy in 2019. Regular checkups maintained.",
        "created_by": "johndoe",
        "created_at": "2024-06-08T10:30:00Z",
        "updated_at": "2024-06-08T11:15:00Z"
    },
    "assigned_doctors": [
        {
            "id": 1,
            "doctor": {
                "id": 1,
                "name": "Sarah Wilson",
                "specialization": "Cardiology",
                "phone": "9876543210",
                "email": "dr.wilson@hospital.com",
                "experience_years": 13,
                "qualification": "MD, FACC, Fellowship in Interventional Cardiology, Board Certified",
                "consultation_fee": "320.00",
                "is_available": true,
                "patient_count": 1,
                "created_at": "2024-06-08T11:30:00Z",
                "updated_at": "2024-06-08T11:45:00Z"
            },
            "assigned_date": "2024-06-08T12:00:00Z",
            "notes": "Initial consultation scheduled for next Tuesday. Patient reported chest pain and shortness of breath.",
            "is_active": true
        }
    ]
}
```

#### 4.4 Remove Doctor from Patient (Soft Delete)

## Postman Configuration:

- **Method:** `DELETE`
- **URL:** `{{base_url}}/mappings/{{mapping_id}}/`
- **Headers:**

`Authorization: Bearer {{access_token}}`

- **Body:** None

## Expected Response (200 OK):

```
json

{
  "message": "Doctor successfully removed from patient"
}
```

---

## 💡 5. ERROR TESTING

### 5.1 Test Without Authentication

#### Postman Configuration:

- **Method:** `GET`
- **URL:** `{{base_url}}/patients/`
- **Headers:** (No Authorization header)

#### Expected Response (401 Unauthorized):

```
json

{
  ...
  "detail": "Authentication credentials were not provided."
}
```

---

### 5.2 Test With Invalid Token

#### Postman Configuration:

- **Method:** `GET`

- **URL:** {{base\_url}}/patients/

- **Headers:**

```
Authorization: Bearer invalid_token_here
```

#### Expected Response (401 Unauthorized):

```
json

{
  "detail": "Given token not valid for any token type",
  "code": "token_not_valid",
  "messages": [
    {
      "token_class": "AccessToken",
      "token_type": "access",
      "message": "Token is invalid or expired"
    }
  ]
}
```

---

### 5.3 Test Validation Errors

#### Postman Configuration:

- **Method:** POST
- **URL:** {{base\_url}}/patients/
- **Headers:**

```
Content-Type: application/json
Authorization: Bearer {{access_token}}
```

- **Body → raw → JSON (Missing required fields):**

```
json

{
  "age": 25,
  "gender": "F"
}
```

#### Expected Response (400 Bad Request):

json

```
{  
... "name": ["This field is required."  
}
```

---

## 6. TESTING CHECKLIST

### Complete Testing Sequence:

#### Phase 1: Authentication

- Register new user
- Login with credentials
- Test token refresh
- Test invalid credentials

#### Phase 2: Patients

- Create patient
- List all patients
- Get specific patient
- Update patient
- Delete patient
- Test permissions (access other user's patients)

#### Phase 3: Doctors

- Create doctor
- List all doctors
- Filter by specialization
- Filter by availability
- Get specific doctor
- Update doctor
- Delete doctor

#### Phase 4: Mappings

- Assign doctor to patient
- List all mappings
- Get patient's doctors
- Remove assignment

Test duplicate assignment prevention

## Phase 5: Error Handling

- Test without authentication
  - Test with invalid token
  - Test validation errors
  - Test permission errors
- 

### Pro Tips for Testing:

1. **Save tokens in environment variables** - Don't copy-paste each time
2. **Test in sequence** - Create data before trying to update/delete
3. **Check response status codes** - 200/201 for success, 400/401/403/404 for errors
4. **Test edge cases** - Empty fields, invalid data, wrong IDs
5. **Use different users** - Test user isolation by creating multiple accounts
6. **Monitor Django logs** - Check your Django console for detailed error info

This guide covers every endpoint with exact configurations. Follow the sequence for best results! 