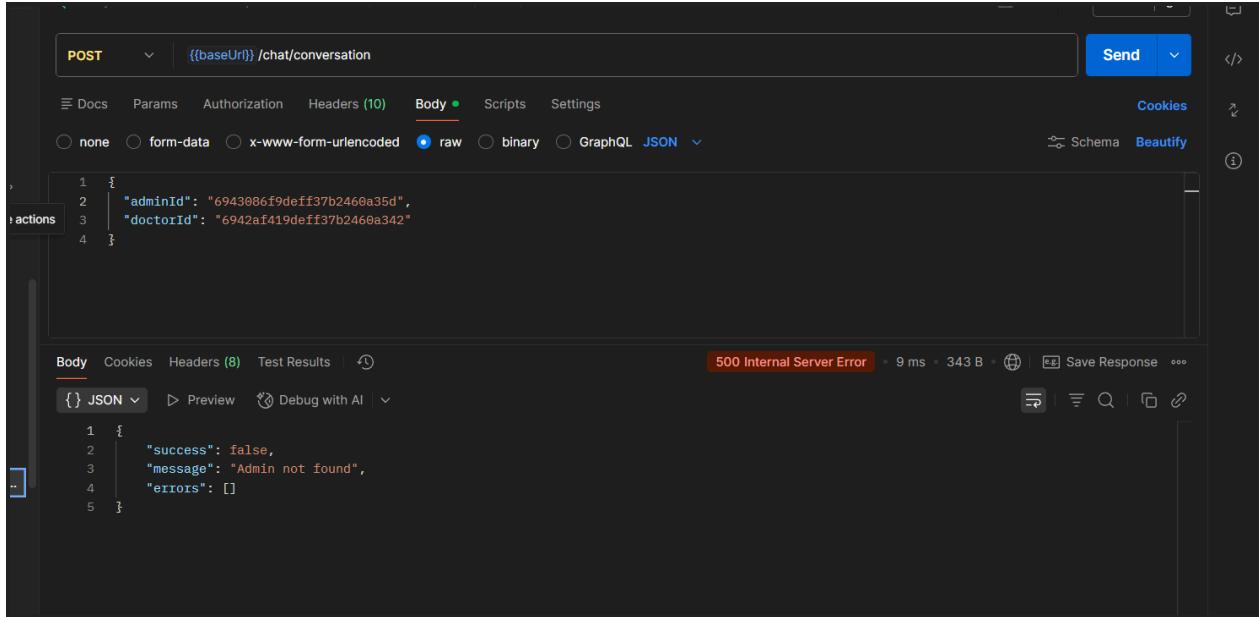


Docker Backend

Post: {{baseUrl}}/chat/conversation



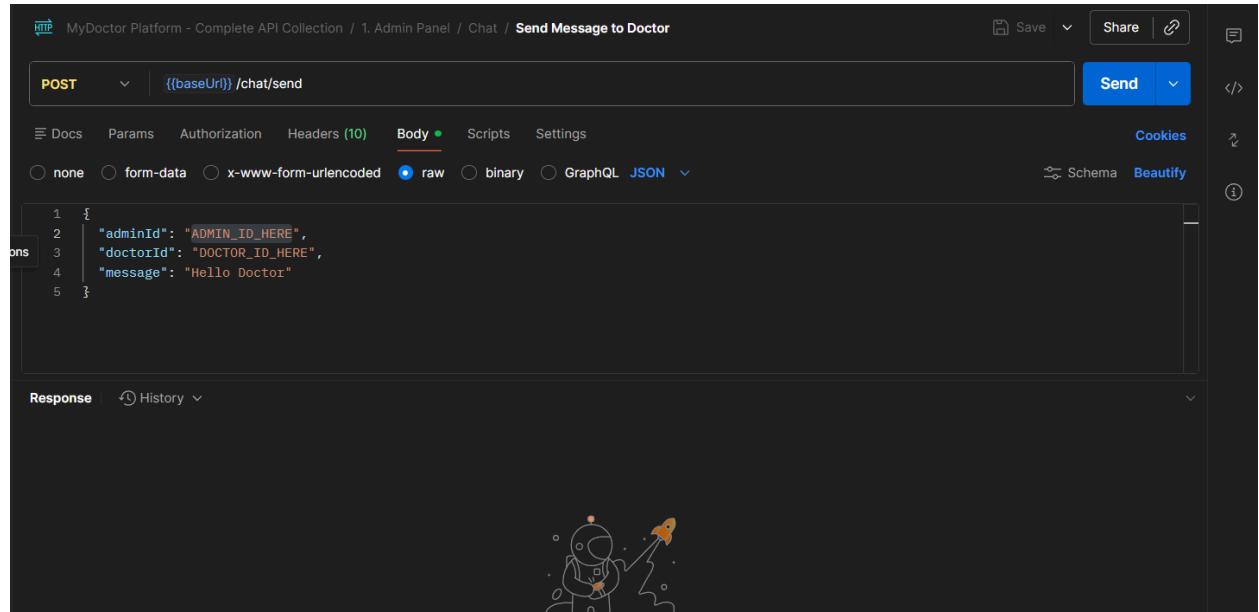
The screenshot shows a POST request to `"/chat/conversation"`. The request body is raw JSON:

```
1 {
2     "adminId": "6943086f9deff37b2460a35d",
3     "doctorId": "6942af419deff37b2460a342"
4 }
```

The response status is `500 Internal Server Error` with a duration of `9 ms` and a size of `343 B`. The response body is:

```
{ } JSON > Preview ⚡ Debug with AI
1 {
2     "success": false,
3     "message": "Admin not found",
4     "errors": []
5 }
```

This route should not take the admin id as it fetch from the token and also incase of not finding return 404/ rather than 500.



The screenshot shows a POST request to `"/chat/send"`. The request body is raw JSON:

```
1 {
2     "adminId": "ADMIN_ID_HERE",
3     "doctorId": "DOCTOR_ID_HERE",
4     "message": "Hello Doctor"
5 }
```

The response status is `500 Internal Server Error` with a duration of `9 ms` and a size of `343 B`. The response body is:

```
{ } JSON > Preview ⚡ Debug with AI
1 {
2     "success": false,
3     "message": "Admin not found",
4     "errors": []
5 }
```

Same as above.

`{{baseUrl}}/doctor/profile`

The screenshot shows the Postman application interface. At the top, there are several tabs: `Get System Activity`, `Get List All Transactions`, `POST Register Doctor`, and `PUT Update Profile`. The `PUT Update Profile` tab is active. Below the tabs, the URL is set to `HTTP MyDoctor Platform - Complete API Collection / 2. Doctor Panel / Profile Management / Update Profile`. The method is set to `PUT` and the endpoint is `PUT {{baseUrl}} /doctor/profile`. The `Body` tab is selected, showing the raw JSON payload:

```
1 {
2   "title": "Senior Cardiologist",
3   "biography": "Experienced doctor",
4   "specializationId": "SPEC_ID_HERE",
5   "experienceYears": 15
6 }
```

Below the body, the response section shows a `400 Bad Request` status with a timestamp of `16 ms` and a size of `368 B`. The response body is:

```
1 {
2   "success": false,
3   "message": "Invalid ID format",
4   "errors": [
5     {
6       "message": "Invalid resource ID"
7     }
8   ]
9 }
```

Kindly confirm the body type.

MyDoctor Platform - Complete API Collection / 2. Doctor Panel / Profile Management / Get My Profile

GET {{baseUrl}}/doctor/profile/:id

Params: Authorization, Headers (8), Body, Scripts, Settings

Query Params:

Key	Value	Description	Bulk Edit
Key	Value	Description	

Path Variables:

Key	Value	Description	Bulk Edit
id	Value	Description	

Body: Cookies, Headers (8), Test Results, Debug with AI

400 Bad Request • 11 ms • 368 B • Save Response

```

1 {
2   "success": false,
3   "message": "Invalid ID format",
4   "errors": [
5     {
6       "message": "Invalid resource ID"
7     }
8   ]
9 }

```

`{{baseUrl}}/doctor/profile/:id`

This route should not take the id - as it will send from the token

Search collections: Products, Pharmacies, Reviews, Transactions, Notifications, Announcements, Chat, Blog, System Activity, 2. Doctor Panel, Profile Management, Dashboard, Subscriptions, Buy Subscription Plan, Get My Subscription, Availability, Weekly Schedule, Appointments.

MyDoctor Platform - Complete API Collection / 2. Doctor Panel / Subscriptions / Buy Subscription Plan

POST {{baseUrl}}/doctor/buy-subscription

Params: Authorization, Headers (10), Body, Scripts, Settings

Body: none, form-data, x-www-form-urlencoded, raw (selected), binary, GraphQL, JSON

```

1 {
2   "planId": "69430af19deff37b2460a395"
3 }

```

Body: Cookies, Headers (8), Test Results, Visualize

200 OK • 19 ms • 1.29 KB • Save Response

```

47 },
48   "subscriptionPlan": {
49     "_id": "69430af19deff37b2460a395",
50     "name": "BASIC",
51     "price": 50,
52     "durationInDays": 30,
53     "features": [
54       "Basic profile",
55       "Limited services"
56     ],
57     "status": "ACTIVE"
58   },
59   "subscriptionExpiresAt": "2026-01-16T20:28:00.978Z"
60 }
61

```

Can you verify something - Like I did buy a plan through the doctor but it is not recorded as the transaction when fetched from the admin..

Important : ``Overall add the balance type logic too like doctor/patient can have a credit / balance and the admin can topup etc too.The doctor can request for the withdraw after the approval he get to cash out his balance..``

`{{baseUrl}}/weekly-schedule`

The screenshot shows a REST API testing interface. On the left, there is a sidebar with a tree view of API endpoints under "MyDoctor Platform - Complete API Collection / 2. Doctor Panel / Weekly Schedule". The "Get Weekly Schedule" endpoint is selected. The main panel shows a "GET" request with the URL `HTTP {{baseUrl}} /weekly-schedule`. Below the URL, there are tabs for "Docs", "Params", "Authorization", "Headers (8)", "Body", "Scripts", and "Settings". The "Params" tab is active, showing a table for "Query Params" with one row: "Key" (Value) and "Description". At the bottom of the main panel, there are tabs for "Body", "Cookies", "Headers (9)", and "Test Results". The "Test Results" tab is active, displaying a 404 Not Found response with a status of 404, a duration of 6 ms, and a body size of 462 B. The body content is a simple HTML page with an error message: "Cannot GET /api/weekly-schedule".

Route not Found.

This route should be id free like it should get from the token:
{{baseUrl}}/appointment?doctorId=:doctorId

{{baseUrl}}/products

This route is okay, just error returning code needs to be adjusted like it should not return the 500 rather send 403/ forbidden.

`{{baseUrl}}/announcements/doctor`

The screenshot shows the Postman application interface. On the left, there's a sidebar with a tree view of collections and specific API endpoints. The main area shows a GET request to `http://{{baseUrl}}/announcements/doctor`. The 'Params' tab is selected, showing a table for 'Query Params'. Below the table, the 'Test Results' tab is open, displaying a JSON response. The response is as follows:

```
1 {  
2     "success": true,  
3     "message": "OK",  
4     "data": {  
5         "announcements": [],  
6         "pagination": {  
7             "page": 1,  
8             "limit": 20,  
9             "total": 0,  
10            "pages": 0  
11        }  
12    }  
13 }
```

I created an announcement through the admin but it is not showing in the doctor section. Can you confirm the logic

`{baseUrl}/notification/:userId?isRead=false&page=1&limit=20`

The screenshot shows the Ollama API workspace interface. On the left, there's a sidebar with a tree view of collections: Blog, System Activity, 2. Doctor Panel, Profile Management, Dashboard, Subscriptions, Availability, Weekly Schedule, Appointments, Products, Chat, Video Sessions, Announcements, and Notifications. Under Notifications, there are two items: "Get My Notifications" (selected) and "Mark Notification as Read". The main area shows a "Get My Notifications" endpoint with the URL `{baseUrl}/notification/:userId?isRead=false&page=1&limit=20`. The "Params" tab is selected, showing query parameters: isRead (false), page (1), and limit (20). Below the URL input, there's a "Send" button and a "Response" section with a cartoon character and a "Click Send to get a response" button.

This route should be id free like it should get from the token:

`{baseUrl}/transaction`

The screenshot shows the Ollama API workspace interface. On the left, there's a sidebar with a tree view of collections: Notifications, Announcements, Chat, Blog, System Activity, 2. Doctor Panel, 3. Patient Panel, Dashboard, Profile, Appointments, Medical Records, Payments, Process Appointment Payment, Process Product Payment, Get My Transactions, Get Transaction by ID (Pay...), Get Transaction by ID (Tran...), Create Transaction, Reviews, Favorites, Chat, Video Sessions, and Notifications. Under Payments, there are three items: "Process Appointment Payment", "Process Product Payment", and "Create Transaction" (selected). The main area shows a "Create Transaction" endpoint with the URL `{baseUrl}/transaction`. The "Body" tab is selected, showing a JSON payload:

```
1 {  
2   "amount": 100,  
3   "type": "APPOINTMENT",  
4   "status": "PENDING"  
5 }
```

. Below the body, there's a "Headers" section with 10 entries. At the bottom, there's a "Test Results" section showing a 400 Bad Request error with the message: "success": false, "message": "Validation error", "errors": [{ "field": "body.userId", "message": "Required" }].

Verify these

The screenshot shows the Postman interface with the following details:

- Left Sidebar (Collections):**
 - Notifications
 - Announcements
 - Chat
 - Blog
 - System Activity
 - 2. Doctor Panel
 - 3. Patient Panel
 - Dashboard
 - Profile
 - Appointments
 - Medical Records
 - Payments
 - Reviews
 - POST Create Review**
 - DEL Delete Review**
 - Favorites
 - Chat
 - Video Sessions
 - Notifications
 - Account
 - 4. File Uploads**
 - POST Upload Profile Image**
 - POST Upload Doctor Documents**
 - POST Upload Clinic Images**
- Request Details:**
 - Method:** POST
 - URL:** {{baseUrl}}/reviews
 - Body:** JSON (raw)


```
{
    "doctorId": "6942af419deff37b2460a342",
    "rating": 5,
    "reviewText": "Excellent doctor"
}
```
- Response:**
 - Status: 400 Bad Request
 - Time: 5 ms
 - Size: 382 B
 - Message: Validation error
 - Errors (JSON):


```
{
    "success": false,
    "message": "Validation error",
    "errors": [
        {
            "field": "body.patientid",
            "message": "Required"
        }
    ]
}
```

Review logic of the patient should be adjusted like we get to overall give the review of the doctor/plus as per the appointment.

The screenshot shows the Postman interface with the following details:

- Left Sidebar (Collections):**
 - Favorites**
 - POST Add Favorite Doctor**
 - GET Get Favorite Doctors**
 - DEL Remove Favorite**
 - Chat**

For these routes getting the body.parameter error- help me fix it.