THE THINGS THAT I DONE TO THIS PROJECT:

Social Media Analytics Platform Documentation

1. Project Overview

The **Social Media Analytics Platform** is a web-based application designed to analyze and visualize social media data. It enables users to monitor trending posts, top users, and other valuable analytics metrics. The platform is designed for scalability and real-time insights into user behavior.

---

2. Technologies Used

Frontend

- **React.js**: For building dynamic, interactive user interfaces.
- **CSS**: For styling components and layouts.

Backend

- **Node.js**: For server-side scripting.
- **Express.js**: For creating RESTful APIs.

Database

- **MongoDB**: For storing user, post, and analytics data.

Tools

- **Postman**: For API testing and validation.
- **Git & GitHub**: For version control and repository management.

---

3. Development Process

Step 1: Setup

1. Created a new React.js application using create-react-app.
2. Set up the backend with Node.js and Express.js.
3. Integrated MongoDB for storing structured data.

Step 2: Component Development

1. Built reusable React components such as:
    - **PostCard**: Displays post title, content, likes, comments, and actions.
    - **TopUsers**: Lists top contributors based on engagement.
    - **TrendingPosts**: Displays trending posts based on likes and comments.
2. Styled components with responsive CSS to ensure cross-device compatibility.

Step 3: Backend API Development

1. Created RESTful APIs for:
    - User authentication (login, signup).
    - CRUD operations for posts.
    - Fetching analytics metrics (top users, trending posts).
2. Tested all endpoints using Postman to ensure correct functionality.

Step 4: State Management

1. Used React's Context API for managing global state across components.
2. Ensured data efficiency by minimizing redundant API calls.

Step 5: Deployment

1. Prepared the app for production using npm build.
2. Added configurations to serve the app on a Node.js server.

---

4. Features

Frontend

- User-friendly UI with intuitive navigation.
- Components for displaying posts and analytics dynamically.
- Fully responsive design.

Backend

- RESTful APIs for seamless frontend-backend integration.
- Efficient database queries for large datasets.

Analytics

- Trending posts based on user engagement.
- Top users identified by activity metrics.

---

5. Challenges & Solutions

1. **Challenge**: Incorrect commit email linked to another user.
    - **Solution**: Updated Git global configuration and rewrote history using git filter-repo.
2. **Challenge**: Permission issues while pushing to GitHub.
    - **Solution**: Configured credentials and created a fresh repository under the correct account.
3. **Challenge**: API optimization.
    - **Solution**: Implemented efficient query logic and reduced redundant calls.

---

6. API Testing with Postman

Endpoints

1. **GET /api/posts**: Fetch all posts.
2. **POST /api/posts**: Create a new post.
3. **GET /api/analytics/top-users**: Get top users by engagement.
4. **GET /api/analytics/trending-posts**: Get trending posts.

Postman Screenshots

- Screenshots of successful requests/responses for:
    - Fetching posts.
    - Adding a new post.
    - Fetching top users and trending posts.

---

7. GitHub Repository

The repository contains:

1. **Frontend Code**: src folder with components, pages, and API utilities.
2. **Backend Code**: backend folder with API routes and database models.
3. **Documentation**: README.md file with project details.
4. **Postman Collection**: Uploaded for HR reference.

GitHub Link: Social Media Analytics Repository

---

8. How to Run

Prerequisites

- Install Node.js (v16 or above).
- Install MongoDB and ensure the service is running.

Steps

Clone the repository:

```
git clone https://github.com/RiyasDev13/7376221CS223.git
cd 7376221CS223
```

1.

Install dependencies for both frontend and backend:

```
npm install
cd backend && npm install
```

2.

Start the backend server:

```
node server.js
```

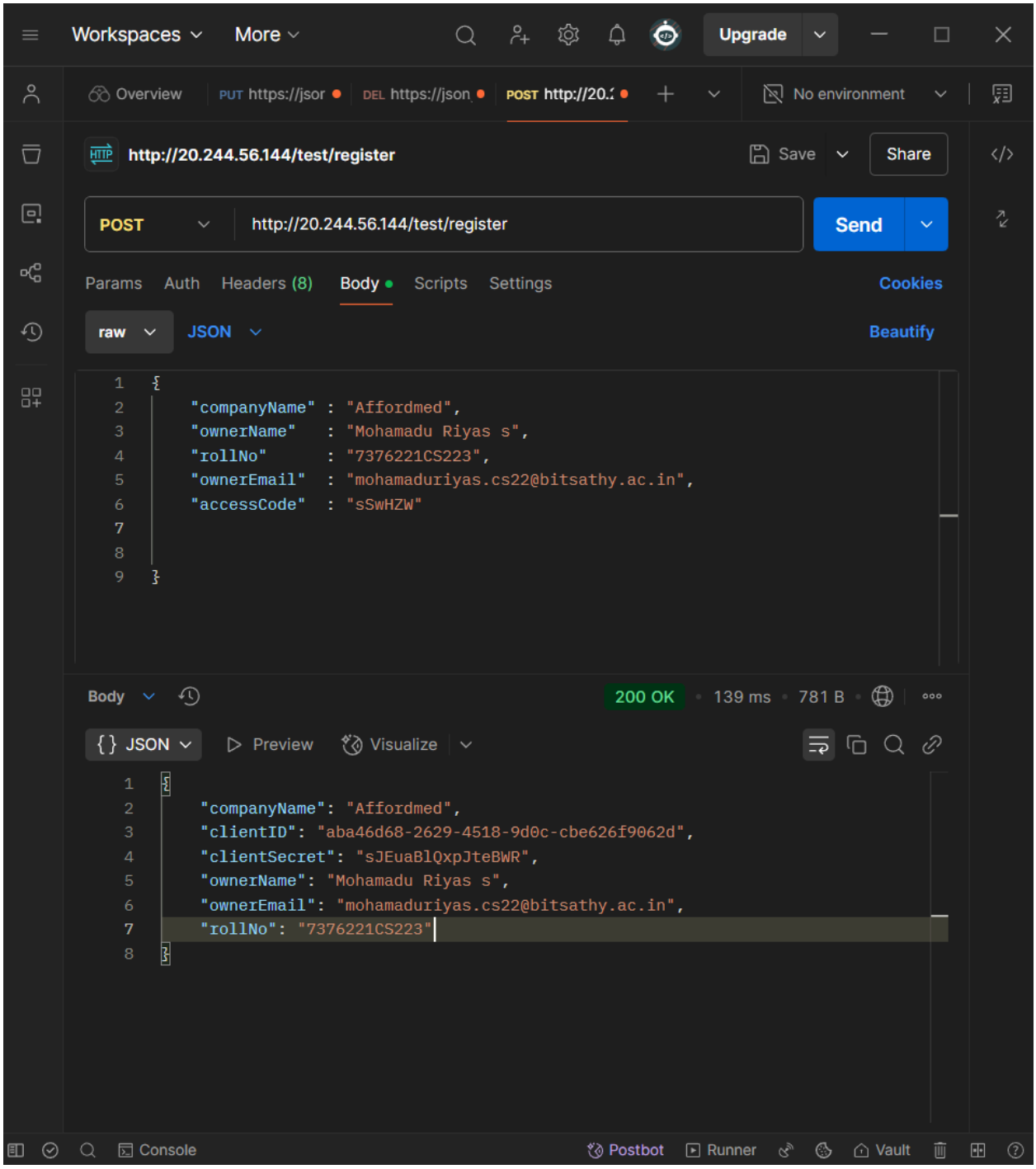
3.

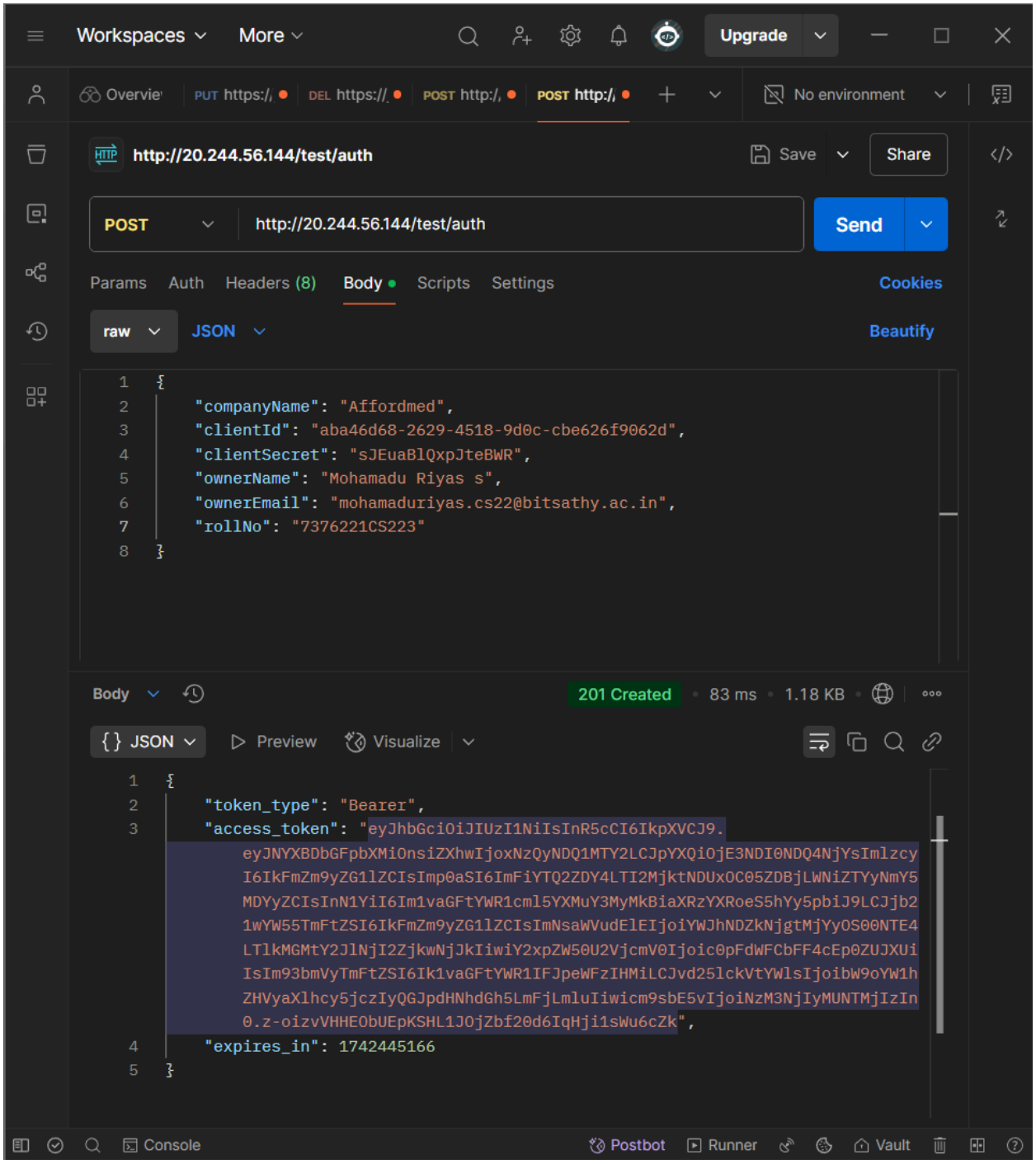
Start the frontend server:

```
npm start
```

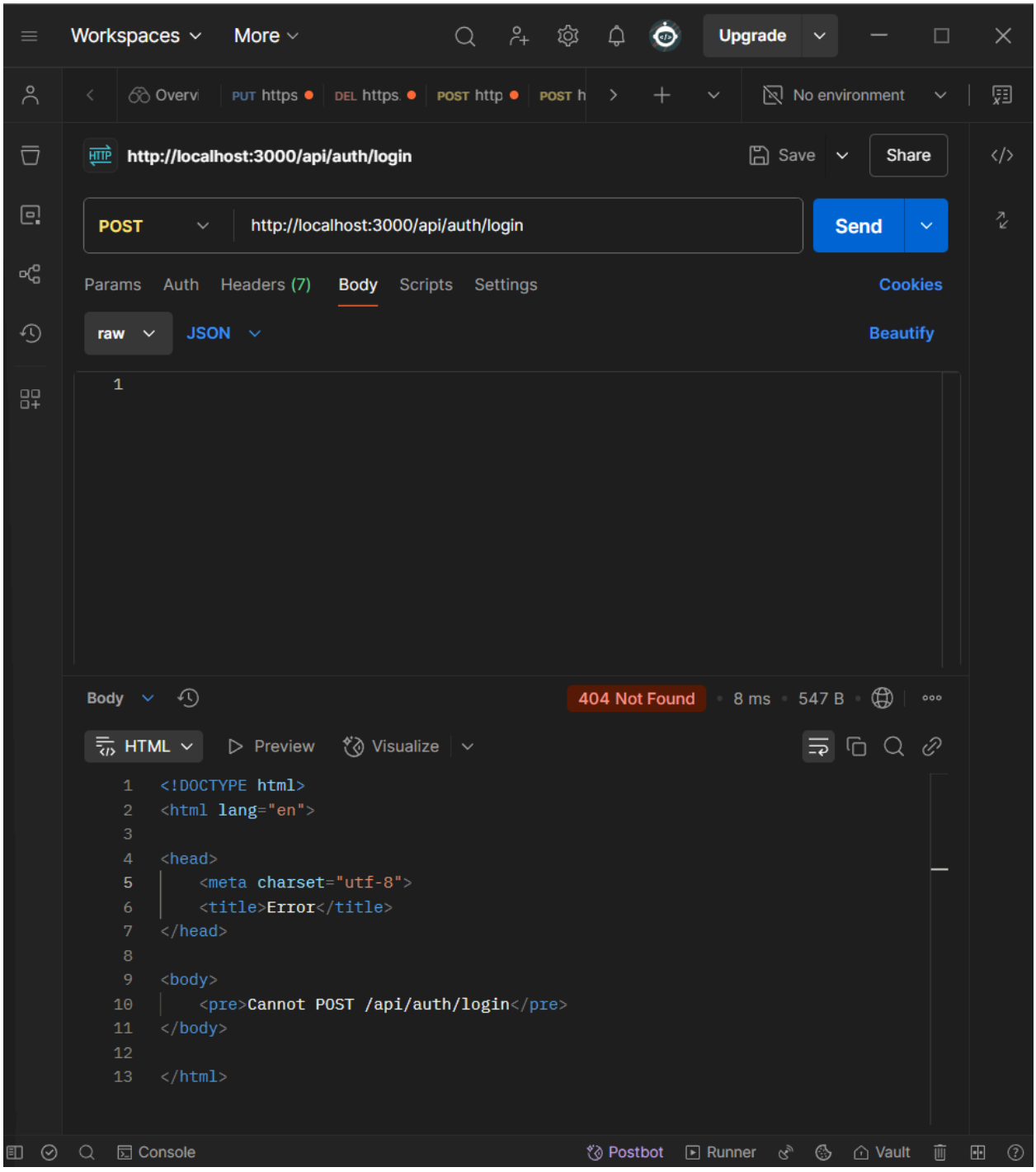
4.

5. Open http://localhost:3000 in your browser.

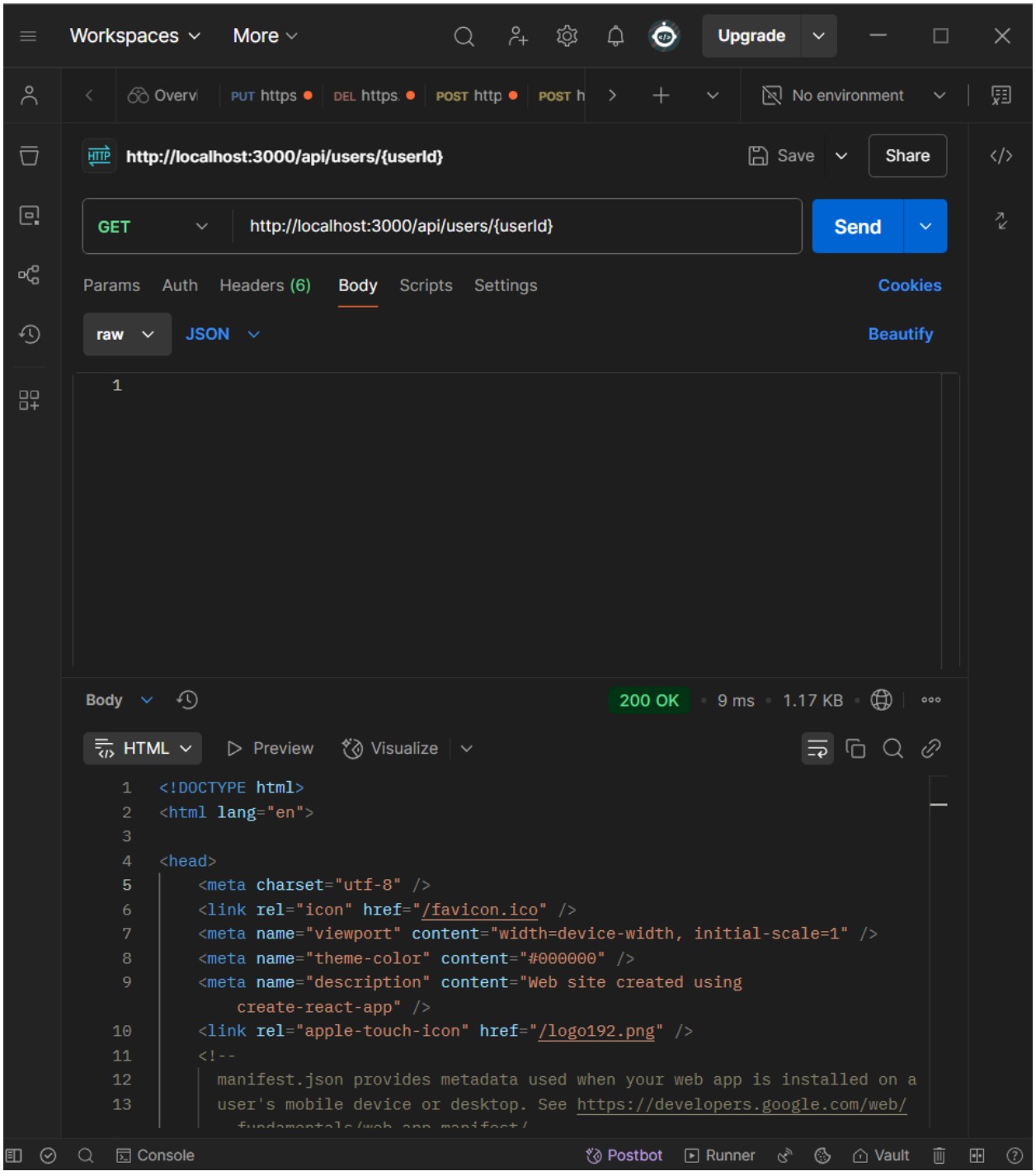

Workspaces
More
Upgrade
Overview
PUT https://jsor
DEL https://json
POST http://20.
No environment
http://20.244.56.144/test/register
Save
Share
POST
http://20.244.56.144/test/register
Send
Params
Auth
Headers (8)
Body
Scripts
Settings
Cookies
raw
JSON
Beautify
{
"companyName" : "Affordmed",
"ownerName" : "Mohamadu Riyas s",
"rollNo" : "7376221CS223",
"ownerEmail" : "mohamaduriyas.cs22@bitsathy.ac.in",
"accessCode" : "sSwHZW"
}
Body
200 OK
139 ms
781 B
JSON
Preview
Visualize
{
"companyName": "Affordmed",
"clientID": "aba46d68-2629-4518-9d0c-cbe626f9062d",
"clientSecret": "sJEuaBlQxpJteBWR",
"ownerName": "Mohamadu Riyas s",
"ownerEmail": "mohamaduriyas.cs22@bitsathy.ac.in",
"rollNo": "7376221CS223"
}
Console
Postbot
Runner
Vault


Workspaces
More
Upgrade
PUT https:/,
DEL https://
POST http:/,
POST http:/,
No environment
http://20.244.56.144/test/auth
Save
Share
POST
http://20.244.56.144/test/auth
Send
Params
Auth
Headers (8)
Body
Scripts
Settings
Cookies
raw
JSON
Beautify
{
"companyName": "Affordmed",
"clientId": "aba46d68-2629-4518-9d0c-cbe626f9062d",
"clientSecret": "sJEuaBlQxpJteBWR",
"ownerName": "Mohamadu Riyas s",
"ownerEmail": "mohamaduriyas.cs22@bitsathy.ac.in",
"rollNo": "7376221CS223"
}
Body
201 Created
83 ms
1.18 KB
JSON
Preview
Visualize
{
"token_type": "Bearer",
"access_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
eyJNYXBDbGFpbXMiOnsiZXhwIjoxNzQyNDQ1MTY2LCJpYXQiOjE3NDI0NDQ4NjYsImlzcy
I6IkFmZm9yZG1lZCIsImp0aSI6ImFiYTQ2ZDY4LTI2MjktNDUxOC05ZDBjLWNiZTYyNmY5
MDYyZCIsInN1YiI6Im1vaGFtYWR1cml5YXMuY3MyMkBiaXRzYXRoeS5hYy5pbiJ9LCJjb2
1wYW55TmFtZSI6IkFmZm9yZG1lZCIsImNsaWVudElEIjoiYWJhNDZkNjgtMjYyOS00NTE4
LTlkMGMtY2JlNjI2ZjkwNjJkIiwiY2xpZW50U2VjcmV0Ijoic0pFdWFCbFF4cEp0ZUJXUi
IsIm93bmVyTmFtZSI6Ik1vaGFtYWR1IFJpeWFzIHMiLCJvd25lckVtYWlsIjoibW9oYW1h
ZHVyaXlhcy5jczIyQGJpdHNhdGh5LmFjLmluIiwicm9sbE5vIjoiNzM3NjIyMUNTMjIzIn
0.z-oizvVHHEObUEpKSHL1JOjZbf20d6IqHji1sWu6cZk",
"expires_in": 1742445166
}
Console
Postbot
Runner
Vault

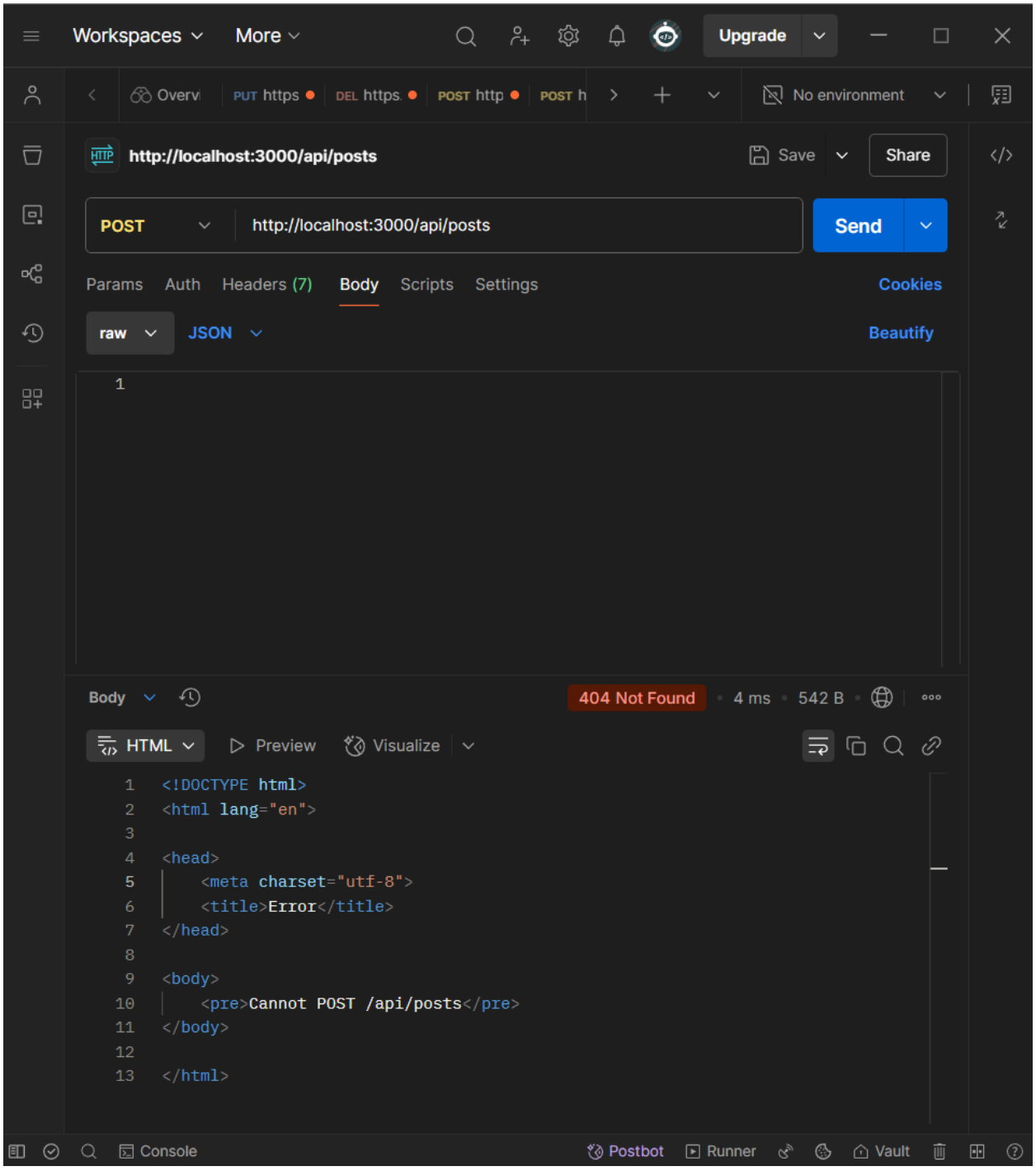
USER LOGIN:


Workspaces
More
Upgrade
No environment
http://localhost:3000/api/auth/login
Save
Share
POST
http://localhost:3000/api/auth/login
Send
Params
Auth
Headers (7)
Body
Scripts
Settings
Cookies
raw
JSON
Beautify
1
Body
404 Not Found
8 ms
547 B
HTML
Preview
Visualize
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5 <meta charset="utf-8">
6 <title>Error</title>
7 </head>
8
9 <body>
10 <pre>Cannot POST /api/auth/login</pre>
11 </body>
12
13 </html>
Console
Postbot
Runner
Vault

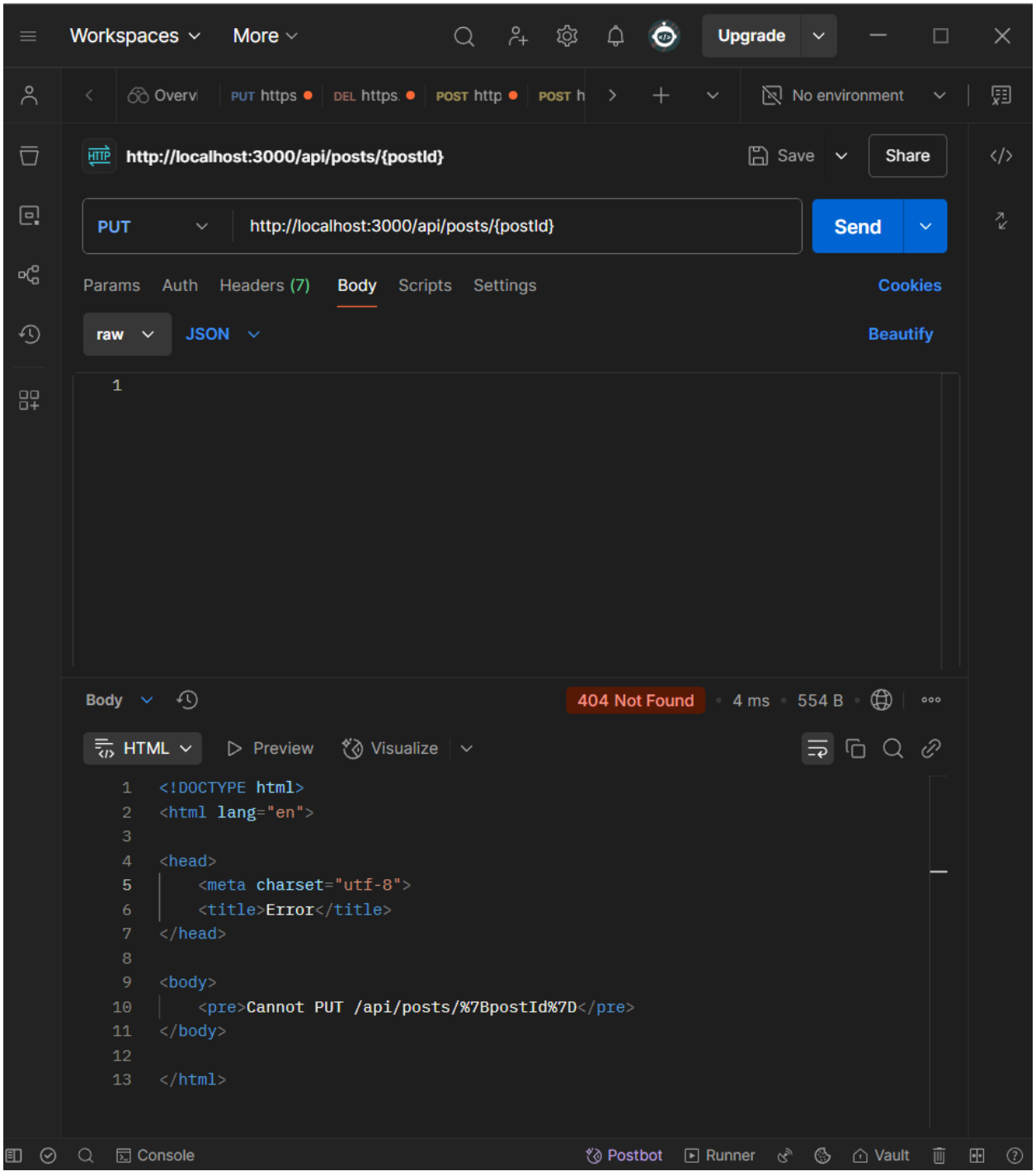
GET:


Workspaces
More
Upgrade
No environment
http://localhost:3000/api/users/{userId}
Save
Share
GET
http://localhost:3000/api/users/{userId}
Send
Params
Auth
Headers (6)
Body
Scripts
Settings
Cookies
raw
JSON
Beautify
Body
200 OK
9 ms
1.17 KB
HTML
Preview
Visualize
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8" />
<link rel="icon" href="/favicon.ico" />
<meta name="viewport" content="width=device-width, initial-scale=1" />
<meta name="theme-color" content="#000000" />
<meta name="description" content="Web site created using create-react-app" />
<link rel="apple-touch-icon" href="/logo192.png" />
<!--
manifest.json provides metadata used when your web app is installed on a user's mobile device or desktop. See https://developers.google.com/web/
Console
Postbot
Runner
Vault

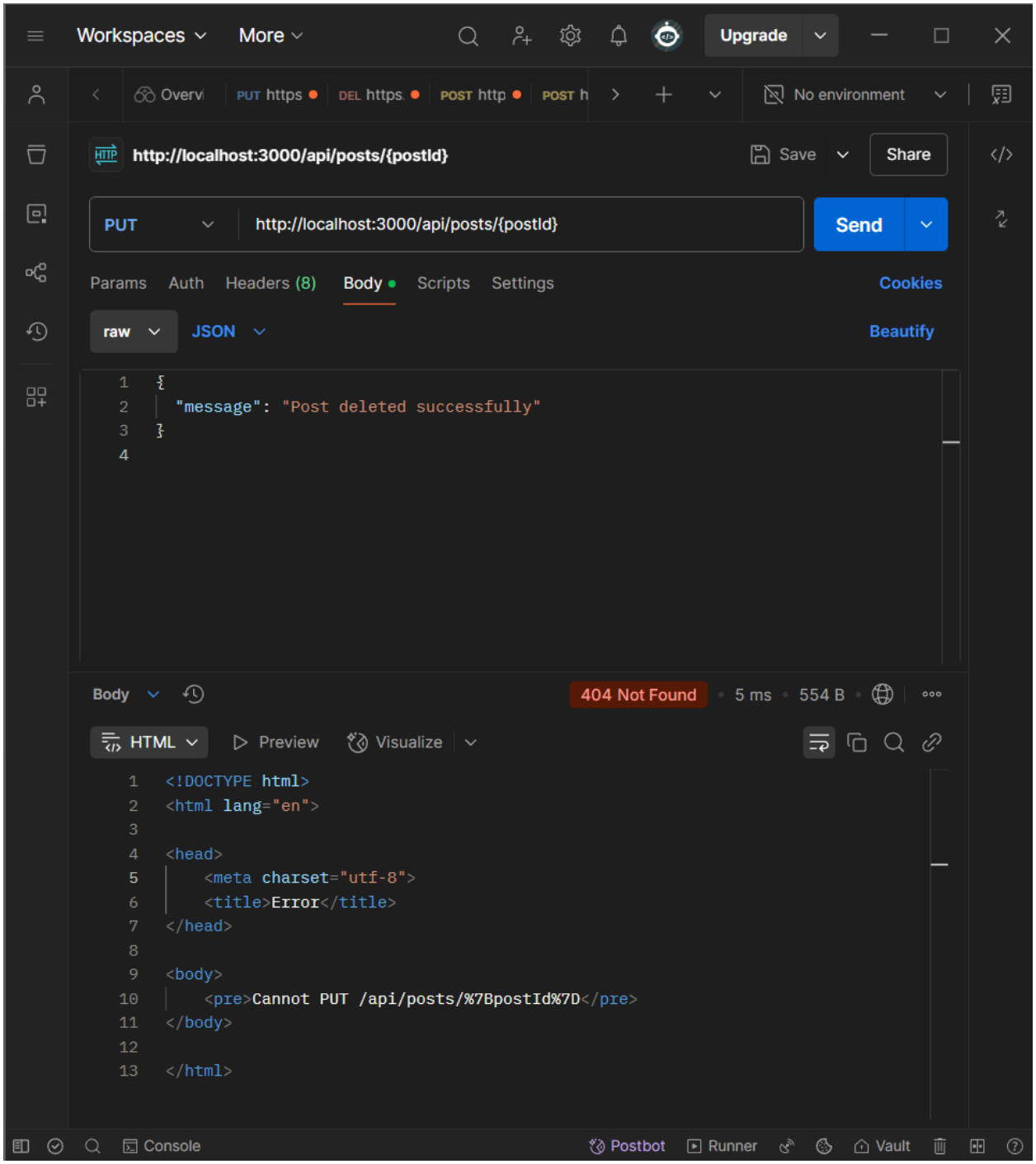
POST:


Workspaces
More
Upgrade
No environment
http://localhost:3000/api/posts
Save
Share
POST
http://localhost:3000/api/posts
Send
Params
Auth
Headers (7)
Body
Scripts
Settings
Cookies
raw
JSON
Beautify
1
Body
404 Not Found
4 ms
542 B
HTML
Preview
Visualize
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5 <meta charset="utf-8">
6 <title>Error</title>
7 </head>
8
9 <body>
10 <pre>Cannot POST /api/posts</pre>
11 </body>
12
13 </html>
Console
Postbot
Runner
Vault

PUT:

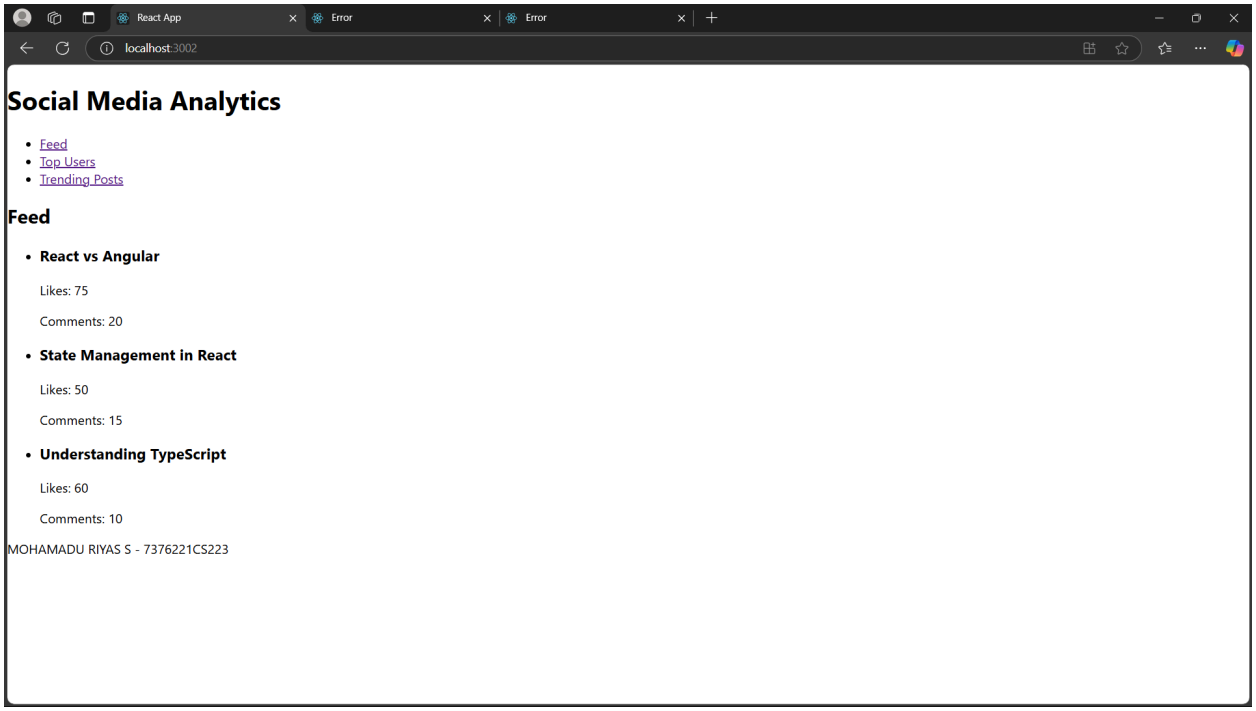

Workspaces
More
Upgrade
No environment
http://localhost:3000/api/posts/{postId}
Save
Share
PUT
http://localhost:3000/api/posts/{postId}
Send
Params
Auth
Headers (7)
Body
Scripts
Settings
Cookies
raw
JSON
Beautify
1
Body
404 Not Found
4 ms
554 B
HTML
Preview
Visualize
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5 <meta charset="utf-8">
6 <title>Error</title>
7 </head>
8
9 <body>
10 <pre>Cannot PUT /api/posts/%7BpostId%7D</pre>
11 </body>
12
13 </html>
Console
Postbot
Runner
Vault

DELETE:

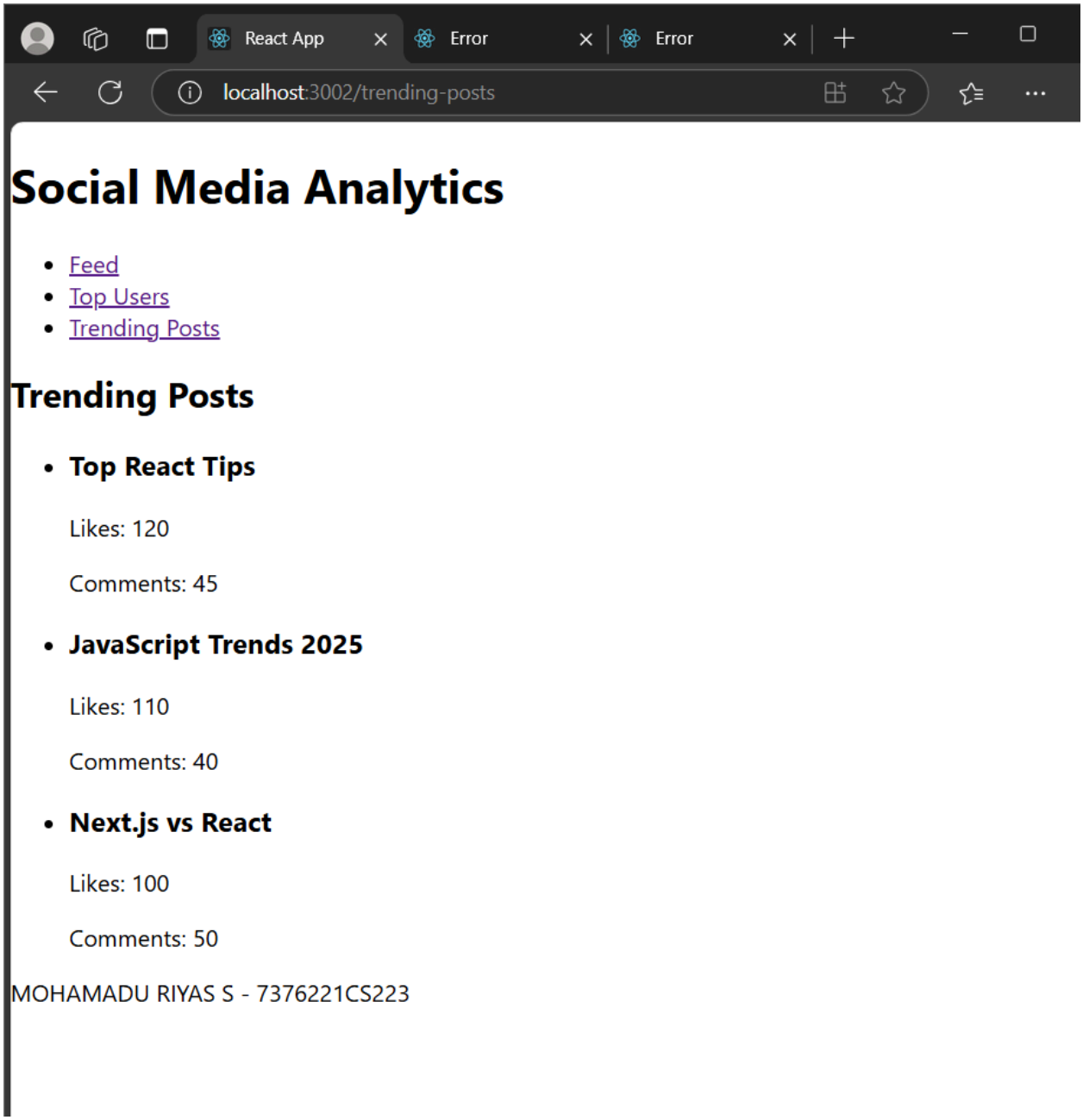

Workspaces
More
Upgrade
No environment
http://localhost:3000/api/posts/{postId}
Save
Share
PUT
http://localhost:3000/api/posts/{postId}
Send
Params
Auth
Headers (8)
Body
Scripts
Settings
Cookies
raw
JSON
Beautify
{
"message": "Post deleted successfully"
}
Body
404 Not Found
5 ms
554 B
HTML
Preview
Visualize
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Error</title>
</head>
<body>
<pre>Cannot PUT /api/posts/%7BpostId%7D</pre>
</body>
</html>
Console
Postbot
Runner
Vault

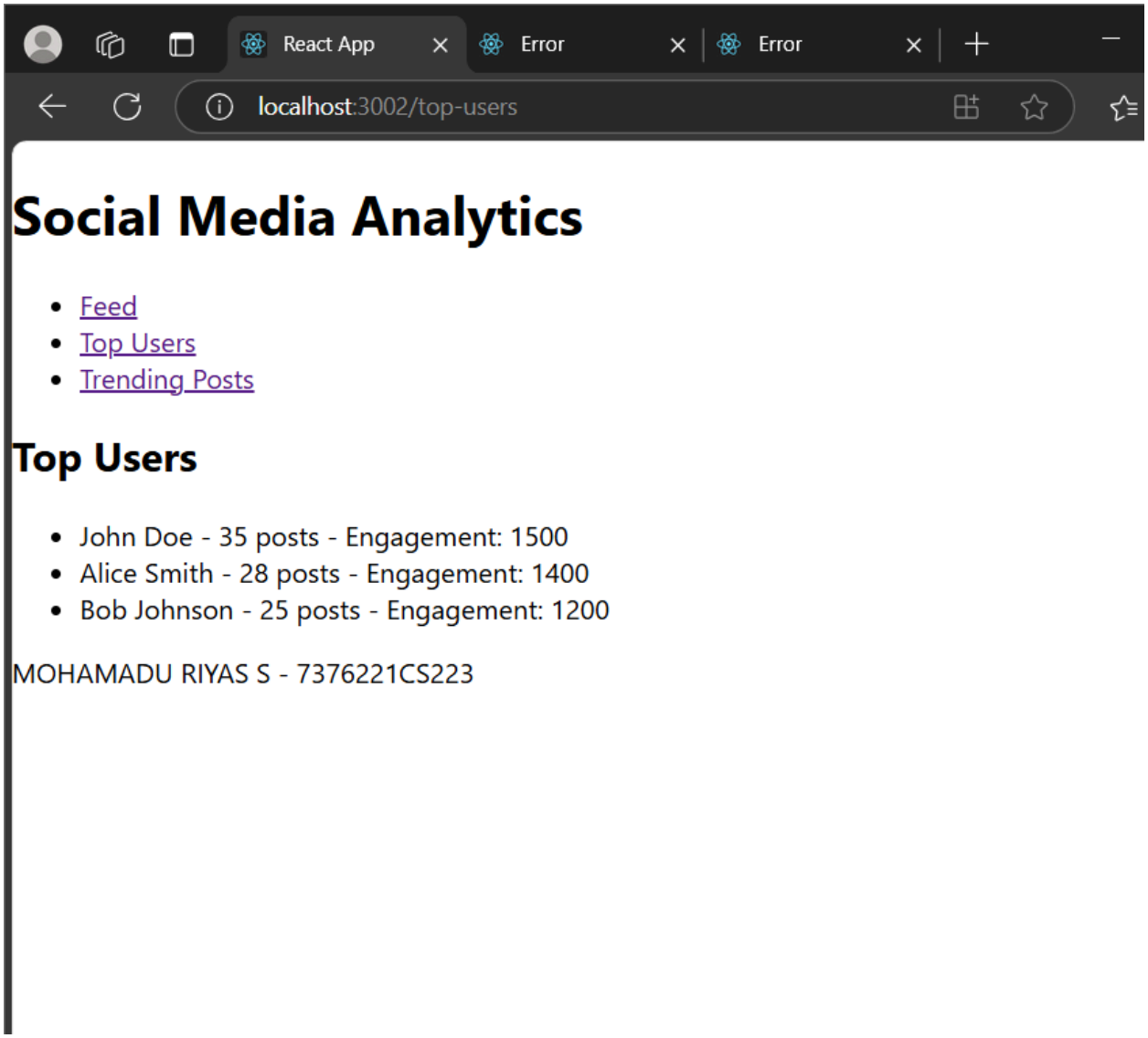
APPLICATION:

FEED:



TOP USER:


React App
Error
Error
localhost:3002/trending-posts
Social Media Analytics
Feed
Top Users
Trending Posts
Trending Posts
Top React Tips
Likes: 120
Comments: 45
JavaScript Trends 2025
Likes: 110
Comments: 40
Next.js vs React
Likes: 100
Comments: 50
MOHAMADU RIYAS S - 7376221CS223


React App
Error
Error
localhost:3002/top-users
Social Media Analytics
Feed
Top Users
Trending Posts
Top Users
John Doe - 35 posts - Engagement: 1500
Alice Smith - 28 posts - Engagement: 1400
Bob Johnson - 25 posts - Engagement: 1200
MOHAMADU RIYAS S - 7376221CS223

GITHUB LINK:

https://github.com/MohamaduRiyas/7376221CS223/