

Register User

The screenshot shows the Postman application interface. A POST request is being made to `http://localhost:8080/register`. The request body contains the following JSON payload:

```
1 {
2   "firstName": "Riya",
3   "lastName": "Sharma",
4   "email": "riya@example.com",
5   "phoneNo": "9876543215",
6   "password": "password123",
7   "confirmPassword": "password123",
8   "role": "CUSTOMER",
9   "address": {
10     "name": "Home",
11     "areaName": "Vastrapuri",
12     "pincode": "3800015",
13     "cityId": 1
14   }
15 }
16 }
```

The response status is 200 OK, with a response time of 242 ms and a response size of 508 B. The response body is:

```
1 {
2   "phone": "9876543215",
3   "email": "riya@example.com",
4   "message": "Successfully Registered"
5 }
```

Login using OTP

The screenshot shows the Postman application interface. A POST request is being made to `http://localhost:8080/login`. The request body contains the following JSON payload:

```
1 {
2   "username": "rahul.sharma@example.com",
3   "loginType": "OTP"
4 }
```

The response status is 200 OK, with a response time of 6.64 s and a response size of 447 B. The response body is:

```
1 OTP sent. Please verify.
```

POST http://localhost:8080/verify-otp

```
9 {  
10   "username": "rahul.sharma@example.com",  
11   "otp": "964115"  
12 }  
13  
14
```

Body Cookies Headers (14) Test Results

```
1 {  
2   "jwtToken": "eyJhbGciOiJIUzI1NiJ9.eyJpczI6IjE1CjZdW10IjYWh1bc5zaGFybWFAZXhhbXBsZS5jb20iLC3pYXQ10jE3NTk40Tk2NzMsiMv4ccI6Mt2MDUuNDE3M38.eyJRMiI6IFYH44_1N1CmP07QG0-whno9jgKLEALDSVAwQj8JaFzJknefFH48gmmqo42Y6fLs2MdueoAIIInzw",  
3   "username": "rahul.sharma@example.com",  
4   "role": "CUSTOMER"  
5 }
```

200 OK 184 ms 705 B

Login using password

POST http://localhost:8080/login

```
1 {  
2   "username": "rahul.sharma@example.com",  
3   "password": "Password@123",  
4   "logInType": "PASSWORD"  
5 }  
6
```

Body Cookies Headers (14) Test Results

```
1 {  
2   "jwtToken": "eyJhbGciOiJIUzI1NiJ9.eyJpczI6IjE1CjZdW10IjYWh1bc5zaGFybWFAZXhhbXBsZS5jb20iLC3pYXQ10jE3NTk40Tk30TQsImV4ccI6Mt2MDUuMjU5NH0.883vgUH1B-e-a54-5sKxtcsxq1VXoFmfZ27J1wIA5m6pcrqANSK-tfp4XYgdzgw9evUPIYsYoIA8t_f_20w",  
3   "username": "rahul.sharma@example.com",  
4   "role": "CUSTOMER"  
5 }
```

200 OK 852 ms 705 B

Fetch all services

My Workspace

POST http://localhost:8080/ | GET http://localhost:8080/st | +

http://localhost:8080/services/all

GET http://localhost:8080/services/all

Params Authorization Headers (7) Body Scripts Settings

none form-data x-www-form-urlencoded raw binary GraphQL

This request does not have a body

Body Cookies Headers (14) Test Results

{ JSON Preview Visualize }

```
[{"serviceId": 1, "serviceName": "Home Cleaning", "serviceImage": null}, {"serviceId": 2, "serviceName": "Plumbing", "serviceImage": null}, {"serviceId": 3, "serviceName": "Electrical Repair", "serviceImage": null}, {"serviceId": 4, "serviceName": "Pest Control", "serviceImage": null}]
```

200 OK 69 ms 747 B

Fetch all subservices for a particular service

My Workspace

POST http://localhost:8080/ | GET http://localhost:8080/st | +

http://localhost:8080/subservices/1

GET http://localhost:8080/subservices/1

Params Authorization Headers (7) Body Scripts Settings

none form-data x-www-form-urlencoded raw binary GraphQL

This request does not have a body

Body Cookies Headers (14) Test Results

{ JSON Preview Visualize }

```
[{"subServiceId": "1", "subServiceName": "Bathroom Cleaning", "price": 150.0}, {"subServiceId": "2", "subServiceName": "Living Room Cleaning", "price": 200.0}, {"subServiceId": "3", "subServiceName": "Bedroom Cleaning", "price": 250.0}]
```

200 OK 58 ms 642 B

Book Order

Home Workspaces API Network

http://localhost:8082/orders/book

POST http://localhost:8082/orders/book

Params Authorization Headers (9) Body Scripts Settings

Body (raw JSON)

```
1 {  
2   "serviceProviderId": "1",  
3   "subServiceId": "2",  
4   "contactName": "Mohan",  
5   "contactPhone": "+9188823231",  
6   "contactAddress": "123, kudasan, Gandhinagar",  
7   "date": "2025-10-15",  
8   "time": "15:00"  
9 }  
10
```

Body Cookies Headers (14) Test Results

200 OK 6.04 s 649 B

{} JSON Preview Visualize

```
1 {  
2   "orderId": 203,  
3   "date": "2025-10-15",  
4   "time": "15:00:00",  
5   "contactName": "Mohan",  
6   "contactPhone": "+9188823231",  
7   "contactAddress": "123, kudasan, Gandhinagar",  
8   "status": "PENDING",  
9   "createdAt": "2025-10-10T19:14:52.954264",  
10  "orderOptList": null  
11 }
```

Bill

Home Workspaces API Network

http://localhost:8080/bills/152

GET http://localhost:8080/bills/152

Params Authorization Headers (7) Body Scripts Settings

Body (raw JSON)

This request does not have a body

Body Cookies Headers (14) Test Results

200 OK 88 ms 500 B

{} JSON Preview Visualize

```
1 {  
2   "invoiceNumber": 1,  
3   "charge": 200.0,  
4   "gst": 6.0,  
5   "total": 206.0,  
6   "status": "PENDING"  
7 }
```

Payment

The screenshot shows the Postman interface with a successful API call. The URL is `http://localhost:8080/api/payments/create/1`. The response status is 200 OK, with a response time of 2.26 ms and a size of 474 B. The response body is a JSON object:

```
1 {  
2   "orderId": "order_PRhReQI4chYwWT",  
3   "amount": 26688.0  
4 }
```

The screenshot shows the Postman interface with a successful API call. The URL is `http://localhost:8080/api/payments/success`. The response status is 200 OK, with a response time of 95 ms and a size of 457 B. The response body is raw text:

```
1 Payment recorded and payouts added
```

Feedback

The screenshot shows the Postman interface with a dark theme. A collection named "Feedback" is selected. A POST request is made to `http://localhost:8080/feedback/submit/1`. The request body is a JSON object:

```
18
19
20
21   "orderId": 1,
22   "rating": 5,
23   "review": "Excellent service, very prompt!",
24   "serviceProviderId": "1"
25 }
```

The response status is 200 OK, with a duration of 220 ms and a size of 454 B. The response body is:

```
1 Feedback submitted successfully
```

Cancel Order

The screenshot shows the Postman interface with a dark theme. A collection named "Orders" is selected. A POST request is made to `http://localhost:8080/cancel/1`. The "Authorization" tab is selected, showing "Bearer Token" as the auth type. A placeholder token is shown in a redacted field.

The response status is 200 OK, with a duration of 4.67 s and a size of 450 B. The response body is:

```
1 Order canceled successfully
```

Update User Profile

The screenshot shows the Postman interface with a successful API call. The URL is `http://localhost:8082/customer/profile/update`. The method is `PUT`. The response status is `200 OK` with a response time of `376 ms` and a size of `457 B`. The response body is:

```
1 {
2   "firstName": "Rohan",
3   "lastName": "Sharma",
4   "email": "rohan.sharma@example.com",
5   "phoneNo": "9876543238"
6 }
```

The response preview shows the message: `User profile updated successfully.`

For My Bookings Screen List Of orders of particular User

The screenshot shows the Postman interface with a successful API call. The URL is `http://localhost:8080/orders/user/1`. The method is `GET`. The response status is `200 OK` with a response time of `48 ms` and a size of `1.04 KB`. The response body is:

```
1 [
2   {
3     "orderId": 103,
4     "contactName": "Rahul",
5     "contactPhone": "+919876543215",
6     "contactAddress": "123, Kudasan, Gandhinagar",
7     "date": "2025-10-15",
8     "time": "15:00:00",
9     "status": "PENDING",
10    "serviceProviderName": "Spark Clean Services"
11  },
12  {
13    "orderId": 152,
14    "contactName": "Rahul",
15    "contactPhone": "+919876543219",
16    "contactAddress": "123, Kudasan, Gandhinagar",
17    "date": "2025-10-15",
18    "time": "15:00:00",
19    "status": "PENDING",
20    "serviceProviderName": "Spark Clean Services"
21  },
22  {
23    "orderId": 1...
```

Change Password

The screenshot shows the Postman application interface. In the top navigation bar, there are links for Home, Workspaces, API Network, and a search bar. On the right side, there are buttons for Invite, Upgrade, and environment selection.

The main workspace displays a collection named "Collections" containing a single item titled "http://localhost:8082/change-password". This item is a PUT request to the URL `http://localhost:8082/change-password`. The request body is set to raw JSON:

```
1 {  
2   "oldPassword": "Password@123",  
3   "newPassword": "password123*",  
4   "confirmPassword": "Password123@"  
5 }  
6
```

Below the body, the "Body" tab is selected, showing the response: "1 Password changed successfully." The status bar at the bottom indicates a 200 OK response with 395 ms latency and 453 B size.