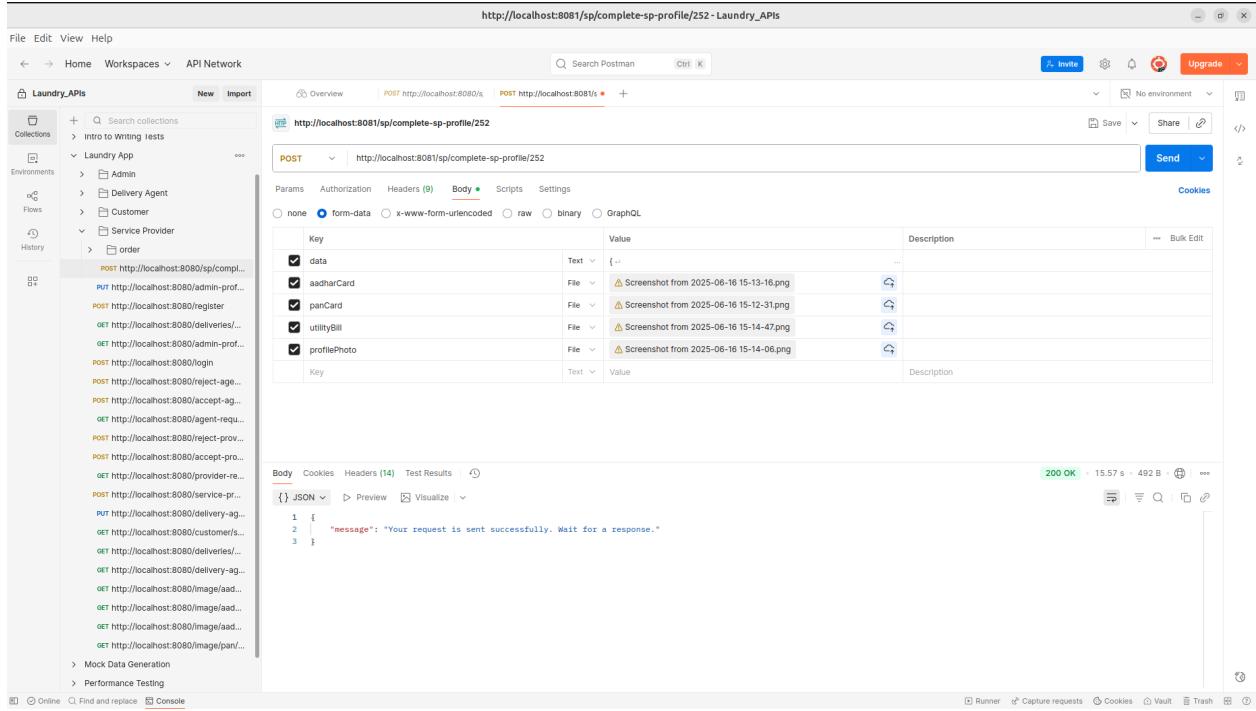


# SERVICE PROVIDER APIs

## 1. Request

- Send request to joining platform.

```
http://localhost:8080/sp/complete-sp-profile/{userId}
```



The screenshot shows the Postman interface with a collection named "Laundry APIs". A POST request is being made to `http://localhost:8081/sp/complete-sp-profile/252`. The request body is set to "form-data" and contains four files:

- data: Screenshot from 2025-06-16 15-13-16.png
- aadharCard: Screenshot from 2025-06-16 15-12-31.png
- panCard: Screenshot from 2025-06-16 15-14-47.png
- utilityBill: Screenshot from 2025-06-16 15-14-06.png

The response status is 200 OK, with a duration of 15.57 s and a size of 492 B.

## 2. Order

- Fetch pending orders

```
http://localhost:8080/provider/orders/pending
```

Status: 200 OK Size: 239 Bytes Time: 185 ms

```

1 [
2   {
3     "orderId": 1,
4     "date": "2025-10-12",
5     "time": "10:30:00",
6     "contactName": "Ramesh Patel",
7     "contactPhone": "9876543210",
8     "contactAddress": "203, Green Park Society, Surat",
9     "serviceName": "Plumbing Services",
10    "subServiceName": "Leak Repair",
11    "charge": 50.0
12  }
13 ]

```

- Get active orders which are not been completed yet

- `http://localhost:8080/provider/orders/active`

Status: 200 OK Size: 239 Bytes Time: 62 ms

```

1 [
2   {
3     "orderId": 1,
4     "date": "2025-10-12",
5     "time": "10:30:00",
6     "contactName": "Ramesh Patel",
7     "contactPhone": "9876543210",
8     "contactAddress": "203, Green Park Society, Surat",
9     "serviceName": "Plumbing Services",
10    "subServiceName": "Leak Repair",
11    "charge": 50.0
12  }
13 ]

```

- Get completed orders

- http://localhost:8080/provider/orders/delivered

The screenshot shows the Postman interface with a GET request to `http://localhost:8080/provider/orders/delivered`. The request is authenticated with a `Bearer` token. The response status is `200 OK`, size is `239 Bytes`, and time is `25 ms`. The response body is a JSON array containing one order object:

```
[{"orderId": 1, "date": "2025-10-12", "time": "10:30:00", "contactName": "Ramesh Patel", "contactPhone": "9876543210", "contactAddress": "203, Green Park Society, Surat", "serviceName": "Plumbing Services", "subServiceName": "Leak Repair", "charge": 50.0}]
```

## - Accept order

- http://localhost:8080/provider/orders/{orderId}/accept

The screenshot shows the Postman interface with a PUT request to `http://localhost:8080/provider/orders/1/accept`. The request is authenticated with a `Bearer` token. The response status is `200 OK`, size is `71 Bytes`, and time is `6.23 s`. The response body is a single-line string:

```
Order 1 accepted by service provider FreshSpin Laundry for customer 200
```

- Reject order(Not implement yet)
- `http://localhost:8080/provider/orders/{orderId}/reject`
- Change order status to IN-PROCESS (Once the service provider reaches the customer's home, they must do otp verification before work begins)
- After hitting this api otp is sent to customer
- `http://localhost:8080/provider/orders/{orderId}/inprocess`

The screenshot shows a Postman interface with the following details:

- Method:** PUT
- URL:** `http://localhost:8080/provider/orders/1/inprocess`
- Auth:** Bearer (selected)
- Response Status:** 200 OK
- Response Size:** 45 Bytes
- Response Time:** 5.63 s
- Response Body:** "Otp successfully send to customer. Verify Now"
- Token Prefix:** Bearer

- Once otp is sent to customer service provider need to verify it by providing otp sended to customer
- `http://localhost:8080/provider/orders/{orderId}/inprocess/verify-otp`

The screenshot shows the Postman application interface. A PUT request is being made to `http://localhost:8081/provider/orders/1/inprocess/verify-otp`. The response status is **200 OK**, size is **32 Bytes**, and time is **582 ms**. The response body contains the message: **1 status is changed to IN\_PROCESS.**

```
PUT http://localhost:8081/provider/orders/1/inprocess/verify-otp
```

```
1 status is changed to IN_PROCESS.
```

- Same for completed

- `http://localhost:8080/provider/orders/{orderId}/complete`

The screenshot shows the Postman application interface. A PUT request is being made to `http://localhost:8081/provider/orders/1/complete`. The response status is **200 OK**, size is **45 Bytes**, and time is **6.22 s**. The response body contains the message: **1 Otp successfully send to customer. Verify Now**.

```
PUT http://localhost:8081/provider/orders/1/complete
```

```
1 Otp successfully send to customer. Verify Now
```

- `http://localhost:8080/provider/orders/{orderId}/complete/verify-otp`

The screenshot shows the Postman application interface. In the top left, there's a navigation bar with File, Edit, Selection, View, Go, Run, etc. A search bar at the top right contains the text "Nov 11 0903" and "202412024\_lab10". Below the navigation is a toolbar with icons for New Request, New Collection, and others. The main area has two tabs: "localhost:8080/provider/o..." and "New Request". The "New Request" tab is active, showing a PUT request to "http://localhost:8081/provider/orders/1/complete/verify-otp". The "Body" tab is selected, showing JSON content:

```
1 {
2   "otp": "085303"
3 }
```

The response panel shows a green status: "Status: 200 OK", "Size: 31 Bytes", and "Time: 151 ms". The response body is: "Status is changed to COMPLETED."

### 3. Profile

- Fetch profile data
- `http://localhost:8080/provider/profile`

pos localhost:8081/register    tc New Request ×    tc New Request

GET <http://localhost:8081/provider/profile> Send

Query Headers 2 Auth 1 Body Tests Pre Run

None Basic Bearer OAuth 2 NTLM AWS

Bearer Token

```
eyJhbGciOiJIUzIwMiJ9.eyJpZCI6MjUyLCJzdWl0iJyYWh1bEBleGFtcGxlLmNvbSlsmIhdC16MTc2Mj4MjkwMywizXhwijoXnYzYzMzg3NzAzfQ.yBejdRH8r0XExp52yFtMKVttmXtYSQByelveXVj8KwPpSrtnzmnSxhnvfU0BnOszyk-35Nev8gYugsqH8kg
```

Token Prefix Bearer

Status: 200 OK Size: 1.06 KB Time: 298 ms

Response Headers 11 Cookies Results Docs

```

1  {
2    "userId": null,
3    "serviceProviderId": null,
4    "firstName": "Rahul",
5    "lastName": "Verma",
6    "phone": "9123456789",
7    "email": "rahul@example.com",
8    "joinedAt": "2025-11-07T19:38:59.087569",
9    "businessName": "Elite Laundry Services",
10   "businessLicenseNumber": "LIC1234567",
11   "gstNumber": "99ABCDE1234F125",
12   "profilePhoto": "https://res.cloudinary.com/dot327hz/image/upload/v1762526887/SmartLaundry/ServiceProvider/252/profile/zaoefslzzsdwcwkht3v.png",
13   "aadharCardPhoto": "https://res.cloudinary.com/dot327hz/image/upload/v1762526879/SmartLaundry/ServiceProvider/252/aadhar/lu9zje1ooovhlfml5w6ap.png",
14   "panCardPhoto": "https://res.cloudinary.com/dot327hz/image/upload/v1762526881/SmartLaundry/ServiceProvider/252/pan/x797zpxnexntqnjz6ek.png",
15   "businessUtilityBillPhoto": "https://res.cloudinary.com/dot327hz/image/upload/v1762526885/SmartLaundry/ServiceProvider/252/utility/eglcsv7w0shitzchcqr.png",
16   "bankName": "HDFC Bank",
17   "ifscCode": "HDFC0001234",
18   "bankAccountNumber": "123406789012",
19   "accountHolderName": "Rahul Verma".

```

Response Chart ↴

Signed out Go Live