

Postman implementation after Api gateway Security and Dockerization

1)user register

The screenshot shows a Postman request for 'Flightbooking_Microservice_Security / register' using a POST method. The request body is a JSON object with fields: name, email, age, gender, password, and role. The response status is '201 Created' with a response time of 809 ms, a response size of 330 B, and a token generated: eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJhbmdhZGlzcmVlbmlkaGlAZ21haWwuY29tIiwiWF0IjoxNzY1MDg5NjExLCJleHAiOjE3NjUxNzYwMTF9.TIKIt03qnFD8rWsDUMRUDn97yIT3LTxEVZogZ3_0i9M.

```
1 {  
2   "name": "Sreenidhi",  
3   "email": "angadisreenidhi@gmail.com",  
4   "age": 20,  
5   "gender": "Female",  
6   "password": "sreenidhi",  
7   "role": "USER"  
8 }
```

2)user already exists validation

The screenshot shows a Postman request for 'Flightbooking_Microservice_Security / register' using a POST method. The request body is identical to the previous registration. The response status is '400 Bad Request' with a response time of 10.00 s, a response size of 344 B, and an error message: 'User already exists with email angadisreenidhi@gmail.com'.

```
1 {  
2   "name": "Sreenidhi",  
3   "email": "angadisreenidhi@gmail.com",  
4   "age": 20,  
5   "gender": "Female",  
6   "password": "sreenidhi",  
7   "role": "USER"  
8 }
```

3)User login

The screenshot shows a Postman request for 'Flightbooking_Microservice_Security / Login' using a POST method. The request body contains email and password. The response status is '200 OK' with a response time of 336 ms, a response size of 444 B, and a token generated: eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJhbmdhZGlzcmVlbmlkaGlAZ21haWwuY29tIiwiWF0IjoxNzY1MDg5NjExLCJleHAiOjE3NjUxNzYwMTF9.TIKIt03qnFD8rWsDUMRUDn97yIT3LTxEVZogZ3_0i9M.

```
1 {  
2   "email": "angadisreenidhi@gmail.com",  
3   "password": "sreenidhi"  
4 }  
5
```

4) get user by email

The screenshot shows a Postman interface with the following details:

- Request URL:** `http://localhost:8765/user-microservice/api/user/angadisreenidhi@gmail.com`
- Method:** GET
- Authorization:** Bearer Token (selected)
- Response Status:** 200 OK
- Response Body (JSON):**

```
1 {  
2   "id": "69351f2967a7e5144f2db250",  
3   "name": "Sreenidhi",  
4   "gender": "Female",  
5   "age": 20,  
6   "password": "$2a$10$JuZrqBfuJEka8i5mqZDMA.H.lpxXahFcFchg244AESsAJhyoVSpIe",  
7   "email": "angadisreenidhi@gmail.com",  
8   "role": "USER"  
9 }
```

5) Adding inventory without authorization (showing one example)

The screenshot shows a Postman interface with the following details:

- Request URL:** `http://localhost:8765/flight-microservice/api/flight/airline/inventory/add`
- Method:** POST
- Body (JSON):**

```
1 {  
2   "airline": "AirAsia",  
3   "fromPlace": "Mumbai",  
4   "toPlace": "Hyderabad",  
5   "departureTime": "2025-12-04T06:00:00",  
6   "arrivalTime": "2025-12-04T09:00:00",  
7   "price": 4800,  
8   "totalSeats": 160,  
9   "availableSeats": 160  
10 }  
11
```
- Response Status:** 401 Unauthorized

6) adding inventory with authorization

The screenshot shows a Postman interface with the following details:

- Request URL:** `http://localhost:8765/flight-microservice/api/flight/airline/inventory/add`
- Method:** POST
- Authorization:** Bearer Token (selected)
- Response Status:** 201 Created
- Response Body (Raw):**

```
1 Flight added successfully with id: 69352313ace9642c67553a1d
```

7)Search flight

The screenshot shows the Postman interface with a POST request to `http://localhost:8765/flight-microservice/api/flight/search`. The request body is a JSON object:

```
1
2 {
3     "fromPlace": "Mumbai",
4     "toPlace": "Hyderabad",
5     "startTime": "2025-11-28T00:00:00",
6     "endTime": "2025-11-28T23:59:59"
7 }
```

The response status is 200 OK, with a response time of 423 ms and a response size of 504 B. The response body is a JSON array containing one flight record:

```
1 [ 2 { 3     "id": "69310ee6e5b8ddb69f9ec145", 4     "airline": "AirAsia", 5     "fromPlace": "Mumbai", 6     "toPlace": "Hyderabad", 7     "departureTime": "2025-11-28T06:00:00", 8     "arrivalTime": "2025-11-28T09:00:00", 9     "price": 4800.0, 10    "totalSeats": 160, 11    "availableSeats": 158 12 } 13 ]
```

8)get all flights

The screenshot shows the Postman interface with a GET request to `http://localhost:8765/flight-microservice/api/flight/getallflights`. The request body is a JSON object:

```
1 {
2     "email": "angadisreenidhi@gmail.com",
3     "password": "sreenidhi"
4 }
```

The response status is 200 OK, with a response time of 55 ms and a response size of 1.37 KB. The response body is a JSON array containing one flight record:

```
1 [ 2 { 3     "id": "692b3a4a6444e2b421e44d2b", 4     "airline": "AirAsia", 5     "fromPlace": "Banglore", 6     "toPlace": "Hyderabad", 7     "departureTime": "2025-11-28T06:00:00", 8     "arrivalTime": "2025-11-28T09:00:00", 9     "price": 4800.0, 10    "totalSeats": 160, 11    "availableSeats": 141 12 }, 13 ]
```

9)Search by airline

The screenshot shows the Postman interface with the following details:

- Request URL:** `http://localhost:8765/flight-microservice/api/flight/search/airline`
- Method:** POST
- Body:** JSON (raw)

```
1 {
2   "fromPlace": "Mumbai",
3   "toPlace": "Hyderabad",
4   "airline": "AirAsia"
5 }
```
- Response Status:** 200 OK
- Response Body:** (JSON)

```
3   {
4     "id": "69310ee6e5b8ddb69f9ec145",
5     "airline": "AirAsia",
6     "fromPlace": "Mumbai",
7     "toPlace": "Hyderabad",
8     "departureTime": "2025-11-28T06:00:00",
9     "arrivalTime": "2025-11-28T09:00:00",
10    "price": 4800.0,
11    "totalSeats": 160,
12    "availableSeats": 158
13  },
14  {
15    "id": "69314b89ec05baf4c110c11a",
16    "airline": "AirAsia",
17    "fromPlace": "Mumbai",
18    "toPlace": "Hyderabad",
19    "departureTime": "2025-12-04T06:00:00",
20    "arrivalTime": "2025-12-04T09:00:00",
21    "price": 4800.0,
22    "totalSeats": 160,
```

10)Booking

The screenshot shows the Postman interface with the following details:

- Request URL:** `http://localhost:8765/booking-microservice/api/flight/booking/69314b89ec05baf4c110c11a`
- Method:** POST
- Authorization:** Bearer Token
- Body:** Raw (Raw)

```
1 PNR: dbeac786
```
- Response Status:** 200 OK
- Response Body:** (Raw)

```
1 PNR: dbeac786
```

11)Ticket by PNR

The screenshot shows a Postman request for a ticket by PNR. The URL is `http://localhost:8765/booking-microservice/api/flight/ticket/dbeac786`. The response is a 200 OK status with a total size of 748 B. The JSON response is as follows:

```
2 "id": "69353007d5c49615a6429808",
3 "pnr": "dbeac786",
4 "userId": null,
5 "departureFlightId": "69314b89ec05baf4c110c11a",
6 "returnFlightId": null,
7 "tripType": "ONE WAY",
8 "bookingTime": "2025-12-07T13:13:03.506",
9 "seatsBooked": "12A",
10 "userEmail": "angadisreenidhi@gmail.com",
11 "mealType": null,
12 "totalPrice": 4800.0,
13 "canceled": true,
14 "passengers": [
15   {
16     "id": "69353008d5c49615a6429809",
17     "name": "Sreenidhi",
18     "gender": null,
19     "age": 20,
20     "seatNumber": "12A",
21     "mealPreference": null,
```

12)get booking history

The screenshot shows a Postman request for booking history. The URL is `http://localhost:8765/booking-microservice/api/flight/booking/history/angadisreenidhi@gmail.com`. The response is a 200 OK status with a total size of 6 KB. The JSON response is as follows:

```
2 {
3   "id": "692c06db78a5e26842e5da35",
4   "pnr": "d66c32a6",
5   "userId": null,
6   "departureFlightId": "692b3a4a6444e2b421e44d2b",
7   "returnFlightId": null,
8   "tripType": "ONE WAY",
9   "bookingTime": "2025-11-30T14:26:59.446",
10  "seatsBooked": "12A",
11  "userEmail": "angadisreenidhi@gmail.com",
12  "mealType": null,
13  "totalPrice": 4800.0,
14  "canceled": false,
15  "passengers": null
16 },
17 {
18   "id": "692c3a017d98e412aefb2478",
19   "pnr": "71c85860",
```

13)Cancel Ticket

The screenshot shows the Postman interface with the following details:

- HTTP Method:** DELETE
- URL:** http://localhost:8082/api/flight/booking/cancel/dbeac786
- Authorization:** Bearer Token (selected)
- Response Status:** 200 OK
- Response Body:** "Cancelled Successfully"

14)cancel ticket validation

The screenshot shows the Postman interface with the following details:

- HTTP Method:** DELETE
- URL:** http://localhost:8765/booking-microservice/api/flight/booking/cancel/9cb8b16a
- Authorization:** Bearer Token (selected)
- Response Status:** 200 OK
- Response Body:** "Cannot cancel ticket within 24 hours of departure"