

# **FLIGHT BOOKING SYSTEM**

**OUTPUTS, REPORTS AND ANALYSIS**

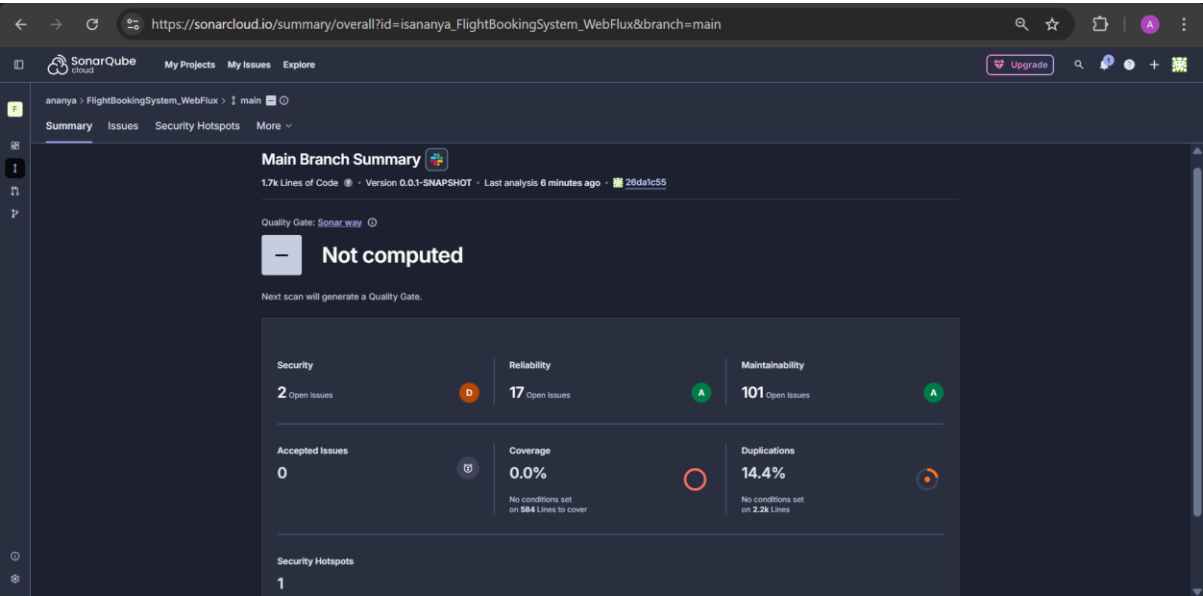
**BY ANANYA NAYAK**

# SonarQube Analysis Reports

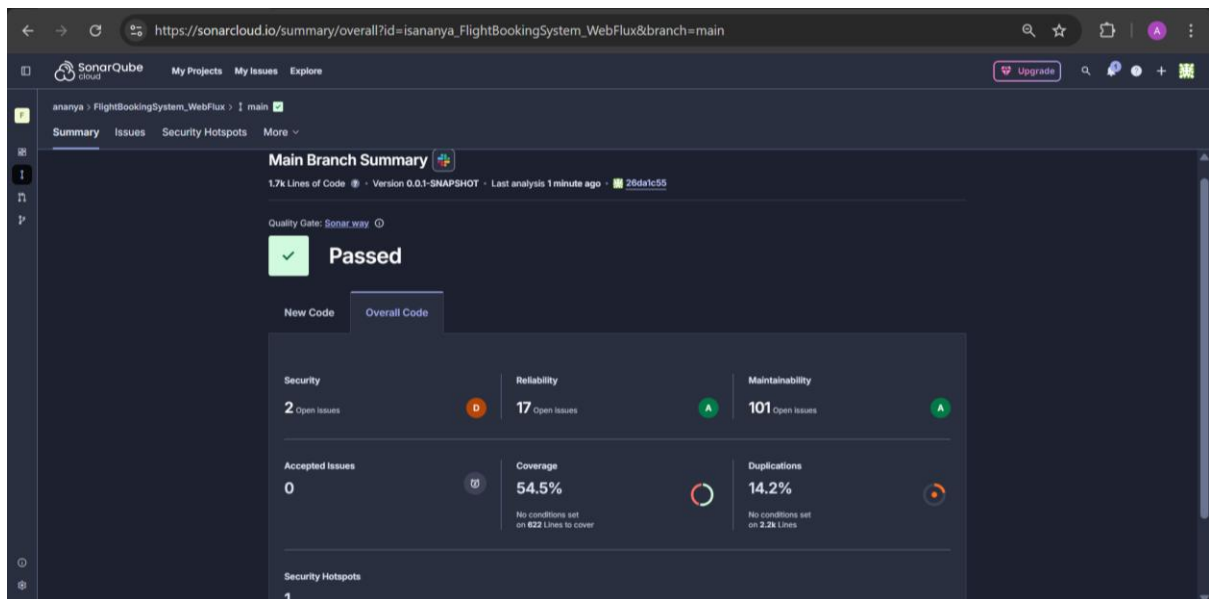
## Running analysis :

```
C:\Windows\System32\cmd.e x + v
[INFO] 22:26:30.011 Architecture analysis is enabled with the following features: legacy
[INFO] 22:26:30.015 * Protobuf reading starting | memory total=1384 | free=1001 | used=382 (MB)
[INFO] 22:26:30.016 * Reading SonarArchitecture IR data from directory "D:\CHUBB_Training\FlightBookingSystem_WebFlux\FlightBookingSystem_WebFlux\target\sonar\architecture\java"
[INFO] 22:26:30.794 * Files successfully loaded: "45" out of "45"
[INFO] 22:26:30.795 * Protobuf reading complete | memory total=1384 | free=982 | used=401 (MB)
[INFO] 22:26:30.827 * Build architecture.graph.java.namespace_graph.default_perspective_hierarchy graph complete (filtered=false) | memory total=1384 | free=974 | used=409 (MB)
[INFO] 22:26:30.857 Sensor JavaArchitectureSensor [architecture] (done) | time=875ms
[INFO] 22:26:30.857 Sensor Zero Coverage Sensor
[INFO] 22:26:30.861 Sensor Zero Coverage Sensor (done) | time=4ms
[INFO] 22:26:30.862 Sensor Java CPD Block Indexer
[INFO] 22:26:30.950 Sensor Java CPD Block Indexer (done) | time=88ms
[INFO] 22:26:30.960 Inferred api base url 'https://api.sonarcloud.io' from host url 'https://sonarcloud.io'.
[INFO] 22:26:31.635 ----- Gather SCA dependencies on project
[INFO] 22:26:31.635 Checking if SCA is enabled for organization ananya
[INFO] 22:26:32.014 Dependency analysis skipped
[INFO] 22:26:32.017 CPD Executor 17 files had no CPD blocks
[INFO] 22:26:32.017 CPD Executor Calculating CPD for 22 files
[INFO] 22:26:32.034 CPD Executor CPD calculation finished (done) | time=16ms
[INFO] 22:26:32.056 SCM Publisher SCM provider for this project is: git
[INFO] 22:26:32.057 SCM Publisher 8 source files to be analyzed
[INFO] 22:26:32.735 SCM Publisher 8/8 source files have been analyzed (done) | time=677ms
[INFO] 22:26:33.822 Analysis report generated in 244ms, dir size=468 KB
[INFO] 22:26:34.080 Analysis report compressed in 224ms, zip size=168 KB
[INFO] 22:26:35.063 Analysis report uploaded in 983ms
[INFO] 22:26:35.066 ANALYSIS SUCCESSFUL, you can find the results at: https://sonarcloud.io/dashboard?id=isananya_FlightBookingSystem_WebFlux
[INFO] 22:26:35.066 Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
[INFO] 22:26:35.066 More about the report processing at https://sonarcloud.io/api/ce/task?id=AZq2y7JYj9xZ-iXtxRZ0
[INFO] 22:26:35.069 ----- Upload SCA dependency files
[INFO] 22:26:36.632 Sensor cache published successfully
[INFO] 22:26:36.706 Analysis total time: 31.803 s
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 01:15 min
[INFO] Finished at: 2025-11-24T22:26:36+05:30
[INFO] -----
D:\CHUBB_Training\FlightBookingSystem_WebFlux\FlightBookingSystem_WebFlux>
```

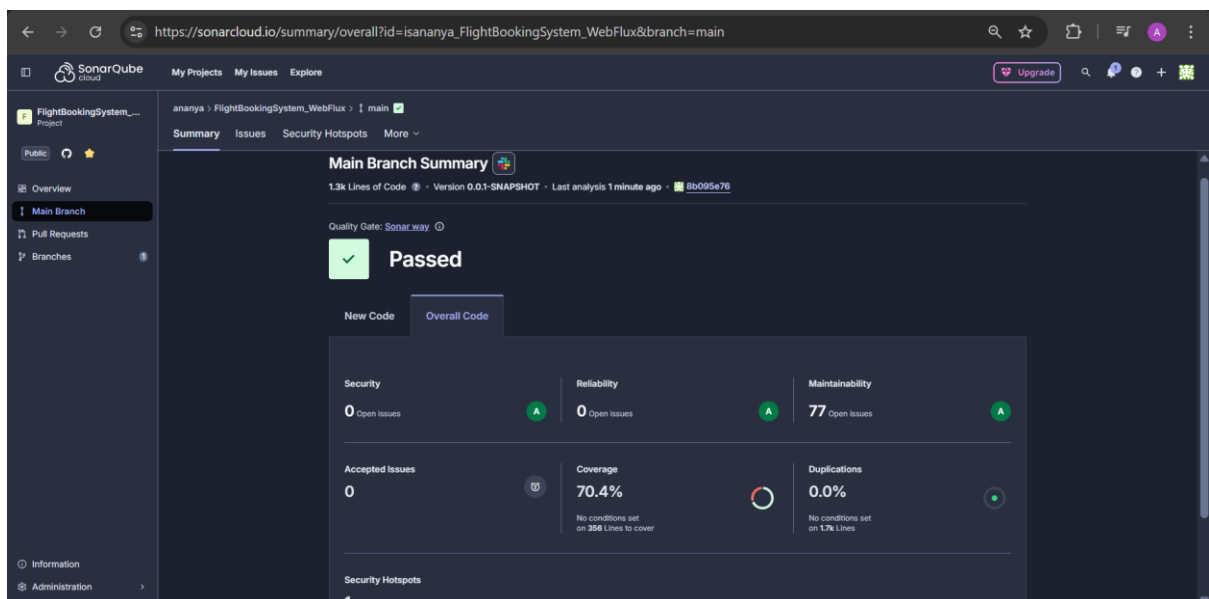
## Before Improvement :



After setting up coverage (jacoco) :



Final Analysis



# JMeter Summary Reports

20 samples :

Summary Report

Name:

Summary Report

Comments:

Write results to file / Read from file

Filename

Browse...

Log/Display Only: ☐ Errors ☐ Successes

Configure

| Label             | # Samples ↓ | Average | Min | Max | Std. Dev. | Error % | Throughput | Received KB... | Sent KB/sec | Avg. Bytes |
|-------------------|-------------|---------|-----|-----|-----------|---------|------------|----------------|-------------|------------|
| Add User          | 20          | 5       | 4   | 16  | 2.55      | 0.00%   | 20.9/sec   | 2.21           | 7.70        | 108.0      |
| Add Flight        | 20          | 3       | 2   | 5   | 1.02      | 0.00%   | 21.2/sec   | 1.83           | 7.99        | 88.5       |
| Add Schedule      | 20          | 4       | 2   | 8   | 1.20      | 100.00% | 21.2/sec   | 244.19         | 7.69        | 11815.0    |
| Search Flights    | 20          | 2       | 2   | 4   | 0.62      | 0.00%   | 21.2/sec   | 8.16           | 7.34        | 395.0      |
| Add Booking       | 20          | 4       | 3   | 6   | 0.81      | 0.00%   | 21.1/sec   | 0.78           | 14.81       | 38.0       |
| Get ticket usi... | 20          | 6       | 4   | 10  | 1.32      | 0.00%   | 21.1/sec   | 11.41          | 3.68        | 555.0      |
| Booking His...    | 20          | 7       | 5   | 10  | 1.26      | 0.00%   | 21.1/sec   | 11.41          | 4.03        | 555.0      |
| Cancel Booki...   | 20          | 2       | 1   | 4   | 0.62      | 100.00% | 21.2/sec   | 0.97           | 4.32        | 47.0       |
| TOTAL             | 160         | 4       | 1   | 16  | 2.14      | 25.00%  | 162.3/sec  | 269.43         | 55.30       | 1700.2     |

50 samples:

Summary Report

Name:

Summary Report

Comments:

Write results to file / Read from file

Filename

Browse...

Log/Display Only:

☐ Errors

☐ Successes

Configure

| Label             | # Samples ↓ | Average | Min | Max | Std. Dev. | Error % | Throughput | Received KB... | Sent KB/sec | Avg. Bytes |
|-------------------|-------------|---------|-----|-----|-----------|---------|------------|----------------|-------------|------------|
| Add User          | 50          | 5       | 3   | 17  | 1.88      | 0.00%   | 50.8/sec   | 5.35           | 18.72       | 108.0      |
| Add Flight        | 50          | 4       | 3   | 12  | 1.49      | 0.00%   | 51.4/sec   | 4.46           | 19.38       | 88.8       |
| Add Schedule      | 50          | 5       | 4   | 8   | 1.21      | 100.00% | 51.5/sec   | 594.13         | 18.71       | 11815.0    |
| Search Flights    | 50          | 3       | 2   | 5   | 0.75      | 0.00%   | 51.7/sec   | 19.92          | 17.91       | 395.0      |
| Add Booking       | 50          | 5       | 4   | 8   | 1.04      | 0.00%   | 51.7/sec   | 1.92           | 36.22       | 38.0       |
| Get ticket usi... | 50          | 7       | 5   | 14  | 1.55      | 0.00%   | 51.7/sec   | 28.00          | 9.03        | 555.0      |
| Booking His...    | 50          | 7       | 5   | 12  | 1.47      | 0.00%   | 51.8/sec   | 28.05          | 9.91        | 555.0      |
| Cancel Booki...   | 50          | 2       | 1   | 6   | 0.89      | 100.00% | 52.2/sec   | 2.40           | 10.66       | 47.0       |
| TOTAL             | 400         | 5       | 1   | 17  | 2.24      | 25.00%  | 395.3/sec  | 656.28         | 134.73      | 1700.2     |

100 Samples :

Summary Report

Name:

Summary Report

Comments:

Write results to file / Read from file

Filename

Browse...

Log/Display Only:

☐ Errors

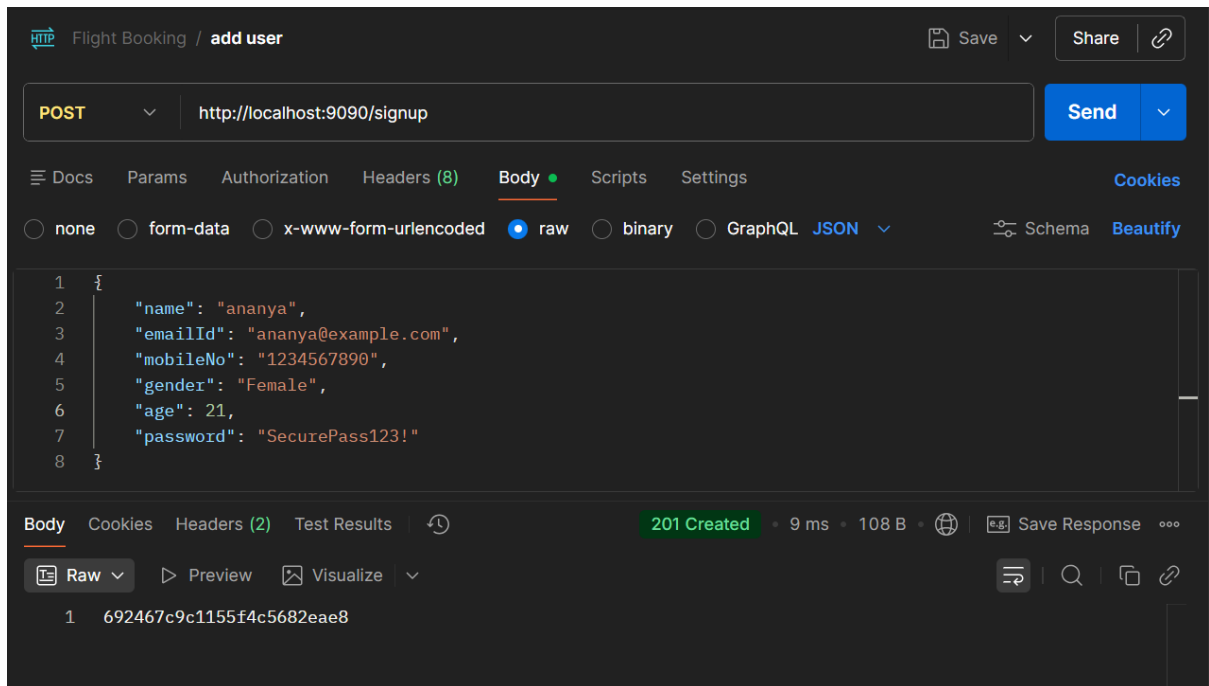
☐ Successes

Configure

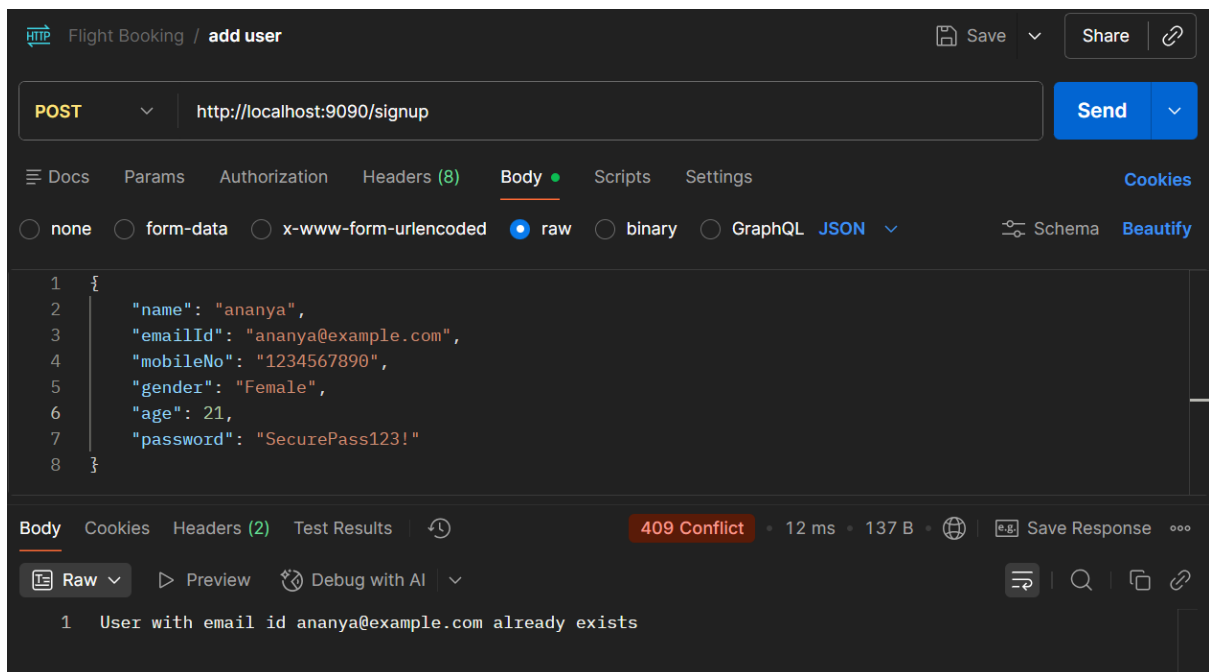
| Label             | # Samples ↓ | Average | Min | Max | Std. Dev. | Error % | Throughput | Received KB... | Sent KB/sec | Avg. Bytes |
|-------------------|-------------|---------|-----|-----|-----------|---------|------------|----------------|-------------|------------|
| Add User          | 100         | 17      | 3   | 63  | 12.71     | 1.00%   | 97.3/sec   | 21.31          | 35.89       | 224.4      |
| Add Flight        | 100         | 15      | 3   | 58  | 11.80     | 0.00%   | 95.8/sec   | 8.32           | 36.10       | 88.9       |
| Add Schedule      | 100         | 16      | 3   | 39  | 10.63     | 100.00% | 93.0/sec   | 1073.31        | 33.79       | 11815.0    |
| Search Flights    | 100         | 10      | 2   | 51  | 8.14      | 0.00%   | 92.5/sec   | 35.68          | 32.07       | 395.0      |
| Add Booking       | 100         | 23      | 4   | 61  | 15.03     | 0.00%   | 90.3/sec   | 3.35           | 63.34       | 38.0       |
| Get ticket usi... | 100         | 28      | 6   | 79  | 18.09     | 0.00%   | 87.3/sec   | 47.29          | 15.25       | 555.0      |
| Booking His...    | 100         | 27      | 6   | 96  | 18.98     | 0.00%   | 86.7/sec   | 46.97          | 16.59       | 555.0      |
| Cancel Booki...   | 100         | 8       | 2   | 46  | 6.96      | 100.00% | 86.8/sec   | 3.98           | 17.72       | 47.0       |
| TOTAL             | 800         | 18      | 2   | 96  | 15.10     | 25.12%  | 664.5/sec  | 1112.69        | 226.52      | 1714.8     |

# Postman Screenshots

## Add User:



## Add User fail:



## Add Flight:

The screenshot shows a REST client interface for a "Flight Booking" API. The endpoint is `http://localhost:9090/airline/route/add` with a `POST` method. The request body is a JSON object: `{ "flightNumber": "AI102", "sourceAirport": "BOM", "destinationAirport": "DEL", "departureTime": "08:30:00", "arrivalTime": "10:45:00", "duration": "PT2H15M" }`. The response status is `201 Created` with a response time of 16 ms and a body size of 88 B. The response body is `AI102`.

Flight Booking / add flight

POST http://localhost:9090/airline/route/add

Send

Docs Params Authorization Headers (9) Body Scripts Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Schema Beautify

```
1 {
2   "flightNumber": "AI102",
3   "sourceAirport": "BOM",
4   "destinationAirport": "DEL",
5   "departureTime": "08:30:00",
6   "arrivalTime": "10:45:00",
7   "duration": "PT2H15M"
8 }
```

Body Cookies Headers (2) Test Results 201 Created • 16 ms • 88 B Save Response

Raw Preview Visualize

```
1 AI102
```

## Add flight fail:

The screenshot shows the same REST client interface as above, but the response status is `409 Conflict` with a response time of 31 ms and a body size of 119 B. The response body is `Flight Number AI102 already exists`.

Flight Booking / add flight

POST http://localhost:9090/airline/route/add

Send

Docs Params Authorization Headers (9) Body Scripts Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Schema Beautify

```
1 {
2   "flightNumber": "AI102",
3   "sourceAirport": "BOM",
4   "destinationAirport": "DEL",
5   "departureTime": "08:30:00",
6   "arrivalTime": "10:45:00",
7   "duration": "PT2H15M"
8 }
```

Body Cookies Headers (2) Test Results 409 Conflict • 31 ms • 119 B Save Response

Raw Preview Debug with AI

```
1 Flight Number AI102 already exists
```

## Add schedule:

The screenshot shows a REST client interface for a "Flight Booking" application. The endpoint is `http://localhost:9090/airline/inventory/add` with a `POST` method. The request body is a JSON object: `{ "airlineName": "Air India Express", "departureDate": "2025-11-27", "basePrice": 4500.0, "totalSeats": 180, "availableSeats": 180, "flightNumber": "AI102" }`. The response status is `201 Created` with a response time of 7 ms and a body size of 108 B. The response body is a long alphanumeric string: `69246880c1155f4c5682eae9`.

Flight Booking / add schedule

Save Share

POST http://localhost:9090/airline/inventory/add Send

Docs Params Authorization Headers (9) Body Scripts Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Schema Beautify

```
1 {
2   "airlineName": "Air India Express",
3   "departureDate": "2025-11-27",
4   "basePrice": 4500.0,
5   "totalSeats": 180,
6   "availableSeats": 180,
7   "flightNumber": "AI102"
8 }
```

Body Cookies Headers (2) Test Results 201 Created • 7 ms • 108 B Save Response

Raw Preview Visualize

```
1 69246880c1155f4c5682eae9
```

## Add schedule fail:

The screenshot shows the same REST client interface as before, but the request has failed. The response status is `409 Conflict` with a response time of 15 ms and a body size of 149 B. The response body is a message: `Schedule with Flight NumberAI102 already scheduled on 2025-11-27`.

Flight Booking / add schedule

Save Share

POST http://localhost:9090/airline/inventory/add Send

Docs Params Authorization Headers (9) Body Scripts Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Schema Beautify

```
1 {
2   "airlineName": "Air India Express",
3   "departureDate": "2025-11-27",
4   "basePrice": 4500.0,
5   "totalSeats": 180,
6   "availableSeats": 180,
7   "flightNumber": "AI102"
8 }
```

Body Cookies Headers (2) Test Results 409 Conflict • 15 ms • 149 B Save Response

Raw Preview Debug with AI

```
1 Schedule with Flight NumberAI102 already scheduled on 2025-11-27
```

## Search flight:

The screenshot shows a REST client interface for a "Flight Booking" API. The endpoint is `http://localhost:9090/flights/search` with a `POST` method. The request body is a JSON object with the following fields: `sourceAirport` ("BOM"), `destinationAirport` ("DEL"), `roundTrip` (false), `departureDate` ("2025-11-27"), and `passengerCount` (2). The response status is `200 OK` with a response time of 5 ms and a body size of 395 B. The response body is a JSON object with the following fields: `departureCount` (1), `returnCount` (0), `departure` (an array of objects), and `arrivalTime` ("10:45:00"). The `departure` array contains one object with `airlineName` ("Air India Express") and `basePrice` (4500.0).

```
1 {
2   "sourceAirport": "BOM",
3   "destinationAirport": "DEL",
4   "roundTrip": false,
5   "departureDate": "2025-11-27",
6   "passengerCount": 2
7 }
```

```
1 {
2   "departureCount": 1,
3   "returnCount": 0,
4   "departure": [
5     {
6       "airlineName": "Air India Express",
7       "arrivalTime": "10:45:00",
8       "basePrice": 4500.0,
9     }
10  ]
11 }
```

## Booking:

The screenshot shows a REST client interface for a "Flight Booking" API. The endpoint is `http://localhost:9090/booking` with a `POST` method. The request body is a JSON object with the following fields: `emailId` ("user@gmail.com"), `departureScheduleId` ("692466ecc1155f4c5682ea81"), `returnScheduleId` ("6923f9cb4a5569b042ffaa45"), `roundTrip` (false), `passengerCount` (2), and `passengers` (an array of objects). The `passengers` array contains one object with `firstName` ("ananya"), `lastName` ("nayak"), `age` (21), `departureSeatNumber` ("2B"), `returnSeatNumber` ("2B"), and `gender` ("FEMALE"). The response status is `201 Created` with a response time of 11 ms and a body size of 116 B. The response body is a plain text message: "Booking Successful! PNR : XUg5FM".

```
1 {
2   "emailId": "user@gmail.com",
3   "departureScheduleId": "692466ecc1155f4c5682ea81",
4   // "returnScheduleId" : "6923f9cb4a5569b042ffaa45",
5   "roundTrip": false,
6   "passengerCount": 2,
7   "passengers": [
8     {
9       "firstName": "ananya",
10      "lastName": "nayak",
11      "age": 21,
12      "departureSeatNumber": "2B",
13      // "returnSeatNumber": "2B",
14      "gender": "FEMALE",
15      "age": 21,
16      "departureScheduleId": "692466ecc1155f4c5682ea81",
17      "returnScheduleId": "6923f9cb4a5569b042ffaa45",
18      "passengerCount": 2,
19      "roundTrip": false
20    }
21  ]
22 }
```

```
1 Booking Successful! PNR : XUg5FM
```

## Seat not available while booking:

The screenshot shows a REST client interface for a "Flight Booking" API. The request is a POST to `http://localhost:9090/booking`. The body is a JSON object with the following fields: `firstName: "Riya"`, `lastName: "Rao"`, `age: 22`, `departureSeatNumber: "2D"`, `returnSeatNumber: "2D"` (commented out), `gender: "FEMALE"`, and `mealOption: "NONVEG"`. The response is a `400 Bad Request` with a message: "Seat Number 2D not available on flight 692466ecc1155f4c5682ea81." The status bar shows a response time of 7 ms and a body size of 152 B.

```
POST http://localhost:9090/booking

{
  "firstName": "Riya",
  "lastName": "Rao",
  "age": 22,
  "departureSeatNumber": "2D",
  // "returnSeatNumber": "2D",
  "gender": "FEMALE",
  "mealOption": "NONVEG"
}
```

400 Bad Request • 7 ms • 152 B

1 Seat Number 2D not available on flight 692466ecc1155f4c5682ea81.

## Get tickets by pnr:

The screenshot shows a REST client interface for a "Flight Booking" API. The request is a GET to `http://localhost:9090/ticket/2ubYPL`. The response is a `200 OK` with a JSON array containing one ticket object. The ticket details are: `firstName: "ananya"`, `lastName: "nayak"`, `age: 21`, `gender: "FEMALE"`, `seatNumber: "2A"`, `mealOption: "VEG"`, `date: "2025-11-29"`, `fromAirport: "BOM"`, `toAirport: "DEL"`, `fromTime: "08:30:00"`, `toTime: "10:45:00"`, and `status: "CONFIRMED"`. The status bar shows a response time of 27 ms and a body size of 533 B.

```
GET http://localhost:9090/ticket/2ubYPL

This request does not have a body
```

200 OK • 27 ms • 533 B

```
[
  {
    "firstName": "ananya",
    "lastName": "nayak",
    "age": 21,
    "gender": "FEMALE",
    "seatNumber": "2A",
    "mealOption": "VEG",
    "date": "2025-11-29",
    "fromAirport": "BOM",
    "toAirport": "DEL",
    "fromTime": "08:30:00",
    "toTime": "10:45:00",
    "status": "CONFIRMED"
  }
]
```

Booking not found (pnr doesn't exist)

Flight Booking / get ticket using pnr

SaveShare

GET

http://localhost:9090/ticket/2uXYZZ

Send

DocsParamsAuthorizationHeaders (6)BodyScriptsSettings

Cookies

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

This request does not have a body

BodyCookiesHeaders (2)Test Results

404 Not Found • 4 ms • 115 B • Save Response

Raw

Preview

Debug with AI

1

Booking not found with 2uXYZZ

Booking history by email:

Flight Booking / booking history

SaveShare

GET

http://localhost:9090/booking/history/user@gmail.com

Send

DocsParamsAuthorizationHeaders (6)BodyScriptsSettings

Cookies

BodyCookiesHeaders (2)Test Results

200 OK • 8 ms • 986 B • Save Response

JSON

Preview

Visualize

1

[

2

{

3

"firstName": "Riya",

4

"lastName": "Rao",

5

"age": 22,

6

"gender": "FEMALE",

7

"seatNumber": "2D",

8

"mealOption": "NONVEG",

9

"date": "2025-11-29",

10

"fromAirport": "BOM",

11

"toAirport": "DEL",

12

"fromTime": "08:30:00",

13

"toTime": "10:45:00",

14

"status": "CONFIRMED"

15

},

16

{

17

"firstName": "ananya",

18

"lastName": "nayak",

19

"age": 21,

## Boooking not found using email:

The screenshot shows a REST client interface for a "Flight Booking" API. The endpoint is `http://localhost:9090/booking/history/user123@gmail.com` and the method is `GET`. The response is a `404 Not Found` status with a response time of 6 ms and a body size of 126 B. The response body, viewed in "Raw" mode, contains the text: "1 Booking not found with user123@gmail.com".

## Delete booking :

The screenshot shows a REST client interface for a "Flight Booking" API. The endpoint is `http://localhost:9090/booking/cancel/XUg5FM` and the method is `DELETE`. The response is a `200 OK` status with a response time of 11 ms and a body size of 38 B. The response body, viewed in "Raw" mode, contains the text: "1".