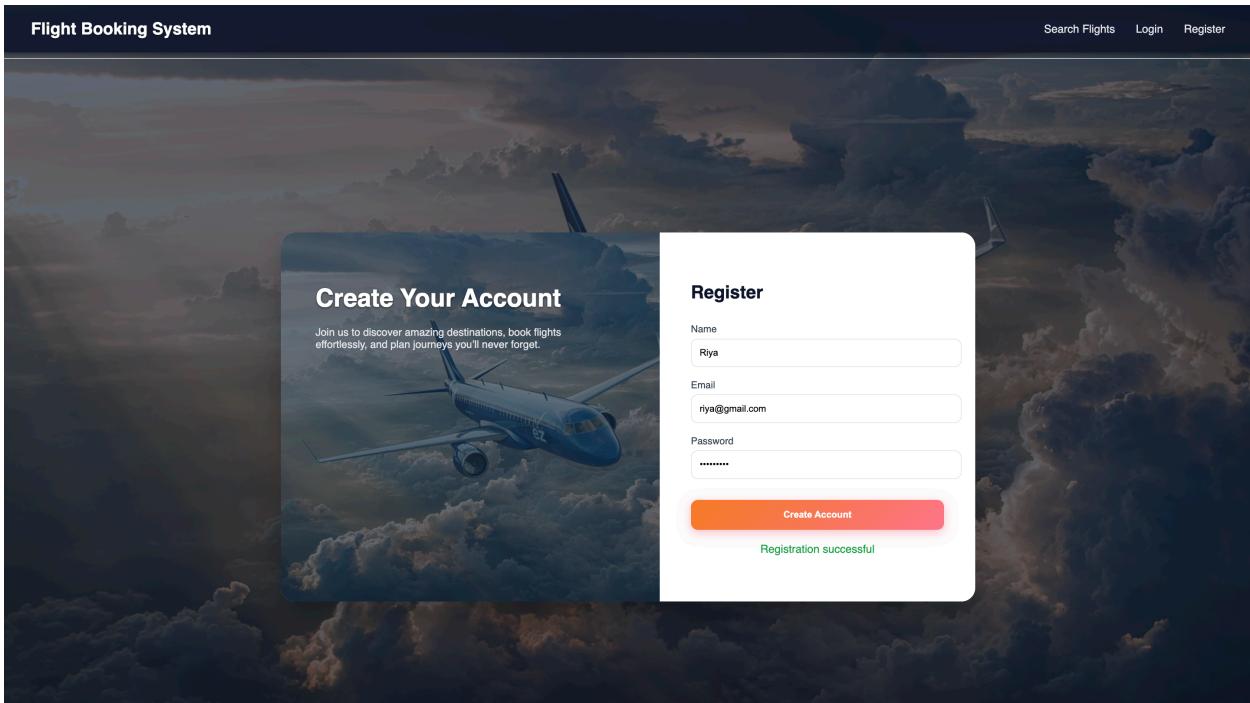
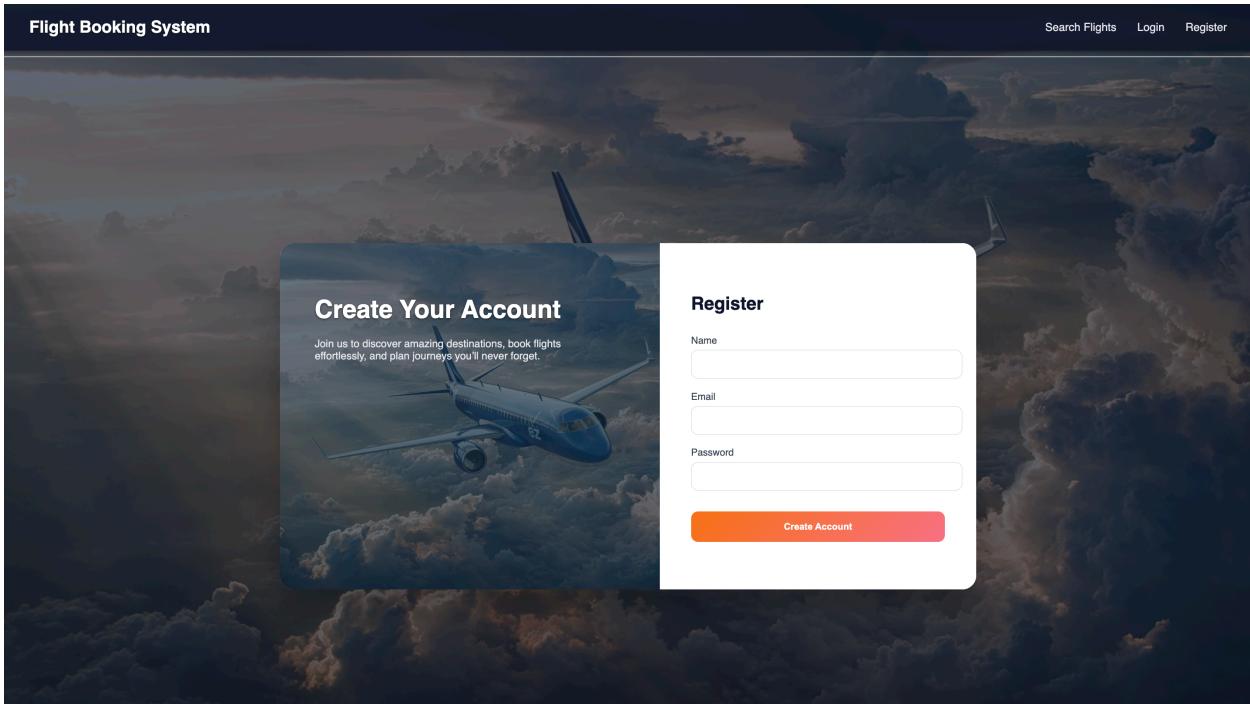
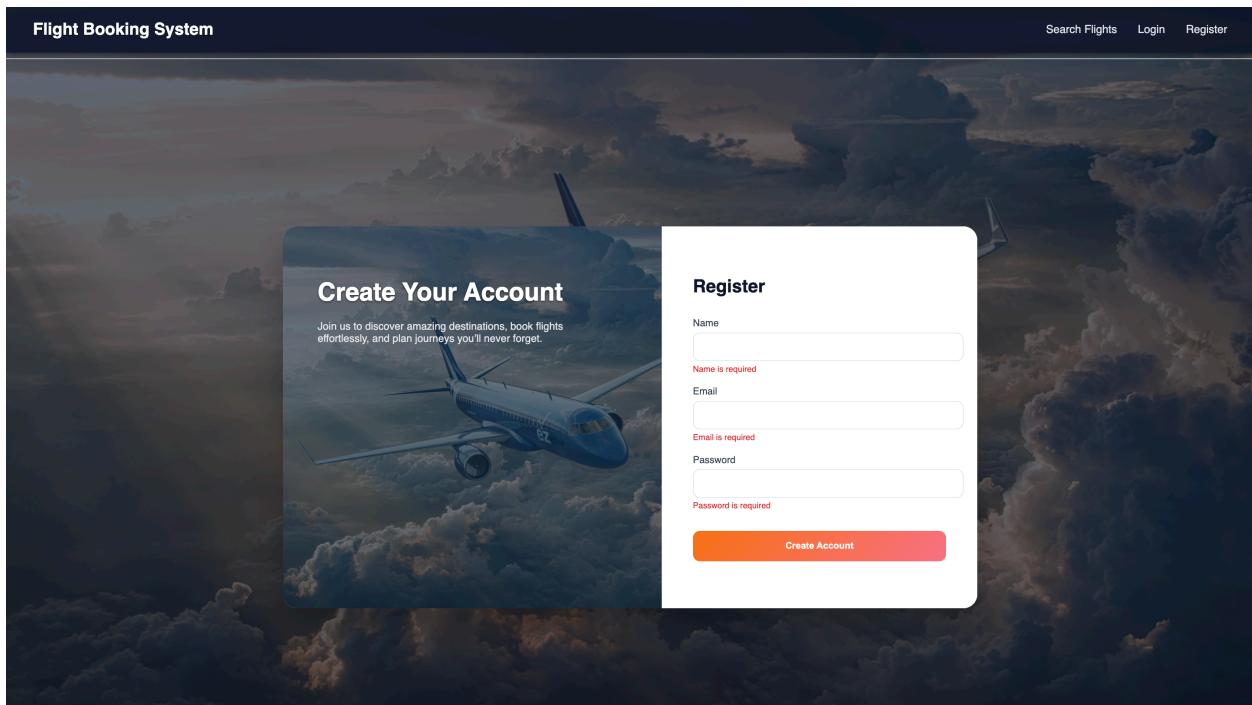


Flight Booking System-UI

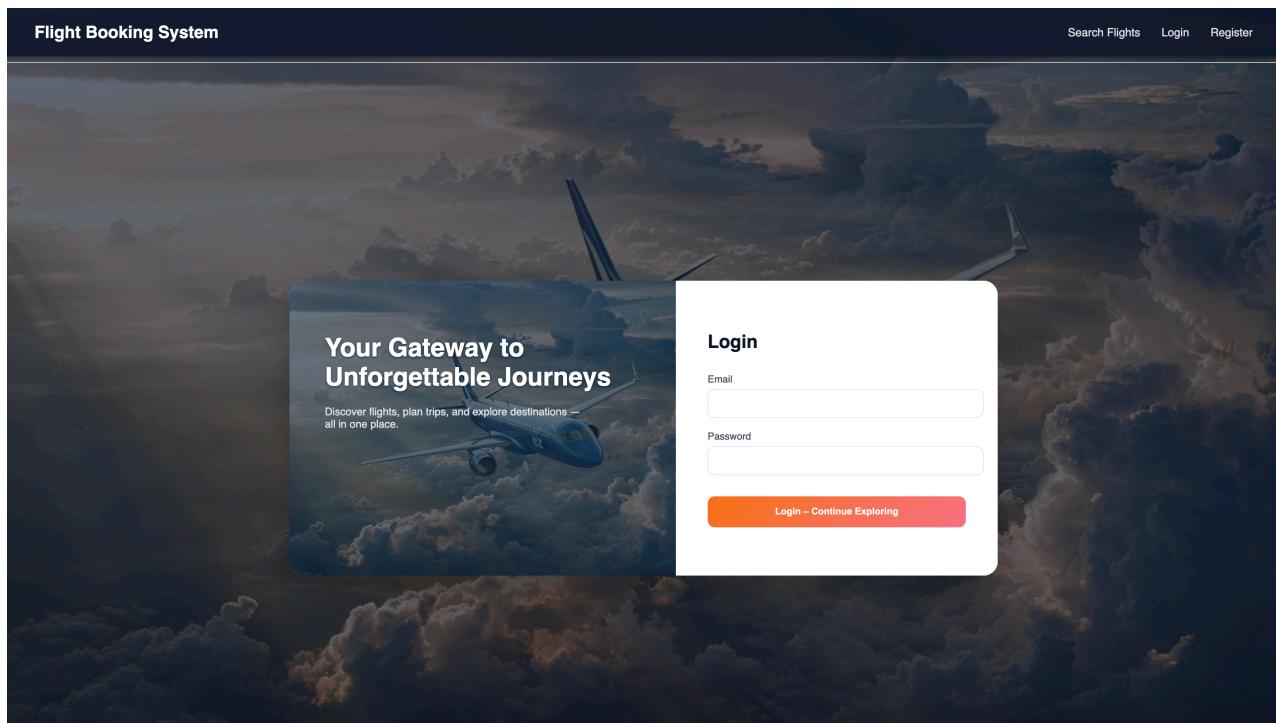
Register:

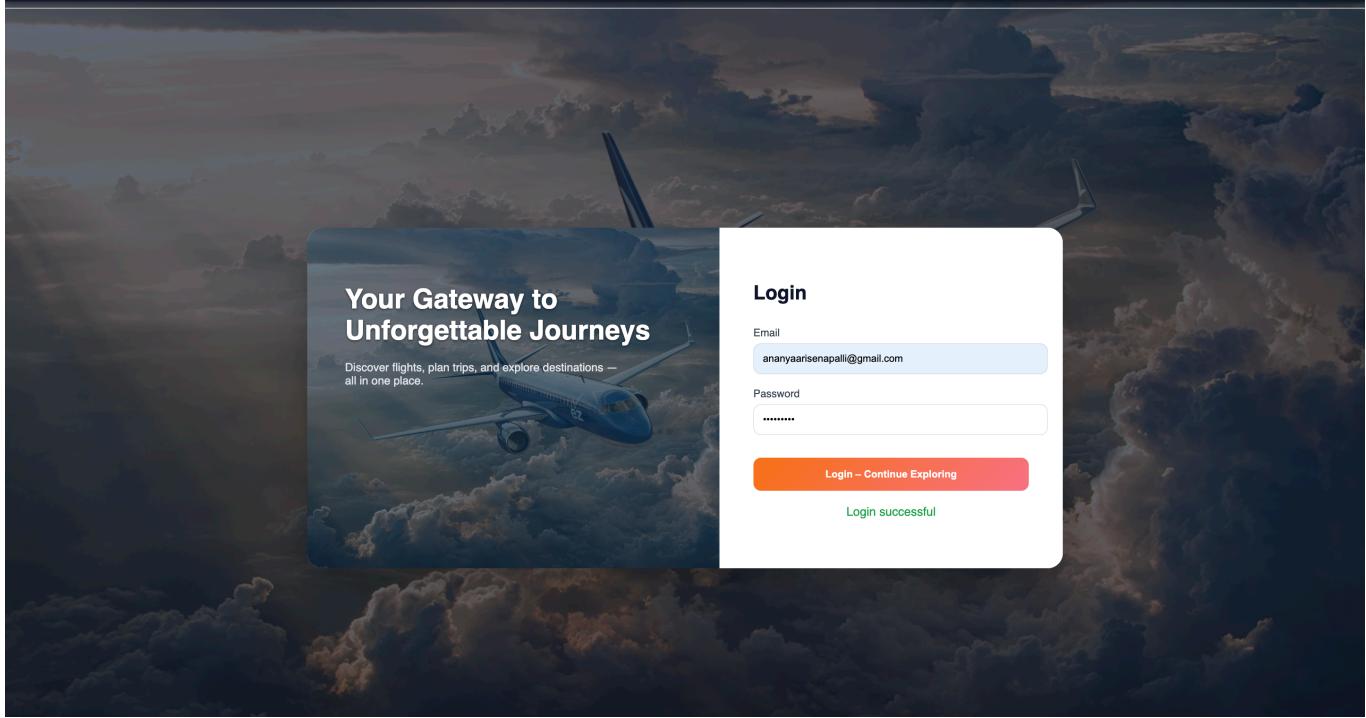


Validations on Registration:

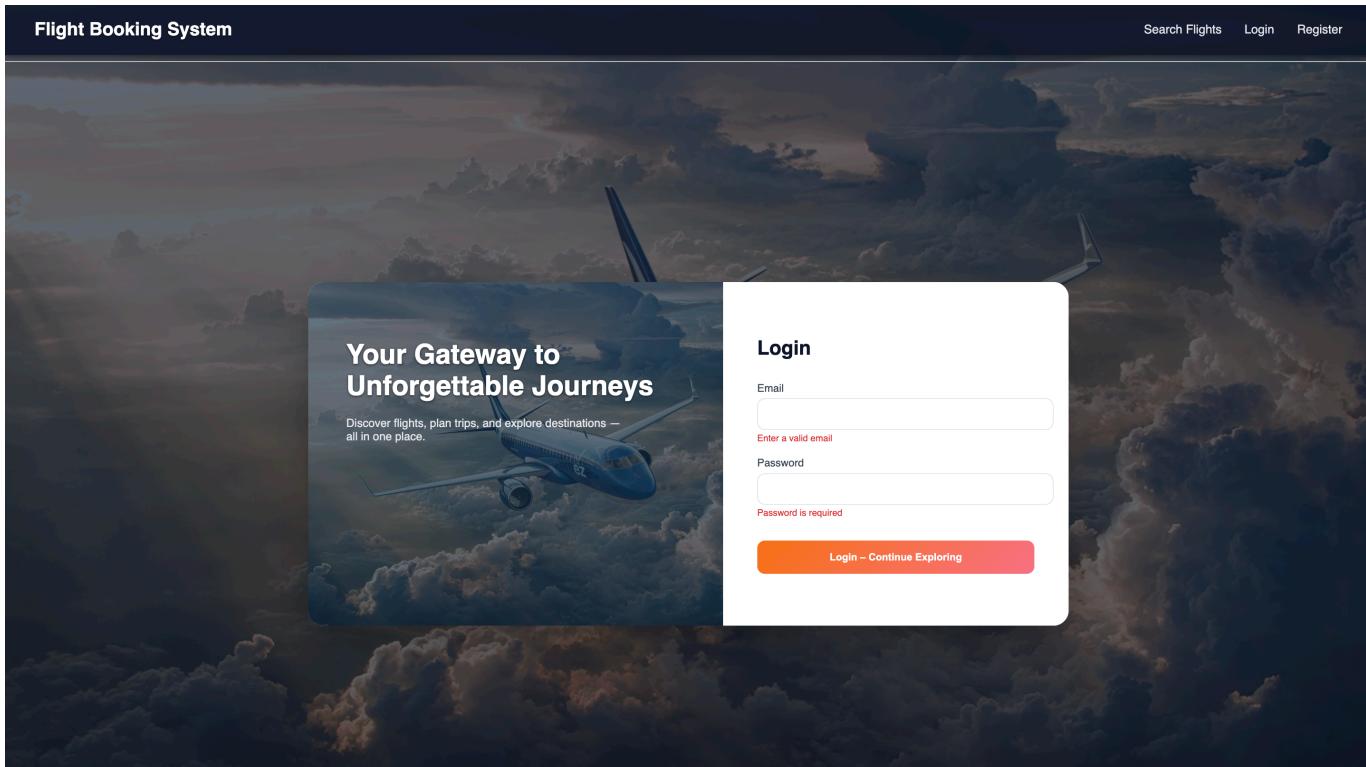


Login:





Validations on Login:



Search Flights:

Flight Booking System

Search Flights Sign Out

Your Search

From: HYD

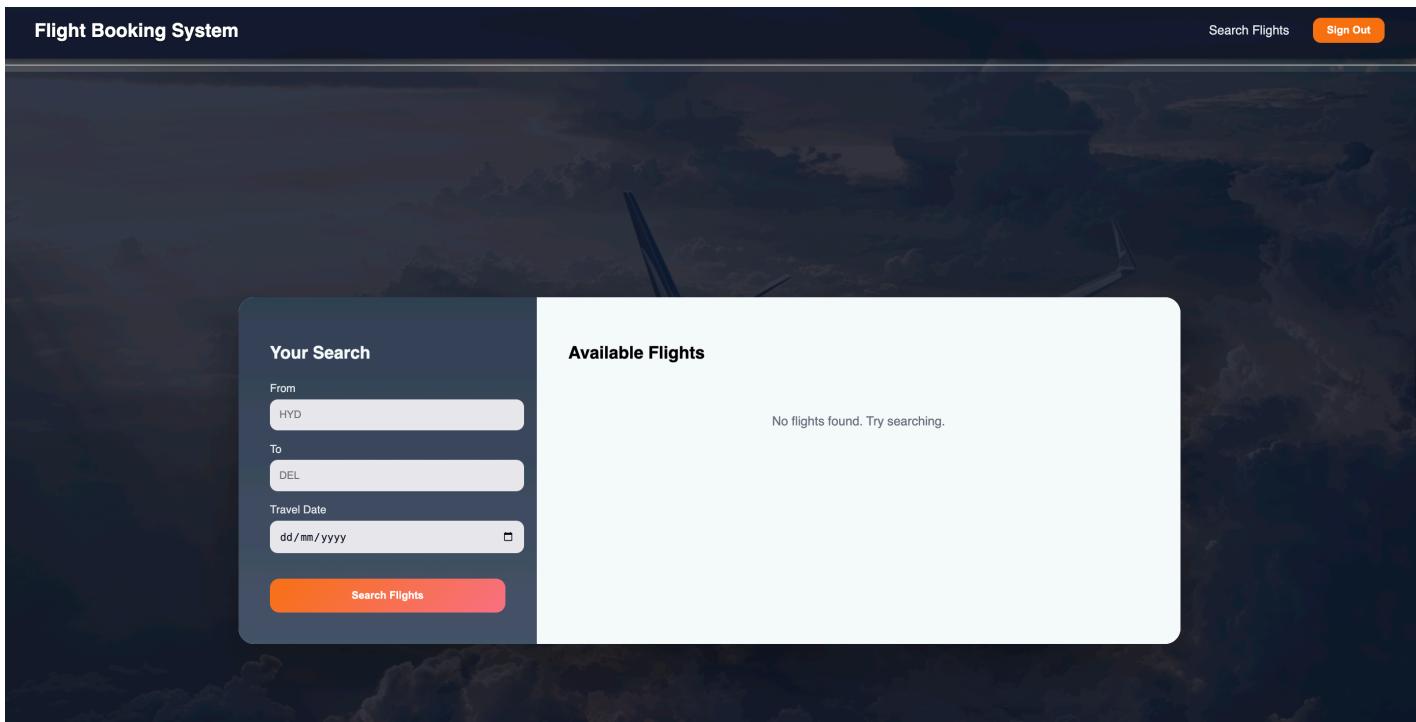
To: DEL

Travel Date: dd/mm/yyyy

Search Flights

Available Flights

No flights found. Try searching.



Flight Booking System

Search Flights Sign Out

Your Search

From: HYD

To: PJB

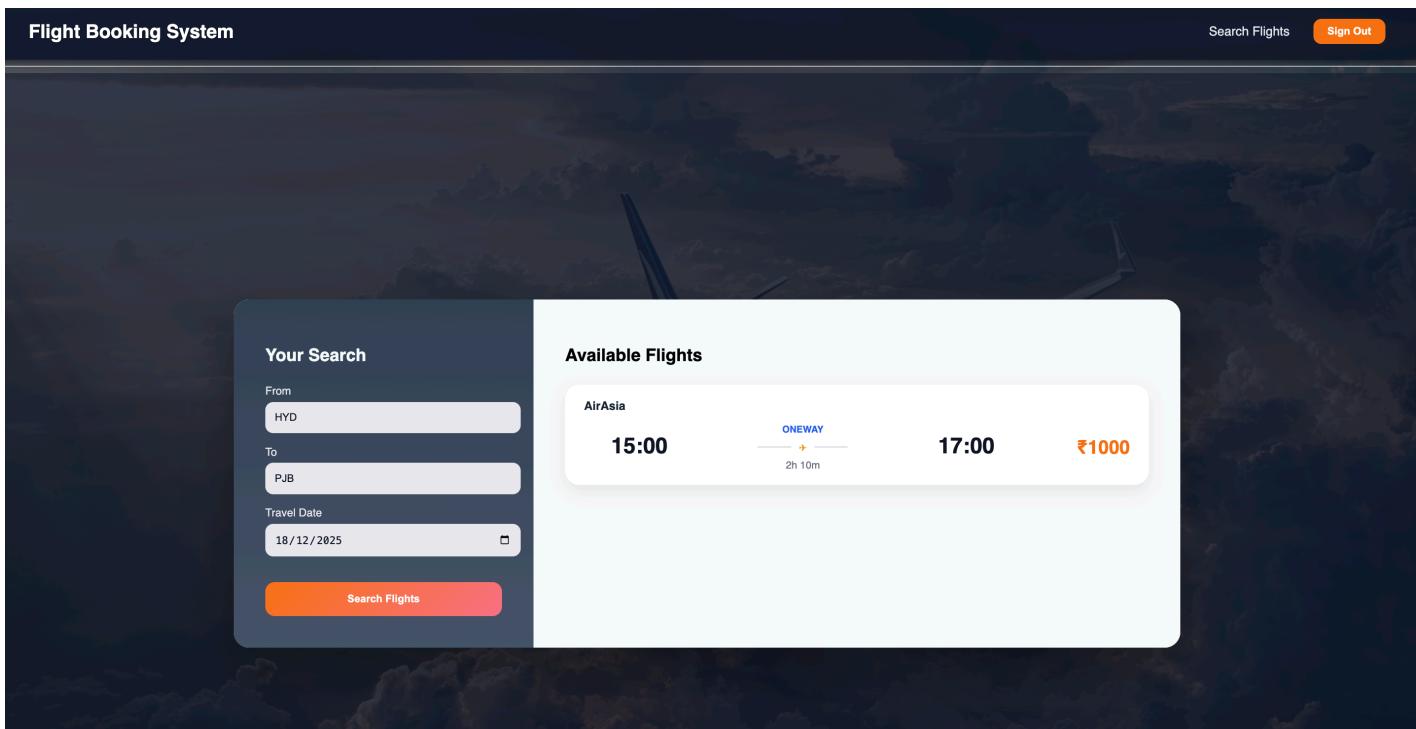
Travel Date: 18/12/2025

Search Flights

Available Flights

AirAsia

15:00 — ONEWAY — 17:00 2h 10m ₹1000



No Flight Available:

The screenshot shows a flight booking application interface. At the top, there's a dark header bar with the text "Flight Booking System" on the left and "Search Flights" and "Sign Out" on the right. Below the header is a large, semi-transparent background image of a cloudy sky with a faint airplane silhouette. In the center, there's a white rectangular search results area. On the left side of this area, under the heading "Your Search", are three input fields: "From" (HYD), "To" (CHN), and "Travel Date" (18/12/2025). Below these is an orange "Search Flights" button. A small message at the bottom of this section says "Failed to fetch flights". To the right of the search form, under the heading "Available Flights", is the text "No flights found. Try searching.".

Eureka Server:

The screenshot shows the Spring Eureka UI. At the top, there's a dark header bar with the "spring Eureka" logo on the left and "HOME LAST 1000 SINCE STARTUP" on the right. Below the header is a "System Status" section with two tables. The first table shows environment details: Environment (test), Data center (default), Current time (2025-12-14T11:10:50 +0000), Uptime (00:02), Lease expiration enabled (true), Renews threshold (11), and Renews (last min) (24). The second table shows DS Replicas, with a single entry for "localhost". Below these is a section titled "Instances currently registered with Eureka", which contains a table listing various services and their status. The table has columns for Application, AMIs, Availability Zones, and Status. Examples of listed services include API-GATEWAY, AUTH-SERVICE, BOOKING-SERVICE, CONFIG-SERVER, FLIGHT-SERVICE, and NOTIFICATION-SERVICE, each with its respective status and unique identifier.

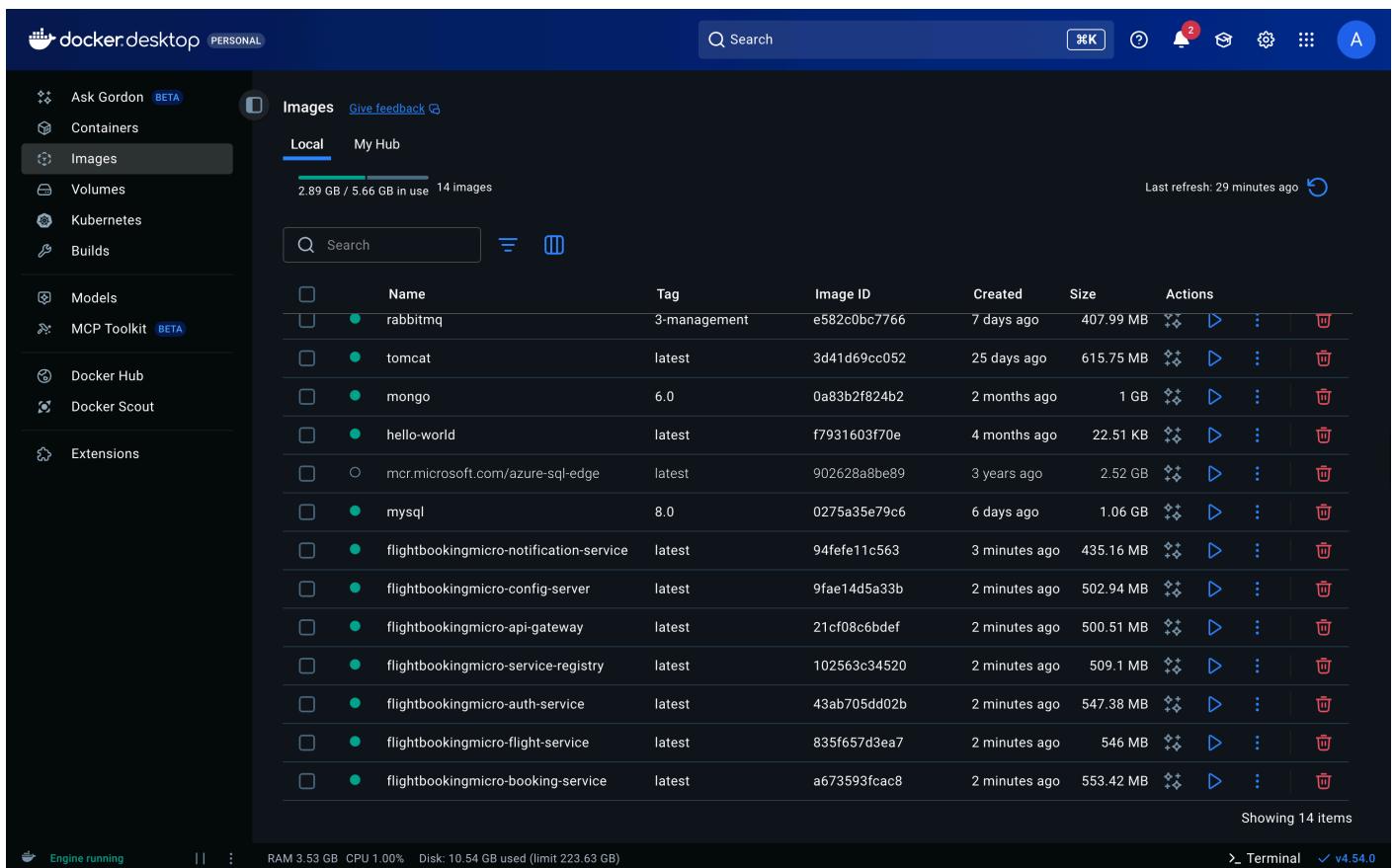
Application	AMIs	Availability Zones	Status
API-GATEWAY	n/a (1)	(1)	UP (1) - 7f09cc55981c:api-gateway:8087
AUTH-SERVICE	n/a (1)	(1)	UP (1) - 786598af6240:auth-service:8085
BOOKING-SERVICE	n/a (1)	(1)	UP (1) - daf7fecf15f2:booking-service:8082
CONFIG-SERVER	n/a (1)	(1)	UP (1) - 4b9a9c6480b4:config-server:8888
FLIGHT-SERVICE	n/a (1)	(1)	UP (1) - a6d92d81a704:flight-service:8081
NOTIFICATION-SERVICE	n/a (1)	(1)	UP (1) - 167ad7ca61c1:notification-service:8090

Docker:

```
[ananyaaraisenapalli@ananya-MacBook-Pro-3 FlightBookingMicro] % docker compose up --build
WARN[0000] /Users/ananyaaraisenapalli/workspace-spring-tools-for-eclipse-4.32.2.RELEASE/FlightBookingMicro/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion
[*] Running 14/14
  ✓ mysql Pulled          20.3s
  ✓ eebabd29ad6b Pull complete   1.1s
  ✓ 842b98544e00 Pull complete   8.3s
  ✓ f7a702cd008e Pull complete   1.4s
  ✓ 05bb5b58cccd Pull complete   9.1s
  ✓ 16e98e12049f Pull complete   1.5s
  ✓ bc6c3e9e9c020 Pull complete   13.2s
  ✓ 44803d6a5f06 Pull complete   1.4s
  ✓ 4b45fb5942a4 Pull complete   1.4s
  ✓ 42c45335a3bb Pull complete   1.4s
  ✓ faefaaeiae1 Pull complete   8.4s
  ✓ 7c8fcfb5feeb8 Pull complete   1.4s
  ✓ f424b1298731 Download complete  0.0s
  ✓ ce3c78665b965 Download complete  1.5s
[+] Building 84.3s (93/93) FINISHED
```

```
[+] Running 18/18
✓ flightbookingmicro-flight-service      Built
✓ flightbookingmicro-auth-service       Built
✓ flightbookingmicro-notification-service Built
✓ flightbookingmicro-api-gateway       Built
✓ flightbookingmicro-service-registry   Built
✓ flightbookingmicro-config-server     Built
✓ flightbookingmicro-booking-service    Built
✓ Network flightbookingmicro_backend   Created
✓ Volume flightbookingmicro_mysql-data Created
✓ Container flight_rabbitmq           Created
✓ Container flight_mysql              Created
✓ Container service-registry          Created
✓ Container notification-service      Created
✓ Container config-server             Created
✓ Container auth-service              Created
✓ Container booking-service           Created
✓ Container flight-service            Created
✓ Container api-gateway              Created

Attaching to api-gateway, auth-service, booking-service, config-server, flight-service, flight_mysql, flight_rabbitmq, notification-service, service-registry
Error response from daemon: ports are not available: exposing port TCP 0.0.0.0:3306 -> 127.0.0.1:0: listen tcp 0.0.0.0:3306: bind: address already in use
```



Containers [Give feedback](#)

Container CPU usage ⓘ 20.25% / 400% (4 CPUs available)

Container memory usage ⓘ 2.62GB / 3.74GB

Show charts

Search Filter Only show running containers

	Name	Container ID	Image	Port(s)	CPU (%)	Memory usage...	Mem	Actions
<input type="checkbox"/>	flightbookingmi -	-	-	-	20.74%	2.62GB / 34.45GB		⋮ ⋮ ⋮ ⋮ ⋮
<input type="checkbox"/>	service-regist f48245f8ac4a	f48245f8ac4a	flightbooki	8761:8761	5.95%	276.2MB / 3.83GB		⋮ ⋮ ⋮ ⋮ ⋮
<input type="checkbox"/>	flight_rabbitm 859cc8fa673a	859cc8fa673a	rabbitmq:3	15672:15672	1.79%	128.7MB / 3.83GB		⋮ ⋮ ⋮ ⋮ ⋮
<input type="checkbox"/>	flight_mysql 60405d45542f	60405d45542f	mysql:8.0	3307:3306	1.08%	384.9MB / 3.83GB		⋮ ⋮ ⋮ ⋮ ⋮
<input type="checkbox"/>	config-server 4b9a9c6480b4	4b9a9c6480b4	flightbooki	8888:8888	0.21%	307.8MB / 3.83GB		⋮ ⋮ ⋮ ⋮ ⋮
<input type="checkbox"/>	booking-servi daf7fecf15f2	daf7fecf15f2	flightbooki	8082:8082	0.44%	452MB / 3.83GB		⋮ ⋮ ⋮ ⋮ ⋮
<input type="checkbox"/>	notification-s 167ad7ca61c1	167ad7ca61c1	flightbooki	8090:8090	8.82%	230MB / 3.83GB		⋮ ⋮ ⋮ ⋮ ⋮
<input type="checkbox"/>	flight-service a6d92d81a704	a6d92d81a704	flightbooki	8081:8081	0.87%	336.3MB / 3.83GB		⋮ ⋮ ⋮ ⋮ ⋮
<input type="checkbox"/>	auth-service 786598af6240	786598af6240	flightbooki	8085:8085	0.12%	307.7MB / 3.83GB		⋮ ⋮ ⋮ ⋮ ⋮
<input type="checkbox"/>	api-gateway 7f09cc55981c	7f09cc55981c	flightbooki	8087:8087	1.46%	254.9MB / 3.83GB		⋮ ⋮ ⋮ ⋮ ⋮

Docker Logs:

```
[11:01:18.030103] from daemon: ports: all not 0.0.0.0:0 exposing port 7071 for 0.0.0.0:7071/tcp + 127.0.0.1:7071/tcp top 0.0.0.0:8080:80.0.0.0:8080, in use
[ananyaarisenapalli@Ananya-MacBook-Pro-3 FlightBookingMicro % docker compose up --build > &1 | tee parent-log.log
time="2025-12-08T23:13:10+05:30" level=warning msg="/Users/ananyaarisenapalli/workspace-spring-tools-for-eclipse-4.32.2.RELEASE/FlightBookingMicro/docker-compose.yml: the attribute `version` is obsolete
, it will be ignored, please remove it to avoid potential confusion"
#1 [internal] load local bake definitions
#1 reading from stdin 4.57kB done
#1 DONE 0.0s

#2 [flight-service internal] load build definition from Dockerfile
#2 transferring dockerfile: 447B done
#2 DONE 0.0s

#3 [api-gateway internal] load build definition from Dockerfile
#3 transferring dockerfile: 447B done
#3 DONE 0.0s

#4 [config-server internal] load build definition from Dockerfile
#4 transferring dockerfile: 447B 0.0s done
#4 DONE 0.0s

#5 [notification-service internal] load build definition from Dockerfile
#5 transferring dockerfile: 447B 0.0s done
#5 DONE 0.0s

#6 [booking-service internal] load build definition from Dockerfile
#6 transferring dockerfile: 447B done
#6 DONE 0.0s

#7 [auth-service internal] load build definition from Dockerfile
#7 transferring dockerfile: 447B done
#7 DONE 0.0s

#8 [service-registry internal] load build definition from Dockerfile
#8 transferring dockerfile: 447B done
#8 DONE 0.0s]
```

Flight Database:

Local instance 3306 - Warning - not supported

Administration **Schemas** **Query 1** **SQL File 3***

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Object Info **Session**

No object selected

```
show tables;
select * from flights;
select * from flight_seats;
use bookingdb_micro;
show tables;
```

Result Grid | Filter Rows: Search | Edit: | Export/Import: | Context Help | Snippets

ID	Airline_Logo_URL	Airline_Name	Arrival_Time	Departure_Time	Destination	Flight_Number	Origin	Price	Total_Seats	Trip_Type
1	https://example.com/logo.png	AirExample	2025-12-15 05:30:00.000000	2025-12-15 04:00:00.000000	BOM	AI-101	HYD	4500	120	ONEWAY
2	http://example.com/logo.png	Air India	2025-12-10 07:00:00.000000	2025-12-10 04:30:00.000000	DEL	AI101	HYD	4500	120	ONEWAY
3	https://indigo.in/logo.png	Indigo Airlines	2025-12-15 05:25:00.000000	2025-12-15 03:15:00.000000	MUM	IE302	BLR	3800.5	150	ROUNDTRIP
4	http://indigo.in/logo.png	Indigo Airlines	2025-12-15 05:25:00.000000	2025-12-15 03:15:00.000000	MUM	IE302	BLR	3800.5	150	ROUNDTRIP
5	https://indigo.in/logo.png	Indigo Airlines	2025-12-15 05:25:00.000000	2025-12-15 03:15:00.000000	MUM	IE302	BLR	3800.5	150	ROUNDTRIP
6	http://indigo.in/logo.png	Indigo Airlines	2025-12-15 05:25:00.000000	2025-12-15 03:15:00.000000	MUM	AE302	BLR	3800.5	150	ROUNDTRIP
7	https://indigo.in/logo.png	Indigo Airlines	2025-12-15 05:25:00.000000	2025-12-15 03:15:00.000000	MUM	AE302	BLR	3800.5	150	ROUNDTRIP
8	http://image.spacexapi.com/logo.png	SpiceJet	2025-12-12 03:20:00.000000	2025-12-12 00:50:00.000000	CCU	SG409	MAA	5200	90	ONEWAY
9	https://images.vistara.com/logo.png	Vistara	2025-12-15 05:15:00.000000	2025-12-15 03:00:00.000000	BOM	UK879	BLR	5200	180	ONEWAY
10	http://airasia/logo.png	AirAsia	2025-12-18 11:30:00.000000	2025-12-18 09:30:00.000000	GOI	IS15	BLR	2500	150	ONEWAY
11	http://airasia/logo.png	AirAsia	2025-12-18 11:30:00.000000	2025-12-18 09:30:00.000000	HYD	IS15	BLR	2000	100	ONEWAY
12	http://airasia/logo.png	AirAsia	2025-12-18 11:30:00.000000	2025-12-18 09:30:00.000000	CHENNAI	IS15	MUM	2000	100	ONEWAY
13	http://airasia/logo.png	AirAsia	2025-12-18 11:30:00.000000	2025-12-18 09:30:00.000000	PJB	IS15	DEL	1000	200	ONEWAY

flights 1 | Apply | Revert

Action Output

Time	Action	Response	Duration / Fetch Time
23:07:27	use flightdb_micro	0 row(s) affected	0.068 sec
23:07:29	select * from flights LIMIT 0, 1000	13 row(s) returned	0.030 sec / 0.000016...

Query Completed

Docker-MySQL MySQL Model* EER Diagram

Administration **Schemas** **Query 1**

SCHEMAS

- authdb
 - Tables
 - Views
 - Stored Procedures
 - Functions
- bookingdb_micro
 - Tables
 - Views
 - Stored Procedures
 - Functions
- flightdb_micro

No object selected

```
use flightdb_micro;
show tables;
select * from flights;
select * from flight_seats;
```

Result Grid | Filter Rows: Search | Edit: | Export/Import: | Context Help | Snippets

ID	Passenger_Name	Seat_Number	Status	Flight...
1	Ananya	1	BOOKED	1
2		2	AVAILABLE	1
3		3	AVAILABLE	1
4		4	AVAILABLE	1
5		5	AVAILABLE	1
6		6	AVAILABLE	1
7		7	AVAILABLE	1
8		8	AVAILABLE	1
9		9	AVAILABLE	1
10		10	AVAILABLE	1
11		11	AVAILABLE	1
12		12	AVAILABLE	1
13		13	AVAILABLE	1
14		14	AVAILABLE	1
15		15	AVAILABLE	1
16		16	AVAILABLE	1
17		17	AVAILABLE	1
18		18	AVAILABLE	1
19		19	AVAILABLE	1
20		20	AVAILABLE	1
21		21	AVAILABLE	1
22		22	AVAILABLE	1
23		23	AVAILABLE	1
24		24	AVAILABLE	1

flight_seats 11 | Apply | Revert

Action Output

Time	Action	Response	Duration / Fetch Time
16:53:53	select * from users LIMIT 0, 1000	1 row(s) returned	0.0045 sec / 0.00001...
16:54:44	select * from flights LIMIT 0, 1000	Error Code: 1146. Table 'authdb.flights' doesn't exist	0.013 sec
16:55:48	use flightdb_micro	0 row(s) affected	0.00078 sec
16:55:50	select * from flights LIMIT 0, 1000	1 row(s) returned	0.0022 sec / 0.00001...
16:55:56	select * from flight_seats LIMIT 0, 1000	100 row(s) returned	0.0087 sec / 0.00003...

Query Completed

Booking Database:

Local instance 3306 - Warning - not supported

Administration Schemas **Query 1** SQL File 3* Limit to 1000 rows Context Help Snippets

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Object Info Session

No object selected

```

57 • select * from flight_seats;
58 • use bookingdb_micro;
59 • show tables;
60 • select * from bookings;
61 • select * from passengers;
100% 24:60

```

Result Grid Filter Rows: Search Edit: Export/Import: Result Grid Form Editor Field Types Query Stats Execution Plan

ID	cancelled_at	created_at	flight_id	num_seats	PNR	Status	Total Price	User Email	Version
27	HULL	2025-12-04 05:35:20.066070	2	1	B894CA5C	ACTIVE	4500	test@gmail.com	HULL
28	HULL	2025-12-04 05:35:37.765510	2	1	5F7B752D	ACTIVE	4500	test@gmail.com	HULL
29	HULL	2025-12-04 05:35:39.163333	2	1	D5BD779D	ACTIVE	4500	test@gmail.com	HULL
30	HULL	2025-12-08 06:41:01.318193	10	2	69EE729F	ACTIVE	0	alice@example.com	0
31	2025-12-08 07:11:48.069455	2025-12-08 06:50:41.076648	10	2	B89C7902	CANCELLED	0	charlie@example.com	0
32	HULL	2025-12-08 06:57:39.187987	10	2	CAF0E05F	ACTIVE	5000	alice@example.com	0
33	HULL	2025-12-08 07:12:43.483474	10	2	D699C21D	ACTIVE	5000	alice@example.com	0
34	HULL	2025-12-08 07:12:57.008338	10	2	4C7A74AA	ACTIVE	5000	alice@example.com	0
35	HULL	2025-12-08 07:12:52.752340	10	2	5314265F	ACTIVE	5000	alice@example.com	0
36	HULL	2025-12-08 07:20:09.009054	10	1	2789568C	CANCELLED	2500	ananyaaisenapalli...	0
37	2025-12-08 10:30:55.187468	2025-12-08 10:21:24.726529	10	1	ED5ECEC1	CANCELLED	2500	ananyaaisenapalli...	0
38	HULL	2025-12-08 10:33:07.769896	10	1	65F64D1F	ACTIVE	2500	ananyaaisenapalli...	0
39	HULL	2025-12-08 10:38:37.603391	10	1	68B87FF6	ACTIVE	2500	shreyaaisenapalli...	0
40	HULL	2025-12-08 10:40:40.695531	10	1	E9AB6722	ACTIVE	2500	chetal7330@gmail...	0
41	2025-12-08 14:46:45.452472	2025-12-08 10:47:44.4739	9	2	1AF828AF	CANCELLED	10400	chetal7330@gmail...	0
42	2025-12-08 11:06:04.634516	2025-12-08 10:46:27.987870	8	2	DDC49E45	CANCELLED	10400	chetal7330@gmail...	0
43	HULL	2025-12-08 10:47:51.028468	8	3	7A173AF3	ACTIVE	15600	shreyaaisenapalli...	0
44	HULL	2025-12-08 11:01:03.304730	8	1	3CEC942C	ACTIVE	5200	palakurad@gmail...	0
45	2025-12-08 14:43:45.771627	2025-12-08 14:42:54.629424	10	1	34BFFFA1	CANCELLED	2500	palakurad@gmail...	0
46	HULL	2025-12-08 14:45:50.343451	10	1	E98F03E1	ACTIVE	2500	palakurad@gmail...	0
47	2025-12-08 15:34:36.604753	2025-12-08 14:48:56.772660	10	1	0B6A0529	CANCELLED	2500	ananyaaisenapalli...	0
48	HULL	2025-12-08 15:34:36.301343	10	1	137A7E16	ACTIVE	2500	ananyaaisenapalli...	0

bookings 3

Action Output

Time	Action	Response	Duration / Fetch Time
1 23:07:27	use flightdb_micro	0 row(s) affected	0.068 sec
2 23:07:29	select * from flights LIMIT 0, 1000	13 row(s) returned	0.030 sec / 0.000016...
3 23:07:57	select * from flight_seats LIMIT 0, 1000	812 row(s) returned	0.0067 sec / 0.00051...
4 23:08:05	use bookingdb_micro	0 row(s) affected	0.00068 sec
5 23:08:09	select * from bookings LIMIT 0, 1000	47 row(s) returned	0.0024 sec / 0.00002...

Query Completed

Local instance 3306 - Warning - not supported

Administration Schemas **Query 1** SQL File 3* Limit to 1000 rows Context Help Snippets

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Object Info Session

No object selected

```

58 • use bookingdb_micro;
59 • show tables;
60 • select * from bookings;
61 • select * from passengers;
62
100% 26:61

```

Result Grid Filter Rows: Search Edit: Export/Import: Result Grid Form Editor Field Types Query Stats Execution Plan

ID	Age	Gender	Meal_Preference	Passenger Name	Seat Number	Booking ID
3	28	F	VEG	Alice	1A	2
4	30	M	VEG	Bob	1B	2
5	20	M	VEG	Bob	3A	3
6	20	M	VEG	Bob	3A	4
7	23	F	VEG	Anu	1A	5
8	20	M	VEG	Bob	3A	6
9	20	M	VEG	Bob	3A	12
10	20	M	VEG	Bob	3A	11
11	20	M	VEG	Bob	3A	7
12	20	M	VEG	Bob	3A	14
13	20	M	VEG	Bob	3A	13
14	20	M	VEG	Bob	3A	8
15	20	M	VEG	Bob	3A	10
16	20	M	VEG	Bob	3A	15
17	20	M	VEG	Bob	3A	9
18	20	M	VEG	Bob	3A	16
19	20	M	VEG	Bob	3A	17
20	20	M	VEG	Bob	3A	18
21	20	M	VEG	Bob	3A	19
22	20	M	VEG	Bob	3A	20
23	20	M	VEG	Bob	3A	21
24	20	M	VEG	Roh	3A	22

passengers 4

Action Output

Time	Action	Response	Duration / Fetch Time
1 23:07:27	use flightdb_micro	0 row(s) affected	0.068 sec
2 23:07:29	select * from flights LIMIT 0, 1000	13 row(s) returned	0.030 sec / 0.000016...
3 23:07:57	select * from flight_seats LIMIT 0, 1000	812 row(s) returned	0.0067 sec / 0.00051...
4 23:08:05	use bookingdb_micro	0 row(s) affected	0.00068 sec
5 23:08:09	select * from bookings LIMIT 0, 1000	47 row(s) returned	0.0024 sec / 0.00002...
6 23:08:20	select * from passengers LIMIT 0, 1000	58 row(s) returned	0.0031 sec / 0.00002...

Query Completed

Auth Database:

The screenshot shows the MySQL Workbench interface with the following details:

- Schemas:** authdb, bookingdb_micro, flightdb_micro
- Tables:** bookings, passengers, user_roles, users
- Query Editor:** Contains the following SQL queries:

```
8 • select * from bookings;
9 • select * from passengers;
10
11 • use authdb;
12 • show tables;
13 • select * from user_roles;
14 • select * from users;
```
- Result Grid:** Shows the data for the users table.

id	created_at	email	password	role	username
1	2025-12-14 11:11:01.427389	ananyaarisenapalli@gmail.com	\$2a\$10\$IZx4NcaBOW1cfUdV7Tje/sV4wXDF7...	ROLE_ADMIN	Ananya
2	2025-12-14 11:26:38.233164	ananyarao1174@gmail.com	\$2a\$10\$XsLpJINq.Y69MWGUbm8lc2i8Ljbdu...	ROLE_USER	Ananya Rao
- Action Output:** Shows the history of actions taken on the database.

Time	Action	Response	Duration / Fetch Time
16:56:13	use bookingdb_micro	0 row(s) affected	0.0050 sec
16:56:15	select * from bookings LIMIT 0, 1000	1 row(s) returned	0.0078 sec / 0.00004...
16:56:24	select * from passengers LIMIT 0, 1000	1 row(s) returned	0.0048 sec / 0.00021...
16:56:29	use authdb	0 row(s) affected	0.0031 sec
16:56:56	select * from users LIMIT 0, 1000	2 row(s) returned	0.0098 sec / 0.00001...

Postman API-Gateway with Security:

Admin Signup:

The screenshot shows a Postman collection named "Flightbooking_security". A POST request is being made to the URL `http://localhost:8087/auth-service/auth/signup`. The request body is set to "raw" JSON and contains the following data:

```
1 {
2     "username": "Ananya",
3     "email": "ananyaarisenapalli@gmail.com",
4     "password": "Passw0rd!",
5     "role": ["admin"]
6 }
```

The response status is 200 OK, and the response body is "User registered successfully!".

Admin Signin:

The screenshot shows a Postman collection named "Flightbooking_security". A POST request is being made to the URL `http://localhost:8087/auth-service/auth/signin`. The request body is set to "raw" JSON and contains the following data:

```
1 {
2     "email": "ananyaarisenapalli@gmail.com",
3     "password": "Passw0rd!"
4 }
```

The response status is 200 OK, and the response body is a JSON object containing a token and user information:

```
1 {
2     "token": "eyJhbGciOiJIUzI1NiJ9.eyJzdWIoIjhbmFueWFhcmLzW5hcGFsbGIAZ21haWwuY29tIiwicm9sZXMiOlisiUk9MRV9BRE1JTJdLCJpYXQiOjE3NjU3MTA2NjgsImV4cCI6MTc2NTcxNDI2OH0.HdVTFDKyYtaiRIrHU9lwUq8wa7QiBrrAqV6YSI7rWcE",
3     "type": "Bearer",
4     "id": 1,
5     "username": "Ananya",
6     "email": "ananyaarisenapalli@gmail.com",
7     "roles": [
8         "ROLE_ADMIN"
9     ]
10 }
```

User Signup:

The screenshot shows the Postman interface for a POST request to `http://localhost:8087/auth-service/auth/signup`. The request body contains the following JSON:

```
1 {
2     "username": "Ananya Rao",
3     "email": "ananyarao1174@gmail.com",
4     "password": "Passw0rd!"
5 }
```

The response status is **200 OK** with a response time of 406 ms and a response size of 345 B. The response body is:

```
1 User registered successfully!
```

User Signin:

The screenshot shows the Postman interface for a POST request to `http://localhost:8087/auth-service/auth/signin`. The request body contains the following JSON:

```
1 {
2     "email": "ananyarao1174@gmail.com",
3     "password": "Passw0rd!"
4 }
```

The response status is **200 OK** with a response time of 218 ms and a response size of 615 B. The response body is a JSON object:

```
1 {
2     "token": "eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJhbmbFueWFyYW8xMTC0QGdtYWlsLmNvbSIsInJvbGVzIjpbIlJPTEVfVVNFUiJdLCJpYXQiOjE3NjU3MTE2MDUsImV4cCI6MTc2NTcxNTIwNX0.1XAwcQ9eXRHWpaUVB1IFGvbSjQhk89YjTJBzaDHs7LI",
3     "type": "Bearer",
4     "id": 2,
5     "username": "Ananya Rao",
6     "email": "ananyarao1174@gmail.com",
7     "roles": [
8         "ROLE_USER"
9     ]
10 }
```

Add Inventory(Only Admin):

The screenshot shows a Postman collection named "Flightbooking_security". A POST request is made to the URL `http://localhost:8087/flight-service/api/flight/airline/inventory/add`. The request body contains the following JSON:

```
1 {
2   "airline": "AirAsia",
3   "airlineLogoUrl": "http://airasia/logo.png",
4   "flightNumber": "I515",
5   "origin": "HYD",
6   "destination": "PJB",
7   "departure": "2025-12-18T15:00:00",
8   "arrival": "2025-12-18T17:00:00",
9   "price": 1000,
10  "totalSeats": 100
11 }
```

The response status is 201 Created, with a response time of 1.08 s and a response size of 354 B. The response body is:

```
1 {
2   "id": 1
3 }
```

Search Flight:

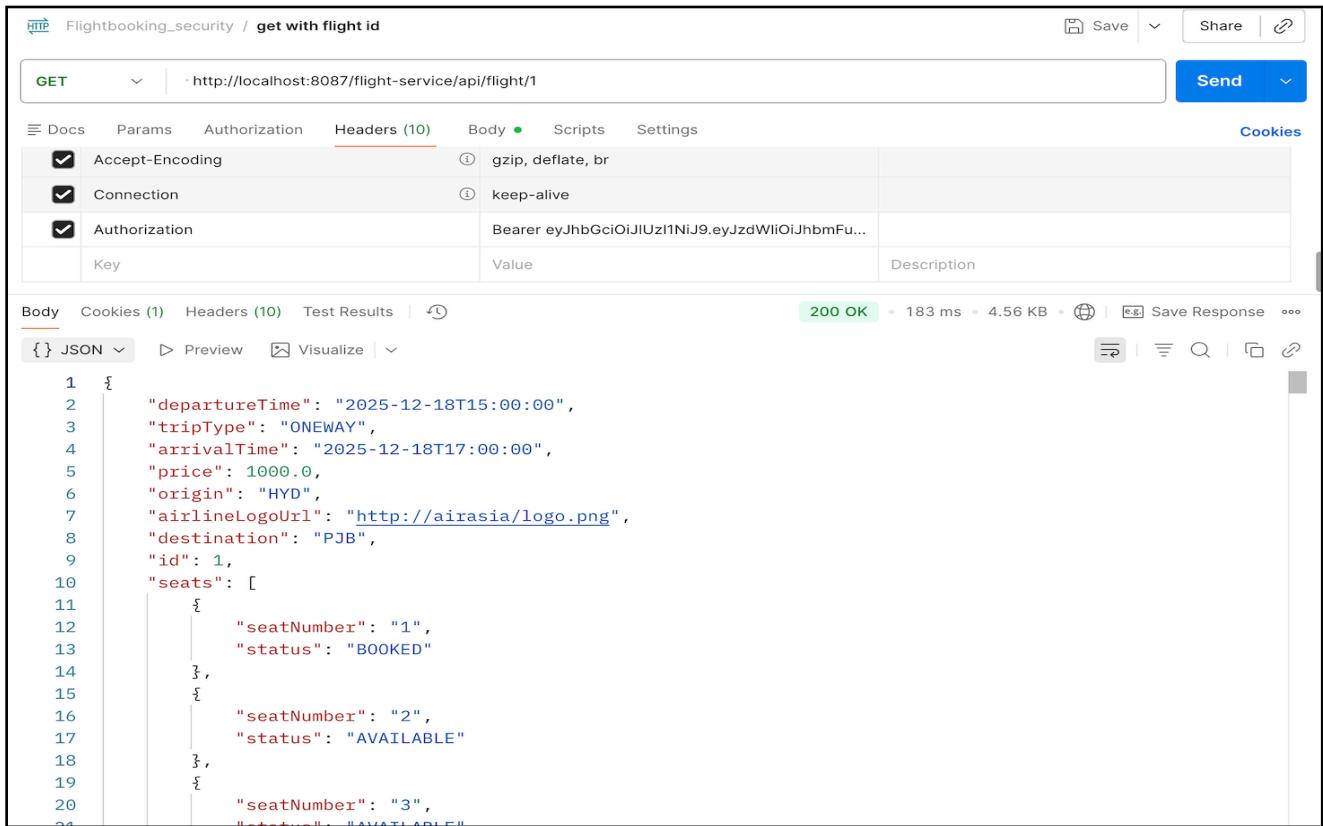
The screenshot shows a Postman collection named "Flightbooking_security". A POST request is made to the URL `http://localhost:8087/flight-service/api/flight/search`. The request includes the following headers:

Header	Value
Cookie	JSESSIONID=1A5BB85303114DBE539B8560C38...
Postman-Token	<calculated when request is sent>
Content-Type	application/json
Content-Length	<calculated when request is sent>
Host	<calculated when request is sent>
User-Agent	PostmanRuntime/7.49.1
Accept	/*
Accept-Encoding	gzip, deflate, br
Connection	keep-alive
Authorization	Bearer eyJhbGciOiJIUzI1NiJ9eyJzdWliOiJhbmfu...

The response status is 201 Created, with a response time of 318 ms and a response size of 534 B. The response body is:

```
2 {
3   "flightId": 1,
4   "departureTime": "2025-12-18T15:00:00",
5   "arrivalTime": "2025-12-18T17:00:00",
6   "airlineName": "AirAsia",
7   "airlineLogoUrl": "http://airasia/logo.png",
8   "price": 1000.0,
9   "tripType": "ONEWAY",
10  "seatsAvailable": 100
11 }
```

Get Flight by ID:



Flightbooking_security / get with flight id

GET http://localhost:8087/flight-service/api/flight/1

Headers (10)

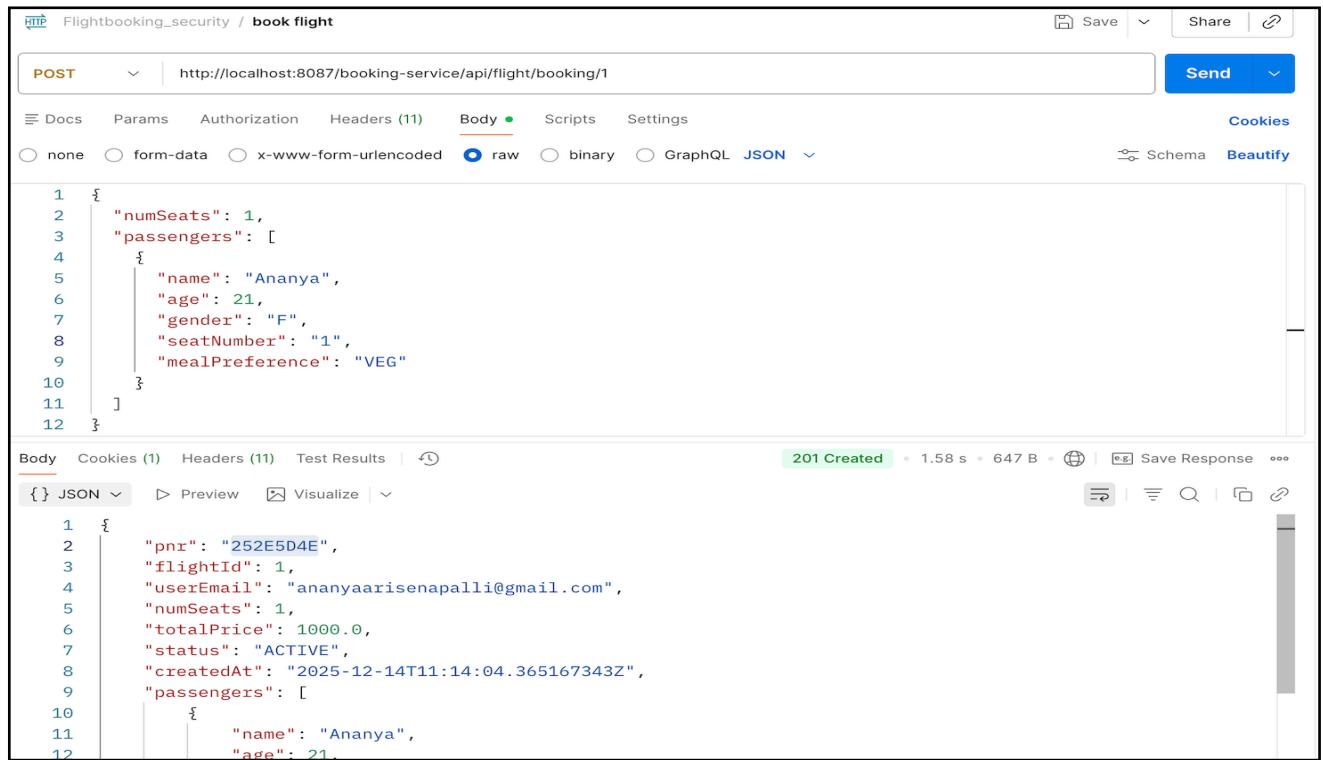
- Accept-Encoding: gzip, deflate, br
- Connection: keep-alive
- Authorization: Bearer eyJhbGciOiJIUzI1NiJ9eyJzdWlOIjhbmFu...

Body Cookies (1) Headers (10) Test Results

200 OK 183 ms 4.56 KB Save Response

```
1 {
2   "departureTime": "2025-12-18T15:00:00",
3   "tripType": "ONEWAY",
4   "arrivalTime": "2025-12-18T17:00:00",
5   "price": 1000.0,
6   "origin": "HYD",
7   "airlineLogoUrl": "http://airasia/logo.png",
8   "destination": "PJB",
9   "id": 1,
10  "seats": [
11    {
12      "seatNumber": "1",
13      "status": "BOOKED"
14    },
15    {
16      "seatNumber": "2",
17      "status": "AVAILABLE"
18    },
19    {
20      "seatNumber": "3",
21      "status": "AVAILABLE"
22    }
23  ]
24}
```

Book Ticket:



Flightbooking_security / book flight

POST http://localhost:8087/booking-service/api/flight/booking/1

Headers (11)

Body raw

```
1 {
2   "numSeats": 1,
3   "passengers": [
4     {
5       "name": "Ananya",
6       "age": 21,
7       "gender": "F",
8       "seatNumber": "1",
9       "mealPreference": "VEG"
10    }
11  ]
12}
```

201 Created 1.58 s 647 B Save Response

Body Cookies (1) Headers (11) Test Results

{ JSON Preview Visualize }

```
1 {
2   "pnr": "252E5D4E",
3   "flightId": 1,
4   "userEmail": "ananyaarisenapalli@gmail.com",
5   "numSeats": 1,
6   "totalPrice": 1000.0,
7   "status": "ACTIVE",
8   "createdAt": "2025-12-14T11:14:04.365167343Z",
9   "passengers": [
10    {
11      "name": "Ananya",
12      "age": 21
13    }
14  ]
15}
```

Get Details by PNR:

The screenshot shows a Postman request for "Flightbooking_security / get ticket by pnr". The method is GET, and the URL is <http://localhost:8087/booking-service/api/flight/ticket/252E5D4E>. In the Headers tab, there is a cookie entry: JSESSIONID=1A5BB85303114DBE539B8560C38... Under the Body tab, the response is displayed as JSON, showing a single passenger record with details like PNR, flight ID, user email, number of seats, total price, status, creation date, and passenger information (name, age, gender, seat number, meal preference). The response status is 200 OK.

```
1 {
2     "pnr": "252E5D4E",
3     "flightId": 1,
4     "userEmail": "ananyaarisenapalli@gmail.com",
5     "numSeats": 1,
6     "totalPrice": 1000.0,
7     "status": "ACTIVE",
8     "createdAt": "2025-12-14T11:14:04.365167Z",
9     "passengers": [
10         {
11             "name": "Ananya",
12             "age": 21,
13             "gender": "F",
14             "seatNumber": "1",
15             "mealPreference": "VEG"
16         }
17     ]
18 }
```

Get History with Email:

The screenshot shows a Postman request for "Flightbooking_security / get history". The method is GET, and the URL is <http://localhost:8087/booking-service/api/flight/booking/history/ananyaarisenapalli@gmail.com>. In the Headers tab, Accept-Encoding, Connection, and Authorization headers are set. The Authorization header value is Bearer eyJhbGciOiJIUzI1NiJ9eyJzdWIiJhbmfu... Under the Body tab, the response is displayed as JSON, showing a list of booking histories for the specified email. Each history entry contains PNR, flight ID, user email, number of seats, total price, status, creation date, and passenger information. The response status is 200 OK.

```
1 [
2     {
3         "pnr": "252E5D4E",
4         "flightId": 1,
5         "userEmail": "ananyaarisenapalli@gmail.com",
6         "numSeats": 1,
7         "totalPrice": 1000.0,
8         "status": "ACTIVE",
9         "createdAt": "2025-12-14T11:14:04.365167Z",
10         "passengers": [
11             {
12                 "name": "Ananya",
13                 "age": 21,
14                 "gender": "F",
15                 "seatNumber": "1",
16                 "mealPreference": "VEG"
17             }
18         ]
19     }
20 ]
```

Cancel Ticket:

The screenshot shows the Postman application interface for a DELETE request to cancel a flight booking.

Request URL: `http://localhost:8087/booking-service/api/flight/booking/cancel/F03847E6`

Headers (7):

Key	Value	Description
Postman-Token	<calculated when request is sent>	
Host	<calculated when request is sent>	
User-Agent	PostmanRuntime/7.49.1	
Accept	*/*	
Accept-Encoding	gzip, deflate, br	
Connection	keep-alive	
Authorization	Bearer eyJhbGciOiJIUzI1NiJ9eyJzdWliOiJhb... (truncated)	

Body:

```
1 {  
2   "pnr": "F03847E6",  
3   "message": "Booking cancelled successfully",  
4   "status": "CANCELLED"  
5 }
```

Response Status: 200 OK | 1.08 s | 487 B | Save Response

Circuit Breaker:

The screenshot shows the Postman interface with the following details:

- Collection:** Ananyarao Arisenapalli's Workspace
- Environment:** No environment
- Request:** POST <http://localhost:8082/api/flight/booking/10>
- Body:** JSON (raw)
- Response Status:** 503 Service Unavailable
- Response Body:**

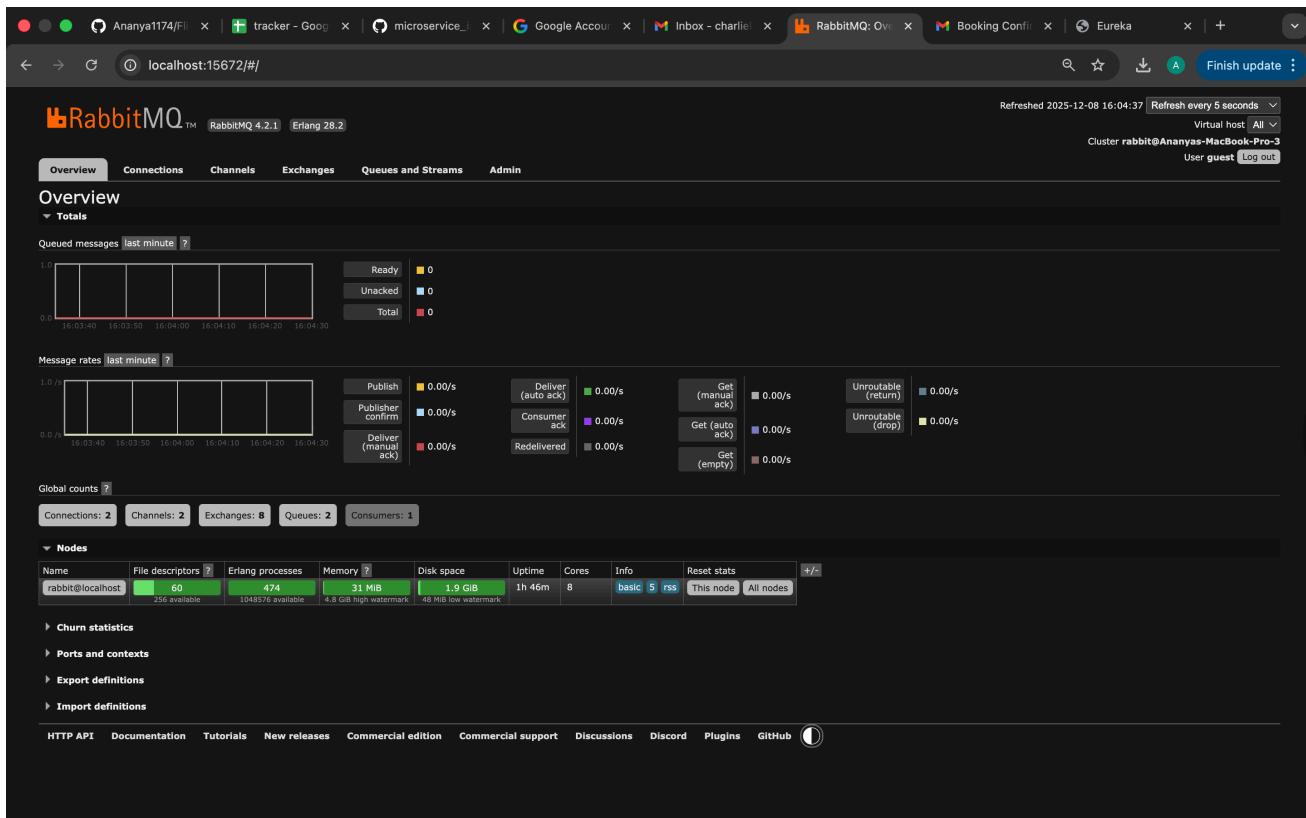
```
1 "timestamp": "2025-12-08T04:52:37.991544Z",
2 "status": 503,
3 "error": "Service Unavailable",
4 "message": "Flight service unavailable. Try again later.",
5 "path": "/api/flight/booking/10"
```

Config-Server:

The screenshot shows a browser window displaying the configuration for the flight-service. The URL is <localhost:8888/flight-service/default>. The configuration is as follows:

```
{"name": "flight-service",
"profiles": [
"default"
],
"label": null,
"version": "ca564e90c0f8e8b786ce53895a4338027dc1d6be",
"test": null,
"propertySources": [
{
"name": "https://github.com/Ananya1174/flight-config-server/flight-service.properties",
"source": {
"spring.application.name": "flight-service",
"server.port": "8080",
"spring.datasource.url": "jdbc:mysql://localhost:3306/flightdb_micro?useSSL=false&serverTimezone=UTC",
"spring.datasource.username": "root",
"spring.datasource.password": "Ananya2084",
"spring.datasource.driver-class-name": "com.mysql.cj.jdbc.Driver",
"spring.jpa.hibernate.ddl-auto": "update",
"spring.jpa.show-sql": "false",
"eureka.client.service-url.defaultZone": "http://localhost:8761/eureka/",
"eureka.client.register-with-eureka": "true",
"eureka.client.fetch-registry": "true"
}
}
```

Message Broker-RabbitMQ:



Successful Booking Email-

```
2025-12-08T16:03:07.826+05:30 INFO 8833 --- [booking-service] [nio-8082-exec-5] c.bookingservice.service.BookingService : Booking saved: pnr=5F64D1FA, flightId=10, user=ananyaarisenapalli@gmail.com  
2025-12-08T16:03:07.839+05:30 INFO 8833 --- [booking-service] [nio-8082-exec-5] c.bookingservice.service.BookingService : Published booking-created email for PNR 5F64D1FA
```

The screenshot shows an email client interface with a single message in the inbox. The message is titled "Booking Confirmed: PNR ED6FCEC1". It is sent from "charliektest11@gmail.com" to "me" at 3:59 PM (0 minutes ago). The message body contains the following text:

Hi,

Your booking is confirmed.

PNR: ED6FCEC1
Flight ID: 10
Seats: 1
Total: 2500.0

Passengers:
- Riya / Seat: 22

Thanks,
Booking Team

At the bottom of the message are three buttons: "Reply", "Forward", and a smiley face icon.

Successful Booking Cancel Email-

```
99caf0 [delegate=amp://guest@127.0.0.1:5672/, localPort=62251]  
2025-12-08T16:00:56.413+05:30 INFO 8833 --- [booking-service] [nio-8082-exec-2] c.bookingservice.service.BookingService : Published booking-cancelled email for PNR ED6FCEC1
```

The screenshot shows an email client interface with a single message in the inbox. The message is titled "Booking Cancelled: PNR ED6FCEC1". It is sent from "charliektest11@gmail.com" to "me" at 4:01 PM (0 minutes ago). The message body contains the following text:

Hi,

Your booking with PNR ED6FCEC1 has been cancelled.

Thanks,
Booking Team

At the bottom of the message are three buttons: "Reply", "Forward", and a smiley face icon.

SonarQube:

Before:

The screenshot shows the SonarQube Main Branch Summary page for the project "flight-service". The page displays the following information:

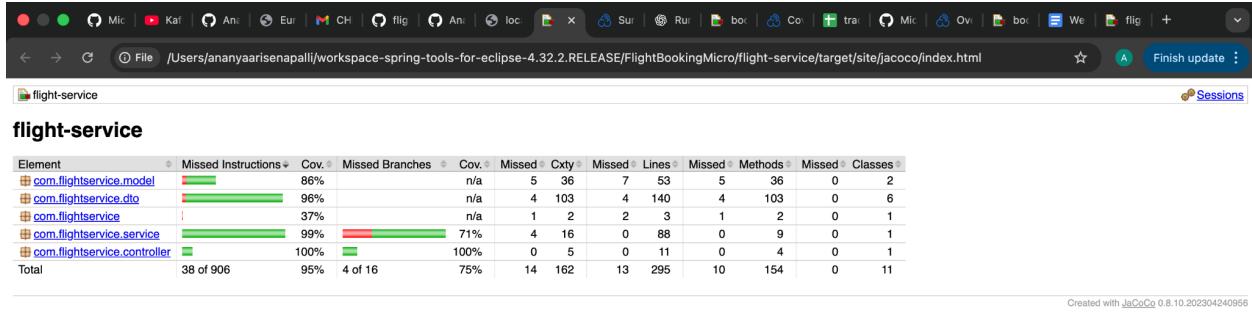
- Project: flight-service
- Branch: main
- Last analysis: 31 seconds ago
- Code Metrics:
 - 778 Lines of Code
 - Version: 0.0.1-SNAPSHOT
- Quality Gate: Sonar way (Passed)
- Status: Passed
- Overall Code Metrics:
 - Security: 0 Open issues (A)
 - Reliability: 0 Open issues (A)
 - Maintainability: 8 Open issues (A)
 - Accepted Issues: 0
 - Coverage: 94.5% (No conditions set on 295 Lines to cover)
 - Duplications: 29.8% (No conditions set on 985 Lines)

After:

The screenshot shows the SonarQube Main Branch Summary page for the project "flight-service". The page displays the following information:

- Project: flight-service
- Branch: main
- Last analysis: 41 seconds ago
- Code Metrics:
 - 522 Lines of Code
 - Version: 0.0.1-SNAPSHOT
- Quality Gate: Sonar way (Passed)
- Status: Passed
- Overall Code Metrics:
 - Security: 0 Open issues (A)
 - Reliability: 0 Open issues (A)
 - Maintainability: 0 Open issues (A)
 - Accepted Issues: 0
 - Coverage: 82.2% (No conditions set on 170 Lines to cover)
 - Duplications: 0.0% (No conditions set on 672 Lines)

Jacoco:



Jmeter:

20 requests:

Summary Report											
Name:	Summary Report										
Comments:											
- Write results to file / Read from file											
Filename	<input type="text"/>				<input type="button" value="Browse..."/>	<input type="checkbox"/> Log/Display Only:	<input type="checkbox"/> Errors	<input type="checkbox"/> Successes	<input type="button" value="Configure"/>		
Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/s...	Sent KB/sec	Avg. Bytes	
HTTP Request	20	26	16	67	10.86	0.00%	1.1/sec	0.59	0.23	577.0	
TOTAL	20	26	16	67	10.86	0.00%	1.1/sec	0.59	0.23	577.0	

50 requests:

Summary Report

Name:

Comments:

Write results to file / Read from file

Filename Log/Display Only: Errors Successes

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/s...	Sent KB/sec	Avg. Bytes
HTTP Request	50	19	8	137	17.22	0.00%	1.7/sec	0.96	0.38	577.0
TOTAL	50	19	8	137	17.22	0.00%	1.7/sec	0.96	0.38	577.0

100 requests:

Summary Report

Name:

Comments:

Write results to file / Read from file

Filename Log/Display Only: Errors Successes

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/s...	Sent KB/sec	Avg. Bytes
HTTP Request	100	15	4	144	13.32	0.00%	1.7/sec	0.95	0.37	577.0
TOTAL	100	15	4	144	13.32	0.00%	1.7/sec	0.95	0.37	577.0