

FLIGHT MANAGEMENT SYSTEM

(Dockerized the microservices)

Debashrita Mandal

This microservices-based Flight Management System uses :

1. **Eureka Service Discovery**: For microservice registry
2. **OpenFeign and API Gateway**: For inter-service communication
3. **Resilience4j Circuit Breaker**: For fault tolerance
4. **Spring Cloud Config Server**: For centralized configuration
5. **RabbitMQ (using Docker)** : as an event-driven messaging layer (message broker)
6. **Notification Service (using Spring Mail)** for asynchronous email delivery.
7. **Docker & Docker Compose** – Containerize all microservices and supporting tools, allowing easy multi-service deployment, scaling, and environment consistency.

The system persists data using **Spring Data JPA** with **MySQL**, supports validation using **Jakarta Validation**, and includes comprehensive **JUnit + Mockito** test coverage with **JaCoCo**.

DOCKER SCREENSHOTS

Building the docker containers for each microservice

```
=> => exporting config sha256:fd7cb25d6eb0c257d01681309d55a7752e95b1a1b06fb784d94720c278c895dd 0.0s
=> => exporting attestation manifest sha256:496af552440ddd4db22ba3ac333f1d3b2cc03a0c8f02df7a5f06e43ba3ed5c8 0.3s
=> => exporting manifest list sha256:8260a49088e36a847e3fbf59fb90269a84e2533478855cdd56a355787304ee5f 0.1s
=> => naming to docker.io/library/fms_microservices-booking-microservice:latest 0.1s
=> => unpacking to docker.io/library/fms_microservices-booking-microservice:latest 0.1s
=> [api-gateway] resolving provenance for metadata file 1.1s
=> [service-registry] resolving provenance for metadata file 1.0s
=> [flight-microservice] resolving provenance for metadata file 0.6s
=> [notification-microservice] resolving provenance for metadata file 0.4s
=> [booking-microservice] resolving provenance for metadata file 0.2s
=> [config-server] resolving provenance for metadata file 0.1s
[+] Building 6/6
✓fms_microservices-config-server      Built 0.0s
✓fms_microservices-flight-microservice Built 0.0s
✓fms_microservices-notification-microservice Built 0.0s
✓fms_microservices-service-registry    Built 0.0s
✓fms_microservices-api-gateway        Built 0.0s
✓fms_microservices-booking-microservice Built 0.0s
PS C:\Users\KIIT\OneDrive\Desktop\chubb_training\FMS_Microservices> docker-compose up -d
time="2025-12-08T19:14:06+05:30" level=warning msg="C:\Users\KIIT\OneDrive\Desktop\chubb_training\FMS_Microservices\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 9/9
✓Network fms_microservices_fms-net   Created 0.1s
✓Container service-registry          Healthy 60.7s
✓Container rabbitmq                 Healthy 123.5s
✓Container mysql                     Healthy 60.7s
✓Container config-server            Healthy 123.2s
✓Container api-gateway              Started 59.5s
✓Container notification-microservice Started 123.9s
✓Container booking-microservice     Started 123.2s
✓Container flight-microservice      Started 123.3s
PS C:\Users\KIIT\OneDrive\Desktop\chubb_training\FMS_Microservices> docker ps
CONTAINER ID   IMAGE                      COMMAND           CREATED          STATUS          P
```

Running the dockerized containers

```
[+] Running 9/9
✓ Network fms_microservices_fms-net      Created          0.1s
✓ Container service-registry             Healthy         60.7s
✓ Container rabbitmq                   Healthy         123.5s
✓ Container mysql                      Healthy         60.7s
✓ Container config-server              Healthy         123.2s
✓ Container api-gateway                Started        59.5s
✓ Container notification-microservice Started        123.9s
✓ Container booking-microservice       Started        123.2s
✓ Container flight-microservice        Started        123.3s
PS C:\Users\KIIT\Desktop\chubb_training\FMS_Microservices> docker ps
CONTAINER ID   IMAGE                               COMMAND                  CREATED     STATUS          NAMES
d03227a0cea3c   fms_microservices-booking-microservice "java -jar app.jar"   2 minutes ago   Up 38 seconds   booking-microservice
0.0.0.0:8082->8082/tcp, [::]:8082->8082/tcp
08e848c18dfe   fms_microservices-flight-microservice "java -jar app.jar"   2 minutes ago   Up 38 seconds   flight-microservice
0.0.0.0:8081->8081/tcp, [::]:8081->8081/tcp
5446d922fc4f   fms_microservices-notification-microservice "java -jar app.jar"  2 minutes ago   Up 38 seconds   notification-microservice
0.0.0.0:8083->8083/tcp, [::]:8083->8083/tcp
38771e10d0f1   fms_microservices-config-server      "java -jar app.jar"   2 minutes ago   Up About a minute (healthy) config-server
0.0.0.0:8888->8888/tcp, [::]:8888->8888/tcp
1d4d63f19348   rabbitmq:3-management               "docker-entrypoint.s..." 2 minutes ago   Up 2 minutes (healthy) rabbitmq
0.0.0.0:5672->5672/tcp, [::]:5672->5672/tcp, 0.0.0.0:15672->15672/tcp, [::]:15672->15672/tcp
09ff251db821   fms_microservices-service-registry "java -jar app.jar"   2 minutes ago   Up 2 minutes (healthy) service-registry
0.0.0.0:8761->8761/tcp, [::]:8761->8761/tcp
07c7607a5b10   mysql:8.0                          "docker-entrypoint.s..." 2 minutes ago   Up 2 minutes (healthy) mysql
0.0.0.0:3306->3306/tcp, [::]:3306->3306/tcp
PS C:\Users\KIIT\Desktop\chubb_training\FMS_Microservices> docker restart flight-microservice booking-microservice api-gateway
flight-microservice
booking-microservice
api-gateway
PS C:\Users\KIIT\Desktop\chubb_training\FMS_Microservices>
```

Creating the databases inside the mysql container

```
at org.hibernate.boot.registry.internal.StandardServiceRegistryImpl.initiateService(StandardServiceRegistryImpl.java:130) ~[hibernate-core-6.3.36.Final.jar!/:6.3.36.Final]
at org.hibernate.service.internal.AbstractServiceRegistryImpl.createService(AbstractServiceRegistryImpl.java:263) ~[hibernate-core-6.3.36.Final.jar!/:6.3.36.Final]
... 37 common frames omitted

PS C:\Users\KIIT> docker exec -it mysql mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 55
Server version: 8.0.44 MySQL Community Server - GPL

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE flightdbnew;
Query OK, 1 row affected (0.01 sec)

mysql> CREATE DATABASE bookingdbnew;
Query OK, 1 row affected (0.04 sec)

mysql> CREATE DATABASE gatewaydb;
Query OK, 1 row affected (0.01 sec)

mysql> exit;
Bye
PS C:\Users\KIIT> docker logs flight-microservice --tail=100
    at org.springframework.boot.loader.launch.Launcher.launch(Launcher.java:106) ~[app.jar:3.1.0]
```

Docker Desktop

The screenshot shows the Docker Desktop interface. On the left sidebar, there are links for Ask Gordon (BETA), Containers, Images, Volumes, Kubernetes, Builds, Models, MCP Toolkit (BETA), Docker Hub, Docker Scout, and Extensions. The main panel displays the 'fms_microservices' stack under 'C:\Users\KIIT\OneDrive\Desktop\chubb_training\FMS_Microservices'. The stack includes services: 'service-fms_microse' (8761:8761), 'rabbitmq' (15672:15672), 'mysql' (3306:3306), 'notifica...' (8083:8083), 'config-s...' (8888:8888), and 'api-gate...' (8765:8765). A log viewer on the right shows Spring Boot logs from 2025-12-08T16:36:12.585Z to 2025-12-08T16:36:23.554Z. The bottom status bar indicates an engine is running with 6.31 GB RAM, 1.51% CPU usage, and 11.63 GB disk usage (limit 1006.85 GB).

EUREKA SERVER

The screenshot shows the Eureka Server dashboard. At the top, it says 'localhost:8761'. Below that is a search bar and a navigation bar with links like Authn | edX, Lecture Videos | Intr..., Analytics Academy, Machine Learning |..., Build Your First Web..., Google Cybersecuri..., AI Ethics | IBM, and All Bookmarks. The main content area has tabs for 'DS Replicas' and 'localhost'. Under 'DS Replicas', it says 'Instances currently registered with Eureka' and lists four services: API-GATEWAY, BOOKING-MICROSERVICE, CONFIG-SERVER-QUIZAPP, and FLIGHT-MICROSERVICE, each with their respective URLs. Under 'General Info', it shows system metrics: total-avail-memory (96mb), num-of-cpus (8), and current-memorv-usaage (70mb (72%)).

POSTMAN TESTING

Registering the User or Admin (depending on role)

The screenshot shows the Postman interface with the 'FMS' collection selected. The 'register' endpoint is highlighted. The request URL is `http://localhost:8765/auth/register`. The request method is POST. The body is set to raw JSON:

```
1 {
2   "username": "aahana",
3   "password": "password123",
4   "role": "USER"
5 }
```

The response status is 200 OK, with a response time of 1.40 s and a response size of 293 B. The response body is:

```
1 {
2   "id": 3,
3   "role": "USER"
4 }
```

Login as User of Admin

The screenshot shows the Postman interface with the 'FMS' collection selected. The 'signin' endpoint is highlighted. The request URL is `http://localhost:8765/auth/login`. The request method is POST. The body is set to raw JSON:

```
1 {
2   "username": "debashrita",
3   "password": "password123"
4 }
```

The response status is 200 OK, with a response time of 3.13 s and a response size of 440 B. The response body is:

```
1 {
2   "token": "eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJkZWJhc2hyaXRhIiwicm9sZSI6Ii1VTRVIiLCJpYXQiOjE3NjUxNDc3MzYsImV4cCI6MTc2NTIzNDEzNn0.zKbfZCLMV8IkY_MgzUNzGapL3nVmPNqhU8SO-1DWmA"
```

Adding inventory (Only by Admin)

workspace New Import < POST Add ● | POST signr ● | POST Sea ● | POST regi ● | POST Boo ● | GET Searc ● | GET Searc ● > + v No environment v

+ Q Search collections

> FMS

✓ FMS Microservice

- POST Add inventory
- POST Search Flights
- POST Book Ticket
- GET Search by pn
- GET Search by mail
- GET Delete flights
- POST register
- POST signin

> FMS_Reactive

> FMS_Reactive_Mongo

> New Collection

> Quiz_microservices

> Redis tutorial

> Scratch_SpringSecurity

> Spring_Mongo_Reactive

> Springboot_Mongodb

> SpringBootSecurity

View Find and replace Console Terminal Postbot Runner Start Proxy Cookies Vault Trash

HTTP FMS Microservice / Add inventory

POST http://localhost:8765/FLIGHT-MICROSERVICE/api/flight

Docs Params Authorization Headers (10) Body Scripts Settings Cookies

Auth Type Bearer Token Token eyJhbGciOiJIUzI1NiJ9.eyJzdWlOiJhY1...

The authorization header will be automatically generated when you send the request. Learn more about [Bearer Token](#) authorization.

Body Cookies Headers (9) Test Results 403 Forbidden 108 ms 285 B e.g. Save Response ...

Raw Preview Debug with AI 1 Access Denied

Postbot Runner Start Proxy Cookies Vault Trash

Searching for a flight (open for any role)

workspace New Import < POST signr ● | POST Sea ● | POST regi ● | POST Boo ● | GET Searc ● | GET Searc ● | DEL Delet ● > + v No environment v

+ Q Search collections

> FMS

✓ FMS Microservice

- POST Add inventory
- POST Search Flights
- POST Book Ticket
- GET Search by pn
- GET Search by mail
- GET Delete flights
- POST register
- POST signin

> FMS_Reactive

> FMS_Reactive_Mongo

> New Collection

> Quiz_microservices

> Redis tutorial

> Scratch_SpringSecurity

> Spring_Mongo_Reactive

> Springboot_Mongodb

> SpringBootSecurity

View Find and replace Console Terminal Postbot Runner Start Proxy Cookies Vault Trash

HTTP FMS Microservice / Search Flights

POST http://localhost:8765/FLIGHT-MICROSERVICE/api/flight/search

Docs Params Authorization Headers (9) Body Scripts Settings Cookies

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

1 { "onwardFlights": [{ "price": 4500.0, "availableSeatNumbers": ["1A", "1B", "2D", "1E", "1F"], "flightId": "79448598", "airline": "IndiGo", "availableSeats": 5, "dateTime": "2025-12-10T09:30" }] }

Body Cookies Headers (10) Test Results 200 OK 2.37 s 1020 B e.g. Save Response ...

JSON Preview Visualize

Postbot Runner Start Proxy Cookies Vault Trash

Booking a flight (only for User)

HTTP FMS Microservice / Book Ticket

POST http://localhost:8765/BOOKING-MICROSERVICE/api/flight/booking/AFA9CCF7

Docs Params Authorization Headers (10) Body Scripts Settings Cookies

Auth Type: Bearer Token

The authorization header will be automatically generated when you send the request. Learn more about [Bearer Token](#) authorization.

Body Cookies Headers (10) Test Results 201 Created 2.63 s 390 B Save Response

```
{
  "pnr": "E434EF57",
  "message": "Booking successful",
  "totalPrice": 9000.0
}
```

Find and replace Console Terminal Postbot Runner Start Proxy Cookies Vault Trash

Search by PNR (only by User)

workspace New Import

HTTP FMS Microservice / Search by pnr

GET http://localhost:8765/BOOKING-MICROSERVICE/api/flight/ticket/6AF5A39E

Docs Params Authorization Headers (8) Body Scripts Settings Cookies

Auth Type: Bearer Token

The authorization header will be automatically generated when you send the request. Learn more about [Bearer Token](#) authorization.

Body Cookies Headers (10) Test Results 200 OK 340 ms 534 B Save Response

```
{
  "pnr": "6AF5A39E",
  "flightId": "AFA9CCF7",
  "status": "CONFIRMED",
  "seatNumbers": [
    "1A",
    "1B"
  ],
  "passengers": [
  ]
```

Find and replace Console Terminal Postbot Runner Start Proxy Cookies Vault Trash

Search by mail (for User)

Workspaces

- + Search collections
- > FMS
- > FMS Microservice
 - POST** Add inventory
 - POST** search flights
 - POST** Book Ticket
 - GET** Search by pnr
 - GET** Search by mail
 - GET** Delete flights
 - POST** register
 - POST** signin
- > FMS_Reactive
- > FMS_Reactive_Mongo
- > New Collection
- > Quiz_microservices
- > Redis tutorial
- > Scratch_SpringSecurity
- > Spring_Mongo_Reactive
- > Springboot_Mongodb
- > SpringBootSecurity

HTTP FMS Microservice / Search by mail

GET http://localhost:8765/BOOKING-MICROSERVICE/api/flight/booking/history/debashritamandal852@gmail.com

Authorization

Auth Type: Bearer Token

The authorization header will be automatically generated when you send the request. Learn more about [Bearer Token](#) authorization.

Body Cookies Headers (11) Test Results

200 OK 135 ms 847 B Save Response

```
{
  "email": "debashritamandal852@gmail.com",
  "history": [
    {
      "pnr": "D4F30643",
      "status": "CANCELLED",
      "flightId": "79448598"
    }
  ]
}
```

Find and replace Console Terminal

Postbot Runner Start Proxy Cookies Vault Trash

Delete the ticket (only by User)

Workspaces

- + Search collections
- > FMS
- > FMS Microservice
 - POST** Add inventory
 - POST** search flights
 - POST** Book Ticket
 - GET** Search by pnr
 - GET** Search by mail
 - GET** Delete flights
 - POST** register
 - POST** signin
- > FMS_Reactive
- > FMS_Reactive_Mongo
- > New Collection
- > Quiz_microservices
- > Redis tutorial
- > Scratch_SpringSecurity
- > Spring_Mongo_Reactive
- > Springboot_Mongodb
- > SpringBootSecurity

HTTP FMS Microservice / Delete flights

DELETE http://localhost:8765/BOOKING-MICROSERVICE/api/flight/booking/cancel/97A2B1B1

Authorization

Auth Type: Bearer Token

The authorization header will be automatically generated when you send the request. Learn more about [Bearer Token](#) authorization.

Body Cookies Headers (10) Test Results

200 OK 833 ms 376 B Save Response

```
{
  "pnr": "97A2B1B1",
  "message": "Ticket cancelled successfully"
}
```

Find and replace Console Terminal

Postbot Runner Start Proxy Cookies Vault Trash