A Report

On

# **“ Flight Management System””**

Submitted by

**Mr. Rohit Sunil Pawar**
**UCS22M1104**
**Roll no. 115**

Under the guidance of

**Prof. P. M. Dhanrao**

**DEPARTMENT OF COMPUTER ENGINEERING**

**SANJIVANI COLLEGE OF ENGINEERING, KOPARGAON, 423603**

**Tal- Kopargaon, Dist: Ahmednagar**

**Academic Year: 2024-25**

**SANJIVANI COLLEGE OF ENGINEERING, KOPARGAON**

**(An Autonomous Institute)**

## DEPARTMENT OF COMPUTER ENGINEERING

# CERTIFICATE

This is to certify that **Mr. Rohit Sunil Pawar** has successfully completed her T.Y B. Report on the topic entitled **“Flight Management System”** This work has been carried out as part of the requirements for the award of the degree of **Bachelor of Computer Engineering** and is submitted in partial fulfillment of the prescribed syllabus for the academic year **2024-25**. We commend her dedication and effort in addressing this innovative and relevant topic, which contributes to the advancement of smart city technologies.

| | |
|---|---|
| **Prof. P. M. Dhanrao** | **Dr. M. A. Jawale** |
| Subject Co-ordinator | Head of Department |

## ACKNOWLEDGEMENT

It gives me immense pleasure to express my deepest and most heartfelt gratitude to my esteemed guide, **Prof. P. M. Dhanrao**, whose invaluable guidance, unwavering support, and continuous encouragement have been the cornerstone of this seminar. His profound knowledge, patience, and insightful suggestions have illuminated every step of my journey throughout this seminar preparation. Prof. Dhanrao has not only imparted his vast academic expertise but also demonstrated exceptional leadership and mentorship, guiding me through every challenge with a steady hand. His constant motivation and constructive feedback have been instrumental in shaping this report, and for that, I am deeply appreciative.

I am also profoundly thankful to **Dr. M. A. Jawale**, our respected Head of the Department (HOD), for his consistent inspiration, vision, and encouragement. His leadership has provided a strong academic foundation, fostering an environment that promotes both learning and innovation. I am immensely grateful for his unwavering belief in my abilities, which has given me the confidence to strive for excellence in my work.

My heartfelt thanks also go out to my Subject Co-ordinator, **Prof. P. M. Dhanrao**, for his invaluable support and guidance throughout the course of this seminar. His meticulous attention to detail and consistent oversight have been critical in ensuring the timely and successful completion of this work. His expert advice and assistance have been pivotal in navigating through the complexities of this project, and I am sincerely grateful for his constant encouragement and cooperation.

**Mr. Rohit Sunil Pawar**

(T.Y B. Tech Computer Engineering)

# Abstract

The Flight Management System is a Java-based application developed to automate and streamline the management of airline crew members, including pilots and cabin staff. The system provides functionalities for registering crew members, assigning them to flights based on availability and qualifications, and managing scheduling constraints to ensure compliance with aviation regulations. It aims to reduce human error, prevent scheduling conflicts, and improve operational efficiency in airline management. By utilizing core Java technologies and database integration, the system offers a scalable and user-friendly solution for crew resource planning in the aviation industry.

The traditional manual process of crew assignment is often prone to scheduling conflicts, underutilization, and regulatory violations regarding working hours and rest periods. This system addresses these issues by offering a structured and automated solution.

# List of Tables

| Figure No. | Name of Chapters | Page No. |
|---|---|---|
| 1 | **Introduction** | 1 |
| 2 | **Objective** | 3 |
| 3 | **Technologies** | 5 |
| 4 | **System Architecture** | 7 |
| 5 | **Conclusion** | 9 |
| 6 | **References** | 13 |

# Chapter 1

# Introduction

In the modern aviation industry, efficient management of airline is crucial for ensuring timely operations, safety compliance, and optimal resource utilization. Airline operations involve a complex interplay between aircraft, schedules, crew members, and regulatory constraints. Traditionally, managing the crew roster manually or through semi-automated spreadsheets often leads to scheduling conflicts, overbooking, underutilization, and non-compliance with legal flight hour regulations.

The Airline Crew Management System, developed using Java, is designed to address these challenges by providing a structured and automated platform for managing crew data, flight assignments, availability tracking, and compliance monitoring. The system enables airline administrators to add, update, and delete crew records, assign crew members to flights based on qualifications and availability, and prevent scheduling conflicts or regulatory violations.

This application leverages the core features of Java, such as object-oriented programming and modular design, along with database connectivity using JDBC to interact with a relational database system. It also uses a graphical user interface (GUI) built with Java Swing or JavaFX for improved usability and user experience.

The goal of this project is not only to optimize crew scheduling and assignment but also to lay a foundation for a scalable, real-world airline operations system. The system can be expanded in the future to support web or mobile platforms, integrate with real-time flight tracking systems, and include advanced features like performance analytics and notification alerts.

This report outlines the system's objectives, design, implementation, and potential for future enhancement, highlighting how Java can be used to solve real-world logistical problems in a critical sector like aviation.

# Chapter 2

# Objectives

The primary objectives of the Flight Management System developed using Java are as follows:

1. Automate Crew Scheduling

To automate the process of assigning airline crew members (pilots, co-pilots, and cabin crew) to flights efficiently and accurately.

2. Manage Crew Records

To maintain detailed records of each crew member, including personal information, role, qualifications, availability, and work history.

3. Prevent Scheduling Conflicts

To avoid double-booking, crew overwork, and ensure adherence to aviation regulations concerning work hours and mandatory rest periods.

4. Ensure Qualification Compliance

To assign crew members only to flights for which they are trained and certified, ensuring flight safety and regulatory compliance.

5. Improve Operational Efficiency

To reduce manual errors, save administrative time, and improve the overall efficiency of crew resource planning.

# Chapter 3

# Technologies

The Flight Management System leverages a combination of programming languages, libraries, frameworks, and tools to deliver a reliable and efficient application. The primary technologies used are:

### 1. Java SE (Standard Edition)

- Java is the core programming language used for building the logic of the application.
- Object-Oriented Programming (OOP) principles such as **inheritance**, **polymorphism**, **encapsulation**, and **abstraction** are used to model real-world entities like CrewMember, Pilot, Flight, and Assignment.
- Java provides robust features for exception handling, collections, threading (if required), and file I/O.

**Why Java?**

- Platform-independent (Write Once, Run Anywhere)
- Strong community support
- Reliable for enterprise-level applications

### 2. JDBC (Java Database Connectivity)

- JDBC is used to establish a connection between the Java application and the relational database (MySQL).
- Through JDBC APIs, SQL queries are executed directly from the application to: o Insert, update, delete crew and flight data.
  - Retrieve assignment records.
  - Perform data validation based on crew availability and qualifications.

**Example Code Snippet:**

java

Copy code

```
Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/airline_db", "root", "password");
PreparedStatement ps = con.prepareStatement("SELECT * FROM crew_members WHERE id = ?");
```

**3. MySQL (Relational Database Management System)**

- MySQL is used as the backend database to persistently store all data related to: o Crew members (ID, name, role, certification, availability) o Flights (flight number, route, date, required crew) o Assignments (link between flight and crew)
- The database schema is normalized to eliminate redundancy and ensure integrity.

**Tables Example:**

- crew_members: stores all crew data.
- flights: stores flight details.
- assignments: maps flights to assigned crew members.

**4. Java Swing / JavaFX (Graphical User Interface - GUI)**

- A user-friendly desktop interface is created using **Swing** or **JavaFX** to allow airline administrators to:
  - Add/edit crew details. o Assign crew to flights.
  - View schedules.

  - Search or filter crew members.

**Java Swing Features Used:**

- JFrame, JTable, JTextField, JButton, JComboBox
- Action listeners for event handling

**JavaFX Features (if used):**

- Layouts like BorderPane, VBox, GridPane
- CSS styling for improved UI appearance

---

### 5. Apache Maven (Optional)

- Maven is used for **build automation** and **dependency management**.
- It simplifies project configuration and ensures all required libraries (like the MySQL JDBC driver) are properly included in the project.

**Benefits of Maven:**

- Handles project builds via a pom.xml file.
- Manages classpaths and libraries.
- Facilitates clean project structure.

### 6. Development Tools

- **IDE**: Eclipse, IntelliJ IDEA, or NetBeans
  Used for writing, debugging, and testing Java code efficiently.
- **Database Tools**: MySQL Workbench / phpMyAdmin
  Used for creating the schema, executing SQL queries, and testing data connectivity.
- **Version Control (Optional)**: Git
  Useful for tracking changes, versioning, and collaboration.

**Optional Tools & Libraries (For Enhancement)**

- **Spring Boot**: For converting the app into a web-based platform.
- **Hibernate ORM**: For simplifying database interactions.
- **JUnit**: For unit testing Java classes.
- **Log4j or SLF4J**: For application logging.

# Chapter 4

## System Architecture

The Flight Management System follows a **layered architecture** that separates the application into distinct modules, making it easier to develop, test, and maintain. The system is designed using core Java components, a relational database (MySQL), and JDBC for data access. The architecture can be broadly divided into the following layers:

**1. Presentation Layer (GUI)**

- Implemented using **Java Swing** or **JavaFX**.
- Provides a graphical user interface for the administrator to:
  - Add/edit crew and flight information.
  - Assign crew to flights.
  - View flight schedules.
- Sends user input to the logic layer and displays output to the user.

**2. Application Logic Layer (Business Logic)**

- Contains core Java classes like:
  - CrewMember, Pilot, CabinCrew
  - Flight, Assignment
- Performs validation and business rules such as:
  - Checking crew availability.
  - Preventing over-assignment.
  - Ensuring crew qualifications match flight requirements.

### 3. Data Access Layer (DAO - Data Access Objects)

- Manages interaction between the application and the database using **JDBC**.
- DAO classes like CrewDAO, FlightDAO, and AssignmentDAO handle: o Insert/update/delete operations.
    - SQL queries to retrieve or filter data

### 4. Database Layer

- MySQL is used as the relational database.
- Stores persistent data in tables: o crew_members: stores crew data. o flights: stores flight details.
    - assignments: maps crew to flights.
- Foreign key relationships ensure data integrity.

### Data Flow Overview

1. The user interacts with the GUI.
2. The GUI sends input to the business logic classes.
3. Logic classes call DAO methods to access or update the database.
4. DAO executes SQL queries and returns results.
5. The GUI displays results or status messages to the user.

# Conclusion

The Flight Management System demonstrates how Java can effectively address real-world challenges in the aviation sector through structured, object-oriented design and reliable database integration. By automating core tasks such as crew scheduling, assignment, and compliance monitoring, the system reduces human error, enhances operational efficiency, and ensures adherence to regulatory standards. The use of Java Swing or JavaFX for the GUI provides an intuitive interface for administrators, making the system both practical and user-friendly. With its modular architecture, the project lays a solid foundation for future scalability— enabling integration with web or mobile platforms, real-time data systems, and advanced analytical tools. Ultimately, this project showcases the potential of Java-based solutions in optimizing complex logistics and improving decision-making in critical industries like aviation.

# References

1. **Java Documentation**
   Oracle. (2025). *The Java™ Programming Language*. Retrieved from https://docs.oracle.com/javase/

2. **JDBC API Guide**
   Oracle. (2025). *JDBC API Tutorial and Reference, Fourth Edition*. Retrieved from https://docs.oracle.com/javase/8/docs/technotes/guides/jdbc/

3. **Java Swing Documentation**
   Oracle. (2025). *Swing Tutorial*. Retrieved from https://docs.oracle.com/javase/tutorial/uiswing/

4. **Effective Java**
   Bloch, J. (2018). *Effective Java (3rd Edition)*. Addison-Wesley Professional.

5. **Database Design and Management**
   Rob, P., & Coronel, C. (2020). *Database Systems: Design, Implementation, & Management (13th Edition)*. Cengage Learning.

6. **Aviation Operations and Scheduling Systems**
   Smith, A. (2019). *Optimizing Airline Crew Scheduling: Challenges and Solutions*. Journal of Aviation Management, 22(3), 45-62.

7. **Introduction to Object-Oriented Programming with Java**
   Liang, Y. D. (2016). *Introduction to Java Programming (10th Edition)*. Pearson.

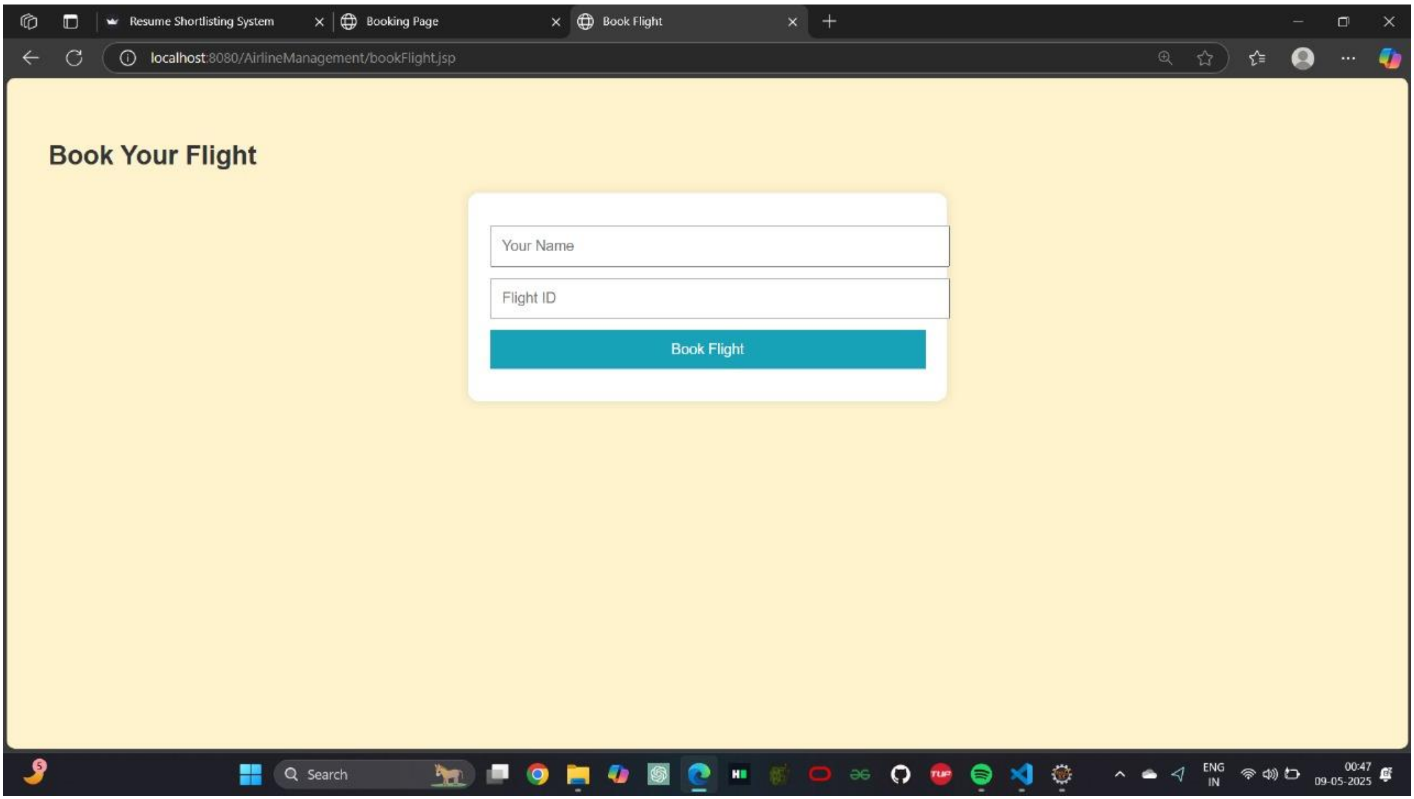
# Future Scope

**Future Scope of the Flight Management System**

1. **Mobile App Development**
   - Launch dedicated Android and iOS applications for easy ticket booking, check-in, and flight tracking.
   - Include push notifications for boarding calls, gate changes, and delays.
2. **Real-Time Flight Tracking & Updates**
   - Integrate GPS and real-time data APIs to track flights and notify passengers of delays, diversions, or cancellations.
   - Improve customer experience with dynamic ETA updates.
3. **Integration with Global Distribution Systems (GDS)**
   - Connect with platforms like Amadeus, Sabre, or Galileo to access real-time inventory and expand booking capabilities.
4. **Online Payment Gateway Integration**
   - Support secure multi-currency transactions through UPI, credit/debit cards, PayPal, and other digital wallets.
   - Include options for EMI or ticket insurance at checkout.
5. **Biometric Check-In and Security**
   - Implement facial recognition and biometric authentication for faster and more secure check-in and boarding processes.
6. **Predictive Analytics and AI**
   - Use machine learning for demand forecasting, dynamic pricing, and personalized travel suggestions.
   - AI chatbots for instant customer support and FAQs.
7. **Automated Baggage Handling System Integration**
   - Enable tracking of checked-in luggage with RFID tags.
   - Notify passengers about baggage status in real-time.
8. **Admin Dashboard and Reporting Tools**
   - Provide analytics on flight occupancy, route profitability, passenger feedback, etc.
   - Help airlines make strategic business decisions.
9. **Code Sharing and Partner Airline Integration**
   - Allow passengers to book connecting flights with partner airlines seamlessly.
   - Simplify international and multi-leg bookings.
10. **Multilingual and Accessibility Features**

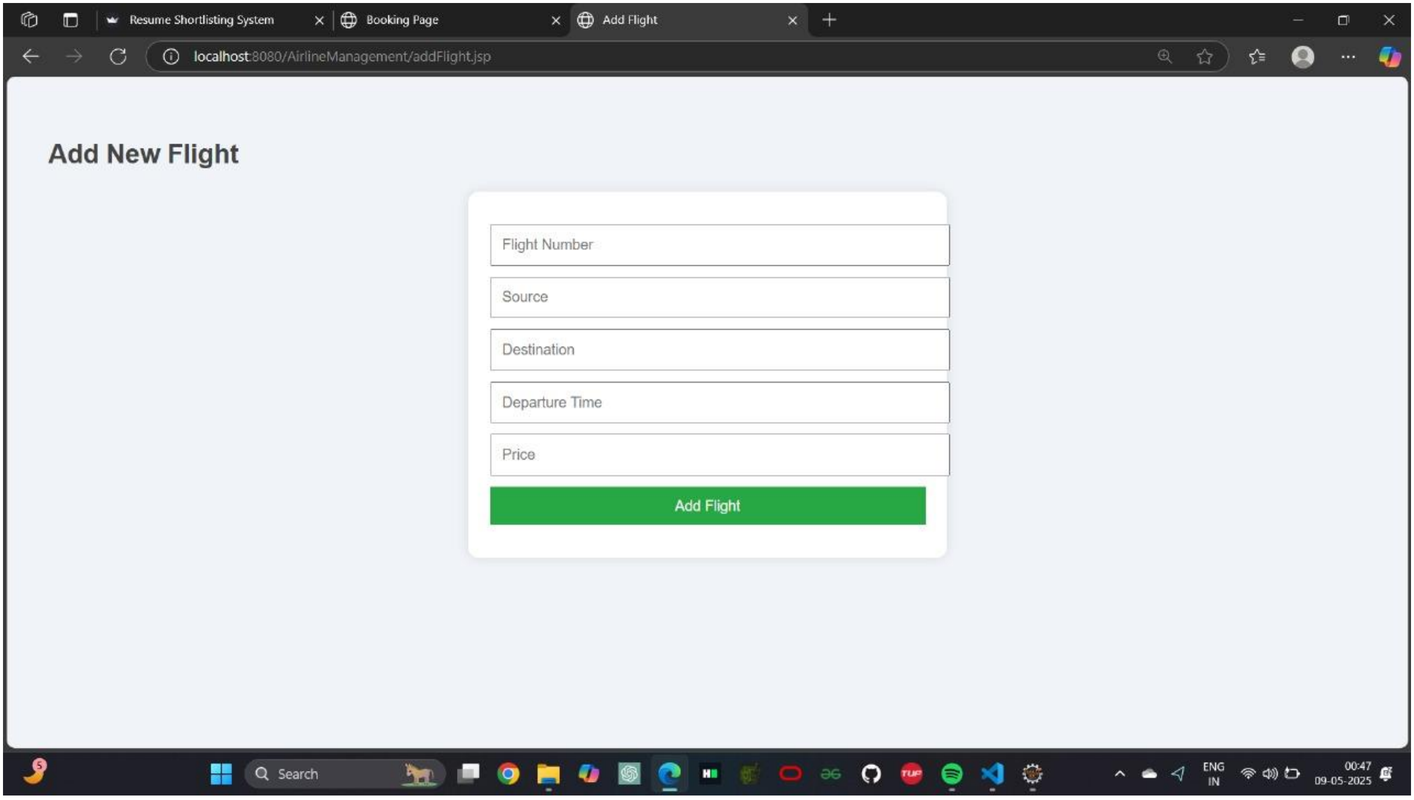
- Support multiple languages and voice-assisted navigation to enhance global user experience.
- Ensure accessibility compliance for differently-abled users.

# Snapshots



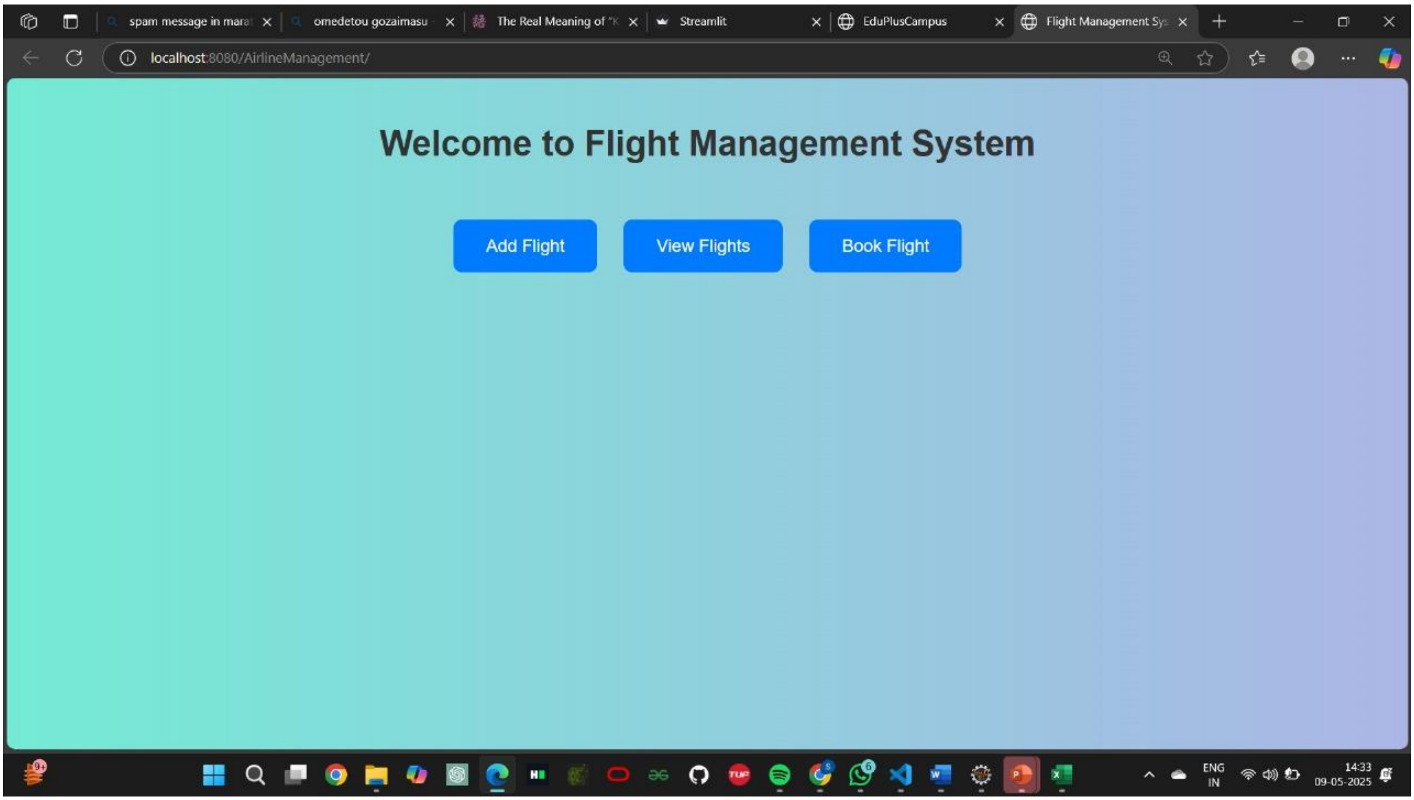
localhost:8080/AirlineManagement/bookFlight.jsp

Book Your Flight

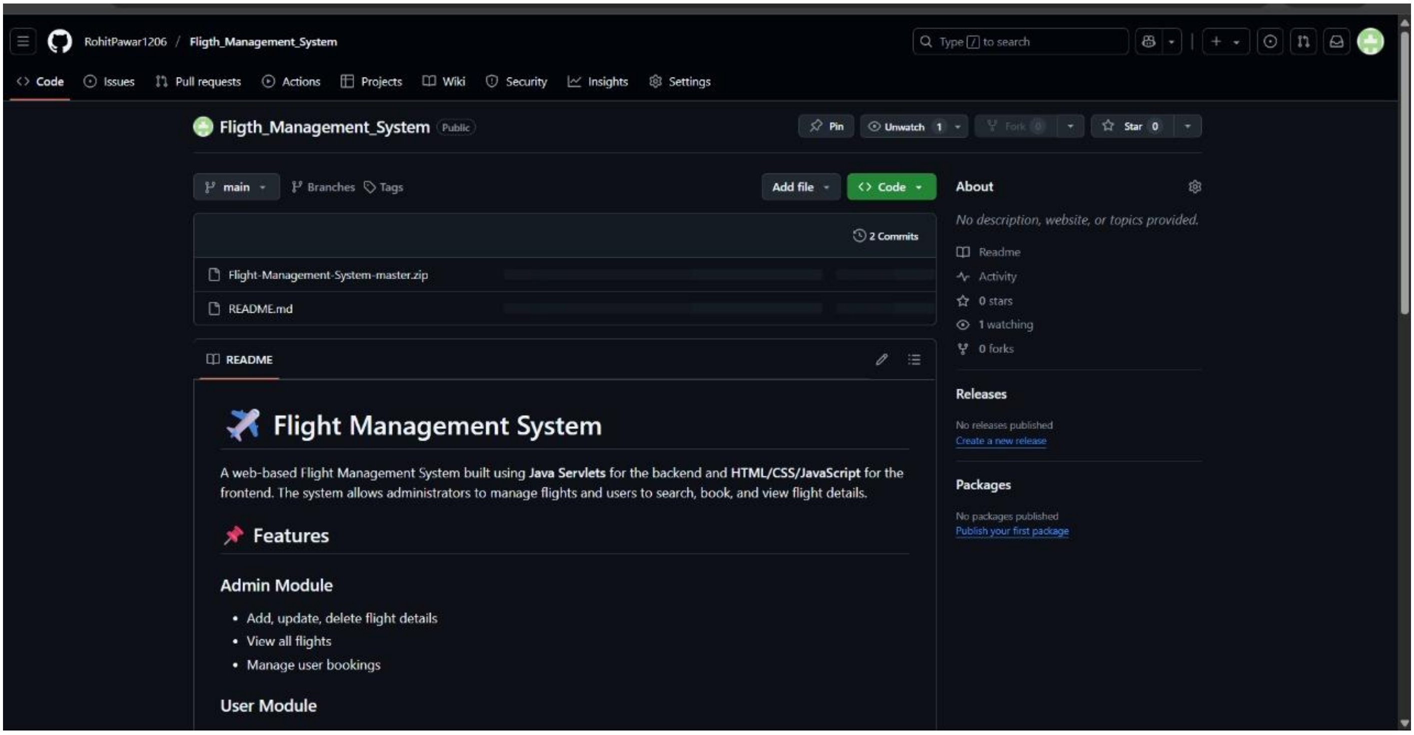


localhost:8080/AirlineManagement/addFlight.jsp

Add New Flight



Welcome to Flight Management System



No description, website, or topics provided.

Flight-Management-System-master.zip

README.md

✈️ Flight Management System

A web-based Flight Management System built using **Java Servlets** for the backend and **HTML/CSS/JavaScript** for the frontend. The system allows administrators to manage flights and users to search, book, and view flight details.

📌 Features

Admin Module

- Add, update, delete flight details
- View all flights
- Manage user bookings

User Module