**Airline Management System - Complete Project Documentation**

## 1. Introduction

The Airline Management System is a comprehensive Java-based application designed to streamline flight bookings, customer interactions, and administrative tasks for Air Axel Airlines. The system provides a user-friendly interface for customers to book flights, view flight details, submit feedback, and access help services. Additionally, it offers an administrative portal for managing flight schedules, updating flight information, and handling user authentication.

# **1.1 Purpose**

Provide a seamless flight booking experience for customers.

Allow administrators to manage flight schedules, add/remove flights, and update flight statuses.

Facilitate customer support through feedback and help sections.

Ensure secure authentication for admin access.

# **1.2 Key Features**

**User Interface (UI)**

Entry Page: A splash screen introducing the airline.

Landing Page: Main dashboard with navigation options.

Flight Booking: Multi-step form for booking tickets.

Search Flights: View available flights with filtering options.

Feedback System: Allows customers to submit complaints/suggestions.

Help & Contact: Provides customer support information.

Our Plans: Displays flight class options (Economy, Business, First Class).

**Admin Panel**

Login System: Secure admin authentication.

Flight Management: Add, delete, and update flights.

Customer Feedback Management: View and manage customer feedback.

Database Integration: Stores flight and admin credentials.

Additional Features

Ticket Generation: Downloads boarding passes in .png and .txt formats.

Responsive UI: Clean and intuitive design with hover effects.

## 2. System Architecture

# **2.1 Technologies Used**

Programming Language: Java (Swing for GUI)

Database: MySQL (for admin authentication)

Libraries:

javax.swing (GUI components)

java.awt (Graphics & event handling)

java.sql (Database connectivity)

javax.imageio (Image handling)

# **2.2 Class Structure**

|  |  |
| --- | --- |
| Class Name | Description |
| Entry.java | Splash screen with "Press to continue" button. |
| Landing.java | Main dashboard with navigation buttons. |
| BookTicket.java | Handles flight booking and ticket generation. |
| Amin\_Login.java | Admin login with database authentication. |
| ManageFlight.java | Admin panel for flight CRUD operations. |
| Menu\_page.java | Admin dashboard with navigation options. |
| ViewFeedback.java | Displays customer feedback in a table. |
| Feedback.java | Customer feedback submission form. |
| Help.java | Help & contact information page. |
| About\_Us.java | Information about Air Axel. |
| OurPlans.java | Displays flight class options. |
| searchFlight.java | Allows users to search and filter flights. |
| DatabaseConnection.java | Manages MySQL database connections. |
|  |  |

## 3. Detailed Module Description

# **3.1 User Modules**

### **3.1.1 Flight Booking**

**Features:**

Two-step form (Flight Details → Passenger Details).

Input validation (ensures required fields are filled).

Generates a unique ticket number (AX-12345678).

Downloads boarding pass as both an image and text file.

### **3.1.2 Search Flights**

**Features:**

Displays all available flights in a table.

Filter options (By Status, From, To, Departure Time).

Search functionality (text-based search).

Special offers highlighted.

### **3.1.3 Feedback System**

**Fields:**

Name, Email, Complain Topic, Feedback, Suggestions.

Submission: Displays a confirmation dialog.

### **3.1.4 Help & Contact**

**Sections:**

Sales Office (Address: 11-E, Egerton Road, Lahore).

Email (airaxal@gmail.com).

Phone (042-36301854).

FAQ (Frequently Asked Questions).

### **3.1.5 Our Plans (OurPlans.java)**

**Flight Classes:**

Economy Class: Budget-friendly option.

Business Class: Premium comfort for professionals.

First Class: Ultimate luxury experience.

# **3.2 Admin Modules**

### **3.2.1 Admin Login (Amin\_Login.java)**

**Authentication:**

Validates username/password against MySQL database.

Shows error messages for invalid credentials.

Redirects to Menu\_page on success.

### **3.2.2 Flight Management**

**Functions:**

Add Flight: Input flight details (Flight No, From, To, Departure, Arrival, Price, Status).

Delete Flight: Remove a flight by Flight No.

Update Flight: Modify existing flight details.

**UI:**

Table view of all flights.

Buttons for CRUD operations.

# **3.2.3 Feedback Management**

**Features:**

Displays customer feedback in a table.

Columns: Name, Email, Complain Topic, Feedback, Suggestions.

Text wrapping for long feedback entries.

### **3.2.4 Admin Dashboard**

**Navigation Options:**

|  |  |
| --- | --- |
| Manage Flights: | Redirects to ManageFlight.java. |
| Manage Customers: | Redirects to ManagePassenger.java (future implementation). |
| Feedback: | Redirects to ViewFeedback.java. |
| Back: | Returns to Landing.java. |

## 4. Database Design

# **4.1 MySQL Database**

The system uses a MySQL database named airline\_db with the following tables:

**Table: flights**

|  |  |  |
| --- | --- | --- |
| Column | Type | Description |
| flight\_no | VARCHAR(10) | Primary key (e.g., PK201). |
| departure\_city | VARCHAR(50) | Departure city (e.g., Karachi). |
| destination\_city | VARCHAR(50) | Destination city (e.g., Lahore). |
| departure\_time | VARCHAR(10) | Departure time (e.g., 10:00 AM). |
| arrival\_time | VARCHAR(10) | Arrival time (e.g., 12:00 PM). |
| duration | VARCHAR(10) | Flight duration (e.g., 2h). |
| price | DECIMAL(10,2) | Ticket price (e.g., 25000.00). |
| status | VARCHAR(20) | Flight status (e.g., Available). |

**Table: passengers**

|  |  |  |
| --- | --- | --- |
| Column | Type | Description |
| passenger\_id | VARCHAR(10) | Primary key (e.g., P001). |
| name | VARCHAR(100) | Passenger name (e.g., Ahmed Khan). |
| email | VARCHAR(100) | Passenger email. |
| phone | VARCHAR(20) | Passenger phone number. |
| flight\_no | VARCHAR(10) | Foreign key referencing flights. |
| departure\_city | VARCHAR(50) | Departure city. |
| destination\_city | VARCHAR(50) | Destination city. |
| boarding\_pass | VARCHAR(20) | Boarding pass number. |
| seat\_no | VARCHAR(10) | Seat number (e.g., 12A). |
| status | VARCHAR(20) | Booking status (e.g., Confirmed). |

**Table: tickets**

|  |  |  |
| --- | --- | --- |
| Column | Type | Description |
| ticket\_number | VARCHAR(20) | Primary key (e.g., AX-87654321). |
| departure\_city | VARCHAR(50) | Departure city. |
| destination\_city | VARCHAR(50) | Destination city. |
| departure\_date | DATE | Departure date (e.g., 2023-12-15). |
| trip\_type | ENUM | One Way or Return. |
| first\_name | VARCHAR(50) | Passenger first name. |
| last\_name | VARCHAR(50) | Passenger last name. |
| nationality | VARCHAR(50) | Passenger nationality. |
| passport\_number | VARCHAR(20) | Passenger passport number. |
| travel\_class | ENUM | Economy, Business, or First Class. |
| issue\_date | TIMESTAMP | Ticket issue timestamp. |

**Table: feedback**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column | Type |  | Description |  |
| id |  | INT |  | Auto-incremented primary key. |
| Name |  | VARCHAR(100) |  | Customer name. |
| email |  | VARCHAR(100) |  | Customer email. |
| topic |  | VARCHAR(100) | Feedback topic (e.g., Service Quality). |  |
| Feedback | TEXT |  | Detailed feedback. |  |
| suggestions |  | TEXT |  | Customer suggestions. |

**Table: Admins**

|  |  |  |  |
| --- | --- | --- | --- |
| Column |  | Type | Description |
| Username |  | VARCHAR(50) | Primary key (e.g., saad). |
| password |  | VARCHAR(50) | Admin password (e.g., 12345678). |
| full\_name |  | VARCHAR(100) | Admin full name (e.g., saad ali). |

5. User Interface (UI) Design

# **5.1 Color Scheme**

|  |  |
| --- | --- |
| Component | Color Code |
| Primary (Red) | #EA2F4A RGB...……….234 47 74 |
| Secondary (Blue) | #30A6C1...……………………RGB 48 166 193 |

# **5.2 Fonts**

Primary: Montserrat (Bold for headings, Regular for text).

# **5.3 Key UI Components**

Buttons: Rounded, hover effects, hand cursor.

Forms: Input validation, error highlighting.

Tables: Centered text, alternating row colors.

## 6. Database Integration

## The system integrates with the MySQL database (airline\_db.sql) to manage:

## Flight Data: CRUD operations for flights.

## Passenger Data: Booking and boarding pass generation.

## Feedback: Submission and viewing.

## Admin Authentication: Secure login for administrators.

## 7. Error Handling & Validation

# **Input Validation:**

Ensures required fields are filled (e.g., name, flight details).

Displays error messages for invalid inputs.

# **Database Errors:**

Shows SQL exceptions in console.

Displays user-friendly error messages.

## 8. Future Enhancements

* Online Payment Integration (e.g., Stripe, PayPal).
* Email Notifications (Booking confirmations).
* Real-time Flight Status Updates (API integration).
* User Accounts (Profile management, booking history).

## 9. Conclusion

The Airline Management System provides a robust solution for flight bookings and administrative tasks. It features a clean UI, secure authentication, and efficient flight management. Future updates can expand functionality with payment gateways and real-time data.