A logo of a person reading a book

AI-generated content may be incorrect.A blue and white logo

AI-generated content may be incorrect. Shri Vile Parle Kelavani Mandal’s

SHRI BHAGUBHAI MAFATLAL POLYTECHNIC

AND COLLEGE OF ENGINEERING

**INFORMATION TECHNOLOGY DEPARTMENT**

SELF LEARNING ASSESSMENT MINI PROJECT REPORT

Submitted in partial fulfilment of the requirements of the

DIPLOMA course by:

|  |
| --- |
| NAME OF THE STUDENT: Bhavika Dhakate |
| ROLL NO : T001 |
| SAP ID : 57498230001 |
| PROJECT TOPIC : Airline Management System |
| COURSE : Programming In Python (PRP230908) |
| SEMESTER : 4th |
| ACADEMIC YEAR : 2024-2025 |
| NAME OF DEPARTMENT: INFORMATION TECHNOLOGY |

THIS IS TO CERTIFY THAT

Ms. Bhavika Dhakate Roll no: T001 has satisfactorily completed her Self Learning Assessment Mini Project and has submitted the report.

Signature of Faculty

A logo of a person reading a book

AI-generated content may be incorrect.A blue and white logo

AI-generated content may be incorrect. Shri Vile Parle Kelavani Mandal’s

SHRI BHAGUBHAI MAFATLAL POLYTECHNIC

AND COLLEGE OF ENGINEERING

**INFORMATION TECHNOLOGY DEPARTMENT**

FOURTH SEMESTER

Course: Programming in Python Course Code: PRP230908

|  |  |
| --- | --- |
| **CO No.** | **COURSE OUTCOME** |
| CO1 | Implement Python fundamentals and control statements |
| CO2 | Use String, List, Tuple, Dictionary and Sets appropriately in Python Scripts |
| CO3 | Execute functions, modules, and packages proficiently |
| CO4 | Code Object Oriented, Exception Handling, Multithreading concepts robustly |
| CO5 | Use Regular expressions to search for patterns in strings and perform file handling operations |
| CO6 | Develop GUI desktop application with database connectivity |

A logo of a person reading a book

AI-generated content may be incorrect.A blue and white logo

AI-generated content may be incorrect.

**Shri Vile Parle Kelvani Mandal’s**

**SHRI BHAGUBHAI MAFATLAL POLYTECHNIC AND COLLEGE OF ENGINEERING**

**IRLA, VILE PARLE(WEST), MUMBAI - 400056**

**CERTIFICATE**

**THIS IS TO CERTIFY THAT**

**Ms. Bhavika Dhakate**

**HAS SATISFACTORILY COMPLETED HER SELF**

**LEARNING ASSESSMENT MINI PROJECT**

**REPORT IN PARTIAL FULFILMENT FOR THE**

**DIPLOMA COURSE IN**

**INFORMATION TECHNOLOGY**

**FACULTY HEAD OF DEPARTMENT (I.T) PRINCIPAL**

**Problem Statement:-**

Celestia Airlines requires an interactive airline management system to streamline flight bookings, seat reservations, and user authentication for both customers and administrators. The application should integrate weather data for safety checks, offer a graphical interface for usability, and ensure real-time management of ticketing operations and user credentials using a robust backend system.

**Features of Celestia Airlines:-**

### **1. Authentication & User Roles**

* **Admin Login:** Secure admin authentication to access management features.
* **Customer Login & Registration:** Allows new user sign-up and returning user login.

**2. Admin Dashboard**

* Centralized panel for managing system data.
* **Features:**
  + Add/update/delete flights.
  + Assign pilots and crew.
  + View system activity (flights, bookings).
  + Access to all customer bookings.

**3. Flight Search & Booking (Customer)**

* Customers can search flights using source and destination filters.
* **Details displayed include:**
  + Flight number, airline, route, departure/arrival time, and fare.
* Booking is handled with:
  + Passenger input form.
  + Real-time validation and confirmation.

**4. Visual Seat Map Selection**

* Graphical interface to choose available seats.
* Selected seats are stored in the booking record.
* Helps avoid double booking and improves user experience.

**5. PDF Ticket Generation**

* After a successful booking, customers receive a downloadable ticket.
* **Ticket includes:**
  + Passenger name, flight number, seat number, route, and booking ID.
  + Date of travel and QR code or digital stamp (optional for future).

**6. Customer Dashboard**

* Allows customers to:
  + View current and past bookings.
  + Reprint/download tickets.
  + Cancel upcoming flights (optional feature).

**7. Weather Integration**

* Real-time weather data for source and destination.
* Integrated using OpenWeatherMap API.
* Shows temperature, humidity, conditions, and wind speed with icons.

**8. MySQL Database Integration**

* Centralized database using MySQL via XAMPP.
* Manages:
  + Users, flights, crew, seats, bookings.
* Secure connection via database.py.

**9. GUI Interface (Tkinter)**

* Clean and modular design with:
  + Pop-up dialogs
  + Dropdown menus
  + Separate admin and customer windows

## **Softwares Used i.e. Python Version, Sublime Text3 or any IDE with logo and brief description and any other third-party libraries:-**

**Python 3.10+**



* Primary programming language used.
* Supports GUI (Tkinter), database connectors (MySQL), and HTTP requests.
* Enables modular OOP-based design and integration with third-party APIs.

**Sublime Text 3**



* Lightweight code editor used for development.
* Features syntax highlighting, split editing, and command palette.
* Fast performance and customizable with plugins.

**XAMPP with phpMyAdmin (MySQL)**

A white and orange logo

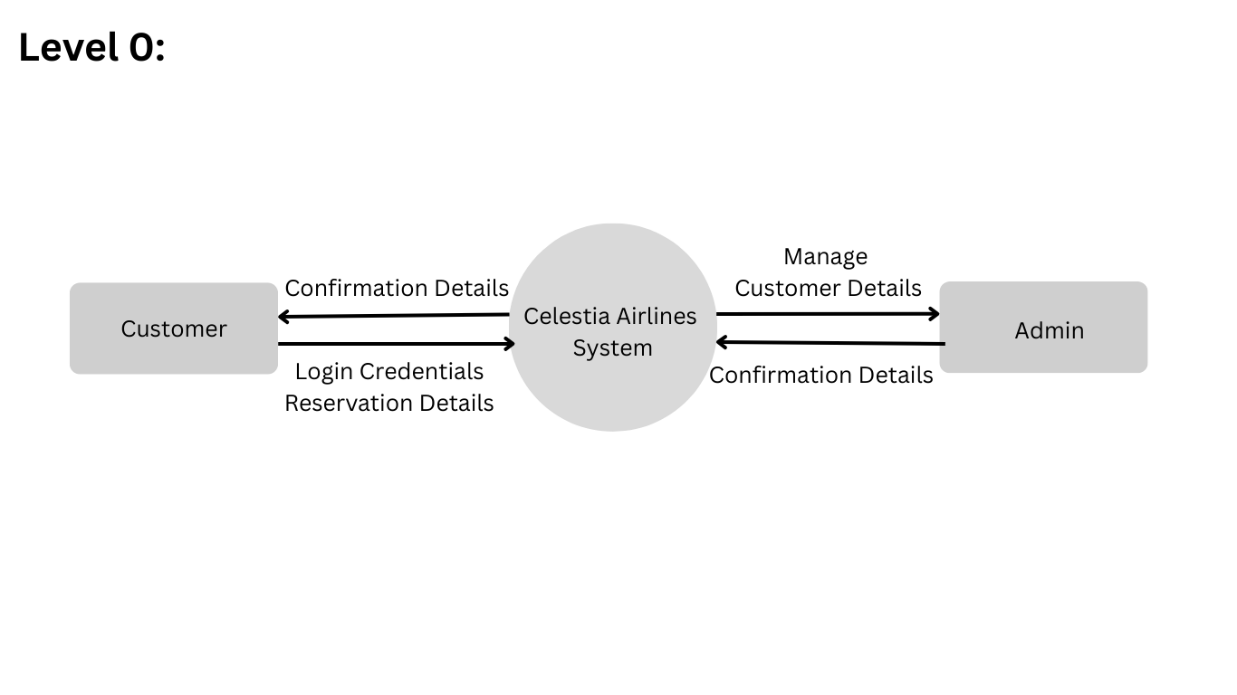
AI-generated content may be incorrect.

* Provides the MySQL server used to manage the relational database.
* phpMyAdmin allows visual control over tables, queries, and relationships.
* Local database connectivity used with mysql.connector in Python.

**📦 Third-Party Libraries**

|  |  |
| --- | --- |
| **Library** | **Description** |
| mysql.connector | For connecting Python scripts to the MySQL database securely. |
| requests | Used to fetch real-time weather data from the OpenWeatherMap API. |
| tkinter | Built-in GUI library in Python for creating windows, forms, and menus. |

**DFD:-**

****

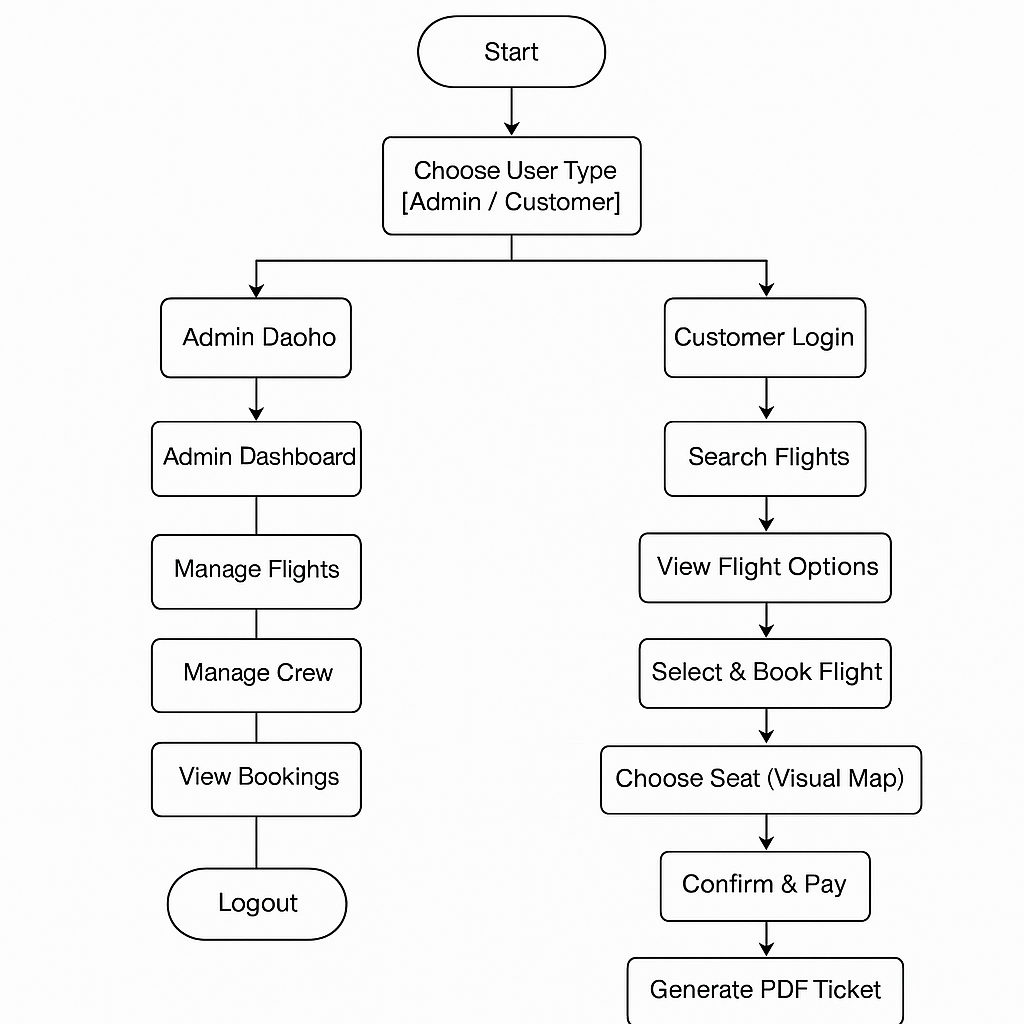
**A diagram of a company

AI-generated content may be incorrect.**

**A diagram of a company

AI-generated content may be incorrect.**

**Activity Diagram:-**

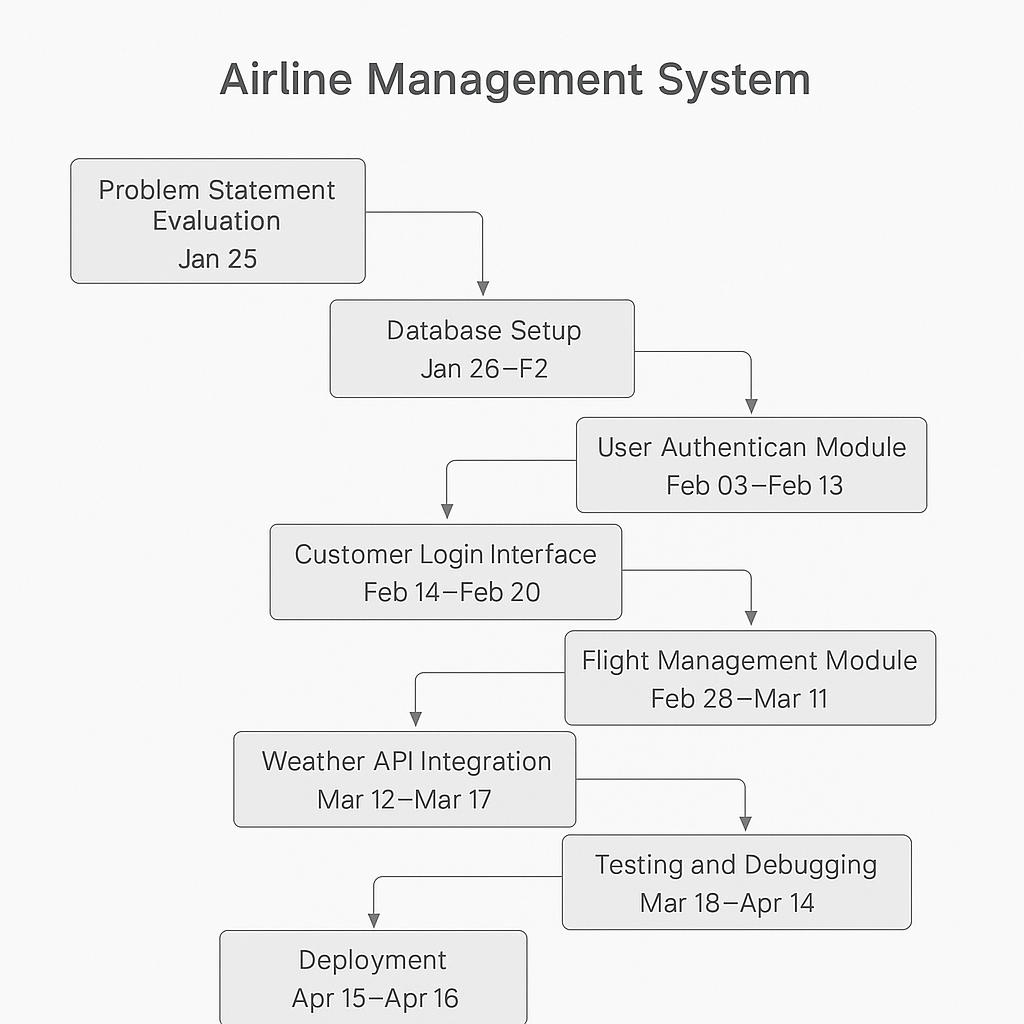
****

**Gantt Chart:-**

**A diagram of a flight schedule

AI-generated content may be incorrect.**

**Flow Chart:-**

****

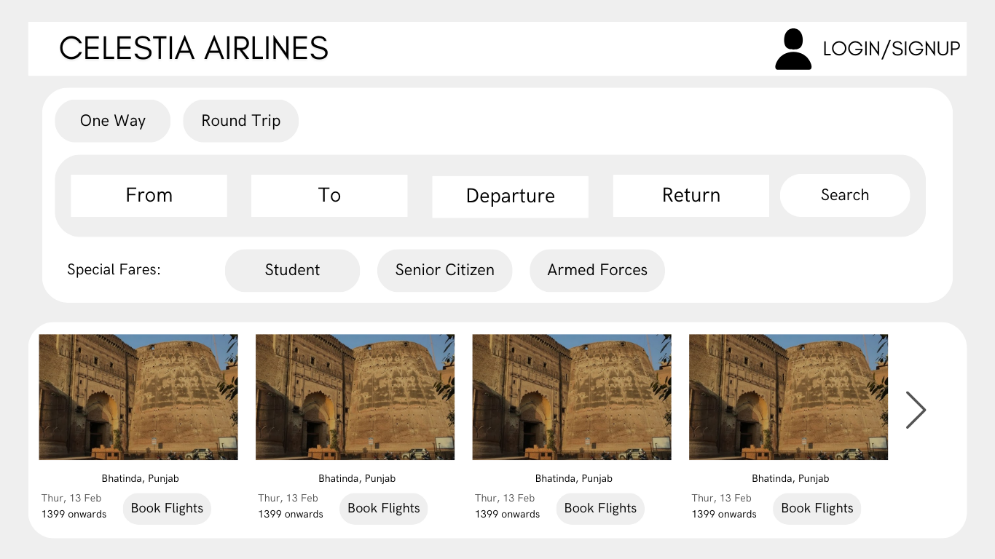
**ER-diagram of tables:-**

**· Modules/Functions Implementation Planning:**

requested:

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No.** | **Module Name** | **Description** | **Implementation Duration** |
| 1 | Problem Statement Evaluation | Understanding and defining the problem | 7 days from 23-01-25 to 30-01-25 |
| 2 | Database Setup | Designing and creating the database | 10 days from 31-01-25 to 09-02-25 |
| 3 | User Authentication Module | Handling user login/logout and security | 18 days from 10-02-25 to 27-02-25 |
| 4 | Customer Login Interface | UI for customer login and access | 11 days from 28-02-25 to 10-03-25 |
| 5 | Admin Login Interface | UI for admin login and management access | 7 days from 11-03-25 to 17-03-25 |
| 6 | Flight Management Module | Module for managing flights | 14 days from 18-03-25 to 31-03-25 |
| 7 | Weather API Integration | Integrating weather data for flights | 6 days from 01-04-25 to 06-04-25 |
| 8 | Testing and Debugging | Identifying and fixing bugs | 14 days from 07-04-25 to 20-04-25 |
| 9 | Deployment | Releasing the final system | 6 days from 21-04-25 to 26-04-25 |

**Implementation Phase 1:-**

****

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a flight

AI-generated content may be incorrect.**

**A login box with a box and a login box

AI-generated content may be incorrect.**

**A white rectangular object with red text

AI-generated content may be incorrect.**

**Implementation Phase 2**

**SOURCE CODE:**

**customer\_login.py**

**Output(snaps shots):-**

**Future Scope:**

**1.Online Payment Gateway Integration**

* Enable secure payments through UPI, cards, or net banking for real-time booking confirmation.

2. **Real-time Flight Tracking**

* Use APIs to show live flight status and delays within the app.

3. **Notification System**

* Integrate SMS/email alerts for booking confirmations, cancellations, or delays.

4. **Frequent Flyer Program**

* Allow users to accumulate points based on bookings and redeem them for discounts or upgrades.

5. **Multi-language Support**

* Expand accessibility by supporting regional and international languages.

6. **Admin Analytics Dashboard**

* Provide data visualizations on bookings, revenues, cancellations, etc., to help management make informed decisions.

7. **In-flight Services Booking**

* Allow passengers to pre-book meals, Wi-Fi, or seat upgrades.

8. **Dynamic Pricing Model**

* Implement an algorithm to adjust ticket prices based on demand, booking time, and seat availability.

9. **Mobile App Version**

* Develop a cross-platform mobile app for easier access and convenience.

**10. AI-Based Chatbot Support**

* Offer a virtual assistant to help users book flights, resolve queries, and manage bookings.

### **Conclusion**

This project really helped me strengthen my core Java concepts like **OOP, exception handling, GUI design, and modular programming**. I applied what I learned in class—like creating classes, using functions, and building user interfaces.

Beyond the syllabus, I explored **API integration**, **PDF generation**, and **real-time weather checks**, which made the project more advanced and realistic. Overall, it improved both my **technical skills** and **problem-solving** ability through a practical, hands-on approach.