SAFWA

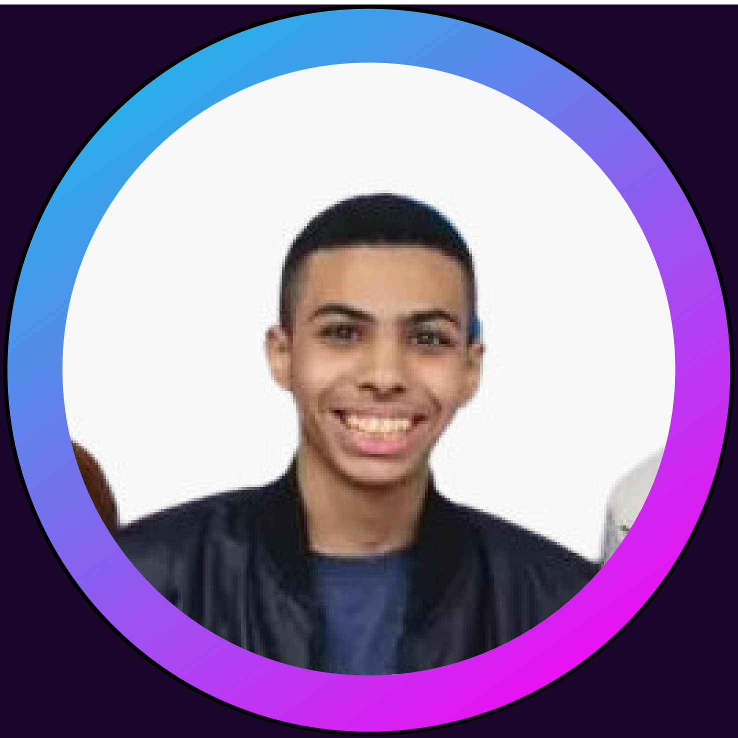
AIRPORT MANAGEMENT SYSTEM

5/5/2024

project airport

The Java airport project is a comprehensive software application designed to facilitate various operations within an airport environment. It provides opportunities for passengers to book flight tickets, pilots to manage their assigned flights, and managers to oversee and administer the system.

Our Team



Mohanad khaled

Project manager flight class and Main class

Lamice ossama

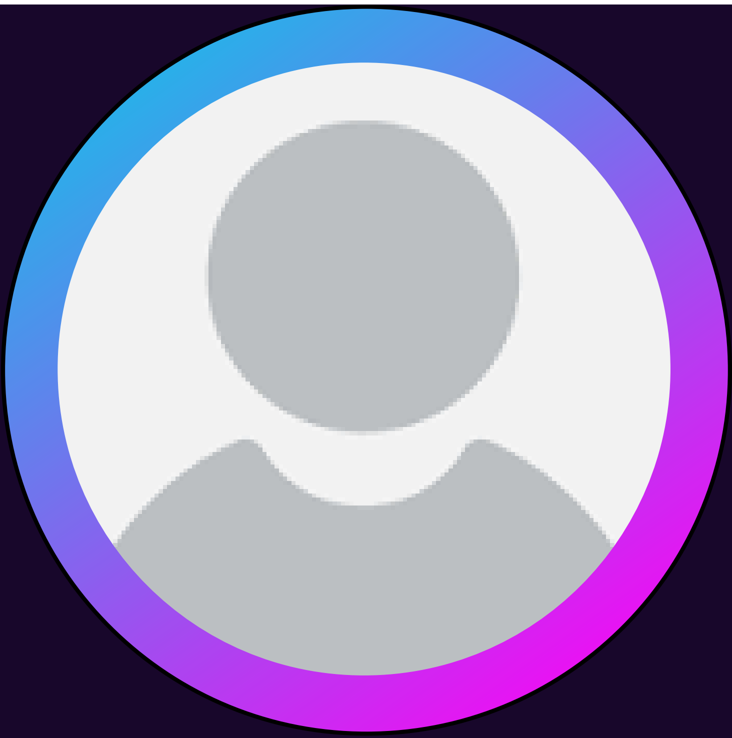
Project manager Manager class

Ahmed shacker

Project manager Main class and passenger class

Maryam islam

Project manager Main class



Sabrina hossam

Project manager pilot class

SAFWA

Classes

01 Passenger class

- Passenger Information: Stores the passenger's name.
- Booking History: Manages a list of Flight objects (class passmanger) representing the passenger's booked flights.
- Provides methods to: View the passenger's name.
- Access the complete list of booked flights.
- Book a new flight.
- View details of all booked flights including: flight number, source-destination, departure time, ticket price, and transit status.

Classes

02 Pilot class

- Pilot Information:
- Stores the pilot's name.
- Assigned Flights:
- Maintains a list of Flight objects (pilot class) representing flights assigned to the
- Provides methods to:View details of all assigned flights (source, destination, departure time, flight number).
- Assign a new flight to the pilot.

Classes

03 Manager class

- Manages: Pilots and Flights
- Manager Attributes:
- Name: Stores the manager's name.
- Pilots: List of Pilot objects representing all pilots in the system.
- Available Flights: Likely stores flights available for booking.
- All Flights: Stores all flights in the system (including available and potentially booked flights).
- Pilot Management:
- Add pilots to the system.
- Remove pilots from the system.
- View a list of all pilots with names and numbers.
- Flight Management:
- Create new flights with details (source, destination, price, transit).
- Set the list of all flights

Classes

04 Flight class

- Flight Detail:
- Stores flight number (automatically assigned), destination, departure time (date and time combined), transit status (true/false), ticket price, and origin.
- Constructors:
- Default constructor: Generates a flight number based on the number of created flights.
- Parameterized constructor: Initializes a flight with provided destination, transit status, ticket price, and origin.
- Other Methods:
- Provides getter and setter methods to access and potentially modify most flight attributes.
- The setDeparture_time method uses user input with validation to ensure a valid date (after today) and time format (HH:MM) for departure.

Objectives

1-Manage the Airport System:

- Passengers can view available flights, book flights, and see their booking history.
- Pilots can view their assigned flights and potentially be assigned new flights.
- The manager can add/remove pilots, add new flights, view information about pilots

2. Facilitate Flight Bookings

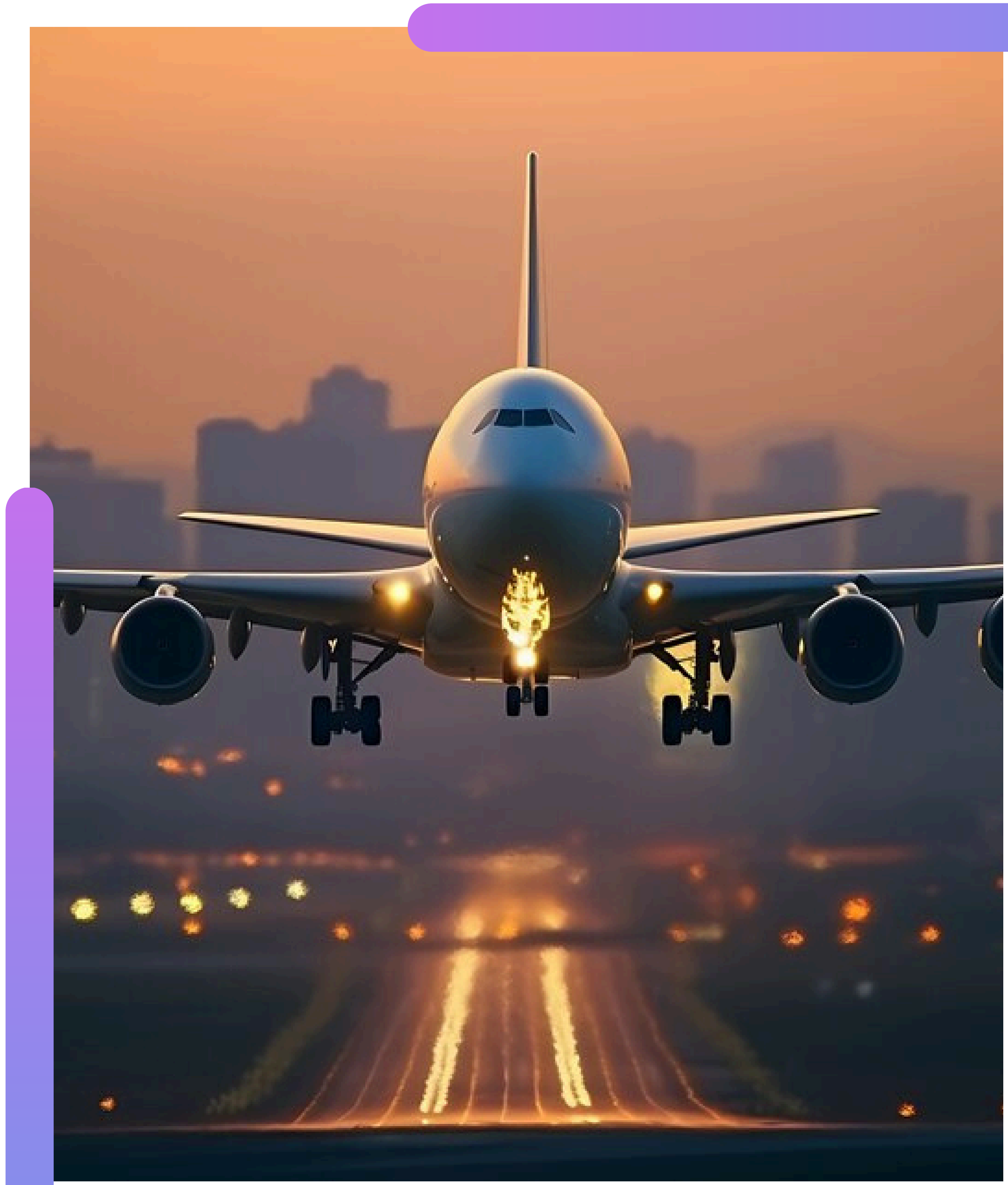
- Provides a menu option for passengers to "book a flight".
- Displays a list of available flights with details like source, destination, and departure time.
- Allows passengers to choose a flight from the available options
- Potentially handles setting the ticket price based on a user selection
- Adds the chosen flight to the passenger's booking history .

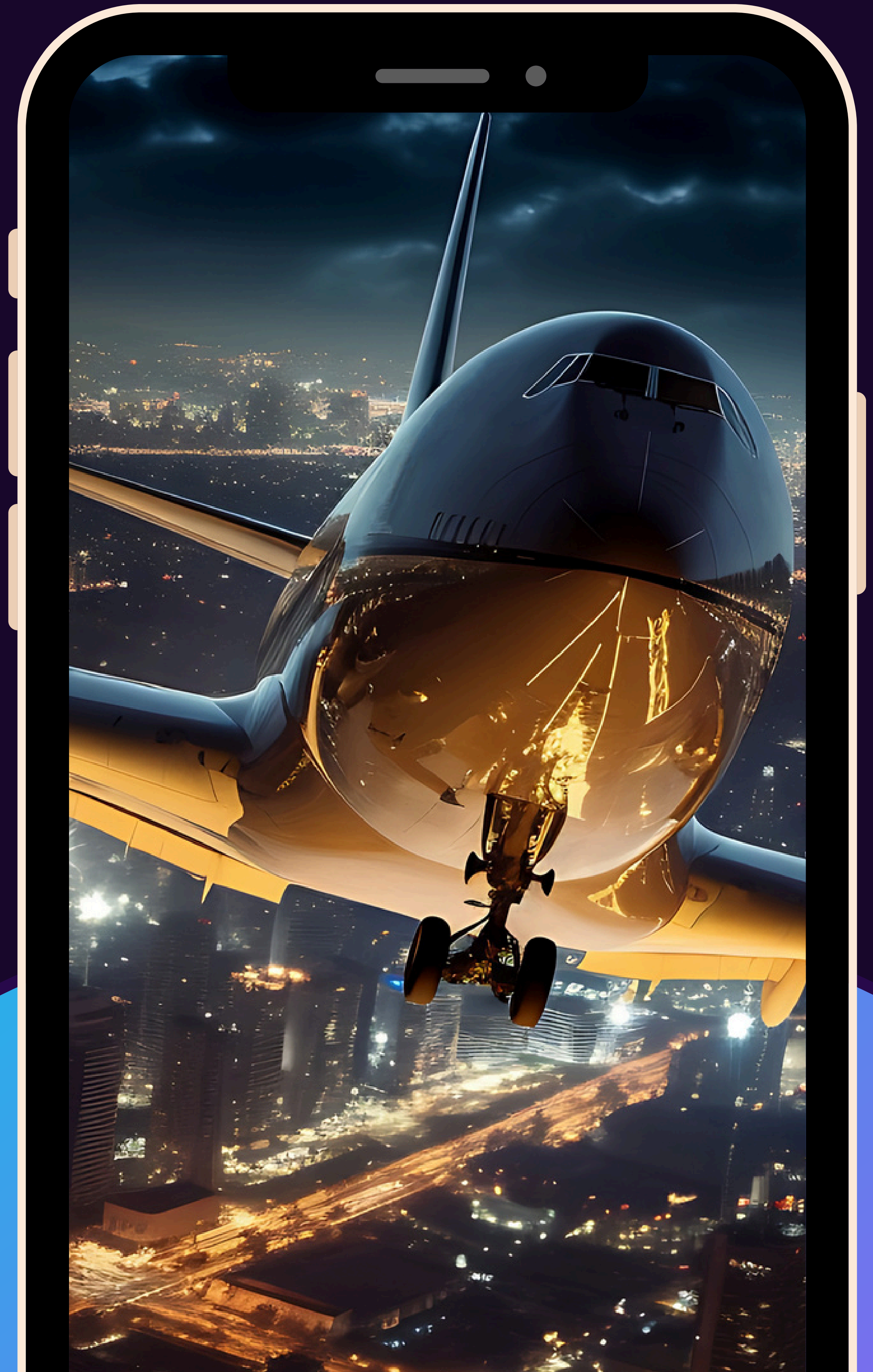
3. Provide System Overview and Information:

- System Insights: Offer various ways to get information about the overall system state.
- Potential Functionalities:Pilot Information: Allow viewing details about pilots in the system, such as a list of all pilots with names .
- Flight Information: Provide ways to view information about all flights or specific flights. Details might include source, destination, departure time, transit status, and potentially the assigned pilot.
- Booking Information: Enable viewing booking history for passengers . This allows passengers to see their booked flights.
- Manager Reports : The manager might have access to reports or summaries of the current system state. These reports could include details like the total number of pilots, flights, or passengers.

conclusion

- Enhances airline efficiency:Streamlines pilot and flight management.
- Lays the groundwork for flight bookings.
- Provides a central information hub.
- Potential for future improvements:Integrate functionalities for flight bookings and passenger management.
- Generate reports for better decision-making.





thank you