Airline Booking System with Dynamic Pricing

Student Name: Irene Charles Koner
Roll Number: 150096725146
Course: B.Tech CSE (2025–29)
Cohort : Jeff Bezos

Introduction to the Case Study

With the rapid growth of the travel and aviation industry, airlines increasingly rely on digital systems to manage flight reservations, seat availability, and ticket pricing efficiently. Modern airline booking platforms help passengers select flights, choose seats, and obtain fair prices while allowing airlines to optimize revenue through demand-based pricing strategies. However, many beginners find it challenging to understand how such large-scale reservation systems are designed and implemented from a programming perspective.

This case study focuses on developing an **Airline Booking System with Dynamic Pricing** using Python. The project demonstrates how core programming concepts can be applied to solve a real-world problem by creating a menu-driven application that manages flight data, performs real-time price calculations, handles bookings and cancellations, and analyzes revenue trends using data visualization and file handling techniques.

Problem Statement / Case Background (Abstract)

The objective of this case study is to design and implement a **Python-based airline booking system** that allows users to:

- View available flights and flight details
- Select seats and manage seat availability
- Calculate ticket prices using dynamic pricing techniques
- Book and cancel airline tickets
- Analyze revenue and booking data

The system is designed as a menu-driven application and uses **CSV files for persistent data storage**.
It emphasizes modularity, clarity, and usability, making it suitable for beginners while demonstrating important programming concepts such as **object-oriented design, dynamic pricing algorithms, decorators, exception handling, and data analysis with visualization**.

Case Study Design

The problem addressed in this case study is the need for a simple, console-based airline booking system that demonstrates real-world Python usage. The design goals of the system are:

- To manage static flight details such as routes, schedules, and aircraft information
- To handle dynamic data such as seat availability, bookings, and ticket prices in real time
- To prevent double booking of seats
- To store flight, booking, and pricing data persistently using CSV files
- To provide meaningful analytical insights through revenue reports and pricing analysis

The system follows a **menu-driven design**, allowing users to interact with different functionalities such as viewing flight details, checking seat layouts, viewing real-time price breakdowns, booking and canceling tickets, and generating revenue reports.

Methods & Algorithms

The following methods, algorithms, and technologies were applied in this case study:

Programming Concepts

- **Object-Oriented Programming (OOP):**
  Used to model real-world airline entities such as flights, seats, bookings, and pricing logic through classes like Flight, Seat, Booking, and PricingEngine.
- **Decorators:**
  Implemented to log booking transactions and ensure better traceability of user

actions within the system.

- **Lambda Functions:**
  Used for compact calculations in the dynamic pricing logic, such as applying demand-based surcharges and time-based pricing adjustments.

- **Exception Handling:**
  Applied to handle invalid user inputs, unavailable seats, and incorrect flight selections gracefully.

Data Handling

- **CSV File Handling:**
  Used for persistent storage of flight details, booking records, and pricing history to ensure data is retained between program executions.

- **Pandas:**
  Applied for analyzing booking data, calculating revenue metrics, and generating summary reports for revenue analysis.

- **NumPy:**
  Used to generate and display structured seat layouts, visually representing seat availability and booking status.

Algorithms

- **Dynamic Pricing Algorithm:**
  Ticket prices are calculated based on base fare, seat class, simulated demand levels, and remaining time before departure.

- **Seat Availability Algorithm:**
  Ensures real-time tracking of booked and available seats while preventing double bookings.

- **Revenue Calculation Algorithm:**
  Aggregates booking prices to compute total revenue, average fare, and load factor per flight.

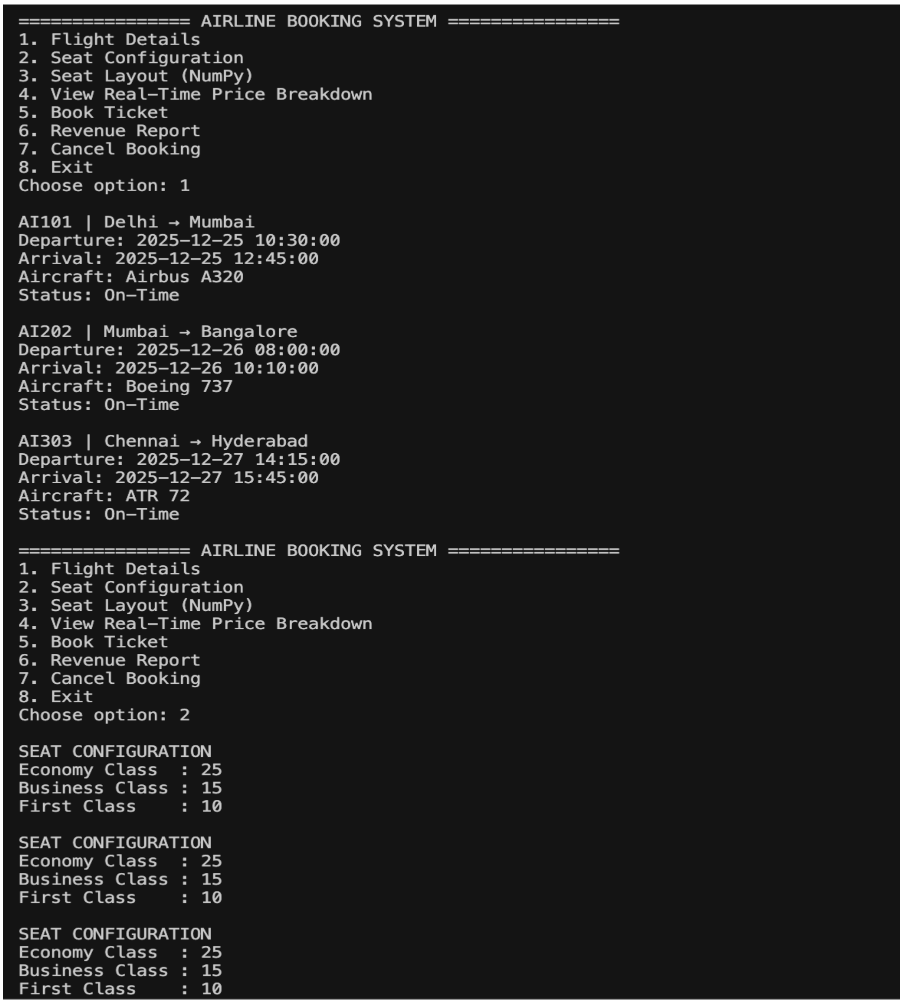
Case Study Implementation Details and

The implementation consists of four main Python files:

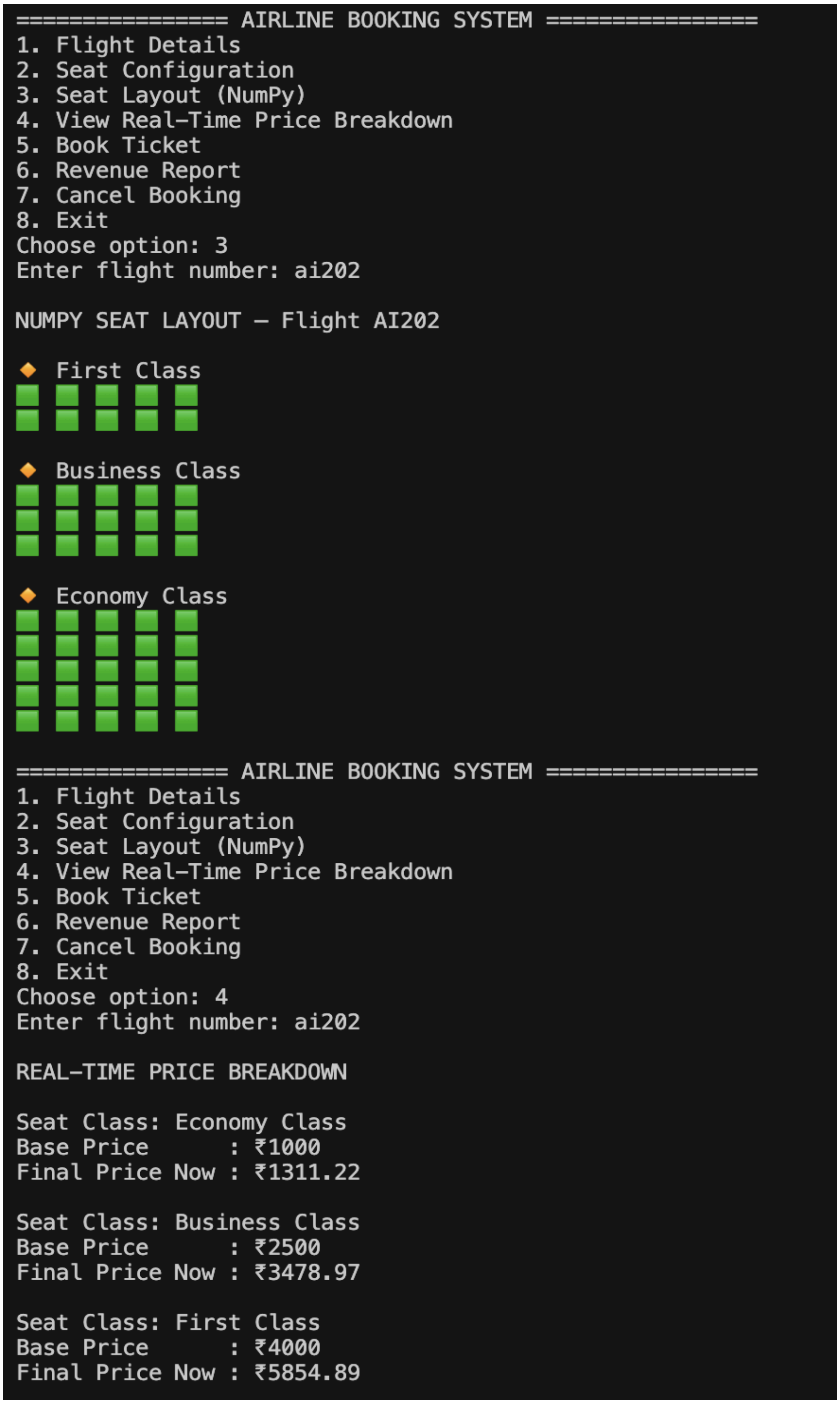
- **booking_main.py** – Handles user interaction, menu logic, and booking flow
- **flight_class.py** – Contains the Flight and Seat classes along with seat layout logic
- **pricing_engine.py** – Implements dynamic pricing and real-time price breakdowns
- **revenue_analyzer.py** – Generates revenue reports and visualizations

Key Implementation Details

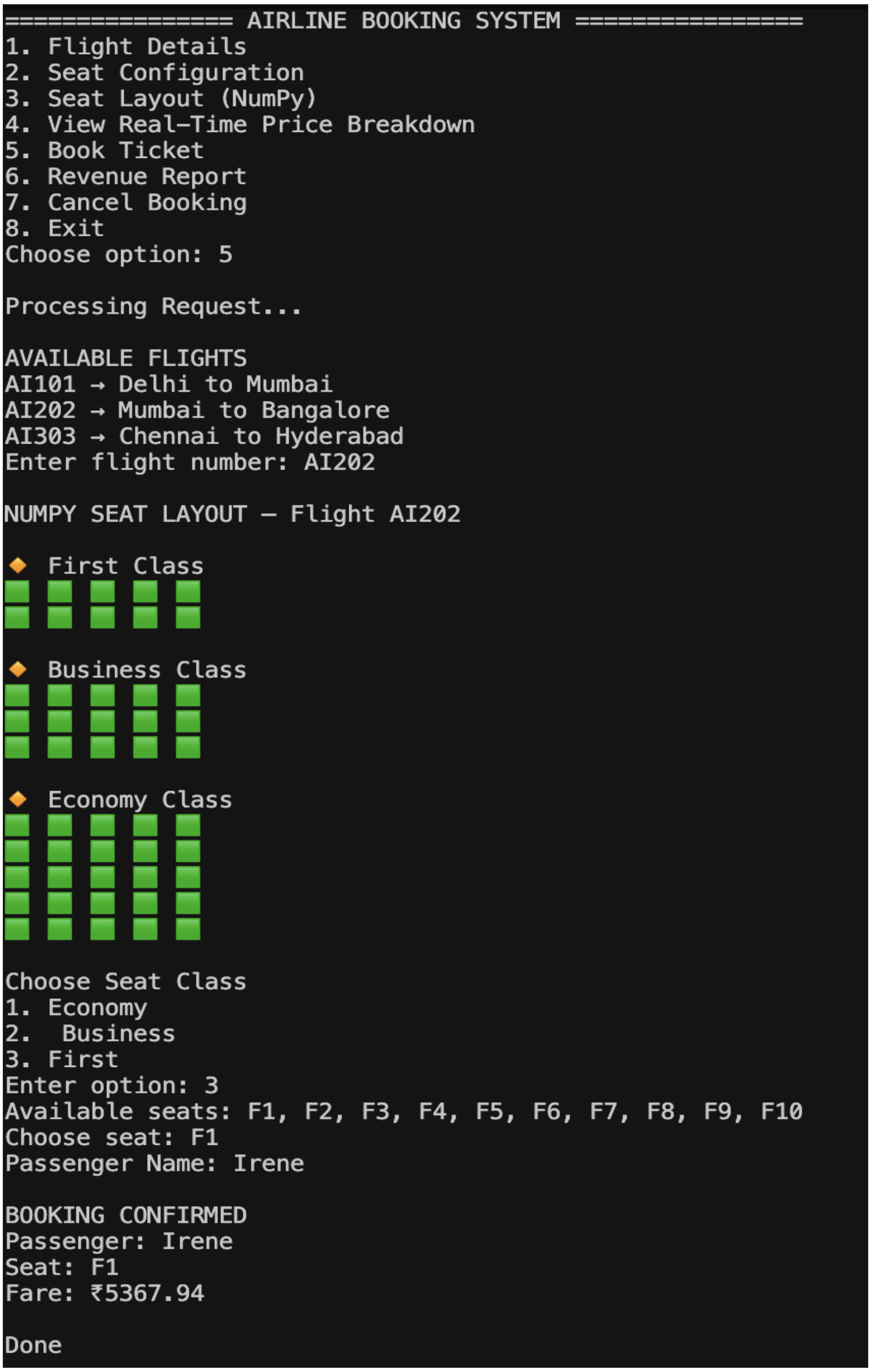
- Flight details (flight number, route, timings, aircraft type) are predefined and loaded at program start.
- Each flight maintains its own seat configuration with seat class and booking status.
- Seat layouts are displayed using NumPy to visually represent availability.
- Ticket prices are calculated dynamically based on seat class and demand factors.
- Real-time price breakdowns are shown before booking confirmation.
- Bookings are stored in memory and optionally exported to CSV files.
- Revenue data is analyzed.



```
================ AIRLINE BOOKING SYSTEM ================
1. Flight Details
2. Seat Configuration
3. Seat Layout (NumPy)
4. View Real-Time Price Breakdown
5. Book Ticket
6. Revenue Report
7. Cancel Booking
8. Exit
Choose option: 1

AI101 | Delhi → Mumbai
Departure: 2025-12-25 10:30:00
Arrival: 2025-12-25 12:45:00
Aircraft: Airbus A320
Status: On-Time

AI202 | Mumbai → Bangalore
Departure: 2025-12-26 08:00:00
Arrival: 2025-12-26 10:10:00
Aircraft: Boeing 737
Status: On-Time

AI303 | Chennai → Hyderabad
Departure: 2025-12-27 14:15:00
Arrival: 2025-12-27 15:45:00
Aircraft: ATR 72
Status: On-Time

================ AIRLINE BOOKING SYSTEM ================
1. Flight Details
2. Seat Configuration
3. Seat Layout (NumPy)
4. View Real-Time Price Breakdown
5. Book Ticket
6. Revenue Report
7. Cancel Booking
8. Exit
Choose option: 2

SEAT CONFIGURATION
Economy Class  : 25
Business Class : 15
First Class    : 10

SEAT CONFIGURATION
Economy Class  : 25
Business Class : 15
First Class    : 10

SEAT CONFIGURATION
Economy Class  : 25
Business Class : 15
First Class    : 10
```



```
================ AIRLINE BOOKING SYSTEM ================
1. Flight Details
2. Seat Configuration
3. Seat Layout (NumPy)
4. View Real-Time Price Breakdown
5. Book Ticket
6. Revenue Report
7. Cancel Booking
8. Exit
Choose option: 3
Enter flight number: ai202

NUMPY SEAT LAYOUT – Flight AI202

🔶 First Class
🟩 🟩 🟩 🟩 🟩
🟩 🟩 🟩 🟩 🟩

🔶 Business Class
🟩 🟩 🟩 🟩 🟩
🟩 🟩 🟩 🟩 🟩
🟩 🟩 🟩 🟩 🟩

🔶 Economy Class
🟩 🟩 🟩 🟩 🟩
🟩 🟩 🟩 🟩 🟩
🟩 🟩 🟩 🟩 🟩
🟩 🟩 🟩 🟩 🟩
🟩 🟩 🟩 🟩 🟩

================ AIRLINE BOOKING SYSTEM ================
1. Flight Details
2. Seat Configuration
3. Seat Layout (NumPy)
4. View Real-Time Price Breakdown
5. Book Ticket
6. Revenue Report
7. Cancel Booking
8. Exit
Choose option: 4
Enter flight number: ai202

REAL-TIME PRICE BREAKDOWN

Seat Class: Economy Class
Base Price      : ₹1000
Final Price Now : ₹1311.22

Seat Class: Business Class
Base Price      : ₹2500
Final Price Now : ₹3478.97

Seat Class: First Class
Base Price      : ₹4000
Final Price Now : ₹5854.89
```



```
================ AIRLINE BOOKING SYSTEM ================
1. Flight Details
2. Seat Configuration
3. Seat Layout (NumPy)
4. View Real-Time Price Breakdown
5. Book Ticket
6. Revenue Report
7. Cancel Booking
8. Exit
Choose option: 5

Processing Request...

AVAILABLE FLIGHTS
AI101 → Delhi to Mumbai
AI202 → Mumbai to Bangalore
AI303 → Chennai to Hyderabad
Enter flight number: AI202

NUMPY SEAT LAYOUT — Flight AI202

🔸 First Class
🟩 🟩 🟩 🟩 🟩
🟩 🟩 🟩 🟩 🟩

🔸 Business Class
🟩 🟩 🟩 🟩 🟩
🟩 🟩 🟩 🟩 🟩
🟩 🟩 🟩 🟩 🟩

🔸 Economy Class
🟩 🟩 🟩 🟩 🟩
🟩 🟩 🟩 🟩 🟩
🟩 🟩 🟩 🟩 🟩
🟩 🟩 🟩 🟩 🟩
🟩 🟩 🟩 🟩 🟩

Choose Seat Class
1. Economy
2.  Business
3. First
Enter option: 3
Available seats: F1, F2, F3, F4, F5, F6, F7, F8, F9, F10
Choose seat: F1
Passenger Name: Irene

BOOKING CONFIRMED
Passenger: Irene
Seat: F1
Fare: ₹5367.94

Done
```

Case Study Results

The Airline Booking System successfully meets all the objectives defined in the problem statement. The application enables users to view flight details, select seats, calculate ticket prices dynamically, and manage bookings while demonstrating key Python programming concepts.

Results

- Accurate display of flight information and seat availability
- Dynamic seat-based pricing with real-time price breakdowns
- Visual representation of seat layouts using NumPy
- Successful booking and cancellation of tickets with proper validation
- Generation of revenue reports and analytical charts
- Structured storage of booking-related data for analysis

Conclusion

This case study demonstrates how Python can be effectively used to develop a real-world airline reservation system using fundamental programming principles. The project strengthened understanding of object-oriented programming, error handling, decorators, and numerical computing. The system is modular, easy to extend, and well-suited for academic evaluation, making it an ideal case study for beginner to intermediate-level computer science students.
