Database System Project (Team 3)

Airline Ticketing System

1. Introduction

Project Description:

The goal of this project is to design and implement a comprehensive and efficient Airline Booking Database System. This system will manage all aspects of airline operations related to customer bookings, flights, tickets, payments, aircraft, and associated entities. The database will serve as the backbone for an airline or travel agency's booking process, ensuring smooth transactions, efficient data management, and seamless user experience.

Motivation for Selecting the Project

The primary motivation for selecting this project is the complex yet essential nature of airline operations and the impact of an efficient booking system on overall customer experience and business operations. An airline booking system must handle large volumes of data, perform real-time transactions, and ensure the integrity of critical information, such as customer details, flight schedules, and payments. By designing this database, I aim to explore solutions for managing complex relationships between entities, ensuring data security, and optimizing the performance of high-demand operations.

How is this database helpful?

The Airline Booking Database System allows customers to book, view, and modify flight reservations, make payments, and manage their tickets and baggage. For airline staff and travel agencies, the system manages flight schedules, customer bookings, passenger information, payments, and reporting. It supports real-time updates, secure payment processing, seat assignments, and detailed reporting to streamline airline operations and enhance the customer experience.

2. Requirements Analysis

Functional Requirements:

1. Search for available flights based on departure and arrival locations, dates, and airlines.
2. Process flight bookings for customers and generate tickets for all passengers.
3. Modify flight schedules, including departure and arrival times.
4. Record payment transactions associated with bookings.

5. Add passengers to bookings and assign seats.
6. Manage and track baggage for each passenger.
7. View and manage booking details, including flight, payment, and ticket information.
8. Update customer information, including contact details.
9. Cancel bookings and handle rescheduling.

Non-functional Requirements:

1. Scalability: Handle thousands of concurrent bookings and queries.
2. Data Security: Encrypt sensitive information and ensure role-based access control.
3. Performance: Ensure real-time response for critical operations such as booking and payment processing.
4. Reliability: Achieve 99.9% system uptime with robust failover mechanisms.

3. ER Model Design

Entities and Attributes

1) Customer
    a) Attributes:
        i) Customer_ID (Primary Key)
        ii) Customer_Name
        iii) DoB (Date of Birth)
        iv) Email_Address
        v) Contact_Number
2) Flight
    a) Attributes:
        i) Flight_Num (Primary Key)
3) Airline
    a) Attributes:
        i) Airline_Code (Primary Key)
        ii) Legal_Name
        iii) ICAO_Code
        iv) IATA_Code
        v) Region
        vi) Country
        vii) Website
4) Airport
    a) Attributes:
        i) Airport_Code (Primary Key)
        ii) Airport_Name
        iii) Country

iv) City

5) Booking
    a) Attributes:
        i) Booking_Number (Primary Key)
        ii) Number_of_Passengers
6) Ticket
    a) Attributes:
        i) Ticket_Number (Primary Key)
7) Payment
    a) Attributes:
        i) Payment_Num (Primary Key)
8) Aircraft
    a) Attributes:
        i) Aircraft_ID (Primary Key)
        ii) Model
        iii) Seating_Capacity
9) Passenger
    a) Attributes:
        i) Passenger_Full_Name
        ii) Passport_Number (Primary Key)
        iii) Passport_Issuance_Country
        iv) DoB (Date of Birth)
10) Seat
    a) Attributes:
        i) Seat_Num (Primary Key)
11) Baggage
    a) Attributes:
        i) Baggage_ID (Primary Key)
12) Type_of_Trip
    a) Attributes:
        i) ToTrip_ID (Primary Key)
        ii) ToTrip_Name
13) Class
    a) Attributes:
        i) Class_ID (Primary Key)
        ii) Class_Name
14) Invoice
    a) Attributes:
        i) Invoice_Num (Primary Key)
        ii) Total_Amount

Relationship among entities:

1) Customer-Booking: A customer can create multiple bookings, but each booking is linked to only one customer.
2) Booking-Passenger: A booking includes one or more passengers, and a passenger can be part of multiple bookings.
3) Passenger-Ticket: A passenger can hold multiple tickets, and each ticket is assigned to a single passenger.
4) Booking-Invoice: Every booking is linked to exactly one invoice.
5) Invoice-Payment: Each invoice corresponds to exactly one payment.
6) Flight-Airline: An airline operates multiple flights, and each flight is associated with one airline.
7) Aircraft-Flight: Flight is an aircraft and aircraft has many flight

5. Relational Model

Relational Schema, Cardinality and Normalization

1) Customer:
    a. Customer(Customer_ID (Primary Key), First_Name, Last_Name,DoB, Email_Address, Contact_Number,gender)
    b. Normalization: No partial or transitive dependencies. This relation in 3NF
2) Registered Customer:
    a. Registered_Customer(Customer_ID (Primary Key, FK), Joined_Date)
    b. Normalization: No partial or transitive dependencies. This relation in 3NF
3) Guest Customer
    a. Guest_Customer(Customer_ID (Primary Key, FK), passport_no)
    b. Normalization: No partial or transitive dependencies. This relation in 3NF
4) Flight
    a. Flight(Flight_Num (Primary Key), Aircraft_ID(FK), Airline_Code(FK), depature_Airport_code(FK),Dep_time,Arrival_Air_port_code(FK), Arriv_Time)
    b. Normalization: No partial or transitive dependencies. This relation in 3NF
5) Airline
    a. Airline(Airline_Code (Primary Key), Legal_Name, ICAO_Code, IATA_Code, Region, Country, Website)
    b. Normalization: No partial or transitive dependencies. This relation in 3NF
6) Airport
    a. Airport(Airport_Code (Primary Key), Airport_Name, Country, City, destination_image)
    b. Normalization: No partial or transitive dependencies. This relation in 3NF
7) Booking
    a. Booking(Booking_Number (Primary Key), Booking_Date, Status, Number_of_Passenger , Customer_ID(FK))
    b. Normalization: No partial or transitive dependencies. This relation in 3NF
8) Ticket
    a. Ticket(Ticket_Number (Primary Key),flight_num(FK),Seat_Num(FK), Booking_Number(FK), Passport_Number(FK), Trip_Type_ID(FK), Baggage_ID(FK))

b. Normalization: No partial or transitive dependencies. This relation in 3NF

9) Payment

a. Payment(Payment_Num (Primary Key),payment_method, payment_satus, payment_date)
b. Normalization: No partial or transitive dependencies. This relation in 3NF

10) Aircraft
    a. Aircraft(Aircraft_ID (Primary Key), Model, Seating_Capacity, manufacturer, flight_range)
    b. Normalization: No partial or transitive dependencies. This relation in 3NF
11) Passenger
    a. Passenger(Passenger_Full_Name, Passport_Number (Primary Key), Id_proof, DoB)
    b. Normalization: No partial or transitive dependencies. This relation in 3NF
12) Seat
    a. Seat(Seat_Num (Primary Key),is_avilable)
    b. Normalization: No partial or transitive dependencies. This relation in 3NF
13) Baggage
    a. Baggage(Baggage_ID (Primary Key), weight, type, dimension)
    b. Normalization: No partial or transitive dependencies. This relation in 3NF
14) Type_of_Trip
    a. Type_of_Trip(ToTrip_ID (Primary Key), ToTrip_Name)
    b. Normalization: No partial or transitive dependencies. This relation in 3NF
15) Class
    a. Class(Class_ID (Primary Key), Class_Name)
    b. Normalization: No partial or transitive dependencies. This relation in 3NF
16) Invoice
    a. Invoice(Invoice_Num (Primary Key), Total_Amount)
    b. Normalization: No partial or transitive dependencies. This relation in 3NF
17) Customer_Review
    a. Customer_Review((Customer_ID (FK),review_id)(Primary Key), review)
    b. Normalization: No partial or transitive dependencies. This relation in 3NF
18) Departure_Flight_Day
    a. Departure_Flight_Day((Flight_Num(FK),Day)(Primary Key))
    b. Normalization: No partial or transitive dependencies. This relation in 3NF
19) Arrival_Flight_Day
    a. Arrival_Flight_Day((Flight_Num(FK),Day)(Primary Key))
    b. Normalization: No partial or transitive dependencies. This relation in 3NF
20) Flight_Seat_Price
    a. Flight_Seat_Price((Flight_num(FK),Seat_Num(FK),Class_ID(FK))(Primary key),price)
    b. Normalization: No partial or transitive dependencies. This relation in 3NF

6. System Design

Security Design

To ensure the security of our online air ticketing system, we have implemented the following measures:

1) Password Hashing

a. All passwords are securely stored using **bcrypt** hashing.

   b. This ensures that passwords are protected against brute force and rainbow table attacks.
   c. Password recovery is handled using secure, token-based reset links, without storing plaintext passwords.

2) **Access Control**
   a. We use **Role-Based Access Control (RBAC)** to restrict access based on user roles:
      i. **Admin**: Full system access for management tasks.
      ii. **Customers**: Access only to personal bookings and payment history.
   b. Each role has the minimum permissions necessary to perform its tasks, ensuring data security.

3) **Encryption**
   a. **Data Transmission**: All data exchanged between the client and server is encrypted using **TLS** for secure communication.
   b. **Data Storage**: Sensitive information, such as payment details, is encrypted using **AES-256**.
   c. Payment processing is handled through **PCI DSS-compliant** payment gateways, ensuring secure transactions.

4) **Additional Measures**
   a. **Monitoring**: Logs of important events (e.g., login attempts, password changes) are maintained and monitored.
   b. **Input Validation**: User inputs are validated to prevent attacks such as SQL injection and XSS.
   c. **Backup and Recovery**: Encrypted backups are maintained, and recovery processes are tested regularly.

7. Implementation

Technology Stack:

The system is built using the following technologies:

1) **Database Management System**:
   a. PostgreSQL with PL/pgSQL extensions for efficient database operations and custom procedures.
2) **Back-end Framework**:
   a. Flask, a lightweight and flexible framework, is used to handle server-side logic and API development.
3) **Frontend Framework**:
   a. The user interface is developed using **HTML**, **CSS**, and **JavaScript** for a responsive and interactive experience.
4) **Security Tools**:

a. **bcrypt**: For secure password hashing.
b. **SSL/TLS**: To ensure secure data transmission between the client and server.

Key Functions and Triggers:

1) **Functions**
   a. **get_arrival**:
      Calculates the arrival time for a flight based on the departure time and route duration.
   b. **get_seat_price**:
      Determines the seat price by considering the route and class of the seat (e.g., economy, business).
2) **Triggers**
   a. **update_customer_bookings**:
      Automatically increments the booking count for registered customers when a new booking is confirmed.
   b. **insert_seats_for_new_model**:
      Automatically inserts seat configurations into the database when a new aircraft model is added.

8. Challenges and resolution

a. **Challenges:**
   i. **Handling Complex Relationships**:
      Managing dependencies between flights, bookings, payments, and customers presented challenges in maintaining consistent data flow and operations.
   ii. **Ensuring Data Integrity**:
      Maintaining referential integrity across multiple tables required careful implementation of foreign keys and constraints.
   iii. **Optimizing Performance**:
      Ensuring fast query responses for real-time operations, such as booking and payment processing, was critical to user experience.

b. **Resolutions:**
   i. **Schema Design:**
      - **Designed a normalized database schema with clearly defined relationships and foreign key constraints to handle complex dependencies.**
   ii. **Indexing:**
      - **Added indexes on frequently queried columns, such as email (for user lookups) and schedule_id (for flight operations), to improve query performance.**

iii. **Transactional Queries:**

- **Used database transactions to ensure atomicity in critical operations like bookings and payments, preventing partial updates in case of errors.**

c. **Future Enhancements**

i. **Integration with Mobile Apps**:

- Develop and expose **API endpoints** to support seamless integration with mobile applications for enhanced customer accessibility.

ii. **Machine Learning Models**:

- Implement demand prediction algorithms to enable dynamic flight pricing, improving revenue generation and seat optimization.

9. Workload

Name	Task done
Amirthavarshani Mahadevan – amahadevan2@student.sgu.edu	• ER Diagram design • Relational Schema • Database design • Presentation slide
Godwin Okwara - gokwara1@student.gsu.edu	• Backend database design • Front end design • Relational model design • Execution of functional queries in the front-end

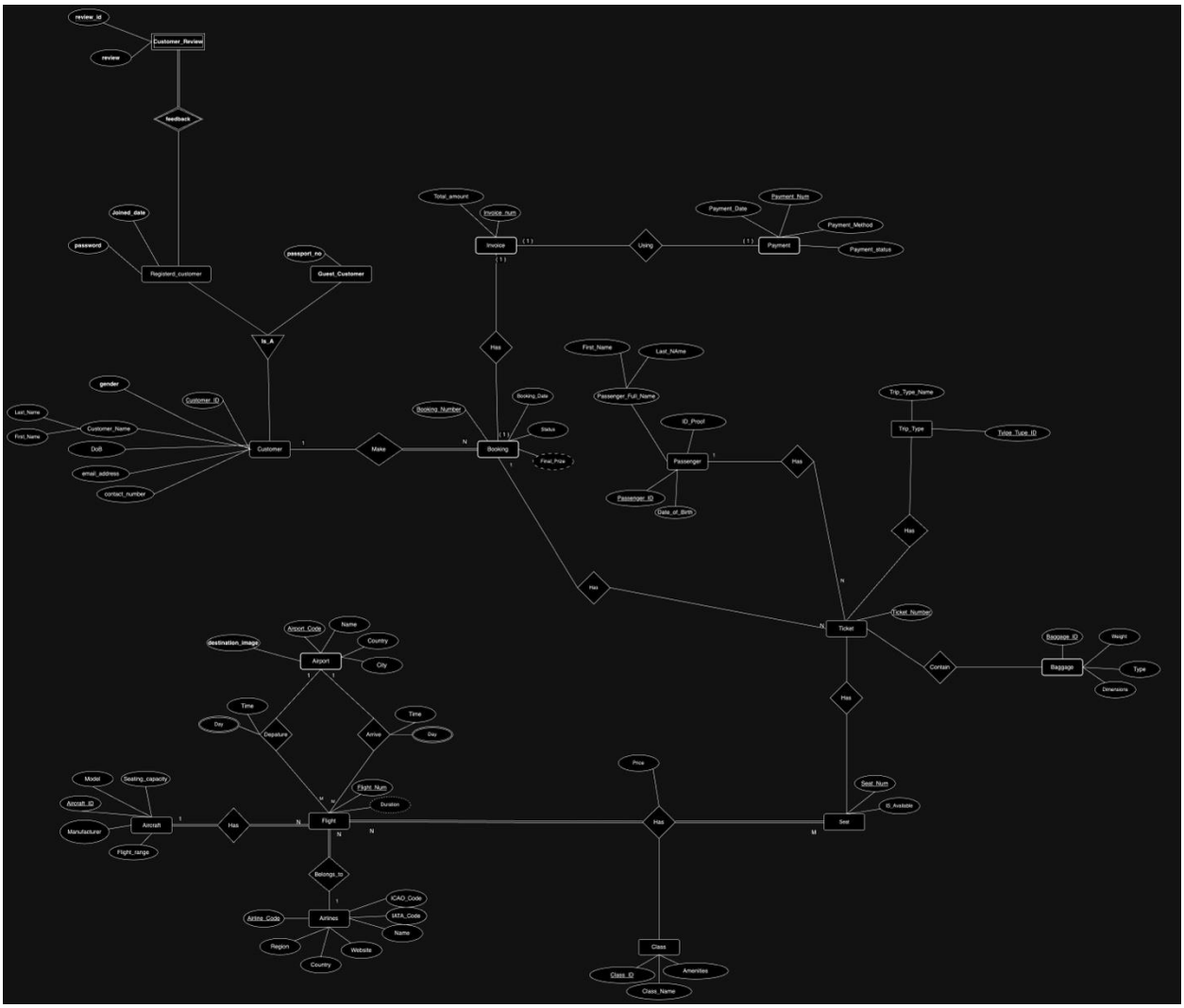
Both of us worked together in the proposal, project report

10. Access Links

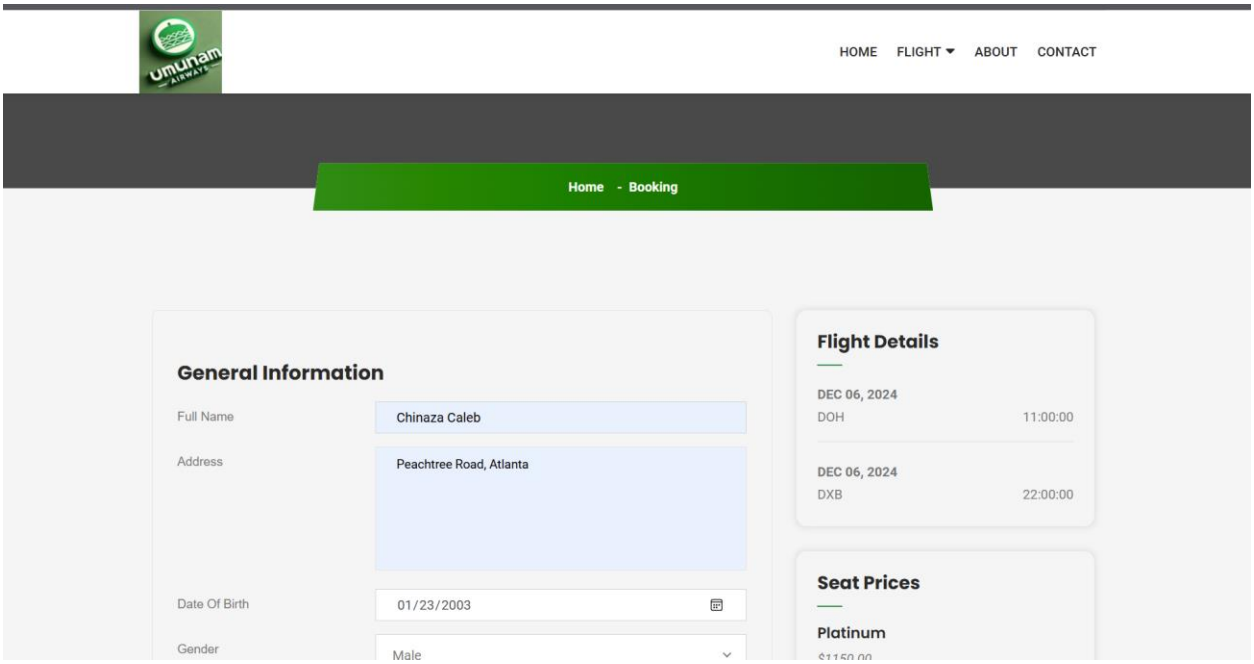
1) SQL folder: https://drive.google.com/drive/folders/1VeTAQ_VqBeuU3ANZNZN3vWaXPRtvWY76?usp=sharing
2) Code folder: https://drive.google.com/drive/folders/1khHfntkyBqMUSjrJDo2zKvc85Xf04JPn?usp=sharing
3) Demo video: https://drive.google.com/drive/folders/1HhNSUcn6Y510qfQRk_yTeNvT1s31SFZj?usp=sharing

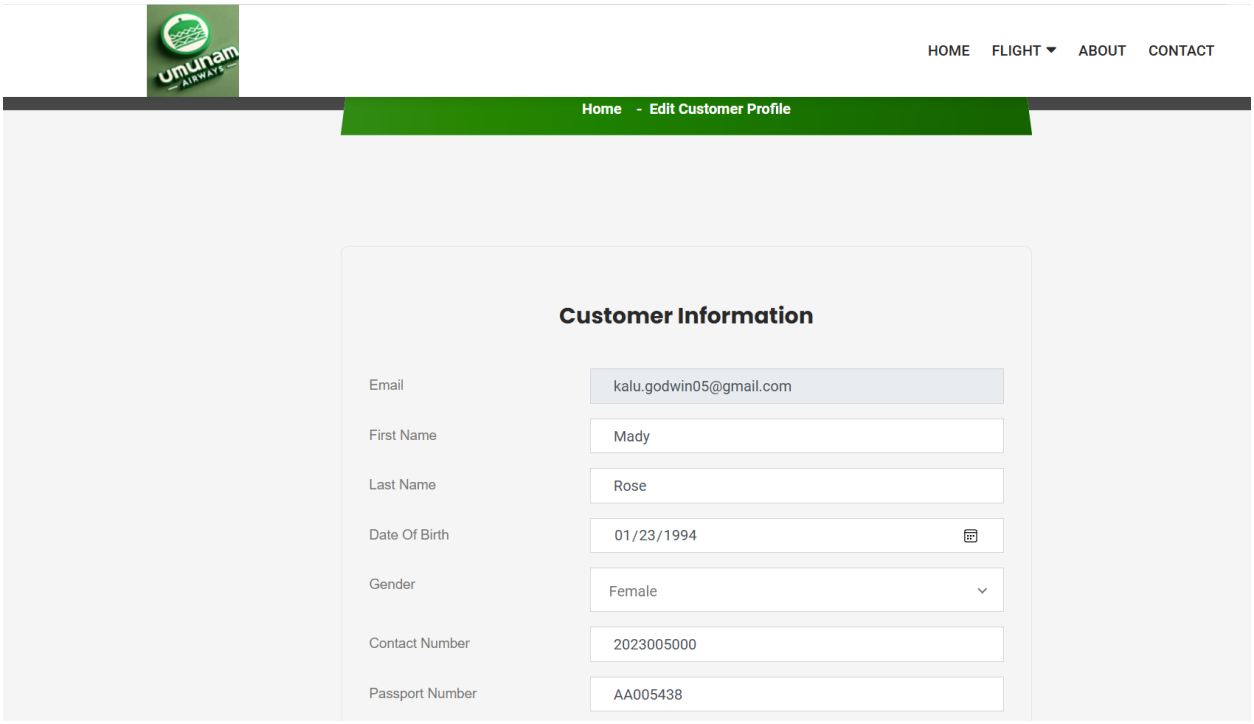
12. Appendix

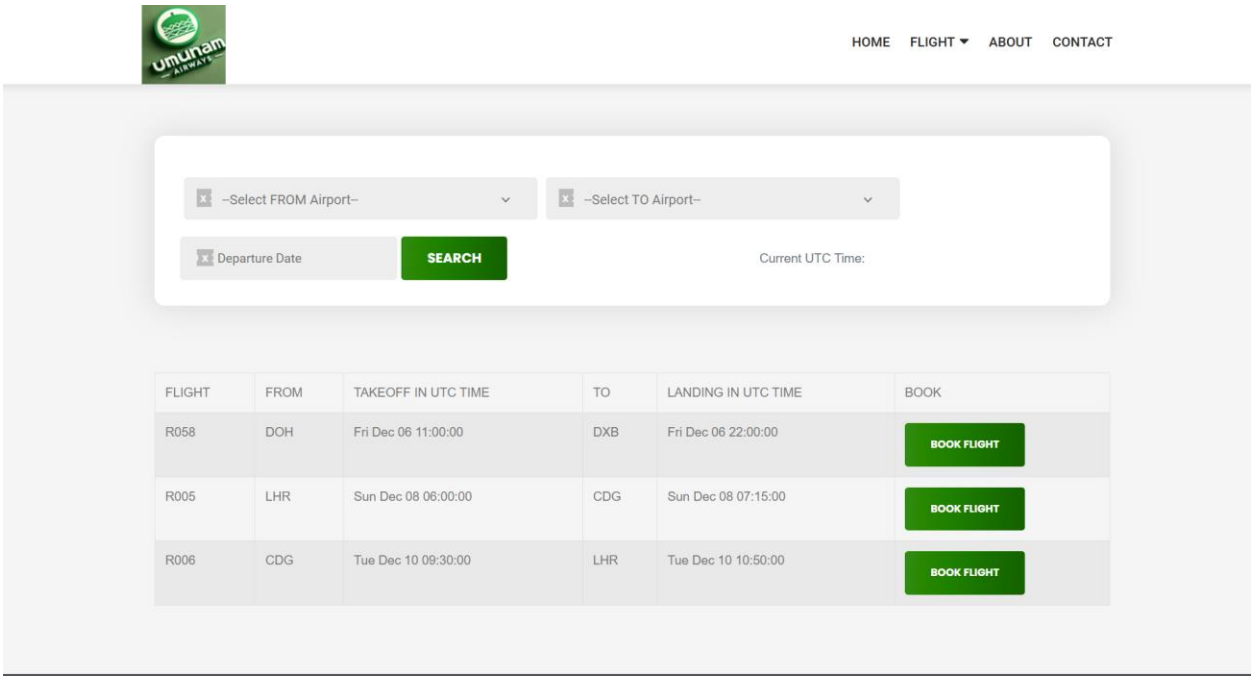
1) ER Diagram

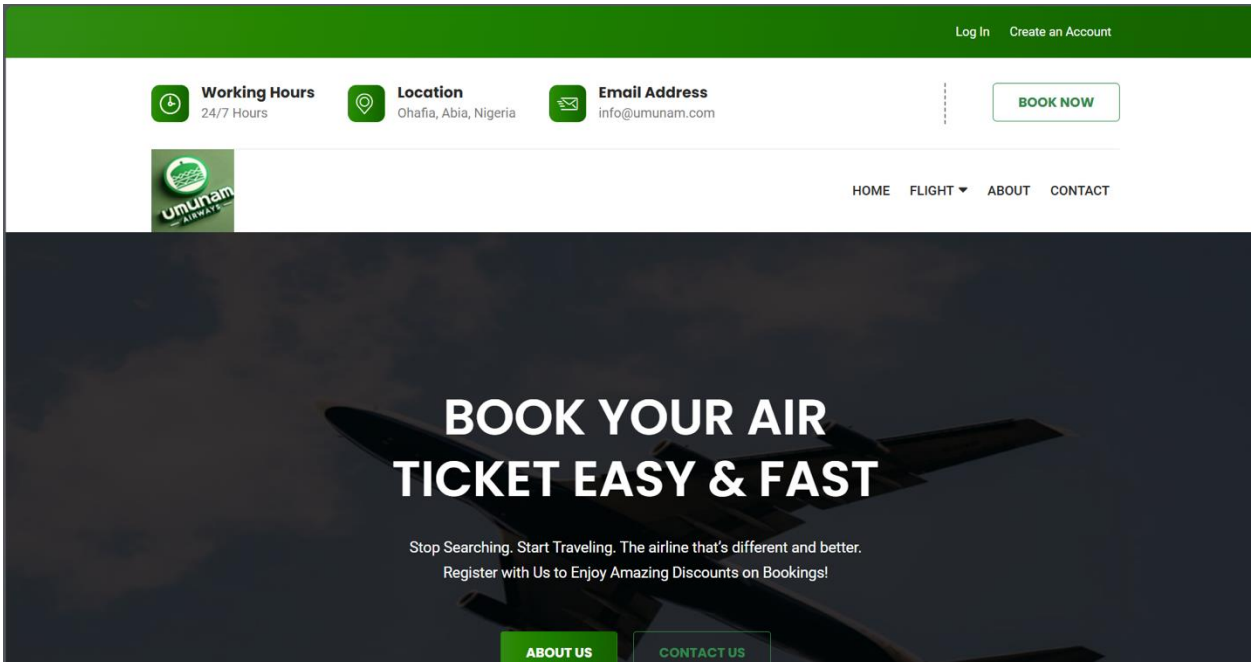


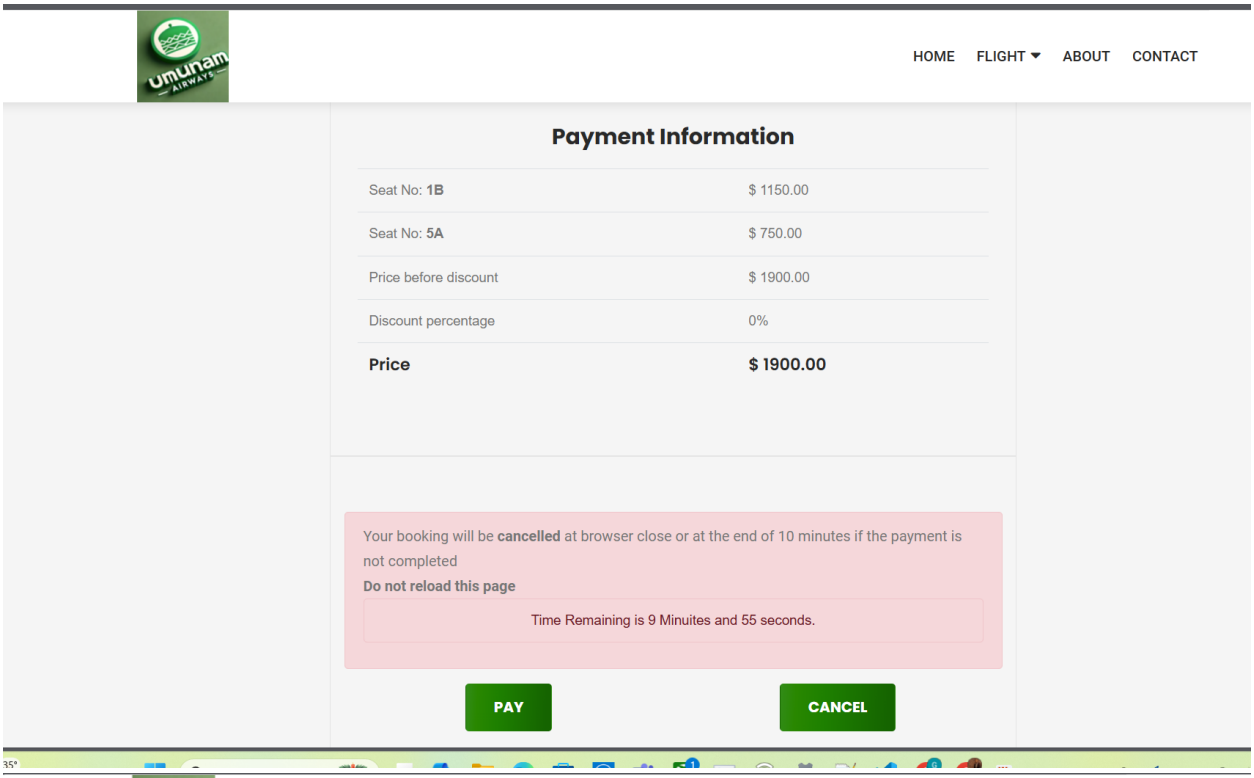
2) *Screenshots of database application:*

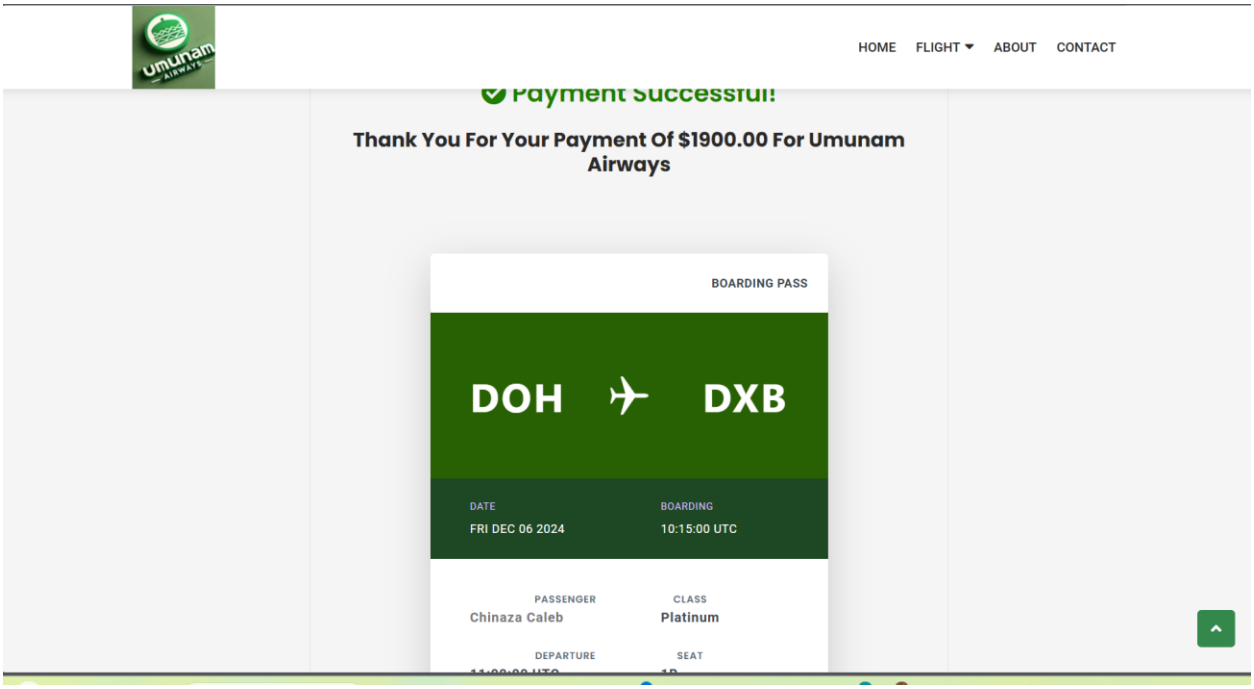


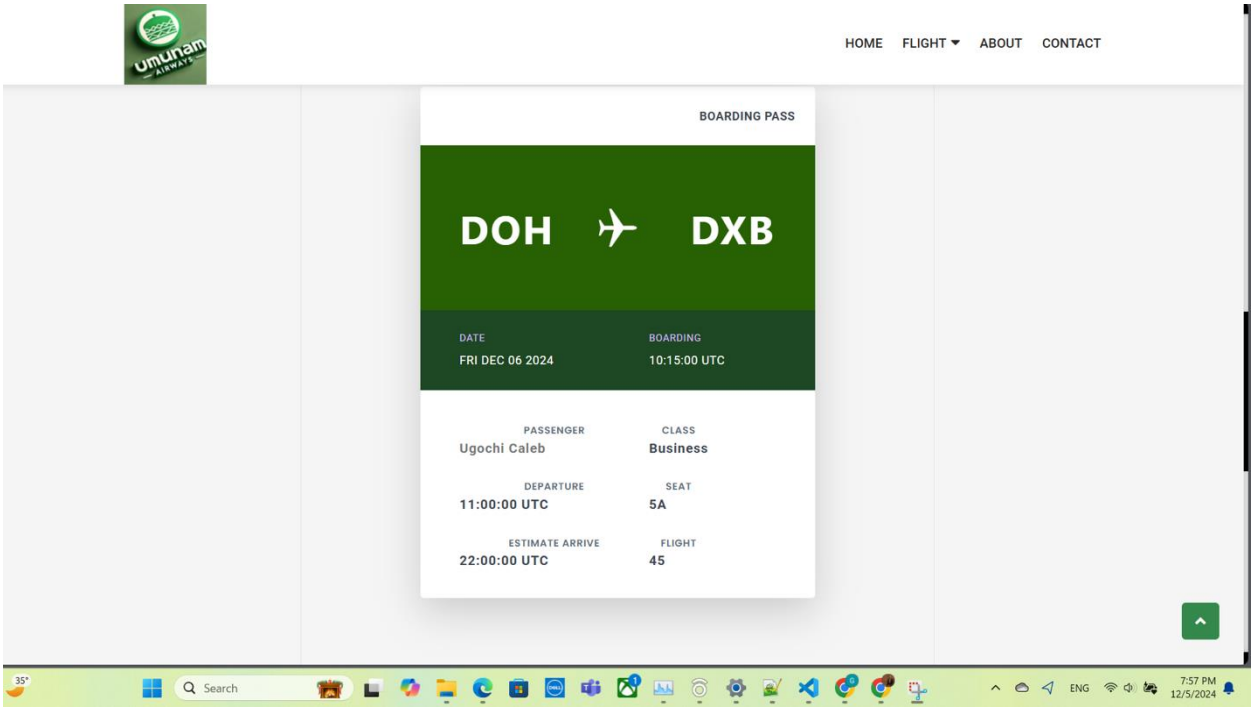


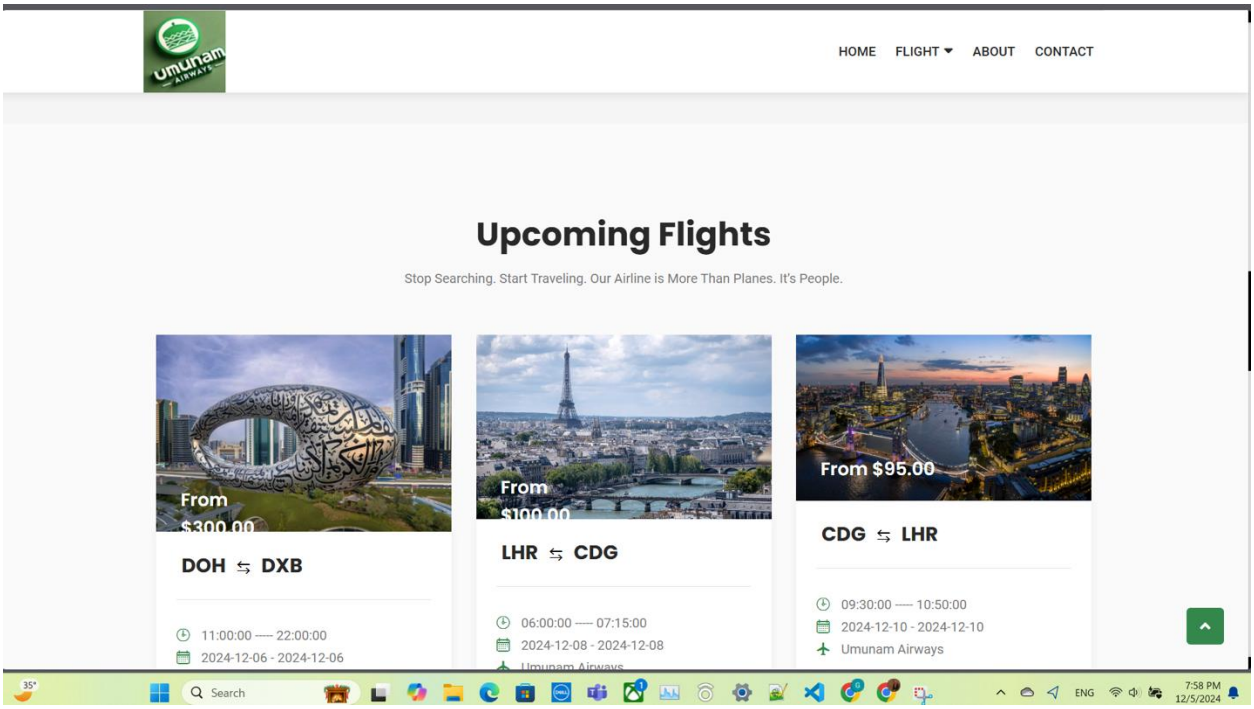

HOME
FLIGHT
ABOUT
CONTACT
--Select FROM Airport--
--Select TO Airport--
Departure Date
SEARCH
Current UTC Time:
FLIGHT
FROM
TAKEOFF IN UTC TIME
TO
LANDING IN UTC TIME
BOOK
R058
DOH
Fri Dec 06 11:00:00
DXB
Fri Dec 06 22:00:00
BOOK FLIGHT
R005
LHR
Sun Dec 08 06:00:00
CDG
Sun Dec 08 07:15:00
BOOK FLIGHT
R006
CDG
Tue Dec 10 09:30:00
LHR
Tue Dec 10 10:50:00
BOOK FLIGHT


Log In
Create an Account
Working Hours
24/7 Hours
Location
Ohafia, Abia, Nigeria
Email Address
info@umunam.com
BOOK NOW
HOME
FLIGHT
ABOUT
CONTACT
BOOK YOUR AIR
TICKET EASY & FAST
Stop Searching. Start Traveling. The airline that's different and better.
Register with Us to Enjoy Amazing Discounts on Bookings!
ABOUT US
CONTACT US


HOME
FLIGHT
ABOUT
CONTACT
Payment Information
Seat No: 1B
$ 1150.00
Seat No: 5A
$ 750.00
Price before discount
$ 1900.00
Discount percentage
0%
Price
$ 1900.00
Your booking will be cancelled at browser close or at the end of 10 minutes if the payment is not completed
Do not reload this page
Time Remaining is 9 Minuites and 55 seconds.
PAY
CANCEL


HOME
FLIGHT
ABOUT
CONTACT
Payment Successful!
Thank You For Your Payment Of $1900.00 For Umunam Airways
BOARDING PASS
DOH
DXB
DATE
FRI DEC 06 2024
BOARDING
10:15:00 UTC
PASSENGER
Chinaza Caleb
CLASS
Platinum
DEPARTURE
SEAT


HOME
FLIGHT
ABOUT
CONTACT
BOARDING PASS
DOH
DXB
DATE
FRI DEC 06 2024
BOARDING
10:15:00 UTC
PASSENGER
Ugochi Caleb
CLASS
Business
DEPARTURE
11:00:00 UTC
SEAT
5A
ESTIMATE ARRIVE
22:00:00 UTC
FLIGHT
45
Search
ENG
7:57 PM
12/5/2024


HOME
FLIGHT
ABOUT
CONTACT
Upcoming Flights
Stop Searching. Start Traveling. Our Airline is More Than Planes. It's People.
From $300.00
DOH ⇆ DXB
11:00:00 —— 22:00:00
2024-12-06 - 2024-12-06
From $100.00
LHR ⇆ CDG
06:00:00 —— 07:15:00
2024-12-08 - 2024-12-08
From $95.00
CDG ⇆ LHR
09:30:00 —— 10:50:00
2024-12-10 - 2024-12-10
Umunam Airways
Search
ENG
7:58 PM
12/5/2024