

**North South University**

Department of Electrical & Computer Engineering

**Project name:**

**Ticketing for all flights of Biman:**

A booking system like for the flights of Biman Bangladesh Airlines.

**Project ID**: 02

**Course:** CSE311

**Section:** 01

**Faculty:** Dr. Abu Sayed Mohammad Latiful Hoque

**Group no:** 22

**Members:**

1. Md. Asaduzzaman Sunny (2211702 642)
2. Kazi Abdullah Al Hasnaine (2211688 642)

**Project Scope**

To manage its ticketing system, **Biman Bangladesh Airlines** needs to develop a website where a **User** (user\_id, us\_f\_name, us\_m\_name, us\_l\_name, email, DOB, phone\_no) can create a profile and book multiple **Tickets** (ticket\_no, from, to, date, price, status). A user can book multiple tickets. Users can also **Reschedule** their flights and view their **Ticketing History** .  
  
A user can give feedback or **Queries**(query\_id, email, query\_text) through the website.

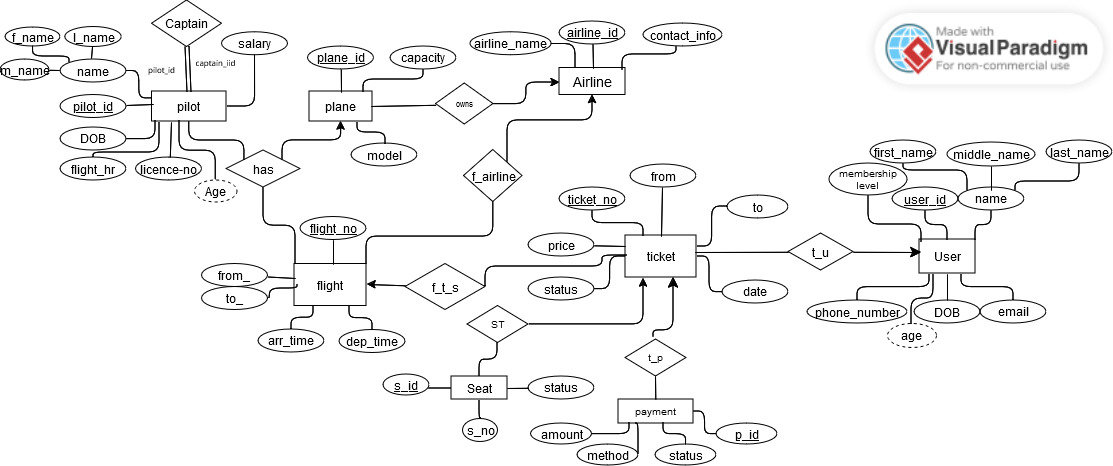
Each ticket requires an associated **Payment** (p\_id, amount,method, status).

A **Flight** (flight\_no, dep\_time, arr\_time,from\_, to\_) contains multiple **Seats** (s\_id, s\_no, status, price), which are allocated to tickets. Each ticket and seat is associated with a single flight.

There are multiple **Airlines** (airline\_id, name, contact\_info), and each airline may operate multiple flights. An airline can own several **Planes** (plane\_id, model, capacity).

Each flight is staffed by multiple **Pilots** (pilot\_id, DOB, first\_name, middle\_name, last\_name, licence\_no, flight\_hours, salary), and each flight has only one assigned plane. A pilot may serve as either a **General Pilot** or **Captain** (captain\_id).  
  
  
In the implicit section of the website, the admins should be able to manage each data from the database.   
That includes managing **Planes, Airlines, Flights, Pilots, Users.**  
  
**Admins** can review the **queries** sent by the users, and can reach out to them using the provided email address.   
  
The **payments** made by the users should be verified by the admins before the ticket associated for the payment can be printed.

**Project ERD**



**Project Schema**

**Flight** (flight\_no, dep\_time, arr\_time, from\_, to\_, plane\_id, airline\_id)

**Airline** (airline\_id, airline\_name, contact\_info)

**Seat** (s\_id, s\_no, s.status, price, ticket\_no, flight\_no)

**Ticket** (ticket\_no, from\_location, to\_location, date, price, t.status, flight\_no, user\_id)

**Plane** (plane\_id, model, capacity, airline\_id)

**Pilot** (pilot\_id, DOB, pi\_f\_name, pi\_m\_name, pi\_l\_name, licence\_no, flight\_hr, salary)

**Captain** (pilot\_id, captain\_id)

**User** (user\_id, us\_f\_name, us\_m\_name, us\_l\_name, email, DOB, phone\_no)

**Payment** (p\_id, amount, method, status, ticket\_no)

**Has** (plane\_id, flight\_no, pilot\_id)

**SQL DDL for the Relation Schema**

1. airline

CREATE TABLE Airline (

airline\_id INT AUTO\_INCREMENT PRIMARY KEY,

airline\_name VARCHAR(100),

contact\_info VARCHAR(150)

);

2. plane

CREATE TABLE Plane (

plane\_id INT AUTO\_INCREMENT PRIMARY KEY,

model VARCHAR(100),

capacity INT,

airline\_id INT,

FOREIGN KEY (airline\_id) REFERENCES Airline(airline\_id)

);

3. Flight

CREATE TABLE Flight (

flight\_no VARCHAR(20) PRIMARY KEY,

dep\_time DATETIME NOT NULL,

arr\_time DATETIME NOT NULL,

from\_ VARCHAR(50) NOT NULL,

to\_ VARCHAR(50) NOT NULL,

plane\_id INT NOT NULL,

airline\_id INT NOT NULL,

FOREIGN KEY (plane\_id) REFERENCES Plane(plane\_id),

FOREIGN KEY (airline\_id) REFERENCES Airline(airline\_id)

);

4. seat

CREATE TABLE Seat (

s\_id INT AUTO\_INCREMENT PRIMARY KEY,

s\_no VARCHAR(10) NOT NULL,

status VARCHAR(20) NOT NULL,

price DECIMAL(10, 2) NOT NULL,

ticket\_no INT,

flight\_no VARCHAR(10),

FOREIGN KEY (ticket\_no) REFERENCES Ticket(ticket\_no),

FOREIGN KEY (flight\_no) REFERENCES Flight(flight\_no)

);

5. pilot

CREATE TABLE Pilot (

pilot\_id INT AUTO\_INCREMENT PRIMARY KEY,

DOB DATE NOT NULL,

f\_name VARCHAR(50) NOT NULL,

m\_name VARCHAR(50),

l\_name VARCHAR(50) NOT NULL,

licence\_no VARCHAR(20) NOT NULL UNIQUE,

flight\_hr INT NOT NULL,

salary DECIMAL(10, 2) NOT NULL

);

6. captain

CREATE TABLE Captain (

captain\_id INT AUTO\_INCREMENT PRIMARY KEY,

pilot\_id INT NOT NULL,

FOREIGN KEY (pilot\_id) REFERENCES Pilot(pilot\_id)

);

7. User

CREATE TABLE User (

user\_id INT AUTO\_INCREMENT PRIMARY KEY,

f\_name VARCHAR(50) NOT NULL,

m\_name VARCHAR(50),

l\_name VARCHAR(50) NOT NULL,

email VARCHAR(100) NOT NULL UNIQUE,

DOB DATE NOT NULL,

phone\_no VARCHAR(15) NOT NULL UNIQUE

);

8. ticket

CREATE TABLE Ticket (

ticket\_no INT AUTO\_INCREMENT PRIMARY KEY,

from\_location VARCHAR(100) NOT NULL,

to\_location VARCHAR(100) NOT NULL,

date DATE NOT NULL,

price DECIMAL(10, 2) NOT NULL,

status VARCHAR(20) NOT NULL,

flight\_no VARCHAR(10) NOT NULL,

user\_id INT NOT NULL,

FOREIGN KEY (flight\_no) REFERENCES Flight(flight\_no),

FOREIGN KEY (user\_id) REFERENCES User(user\_id)

);

9. payment

CREATE TABLE Payment (

p\_id INT AUTO\_INCREMENT PRIMARY KEY,

amount DECIMAL(10, 2) NOT NULL,

method VARCHAR(50) NOT NULL,

status VARCHAR(20) NOT NULL,

ticket\_no INT NOT NULL,

FOREIGN KEY (ticket\_no) REFERENCES Ticket(ticket\_no)

);

10. has

CREATE TABLE Has (

plane\_id INT NOT NULL,

flight\_no VARCHAR(10) NOT NULL,

pilot\_id INT NOT NULL,

PRIMARY KEY (plane\_id, flight\_no, pilot\_id),

FOREIGN KEY (plane\_id) REFERENCES Plane(plane\_id),

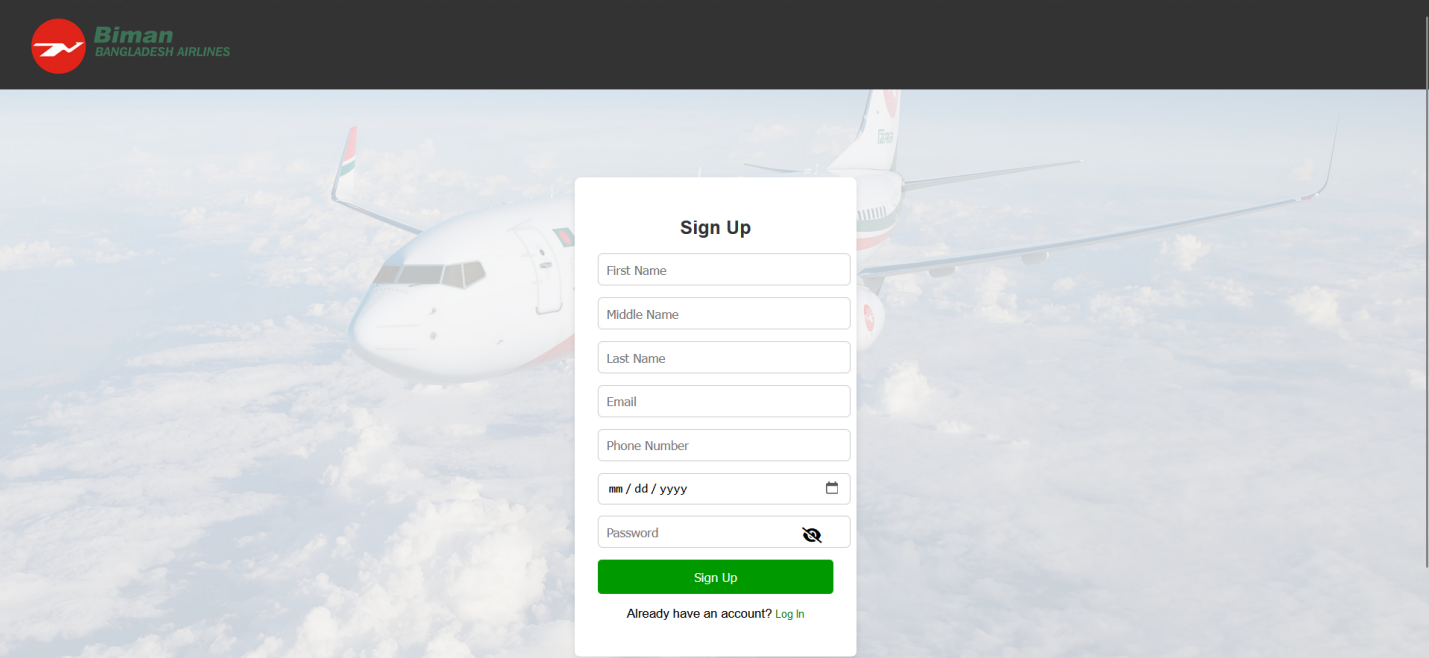
FOREIGN KEY (flight\_no) REFERENCES Flight(flight\_no),

FOREIGN KEY (pilot\_id) REFERENCES Pilot(pilot\_id)

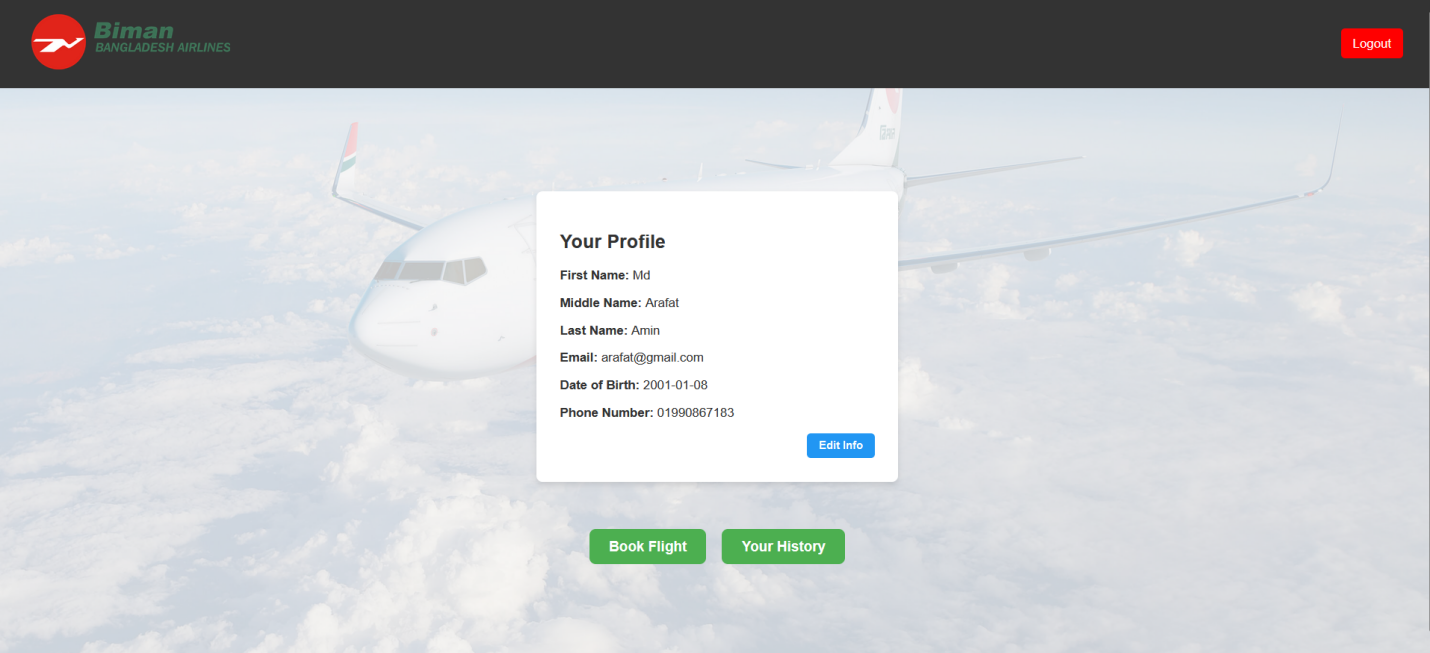
);

**UI Screenshot:**

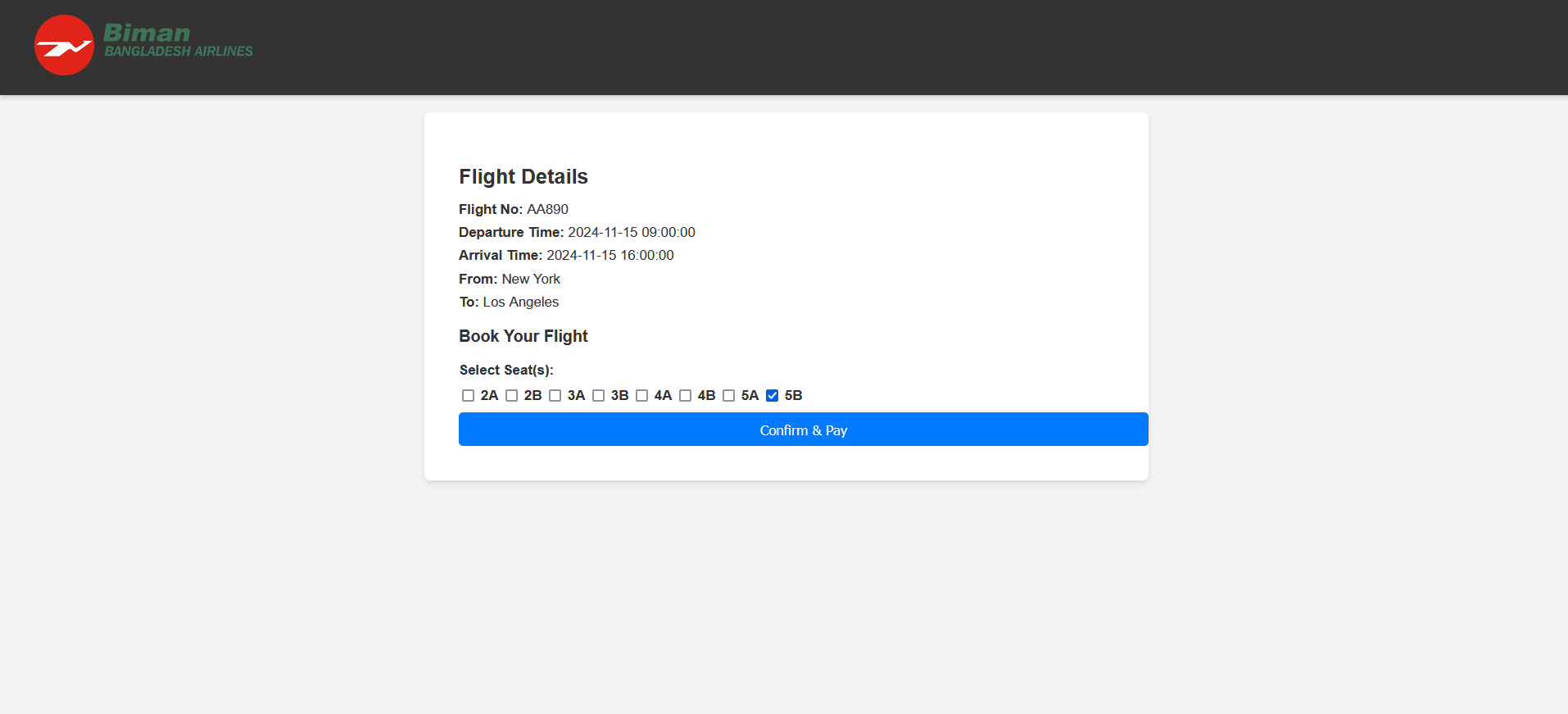
**User signup:**

****

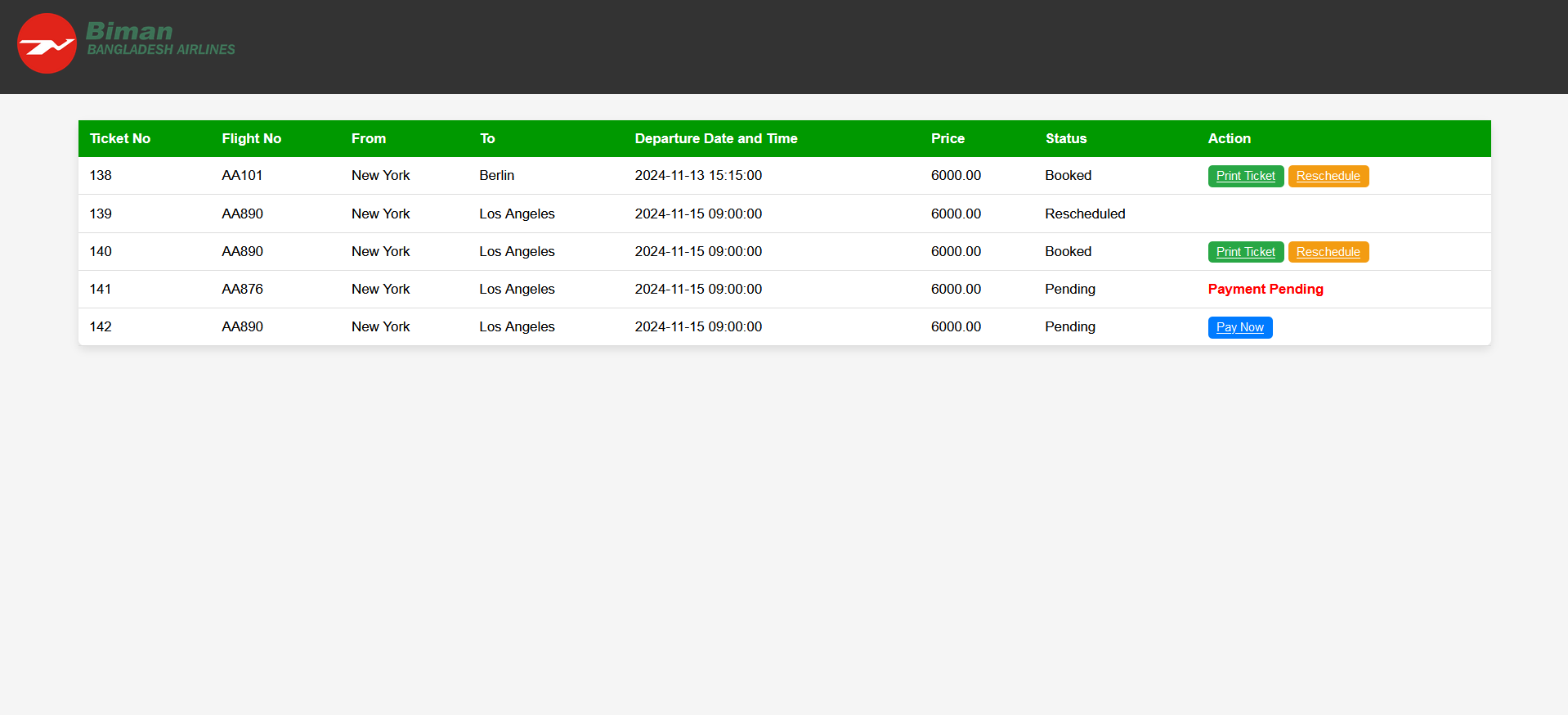
**User Profile:**

****

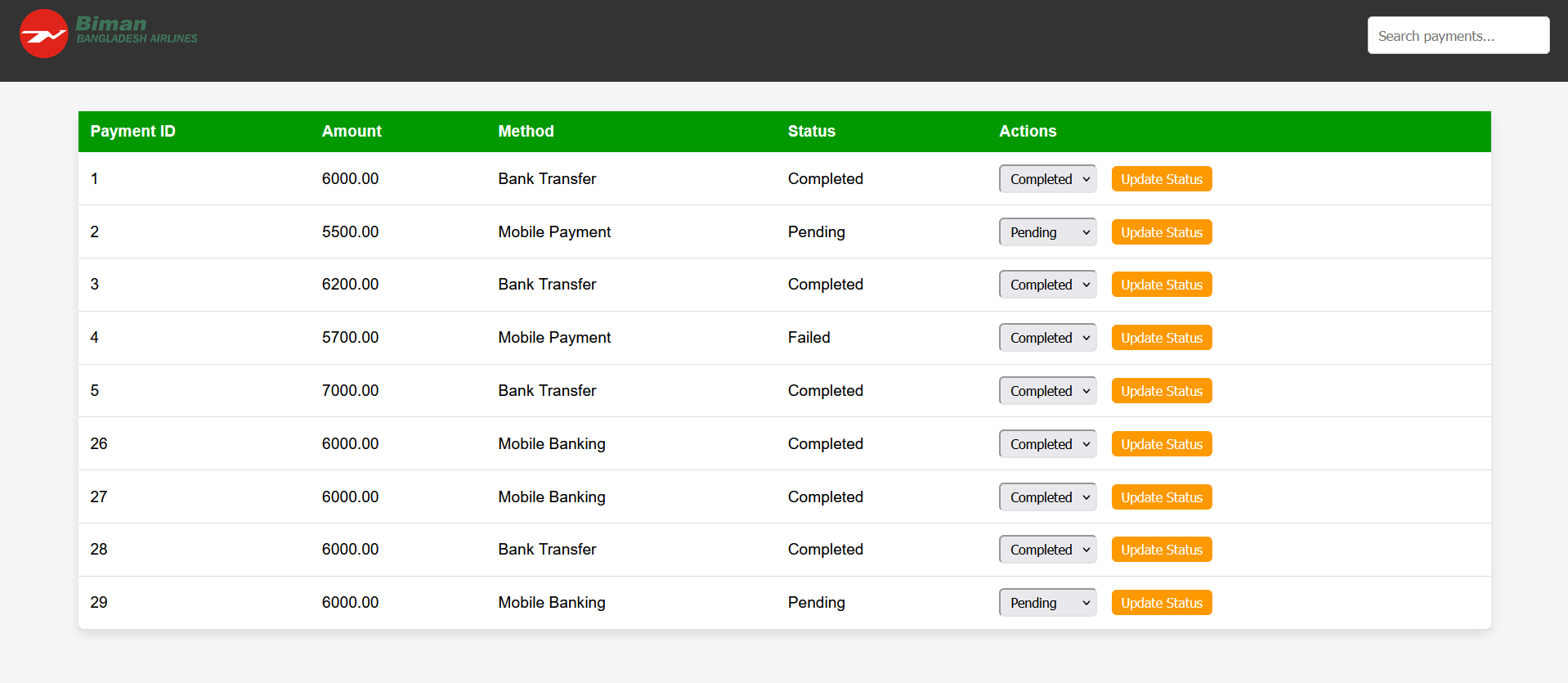
**User Ticket Seat Selection:**

****

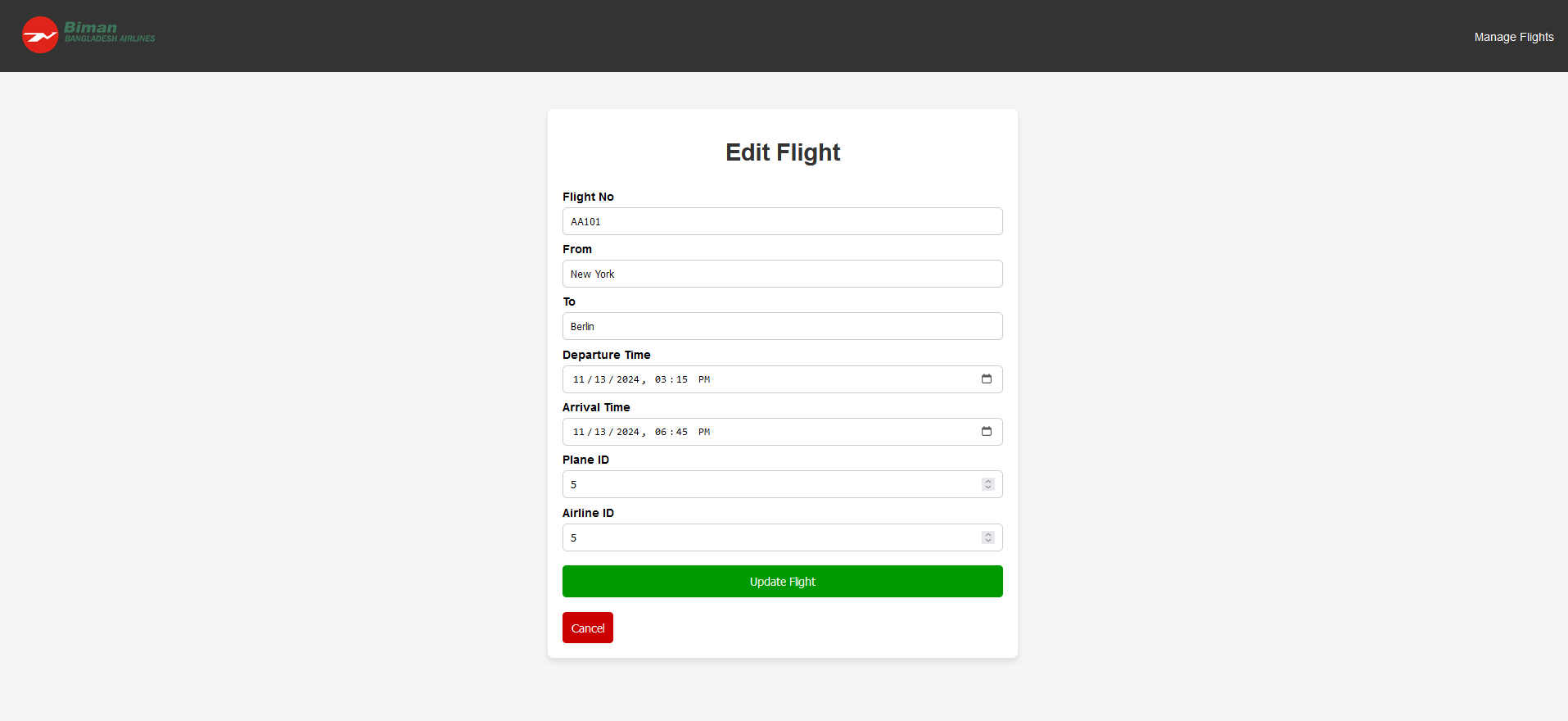
**User History:**

****

**Admin Payment Verification:**

****

**Admin Edit Flight:**

****

**Conclusion**

This project provides a complete ticketing system for Biman Bangladesh Airlines, making it easier to manage user profiles, flight bookings, rescheduling, and payments. The website allows users to create profiles, book tickets, and manage their travel history. Users can also reschedule flights, provide feedback, and submit queries.

Each ticket is linked to a flight, allowing users to select seats, make payments, and check their booking status. Payments are processed securely and must be verified by admins before tickets can be printed. The flight system includes flight schedules, seat availability, and connections to different airlines and planes.

Admins have full control over the system, including managing flights, users, airlines, planes, pilots, and user queries. They can also verify payments to ensure all bookings are valid. The admin interface helps manage flights, pilots, and other key parts of the ticketing system.

This system simplifies flight booking, rescheduling, and payment verification, offering a smooth experience for both users and admins. Biman Bangladesh Airlines can provide a reliable, user-friendly platform for managing air travel with easy administrative control.

However, there were some limitations faced during the development of the project. Payments need to be managed manually, as the system does not currently support automatic processing or OTP verification for confirmation. Additionally, the project could not be synchronized with the system date, which limits some dynamic features. Due to limitations in knowledge, we were unable to implement proper cascading deletes for foreign key relationships, which could affect the integrity of related data when records are deleted. These are areas for improvement in the future.