Flight Management Database System

TEAM NO. 7
TEAM MEMBERS:

SRIJAN JHA - PES1UG23AM316
SRUTHI MAHADEVAN - PES1UG23AM318

1. User Requirement Specifications

Purpose:

The Flight Management System (FL_Management) is designed to streamline and automate key airline operations including flight scheduling, ticket booking, passenger handling, and finance management. The purpose of the project is to centralize all flight-related data into a secure, consistent, and relational structure while providing an intuitive, modern interface for users. Built using MySQL as the backend and a ReactJS-based frontend, the system ensures efficient data processing, easy accessibility, and minimal manual intervention.

Scope:

The project's scope covers complete airline operations management — from creating and updating flight details to handling passenger records, transactions, and administrative roles. It supports full CRUD operations through an interactive web interface and enforces business rules using SQL constraints, triggers, functions, and stored procedures. The frontend is designed to be ultra-modern and user-friendly, featuring an airplane-themed dashboard, icons, and responsive layouts for real-time interaction.

2. Abstract/Description

The Flight Management Database System efficiently manages all flight operations, including passenger bookings, airport and aircraft data, financial transactions, and admin authentication. It ensures data consistency, automation, and security using SQL-based triggers, stored procedures, and functions. The project simulates real-world airline operations, covering booking, payments, and admin verification under a fully normalized relational model.

3. User Requirement Specification

Functional Requirements:

1. Add, update, and delete flight records.
2. Manage airports, aircrafts, passengers, and ticket details.
3. Automatically compute flight duration.
4. Book and cancel tickets via stored procedures.
5. Record all financial transactions per booking.

6. Maintain login access for admins with OTP-based 2FA.
7. Auto-update passenger total tickets upon booking/cancellation.
8. Validate finance records (amount > 0).

Non-Functional Requirements:

- Database must support referential integrity.
- Responses for booking and update operations should execute in under 1 second.
- Secure data access restricted to authorized admins.
- Scalable for multi-airport and multi-aircraft operations.

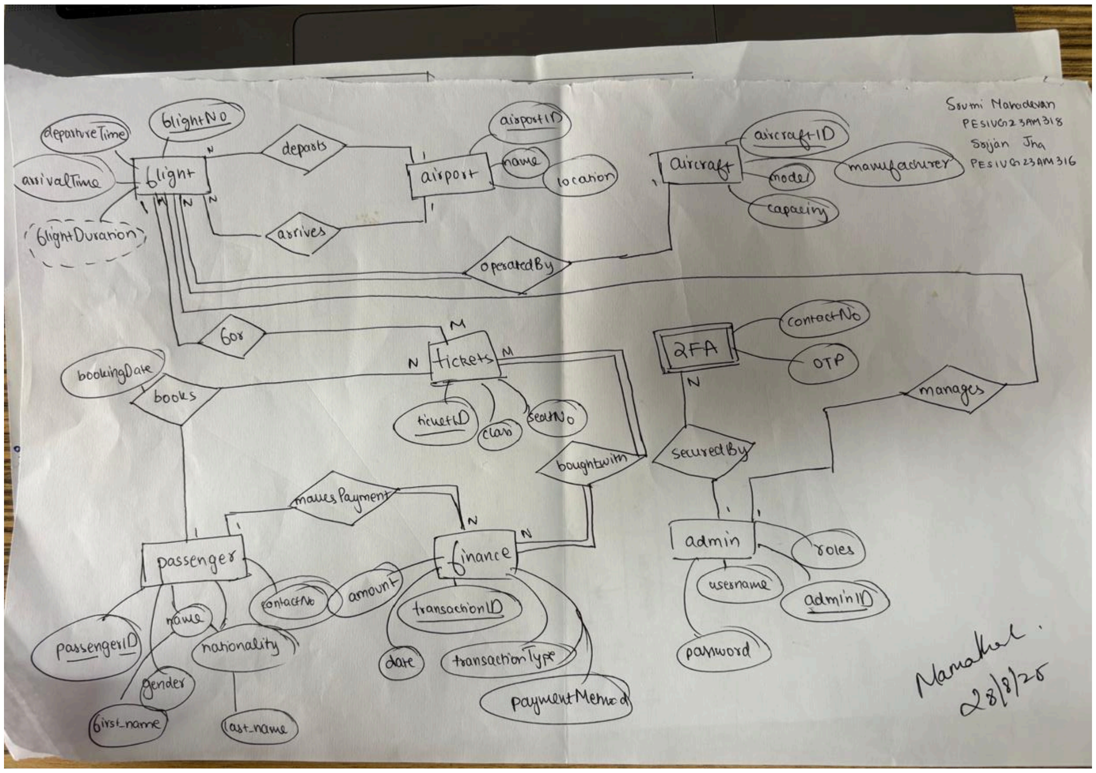
4. Software Used

Component	Tool / Language
Database	MySQL
Interface	MySQL Workbench
Language	SQL (DDL, DML, PL/SQL), ReactJS, Node,HTML, CSS
ER Diagram	Hand-drawn
Testing	SQL Command Line, Workbench Console
OS	MacOS

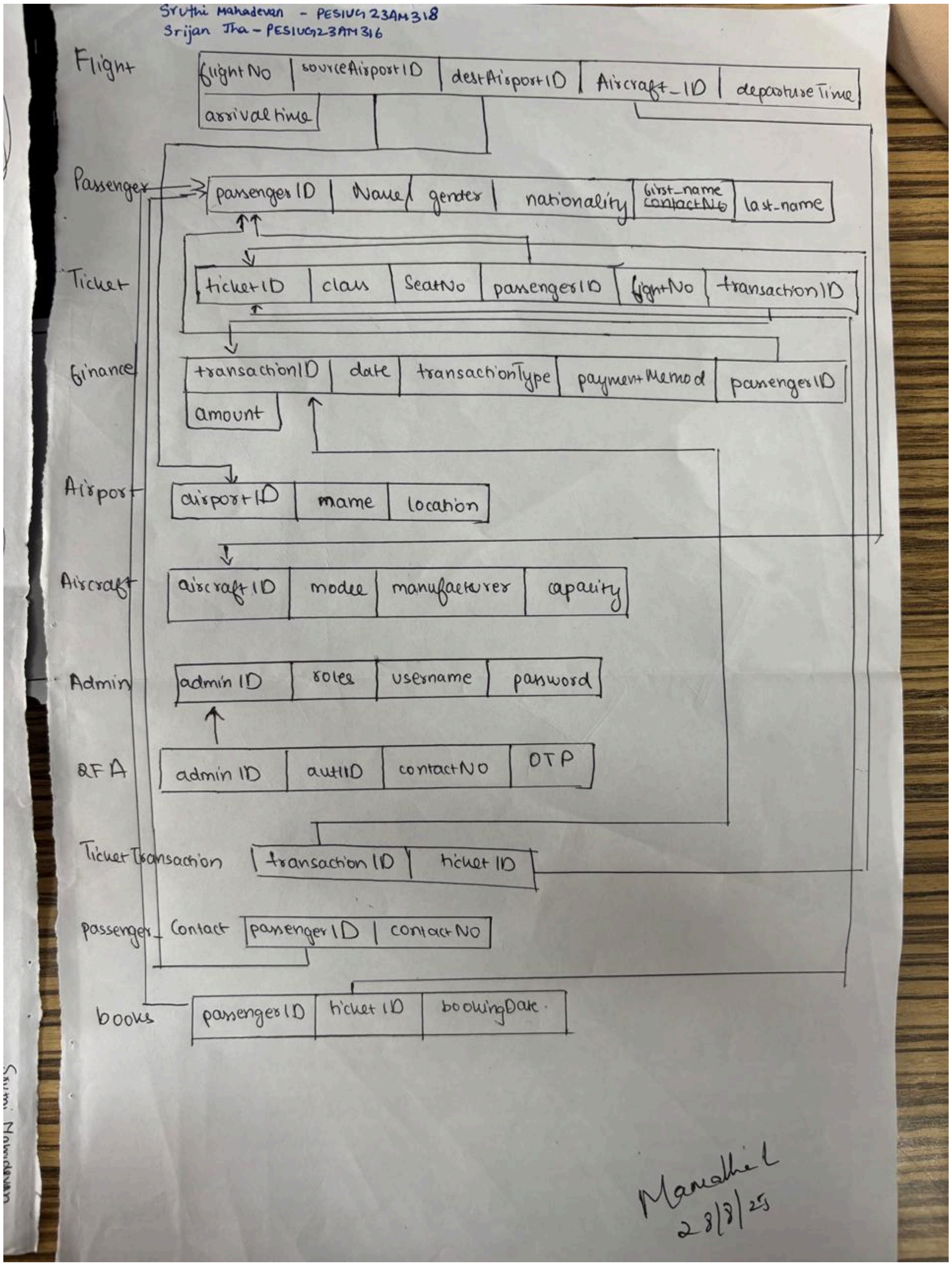
5. ER Diagram

Entities:

- `Flight`, `Airport`, `Aircraft`, `Tickets`, `Passenger`, `Finance`, `Admin`, `2FA`.

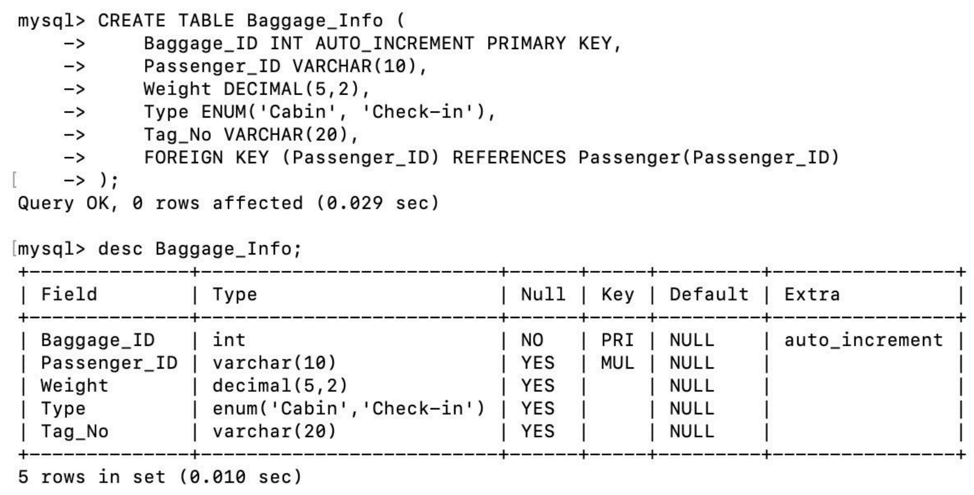


6. Relational Schema



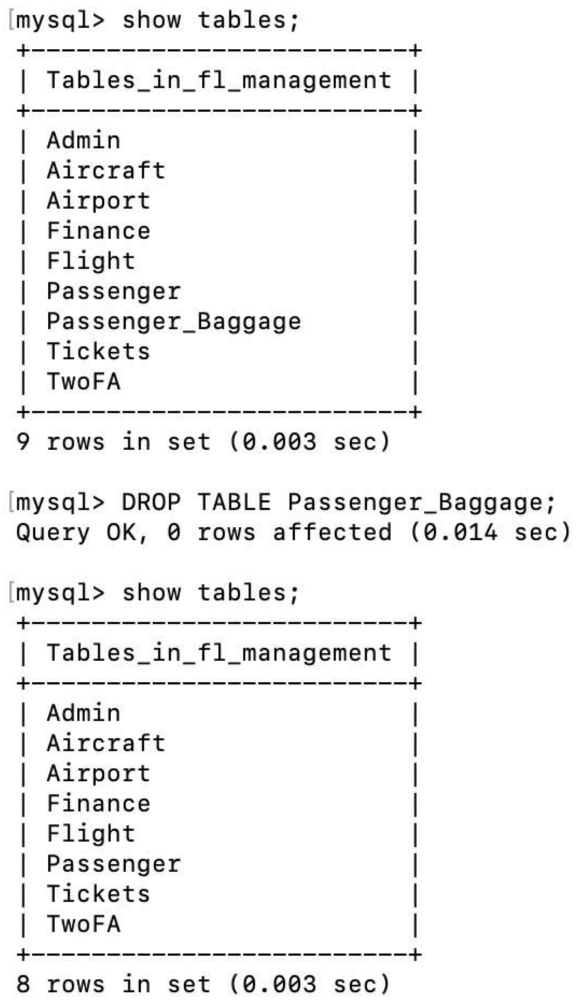
7. DDL Commands (Schema Creation)

CREATE:



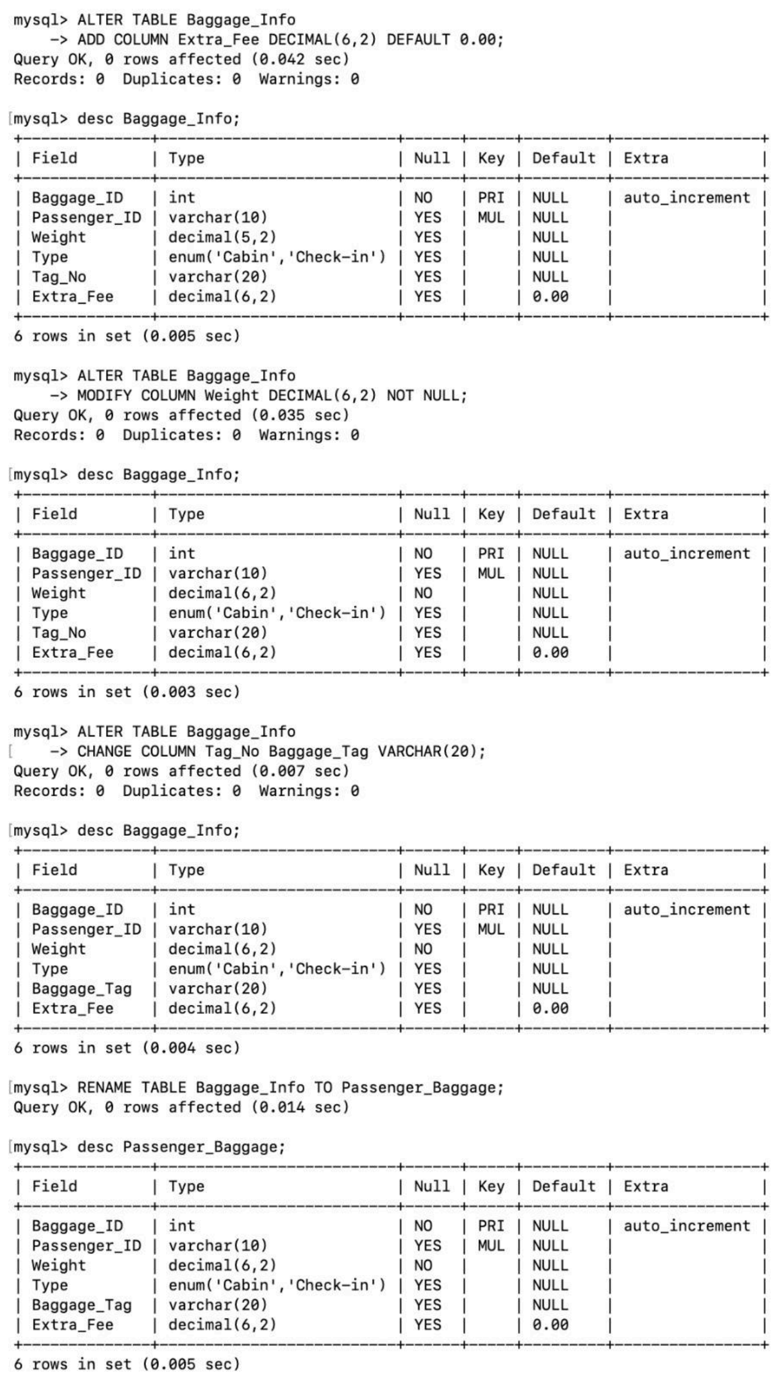
```
mysql> CREATE TABLE Baggage_Info (
    ->     Baggage_ID INT AUTO_INCREMENT PRIMARY KEY,
    ->     Passenger_ID VARCHAR(10),
    ->     Weight DECIMAL(5,2),
    ->     Type ENUM('Cabin', 'Check-in'),
    ->     Tag_No VARCHAR(20),
    ->     FOREIGN KEY (Passenger_ID) REFERENCES Passenger(Passenger_ID)
[    -> );
Query OK, 0 rows affected (0.029 sec)

[mysql> desc Baggage_Info;
+--------------+-------------------------+------+-----+---------+----------------+
| Field        | Type                    | Null | Key | Default | Extra          |
+--------------+-------------------------+------+-----+---------+----------------+
| Baggage_ID   | int                     | NO   | PRI | NULL    | auto_increment |
| Passenger_ID | varchar(10)             | YES  | MUL | NULL    |                |
| Weight       | decimal(5,2)            | YES  |     | NULL    |                |
| Type         | enum('Cabin','Check-in') | YES  |     | NULL    |                |
| Tag_No       | varchar(20)             | YES  |     | NULL    |                |
+--------------+-------------------------+------+-----+---------+----------------+
5 rows in set (0.010 sec)
```

DROP:



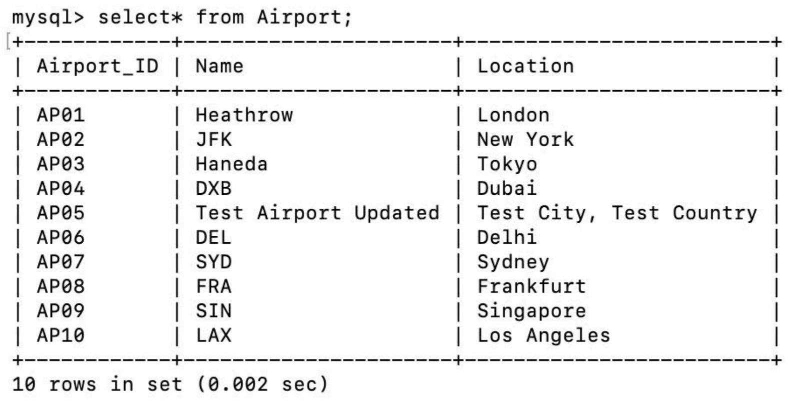
```
[mysql> show tables;
+-------------------------+
| Tables_in_fl_management |
+-------------------------+
| Admin                   |
| Aircraft                |
| Airport                 |
| Finance                 |
| Flight                  |
| Passenger               |
| Passenger_Baggage       |
| Tickets                 |
| TwoFA                   |
+-------------------------+
9 rows in set (0.003 sec)

[mysql> DROP TABLE Passenger_Baggage;
Query OK, 0 rows affected (0.014 sec)

[mysql> show tables;
+-------------------------+
| Tables_in_fl_management |
+-------------------------+
| Admin                   |
| Aircraft                |
| Airport                 |
| Finance                 |
| Flight                  |
| Passenger               |
| Tickets                 |
| TwoFA                   |
+-------------------------+
8 rows in set (0.003 sec)
```

ALTER:

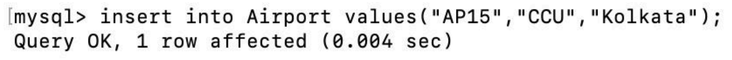


```
mysql> ALTER TABLE Baggage_Info
    -> ADD COLUMN Extra_Fee DECIMAL(6,2) DEFAULT 0.00;
Query OK, 0 rows affected (0.042 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc Baggage_Info;
+--------------+--------------------------+------+-----+---------+----------------+
| Field        | Type                     | Null | Key | Default | Extra          |
+--------------+--------------------------+------+-----+---------+----------------+
| Baggage_ID   | int                      | NO   | PRI | NULL    | auto_increment |
| Passenger_ID | varchar(10)              | YES  | MUL | NULL    |                |
| Weight       | decimal(5,2)             | YES  |     | NULL    |                |
| Type         | enum('Cabin','Check-in') | YES  |     | NULL    |                |
| Tag_No       | varchar(20)              | YES  |     | NULL    |                |
| Extra_Fee    | decimal(6,2)             | YES  |     | 0.00    |                |
+--------------+--------------------------+------+-----+---------+----------------+
6 rows in set (0.005 sec)

mysql> ALTER TABLE Baggage_Info
    -> MODIFY COLUMN Weight DECIMAL(6,2) NOT NULL;
Query OK, 0 rows affected (0.035 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc Baggage_Info;
+--------------+--------------------------+------+-----+---------+----------------+
| Field        | Type                     | Null | Key | Default | Extra          |
+--------------+--------------------------+------+-----+---------+----------------+
| Baggage_ID   | int                      | NO   | PRI | NULL    | auto_increment |
| Passenger_ID | varchar(10)              | YES  | MUL | NULL    |                |
| Weight       | decimal(6,2)             | NO   |     | NULL    |                |
| Type         | enum('Cabin','Check-in') | YES  |     | NULL    |                |
| Tag_No       | varchar(20)              | YES  |     | NULL    |                |
| Extra_Fee    | decimal(6,2)             | YES  |     | 0.00    |                |
+--------------+--------------------------+------+-----+---------+----------------+
6 rows in set (0.003 sec)

mysql> ALTER TABLE Baggage_Info
    -> CHANGE COLUMN Tag_No Baggage_Tag VARCHAR(20);
Query OK, 0 rows affected (0.007 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> desc Baggage_Info;
+--------------+--------------------------+------+-----+---------+----------------+
| Field        | Type                     | Null | Key | Default | Extra          |
+--------------+--------------------------+------+-----+---------+----------------+
| Baggage_ID   | int                      | NO   | PRI | NULL    | auto_increment |
| Passenger_ID | varchar(10)              | YES  | MUL | NULL    |                |
| Weight       | decimal(6,2)             | NO   |     | NULL    |                |
| Type         | enum('Cabin','Check-in') | YES  |     | NULL    |                |
| Baggage_Tag  | varchar(20)              | YES  |     | NULL    |                |
| Extra_Fee    | decimal(6,2)             | YES  |     | 0.00    |                |
+--------------+--------------------------+------+-----+---------+----------------+
6 rows in set (0.004 sec)

mysql> RENAME TABLE Baggage_Info TO Passenger_Baggage;
Query OK, 0 rows affected (0.014 sec)

mysql> desc Passenger_Baggage;
+--------------+--------------------------+------+-----+---------+----------------+
| Field        | Type                     | Null | Key | Default | Extra          |
+--------------+--------------------------+------+-----+---------+----------------+
| Baggage_ID   | int                      | NO   | PRI | NULL    | auto_increment |
| Passenger_ID | varchar(10)              | YES  | MUL | NULL    |                |
| Weight       | decimal(6,2)             | NO   |     | NULL    |                |
| Type         | enum('Cabin','Check-in') | YES  |     | NULL    |                |
| Baggage_Tag  | varchar(20)              | YES  |     | NULL    |                |
| Extra_Fee    | decimal(6,2)             | YES  |     | 0.00    |                |
+--------------+--------------------------+------+-----+---------+----------------+
6 rows in set (0.005 sec)
```

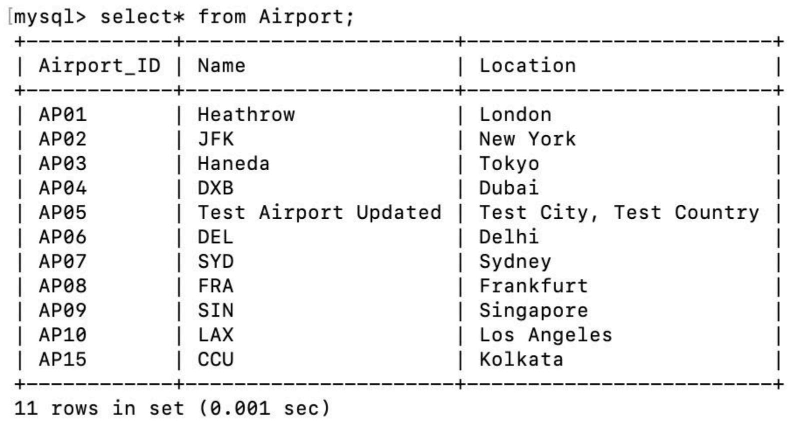
8. CRUD operation Screenshots



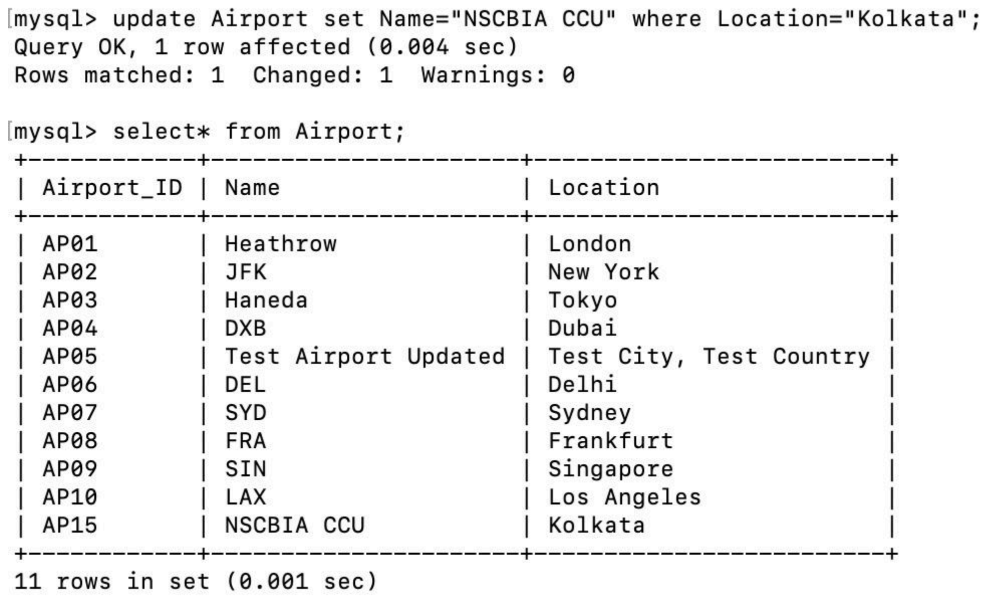
```
mysql> select* from Airport;
+------------+-----------------------+-------------------------+
| Airport_ID | Name                  | Location                |
+------------+-----------------------+-------------------------+
| AP01       | Heathrow              | London                  |
| AP02       | JFK                   | New York                |
| AP03       | Haneda                | Tokyo                   |
| AP04       | DXB                   | Dubai                   |
| AP05       | Test Airport Updated  | Test City, Test Country |
| AP06       | DEL                   | Delhi                   |
| AP07       | SYD                   | Sydney                  |
| AP08       | FRA                   | Frankfurt               |
| AP09       | SIN                   | Singapore               |
| AP10       | LAX                   | Los Angeles             |
+------------+-----------------------+-------------------------+
10 rows in set (0.002 sec)
```



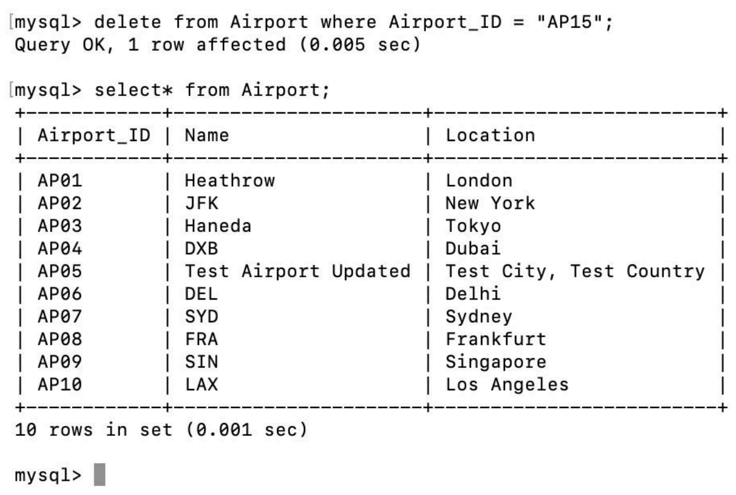
```
mysql> insert into Airport values("AP15","CCU","Kolkata");
Query OK, 1 row affected (0.004 sec)
```



```
mysql> select* from Airport;
+------------+-----------------------+-------------------------+
| Airport_ID | Name                  | Location                |
+------------+-----------------------+-------------------------+
| AP01       | Heathrow              | London                  |
| AP02       | JFK                   | New York                |
| AP03       | Haneda                | Tokyo                   |
| AP04       | DXB                   | Dubai                   |
| AP05       | Test Airport Updated  | Test City, Test Country |
| AP06       | DEL                   | Delhi                   |
| AP07       | SYD                   | Sydney                  |
| AP08       | FRA                   | Frankfurt               |
| AP09       | SIN                   | Singapore               |
| AP10       | LAX                   | Los Angeles             |
| AP15       | CCU                   | Kolkata                 |
+------------+-----------------------+-------------------------+
11 rows in set (0.001 sec)
```



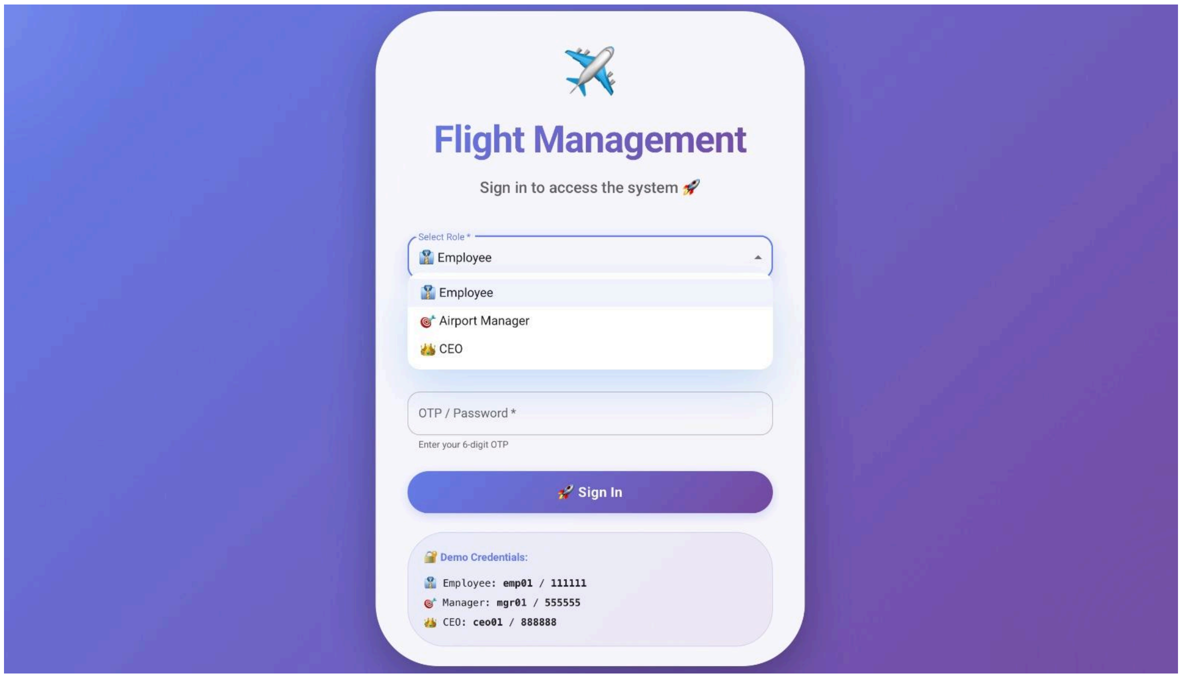
```
mysql> update Airport set Name="NSCBIA CCU" where Location="Kolkata";
Query OK, 1 row affected (0.004 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select* from Airport;
+------------+-----------------------+-------------------------+
| Airport_ID | Name                  | Location                |
+------------+-----------------------+-------------------------+
| AP01       | Heathrow              | London                  |
| AP02       | JFK                   | New York                |
| AP03       | Haneda                | Tokyo                   |
| AP04       | DXB                   | Dubai                   |
| AP05       | Test Airport Updated  | Test City, Test Country |
| AP06       | DEL                   | Delhi                   |
| AP07       | SYD                   | Sydney                  |
| AP08       | FRA                   | Frankfurt               |
| AP09       | SIN                   | Singapore               |
| AP10       | LAX                   | Los Angeles             |
| AP15       | NSCBIA CCU            | Kolkata                 |
+------------+-----------------------+-------------------------+
11 rows in set (0.001 sec)
```



```
mysql> delete from Airport where Airport_ID = "AP15";
Query OK, 1 row affected (0.005 sec)

mysql> select* from Airport;
+------------+-----------------------+-------------------------+
| Airport_ID | Name                  | Location                |
+------------+-----------------------+-------------------------+
| AP01       | Heathrow              | London                  |
| AP02       | JFK                   | New York                |
| AP03       | Haneda                | Tokyo                   |
| AP04       | DXB                   | Dubai                   |
| AP05       | Test Airport Updated  | Test City, Test Country |
| AP06       | DEL                   | Delhi                   |
| AP07       | SYD                   | Sydney                  |
| AP08       | FRA                   | Frankfurt               |
| AP09       | SIN                   | Singapore               |
| AP10       | LAX                   | Los Angeles             |
+------------+-----------------------+-------------------------+
10 rows in set (0.001 sec)

mysql> 
```

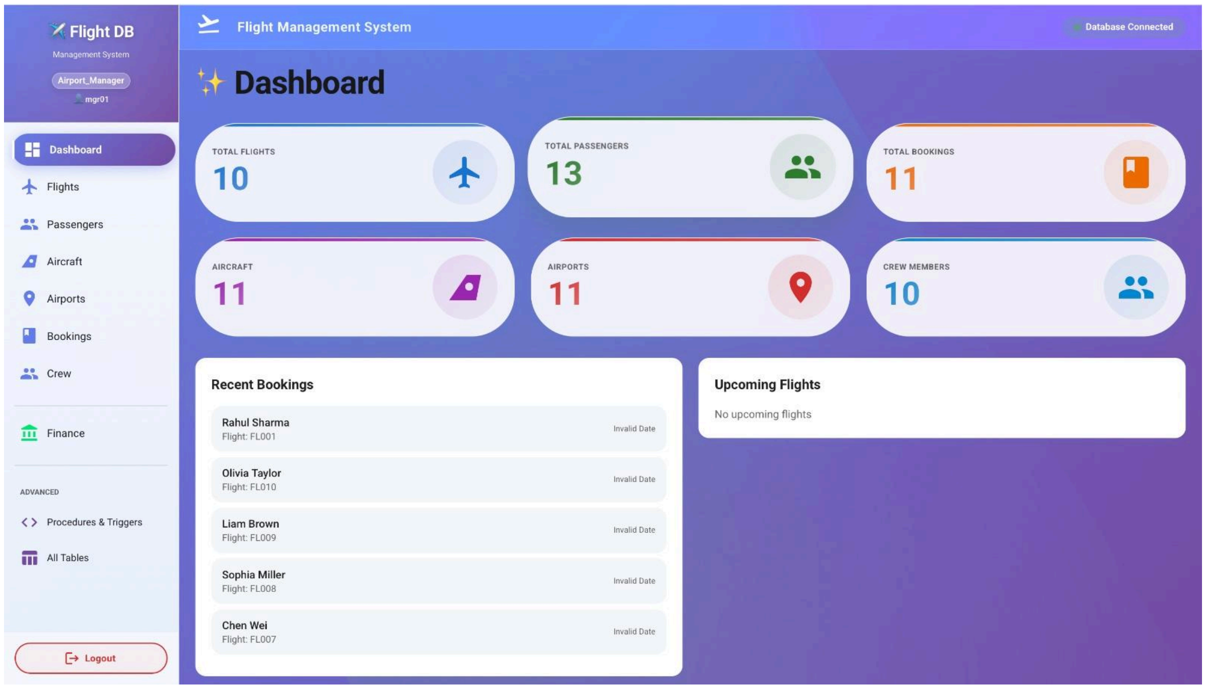
9. List of functionalities/features of the application and its associated screenshots using front end

F1: Secure Login and Role-Based Access: Login page with user authentication. Only managers and the CEO can access the firm's financial tables — ensuring security and restricted access.



Demo Credentials:
Employee: emp01 / 111111
Manager: mgr01 / 555555
CEO: ceo01 / 888888

F2: Interactive Dashboard: Central dashboard displaying key navigation and summaries for easy access to different database tables and functionalities.



Dashboard

TOTAL FLIGHTS 10

TOTAL PASSENGERS 13

TOTAL BOOKINGS 11

AIRCRAFT 11

AIRPORTS 11

CREW MEMBERS 10

Recent Bookings

Rahul Sharma
Flight: FL001 — Invalid Date

Olivia Taylor
Flight: FL010 — Invalid Date

Liam Brown
Flight: FL009 — Invalid Date

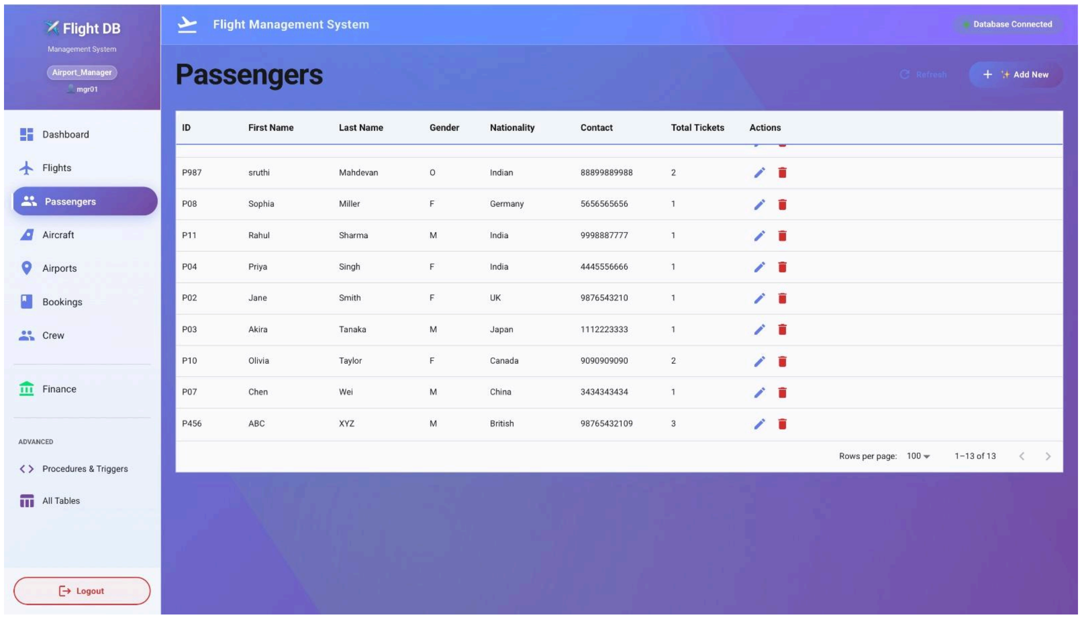
Sophia Miller
Flight: FL008 — Invalid Date

Chen Wei
Flight: FL007 — Invalid Date

Upcoming Flights

No upcoming flights

F3: View Data Tables: Users can view records from any of the seven visible database tables. The *TwoFA* table remains hidden for security reasons.

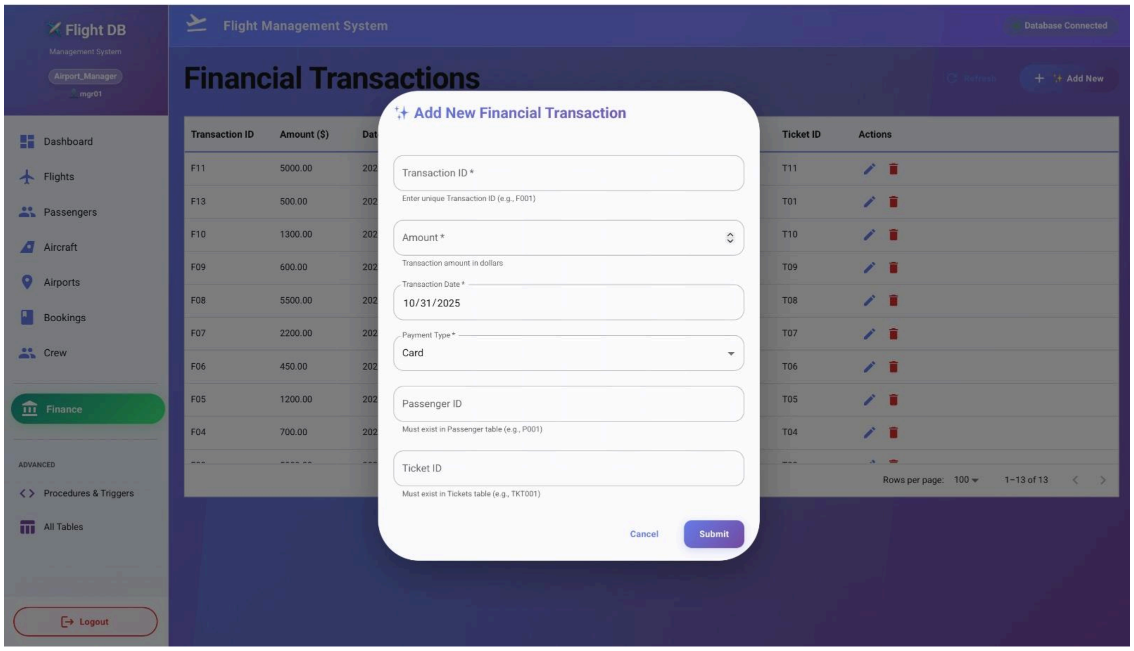


Passengers

ID	First Name	Last Name	Gender	Nationality	Contact	Total Tickets	Actions
P987	sruthi	Mahdevan	O	Indian	88899889988	2	
P08	Sophia	Miller	F	Germany	5656565656	1	
P11	Rahul	Sharma	M	India	9998887777	1	
P04	Priya	Singh	F	India	4445556666	1	
P02	Jane	Smith	F	UK	9876543210	1	
P03	Akira	Tanaka	M	Japan	1112223333	1	
P10	Olivia	Taylor	F	Canada	9090909090	2	
P07	Chen	Wei	M	China	3434343434	1	
P456	ABC	XYZ	M	British	98765432109	3	

Rows per page: 100 1–13 of 13

F4: Add New Records: Users can add new entries to any table. All additions are reflected and stored in the backend database instantly.



Financial Transactions

Transaction ID	Amount ($)	Dat	Ticket ID	Actions
F11	5000.00	202	T11	
F13	500.00	202	T01	
F10	1300.00	202	T10	
F09	600.00	202	T09	
F08	5500.00	202	T08	
F07	2200.00	202	T07	
F06	450.00	202	T06	
F05	1200.00	202	T05	
F04	700.00	202	T04	

Rows per page: 100 1–13 of 13

Add New Financial Transaction

Transaction ID *
Enter unique Transaction ID (e.g., F001)

Amount *
Transaction amount in dollars

Transaction Date *: 10/31/2025

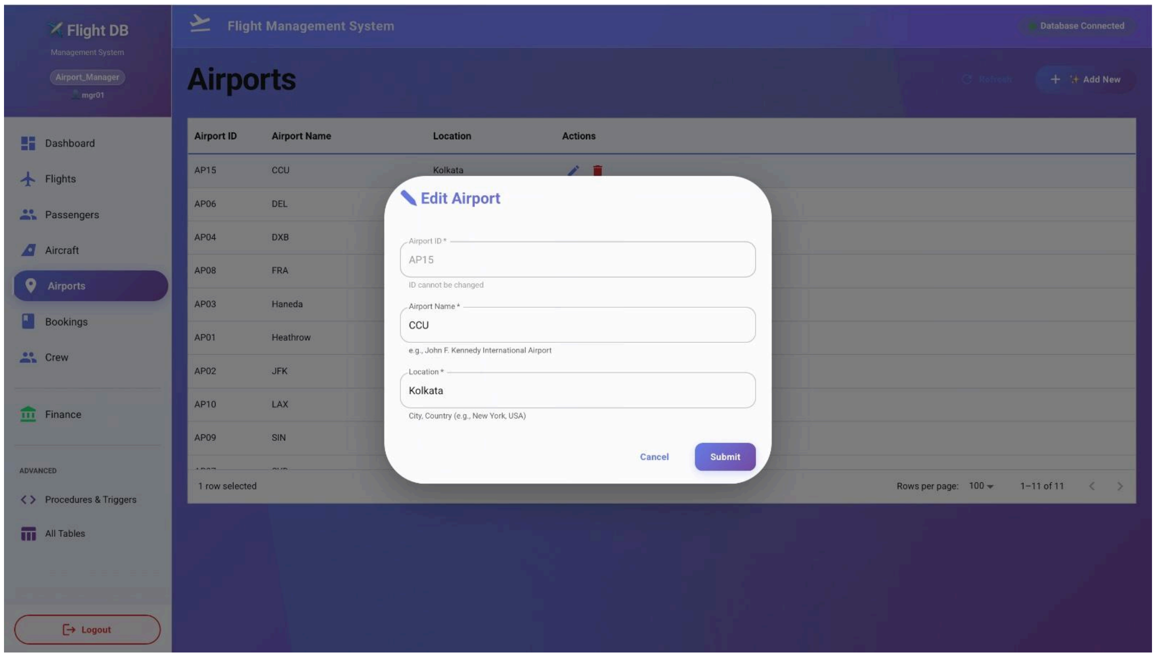
Payment Type *: Card

Passenger ID
Must exist in Passenger table (e.g., P001)

Ticket ID
Must exist in Tickets table (e.g., TKT001)

Cancel Submit

F5: Edit Existing Records: Allows editing of existing table rows. Any modification made on the frontend is automatically updated in the backend.



Airports

Airport ID	Airport Name	Location	Actions
AP15	CCU	Kolkata	
AP06	DEL		
AP04	DXB		
AP08	FRA		
AP03	Haneda		
AP01	Heathrow		
AP02	JFK		
AP10	LAX		
AP09	SIN		

1 row selected   Rows per page: 100 1–11 of 11

Edit Airport

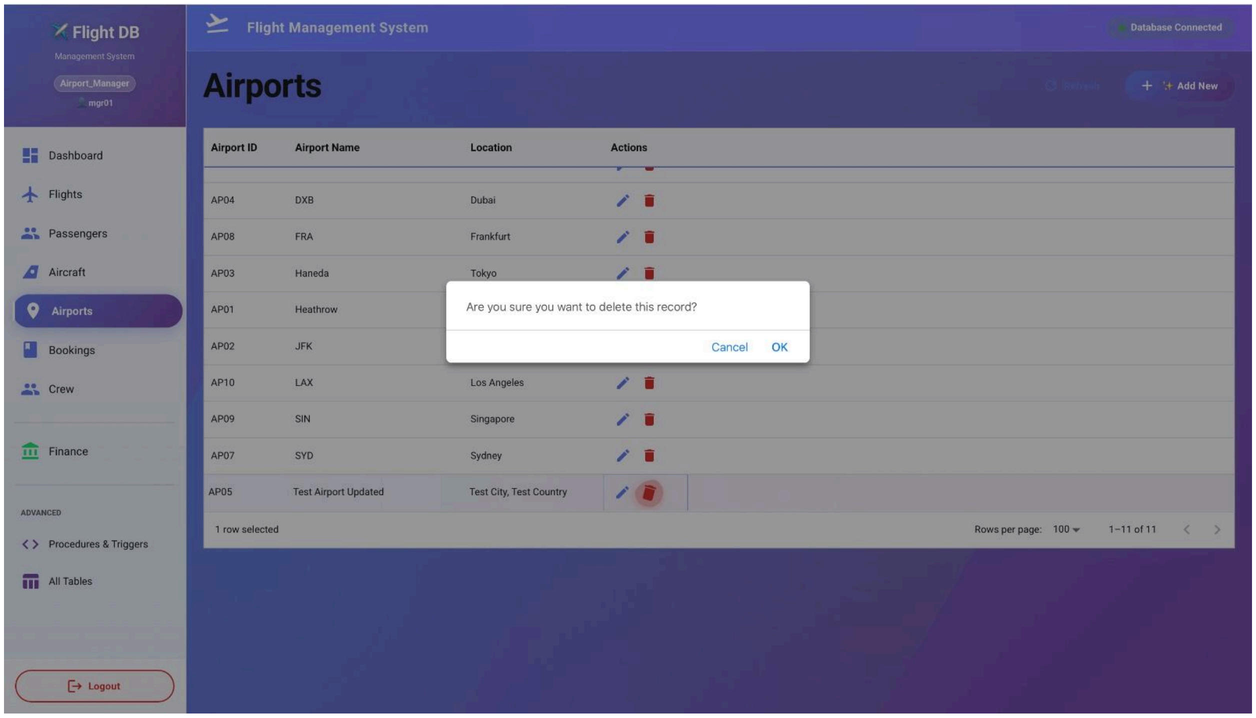
Airport ID *: AP15
ID cannot be changed

Airport Name *: CCU
e.g., John F. Kennedy International Airport

Location *: Kolkata
City, Country (e.g., New York, USA)

Cancel Submit

F6: Delete Records: Users can delete records from any table. Deletions are synchronized with the backend to maintain data consistency.



Airports

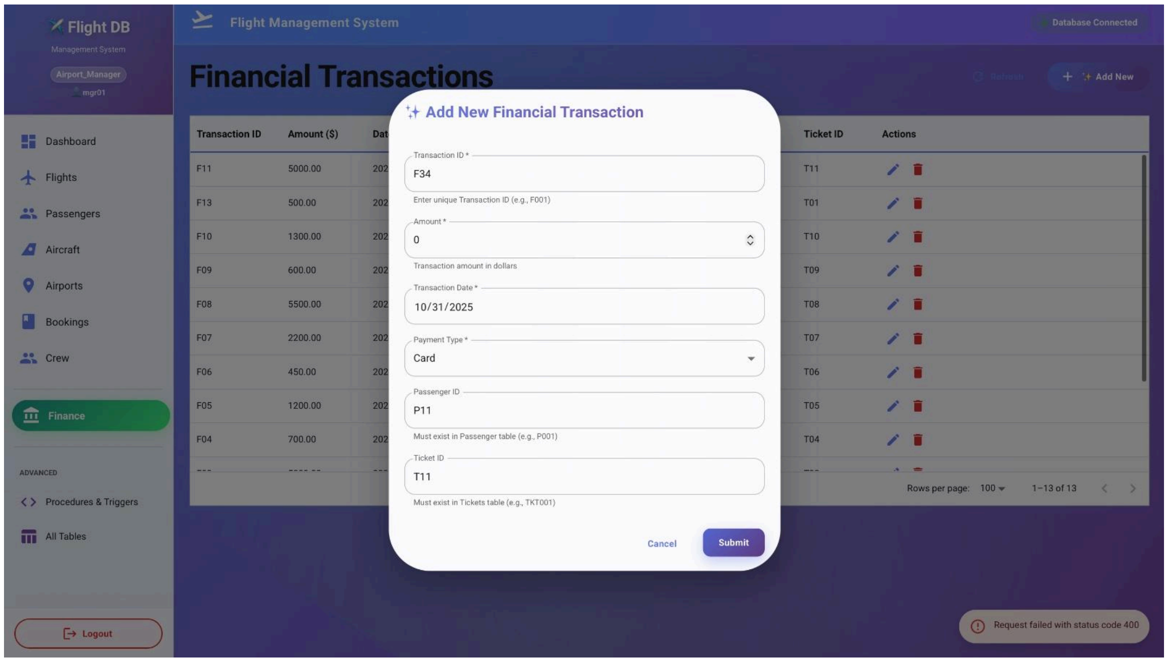
Airport ID	Airport Name	Location	Actions
AP04	DXB	Dubai	
AP08	FRA	Frankfurt	
AP03	Haneda	Tokyo	
AP01	Heathrow		
AP02	JFK		
AP10	LAX	Los Angeles	
AP09	SIN	Singapore	
AP07	SYD	Sydney	
AP05	Test Airport Updated	Test City, Test Country	

1 row selected   Rows per page: 100   1–11 of 11

Are you sure you want to delete this record?

Cancel   OK

F7: Trigger Validation on Transactions: Demonstrates a database trigger in action — an error appears (bottom right) if a transaction amount entered or updated is less than or equal to zero.



Financial Transactions

Transaction ID	Amount ($)	Dat	Ticket ID	Actions
F11	5000.00	202	T11	
F13	500.00	202	T01	
F10	1300.00	202	T10	
F09	600.00	202	T09	
F08	5500.00	202	T08	
F07	2200.00	202	T07	
F06	450.00	202	T06	
F05	1200.00	202	T05	
F04	700.00	202	T04	

Rows per page: 100   1–13 of 13

Add New Financial Transaction

Transaction ID *: F34
Enter unique Transaction ID (e.g., F001)

Amount *: 0
Transaction amount in dollars

Transaction Date *: 10/31/2025

Payment Type *: Card

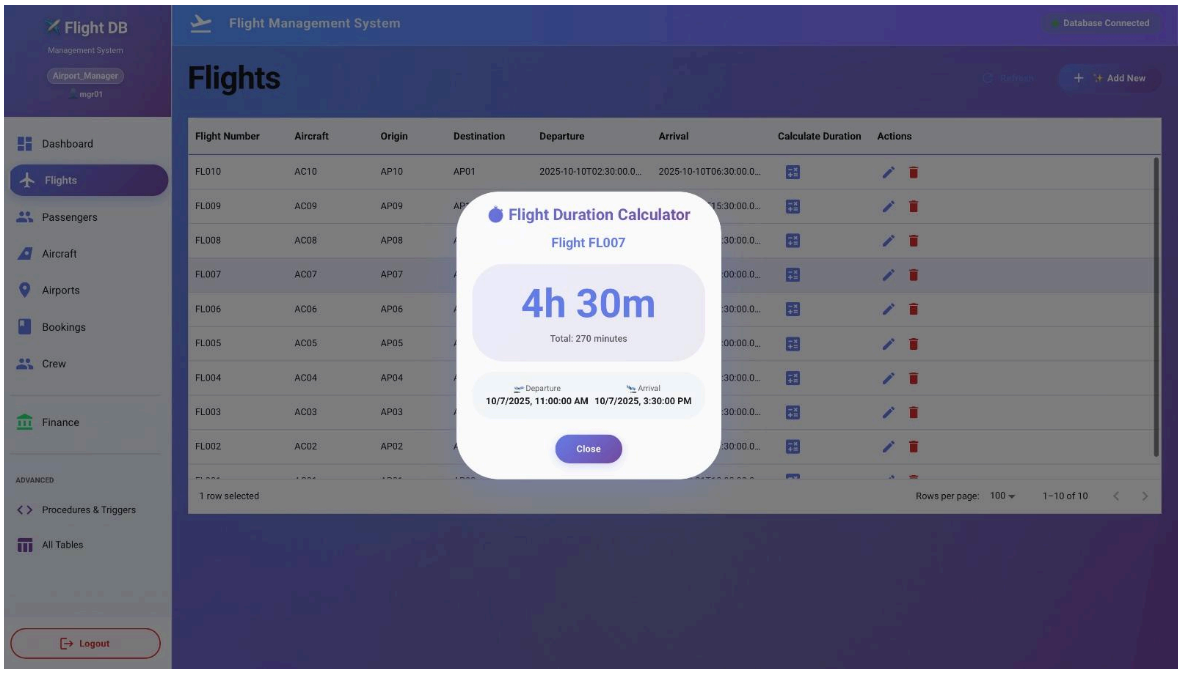
Passenger ID: P11
Must exist in Passenger table (e.g., P001)

Ticket ID: T11
Must exist in Tickets table (e.g., TKT001)

Cancel   Submit

Request failed with status code 400

F8: Automated Flight Duration Update: The function calculates flight duration, while the trigger automatically updates the duration whenever related values change.



Flights

Flight Number	Aircraft	Origin	Destination	Departure	Arrival	Calculate Duration	Actions
FL010	AC10	AP10	AP01	2025-10-10T02:30:00.0...	2025-10-10T06:30:00.0...		
FL009	AC09	AP09	[illegible]	[illegible]	15:30:00.0...		
FL008	AC08	AP08	[illegible]	[illegible]	:30:00.0...		
FL007	AC07	AP07	[illegible]	[illegible]	:00:00.0...		
FL006	AC06	AP06	[illegible]	[illegible]	:30:00.0...		
FL005	AC05	AP05	[illegible]	[illegible]	:00:00.0...		
FL004	AC04	AP04	[illegible]	[illegible]	:30:00.0...		
FL003	AC03	AP03	[illegible]	[illegible]	:30:00.0...		
FL002	AC02	AP02	[illegible]	[illegible]	:30:00.0...		

1 row selected   Rows per page: 100   1–10 of 10

Flight Duration Calculator
Flight FL007
4h 30m
Total: 270 minutes
Departure 10/7/2025, 11:00:00 AM
Arrival 10/7/2025, 3:30:00 PM
Close

10. Triggers, Procedures, Functions, Nested Queries, Joins, Aggregates

Functions:

- GetFlightDuration(flight_no)
- CountPassengerTickets(passenger_id)
- TotalRevenueByPassenger(passenger_id)

Procedures:

- BookTicket()
- AddPayment()
- UpdateFlightTimes()

Triggers:

- trg_UpdateFlightDuration – auto-updates Duration.
- trg_ValidateFinanceAmount – prevents invalid amounts.
- trg_UpdateTicketsAfterInsert/Delete – manages Total_Tickets

11. Code Snippets for Invoking Procedures / Functions / Triggers

```
CALL BookTicket('T12','FL001','P01','Eco','16A','2025-10-02');
CALL AddPayment('F13','P01','T01',500,'2025-10-01','Card');
CALL UpdateFlightTimes('FL001','2025-10-01 11:00:00','2025-10-01 15:30:00');

SELECT GetFlightDuration('FL001');
SELECT TotalRevenueByPassenger('P01');
```

12. SQL File Deliverable

File uploaded in github repository.

13. GITHUB REPOSITORY LINK

https://github.com/srvuthi/Flight-Management-System-DBMS-Project