

**Group Task:**

Syed Muhammad Waiz Rizvi [F2023376117]

Hamza Halim [F2023376113]

**Subject: Database Lab**

**Submitted To: Sir Usama Amjad**

Airline management system

**Database design**

### **Introduction**

In the modern airline industry, the efficient handling of booking, scheduling, aircraft management, and staff coordination is critical. The Airline Management System (AMS) aims to provide a centralized platform to manage these operations through a robust and scalable database design. This document outlines the key entities, relationships, and structural schema required to develop a comprehensive database for AMS, ensuring data consistency, integrity, and optimal performance.

### **Problem Statement**

Airlines often face challenges in managing passenger bookings, flight schedules, aircraft information, staff allocation, and payments due to scattered and inefficient systems. These issues can lead to overbookings, scheduling conflicts, and errors in passenger records. The absence of an integrated system can also hamper customer service and operational efficiency. The primary objective of this project is to design a normalized, relational database model that supports all core airline functions, providing accurate, real-time data management and streamlined operations.

### **Scope**

The scope of the Airline Management System (AMS) database design includes:

* Managing passenger information including personal and contact details.
* Handling flight bookings and associating each booking with a specific flight and passenger.
* Storing flight details, including aircraft used, pilots assigned, and schedules.
* Managing flight schedules with source and destination airports and timing.
* Organizing information about aircrafts, their models, and capacities.
* Recording employee data, including pilots and crew members.
* Capturing payment details linked to specific bookings.
* Providing a structured and normalized database to ensure data integrity, reduce redundancy, and allow easy scalability for future enhancements such as loyalty programs or mobile integration.

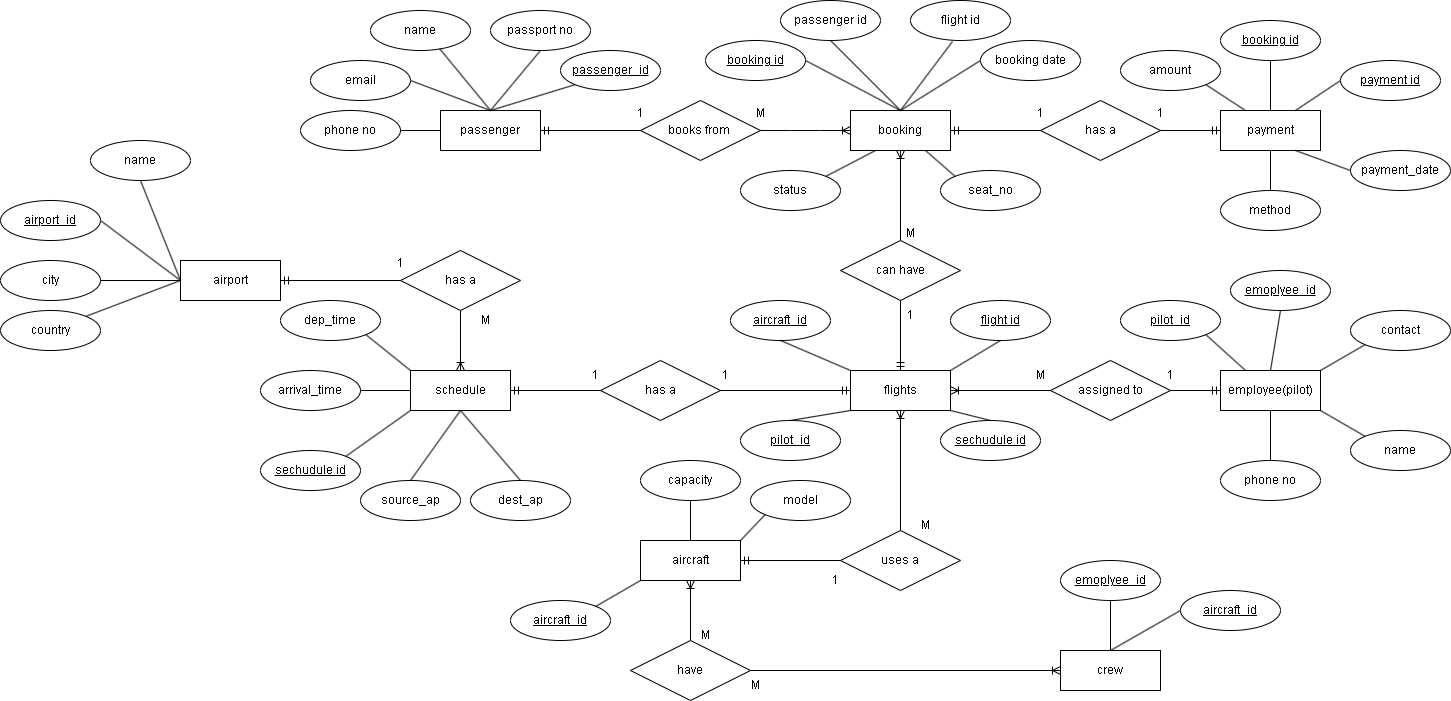
# **Entities:**

| **Entity** | **Primary Key** | **Key Attributes** |
| --- | --- | --- |
| Passenger | passenger\_id | name, email, phone, passport\_no |
| Booking | booking\_id | passenger\_id (FK), flight\_id (FK), seat\_no, status |
| Flight | flight\_id | aircraft\_id (FK), pilot\_id (FK), schedule\_id (FK) |
| Aircraft | aircraft\_id | model, capacity |
| Schedule | schedule\_id | departure\_time, arrival\_time, source\_airport, dest\_airport |
| Airport | airport\_id | name, city, country |
| Employee | employee\_id | name, role, contact |
| Payment | payment\_id | booking\_id (FK), amount, payment\_date, method |

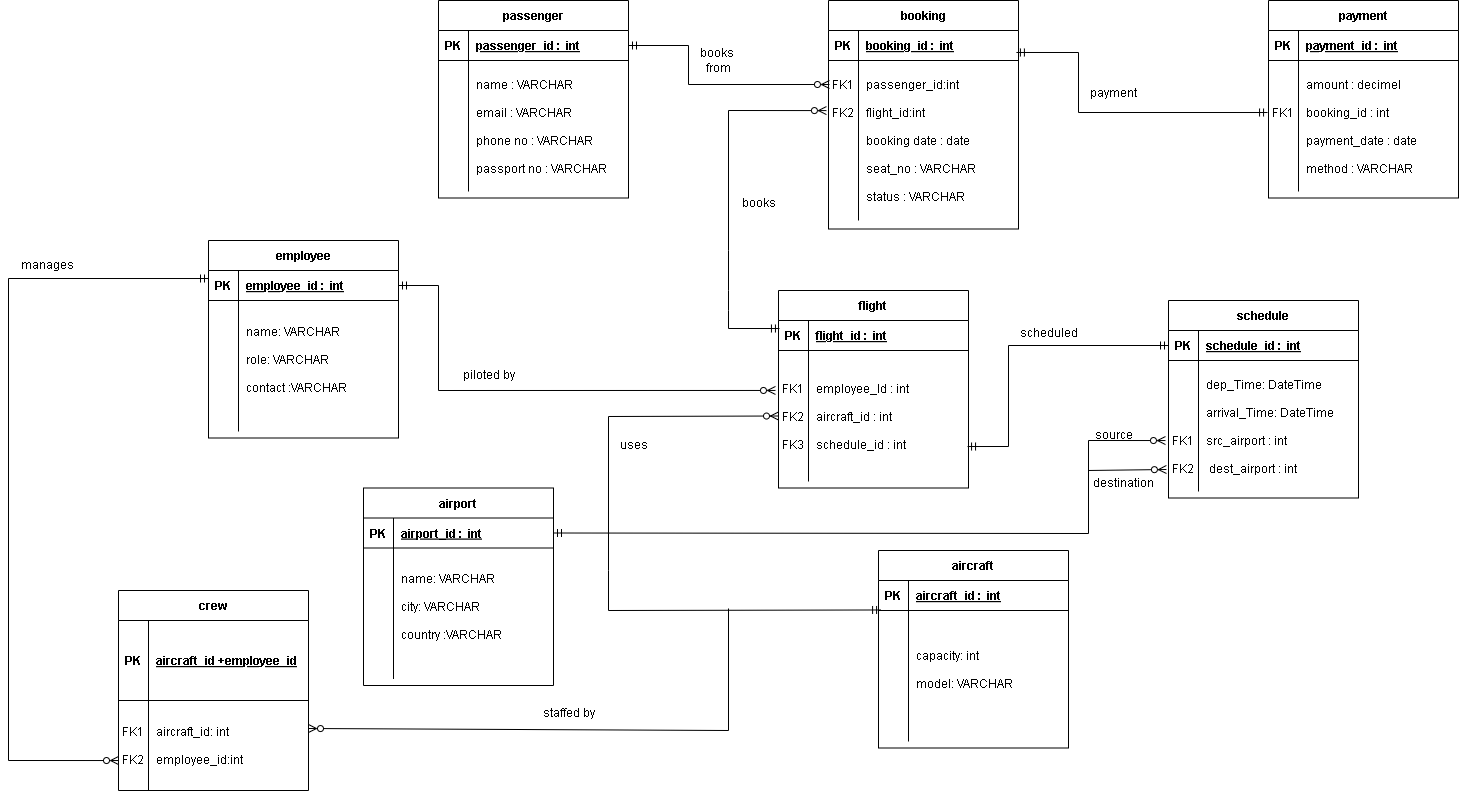
# **Relationship:**

| **Relationship** | **Type** | **Details** |
| --- | --- | --- |
| Passenger – Booking | 1 to many | A passenger can have multiple bookings |
| Booking – Flight | many to 1 | A booking is for one flight |
| Flight – Schedule | 1 to 1 | A flight has one schedule |
| Schedule – Airport (src & dest) | many to 1 (each) | Each schedule has a source and destination |
| Flight – Aircraft | many to 1 | A flight uses one aircraft |
| Flight – Employee (pilot) | many to 1 | A flight is assigned to one pilot |
| Aircraft – Employee (crew) | many to many | Aircraft has multiple crew members |
| Booking – Payment | 1 to 1 (optional) | Each booking has one payment record |

# ERD(entity relationship Diagram):



# Chen view:



### **Conclusion**

This database design for the Airline Management System (AMS) addresses the complexities of managing key airline operations such as passenger handling, flight scheduling, aircraft management, and payment tracking. By defining clear relationships and normalizing data entities, the system ensures efficiency, reliability, and scalability. Implementing this design will significantly enhance operational coordination, reduce redundancy, and provide a solid foundation for future feature expansions in the airline management domain.