(Airline Ticket Reservation System)

FINAL PROJECT REPORT

Presented by:

Thota Sai Navaneeth Kumar- C0820794.

Chaitanya Polamalla- C0824156.

Rakesh Reddy Kotha- C0812619.

Shiva Sai Krishna Kothapalli-C0818766.

Institution:

Lambton College in Toronto.

Instructor:

Mr. Emad Nasrallah.

Course:

Database Design (2303).

Contents

- Introduction
- Existing System
- Proposed System
- Software Purpose Requirements
- Operating Environment
- Hardware requirements
- User characteristics
- Schema diagram
- ER diagram
- Implementation
- Description of tables
- Creation of tables
- Crud operations
- Conclusion
- References

Abstract

Our Project 'Airline Ticket reservation System' is a computerized system used to reserve and get the data and perform transactions related to air travel. The aim of the project is to put forth the relevance and significance of Airline Ticket Reservation Systems.

The system facilitates the passenger to search for flights that are available between the two locations, namely the "Departure city" and "Arrival city" on a specific date. The system is modeled such that flights are available on all days. The system displays all the flight's details such as flight number, name, fare etc.

As the next step the software searches for the seats which are available to be booked. If it finds any then the it allows the passenger to reserve a seat else it asks the user to choose a different flight.

The system asks the customer to enter his details such as name, age, gender, email and contact number to book a flight. The system also allows the customer to cancel his/her booking, if any issues arise.

The primary motive of this software is to reduce the manual errors that occur in the airline reservation process and make it more user friendly for the customers to book the flights as and when they require. The software allows customer to make bookings, make changes to reservations or cancel a particular reservation.

Introduction

Considering the tremendous amounts of data that is required to be tracked and accessed, it would be a laborious task to manage the precision and standard of data manually and deliver them accordingly. It would be well-nigh impossible to retrieve the data required if it is maintained manually. The TWC (Travel with Comfort) is a smart solution that helps in managing enormous loads of flight reservation data. The Airline Ticket Reservation System reduces the complications of the manual work and allows seamless administration of the operations of transportation.

Purpose

This project is designed to significantly cut the manual work associated with the data maintenance in the Flight Booking and automates the Airline Reservation System. The purpose of the project is to computerize the administrative operations of a Flight Booking and to develop software which is user friendly, simple, fast, and cost – effective. It deals with the collection of Users, Employees, Flights and Booking information,

Fare details, etc. conventionally, it was done manually. The main function of the system is to enter and book Flights and retrieve these details as and when required, and also to manipulate these details meaningfully.

Existing System

In few countries if a person wants to reserve a flight ticket, it was necessary for the person to go the Airport and book his ticket, down the ticket in the form of paper document, fill it and submit it at Airport. Then these details are entered into the online form and print the ticket.

In the above process the Passenger may not have much freedom over this approach. A candidate has to travel to airport to make any kind of changes to his /booking making the process more cumbersome and time taking. Hence the Passenger may or may not be satisfied with this process as it includes manual intervention like travelling to Airport for booking his ticket, cannot upload and download the latest updates. It also makes person intervening physically, so this system should be reduced.

Proposed System

The Proposed system ensures the complete freedom for users, where, user himself can login and book the ticket at his own comfort. This proposed system allows only the registered persons to book the tickets, view timings and cancel their tickets. In this system user can reserve a ticket from any location which means the person need not travel to any particular booking facilities to book his/her ticket. This system also facilitates the passengers to send their queries and suggestions using a feedback form. This system is also safe to use with strong password mechanism to protect the user data. Advantages are User friendliness provided in the application with various controls, the system makes the overall project management much easier and flexible and it provides high level of security with different level of authentication.

Software Requirements

The Airline Reservation System operates with a client-server architecture, and as such, must have minimum hardware and software to run the server.

The system is operated by Admins who can add or delete an aircraft or manipulates data using a computer with a HTML compatible browser. The system is also used by the passengers of the Airline, who can book the tickets of his own by using a desktop or computer with a php compatible browser.

The server software runs in a dedicated centralized server hosting center for the Airline database. The scripts and http server run on the server, and require a php interpreter, along with the dependencies for the scripts, as well as the MySQL server.

Hardware Requirements

The Hardware requirements are very minimal and the program can be run on most of the machines.

- Processor - Intel 486/Pentium processor or better
- Processor Speed - 500 MHz or above
- Hard Disk - 20GB (approx.)
- RAM - 64MB or above
- Storage Space - Approx. 2MB

Software Requirements

- Technology Implemented : MySQL Server
- Database : My SQL

User characteristics

• The first user who holds the administrative powers is admin which include:

> adding/deleting aircrafts in the database.

> can view all the records in the database.

• The second user, who can book tickets to a flight sitting at home or from any place is customer, his functions include:

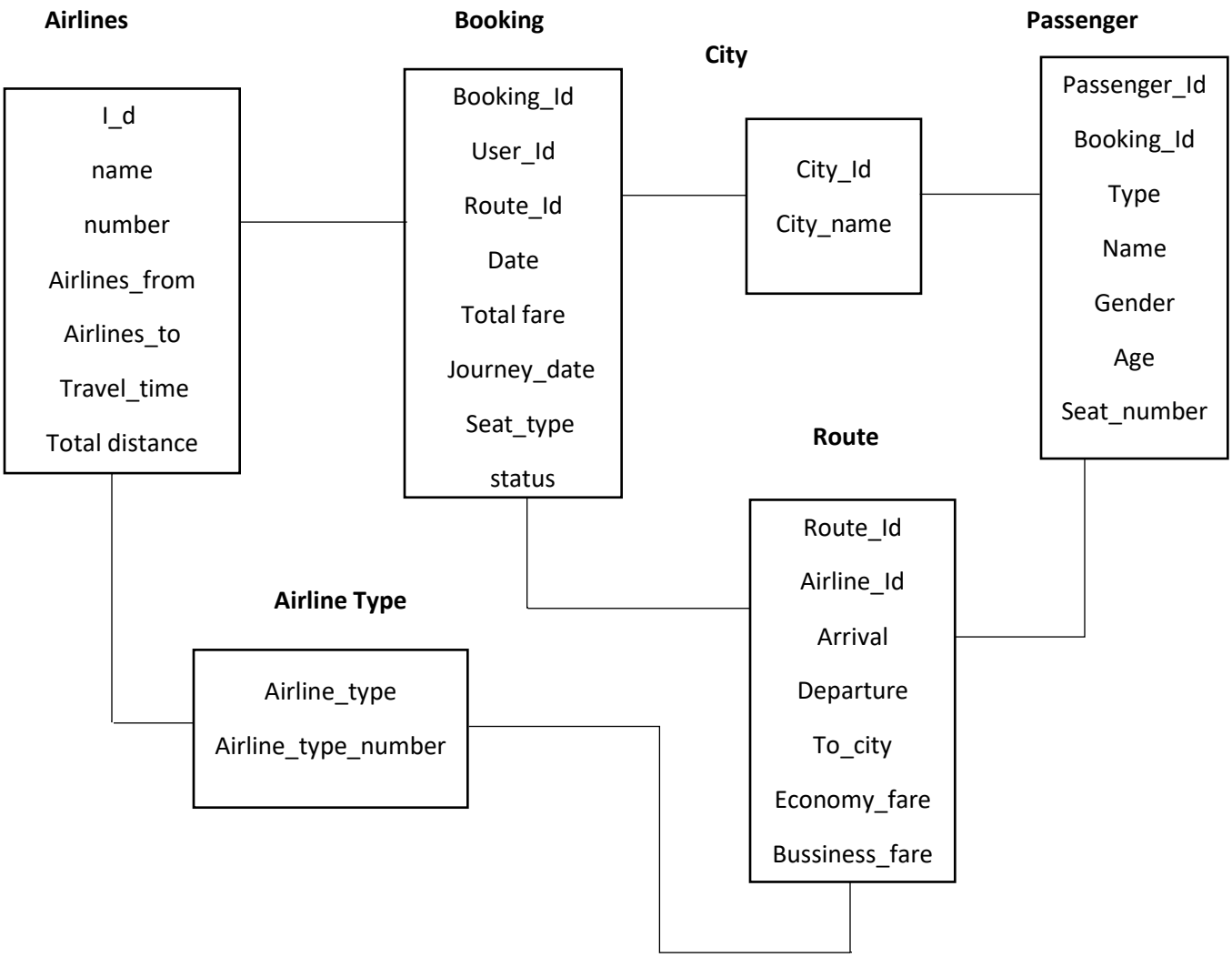
> checking the flights based on the source and destination.

> booking a ticket to fly.

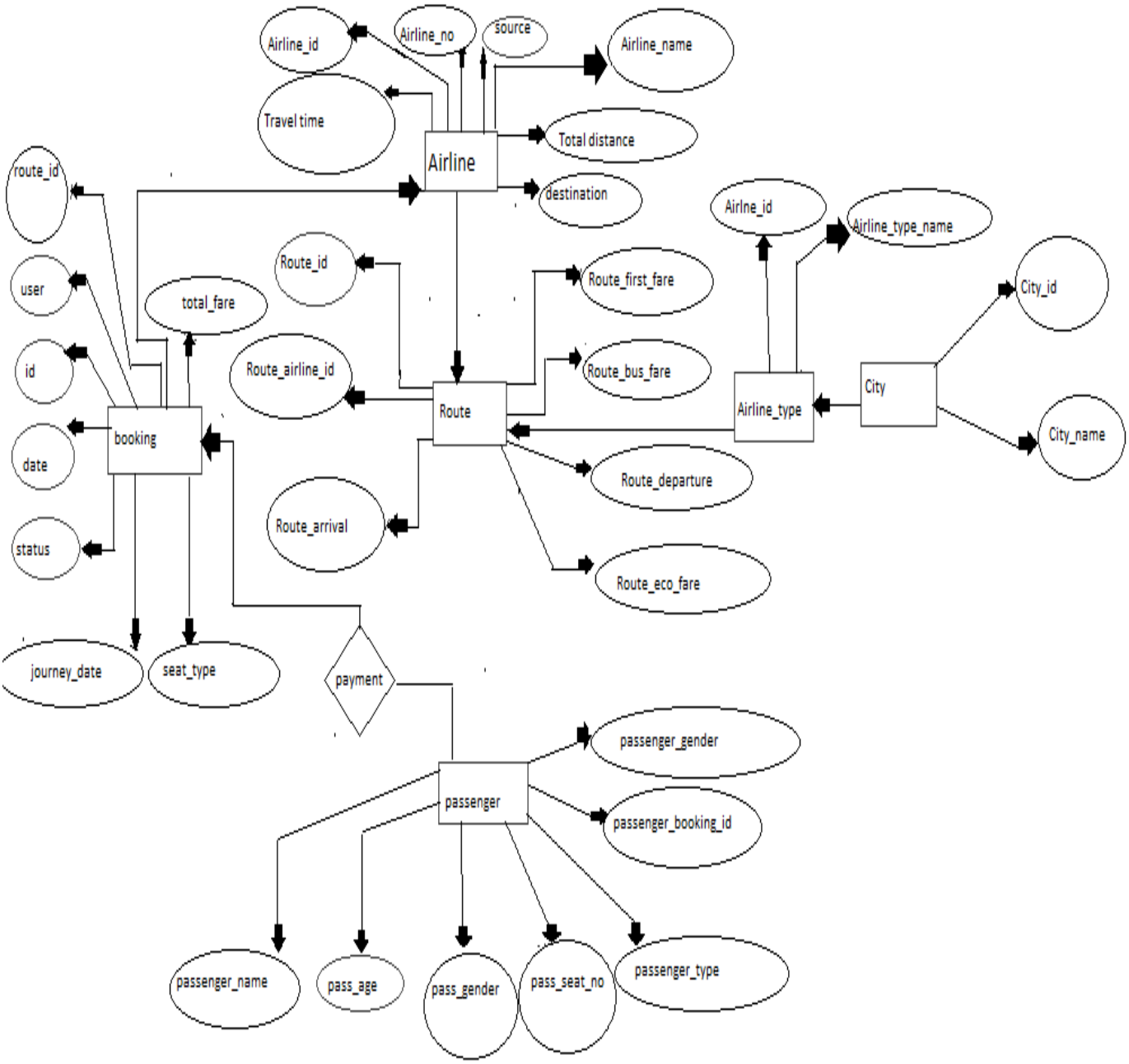
> logging out from his current account.

> viewing his booking which are done in his account.

SCHEMA DIAGRAM



ER DIAGRAM


Airline_id
Airline_no
source
Airline_name
Travel time
Total distance
Airline
destination
route_id
user
total_fare
Route_id
Route_first_fare
id
Route_airline_id
Route_bus_fare
booking
Route
date
Route_departure
status
Route_arrival
Route_eco_fare
journey_date
seat_type
payment
passenger_gender
passenger
passenger_booking_id
passenger_name
pass_age
pass_gender
pass_seat_no
passenger_type
Airlne_id
Airline_type_name
City_id
Airline_type
City
City_name

Description of tables

Airline

Attributes	Data type	Length	Constraints	Comments
Airline_id	Int	20	Not null	Id of airline
Airline_name	Varchar	100	Not null	Name of the airline
Airline_no	Varchar	100	Not null	Number of the airline
Source	Varchar	100	Not null	Source city
Destination	Varchar	100	Not null	Destination city
Travel time	Varchar	100	Not null	Time period of travel
Total distance	Int	20	Not null	Distance travelled

Airline's type

Attributes	Datatype	Length	Constraints	Time	comments
Airline_id	Int	10	Not null	-	Id of airline type
Airline_type_name	Varchar	100	Not null	-	Name of airline type

Booking

Attributes	Datatype	Length	Constraints	Time
Booking_id	Int	10	Not null	-
Booking_user	Varchar	100	Not null	-
Booking_route_id	Varchar	100	Not null	-
Booking_date	Date	-	Not null	'YYYY-MM-DD'
Booking_total_fare	Int	20	Not null	-
Booking_journey_date	Date	-	Not null	'YYYY-MM-DD'
Booking_seat_type	Varchar	100	Not null	-
Booking_status	Varchar	100	Not null	-

City

Attributes	Datatype	Length	Constraints	Time	Comments
City_id	int	10	Not null	-	Id of the city
City_name	Varchar	100	Not null	-	Name of the city

Route

Attributes	**Datatype**	**Length**	**Constraints**	**Time**
Route_id	Int	10	Not null	-
Route_airline_id	Varchar	100	Not null	-
Route_arrival	Varchar	100	Not null	-
Route_departure	Varchar	100	Not null	-
Route_economy_fare	Varchar	100	Not null	-
Route_business_fare	Varchar	100	Not null	-
Route _first_fare	Varchar	100	Not null	-

Passenger

ATTRIBUTES	DATA TYPE	LENGTH	CONSTRAINS	TIME	COMMENTS
Passenger	Int	10	Not null	-	Id of the passenger
Passenger_Booking_Id	Varchar	100	Not null	-	Booking Id of passenger
Passenger_Type	Varchar	100	Not null	-	Type of passenger
Passenger_Name	Varchar	100	Not null	-	Name of passenger
Passenger_Gender	Varchar	100	Not null	-	Gender of passenger
Passenger_Age	Int	10	Not null	-	Age of passenger
Passenger_Seat_no	Int	10	Not null	-	Seat number of passenger

Creation of tables

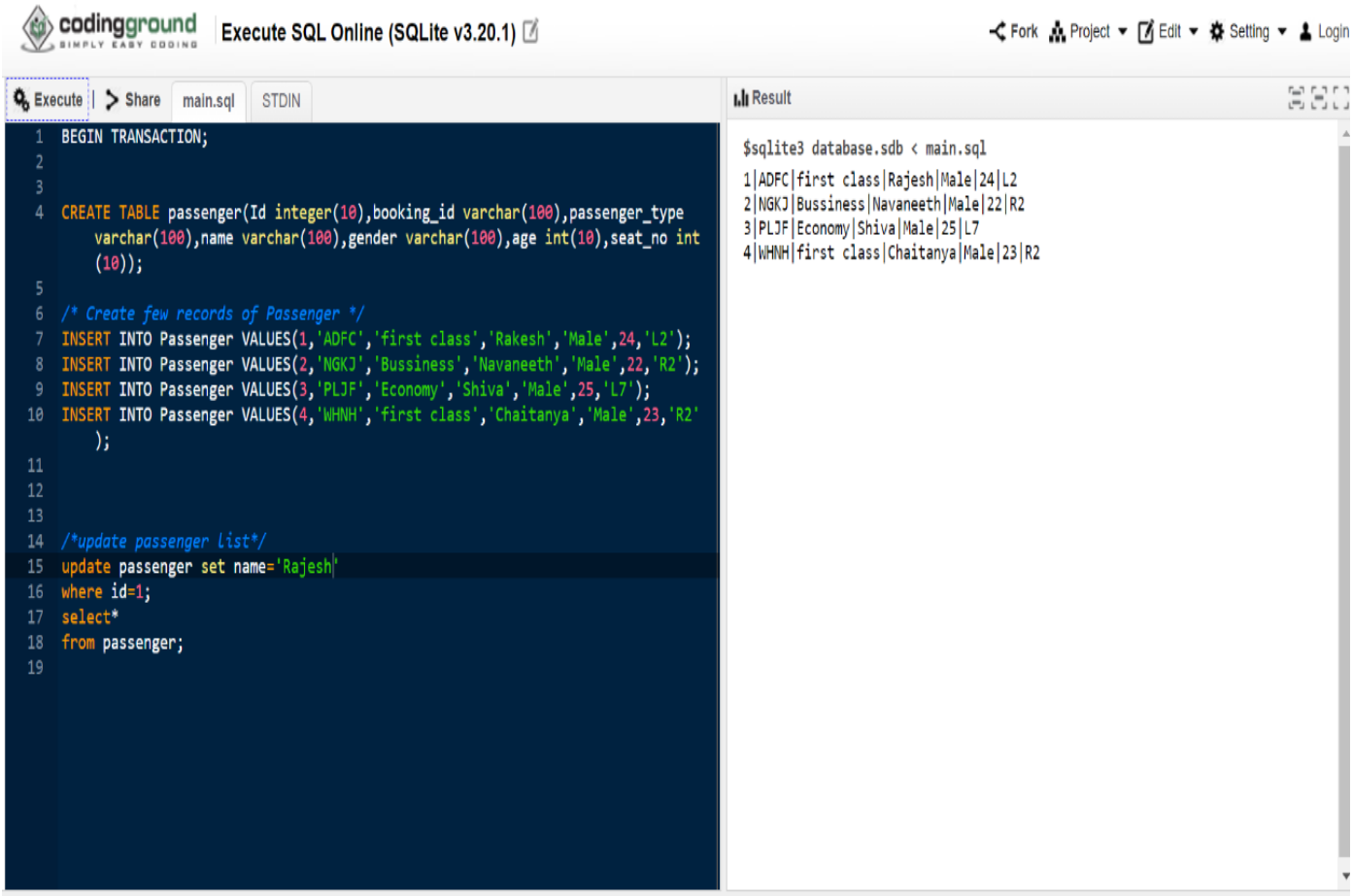
```
CREATE TABLE airlines (
airlines_Id int (10) NOT NULL,
airlines_name Varchar (50) NOT NULL,
airlines_no Int (10) NOT NULL,
airlines_source Varchar (50) NOT NULL,
airlines_destination Varchar (20) NOT NULL,
airlines_travel_time Varchar (30) NOT NULL,
airlines_total_distance int (20) NOT NULL
PRIMARY KEY (airlines_Id)
);


CREATE TABLE airline_type (
airline_type_id int (10) NOT NULL,
airline_type_name varchar (100) NOT NULL
);
CREATE TABLE booking (
booking_id int (10) NOT NULL,
booking_user varchar (100) NOT NULL,
booking_route_id varchar (100) NOT NULL,
booking_date varchar (100) NOT NULL,
booking_total_fare int (10) NOT NULL,
booking_journey_date varchar (100) NOT NULL,
booking_seat_type varchar (100) NOT NULL,
booking_status varchar (100) NOT NULL
);
```

```
CREATE TABLE city (

city_id int (10) NOT NULL,
city_name varchar (100) NOT NULL
);

CREATE TABLE Passenger (
Passenger int (10) NOT NULL,
Passenger_booking_id varchar (100) NOT NULL,
Passenger_type varchar (100) NOT NULL,
Passenger_name varchar (100) NOT NULL,
Passenger_gender varchar (100) NOT NULL,
Passenger_age int(100) NOT NULL,
Passenger_seat_no int (100) NOT NULL
);

CREATE TABLE route (
route_id int (10) NOT NULL,
route_airlines_id varchar (100) NOT NULL,
route_from_arrival varchar (100) NOT NULL,
route_to_city varchar (100) NOT NULL,
route_economy_fare varchar (100) NOT NULL,
route_business_fare varchar (100) NOT NULL
);
```

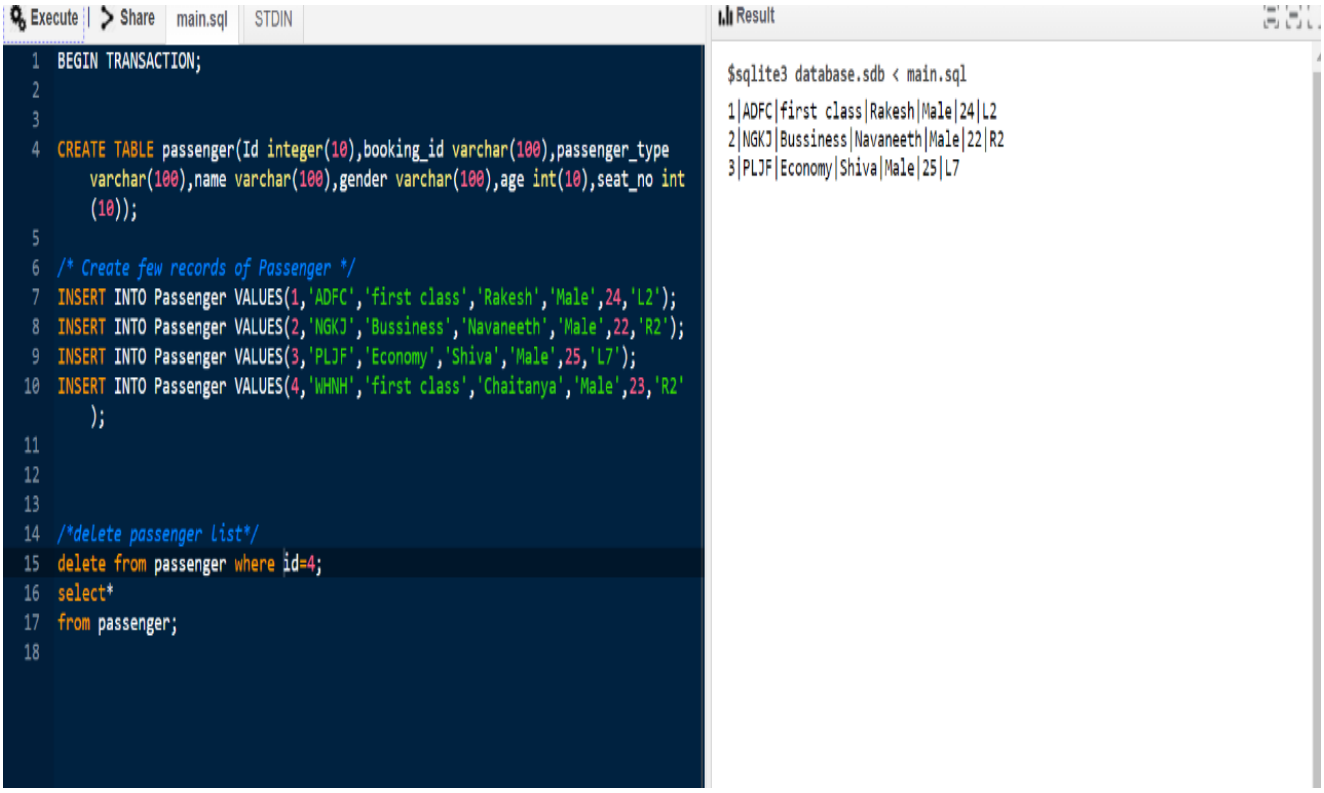
CRUD OPERATIONS

Here we have created the airlines table and performed various operations such as insert, update, delete and select functions on the airlines table.

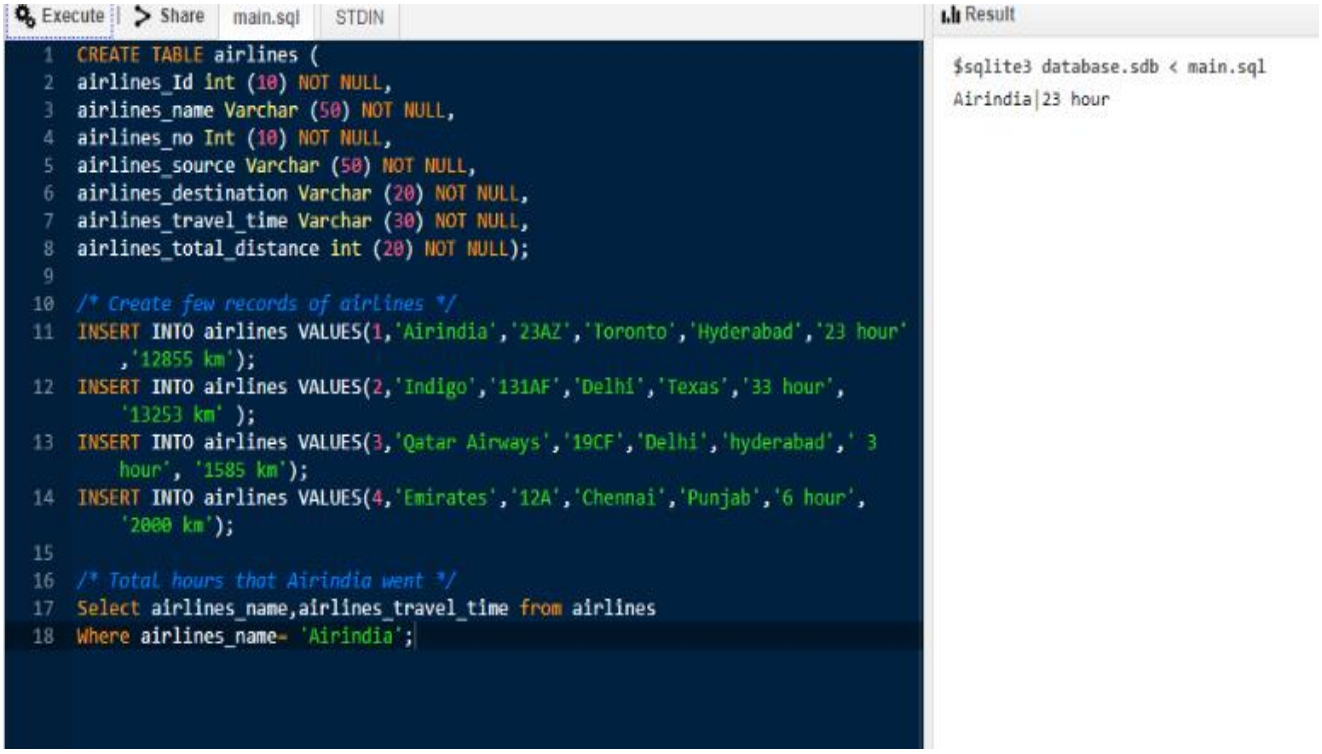
- **UPDATE OPERATIONS**



- **Delete operation**

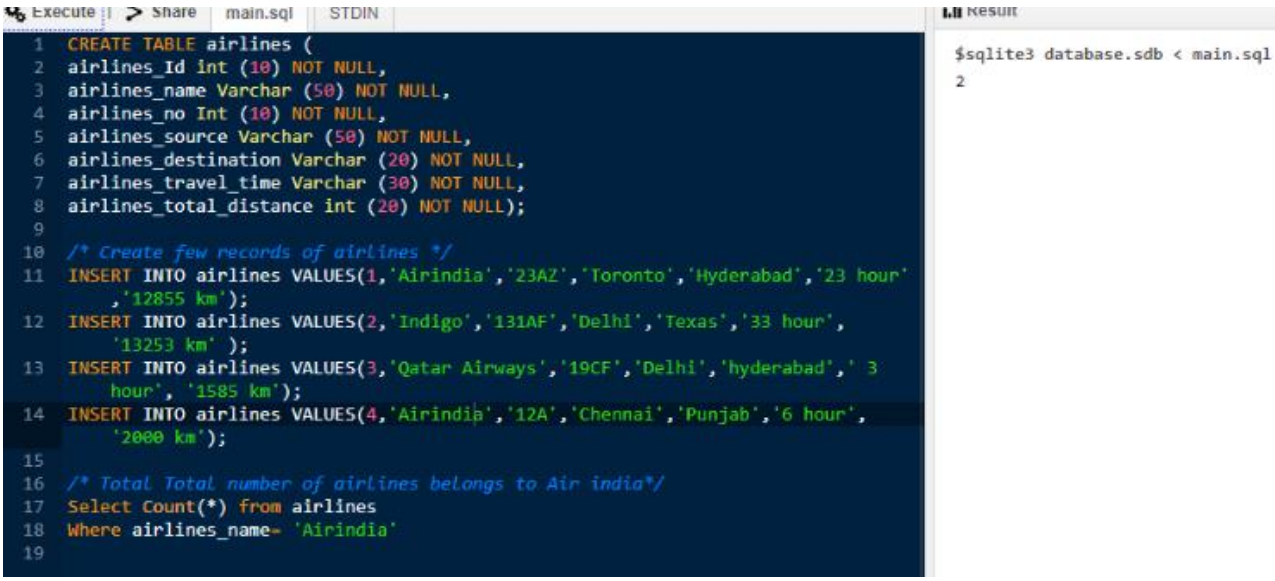


- Total hours that air India spent in Air



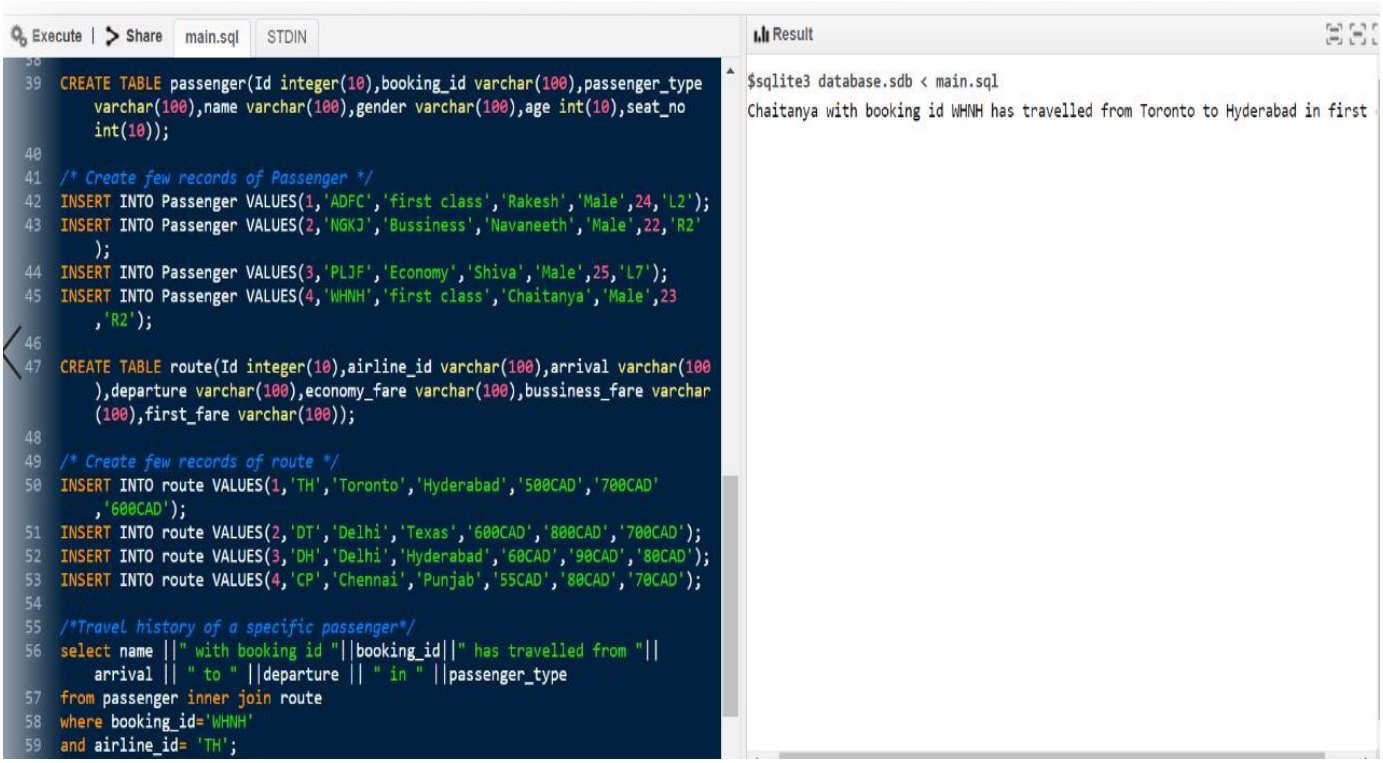

```sql
CREATE TABLE airlines (
airlines_Id int (10) NOT NULL,
airlines_name Varchar (50) NOT NULL,
airlines_no Int (10) NOT NULL,
airlines_source Varchar (50) NOT NULL,
airlines_destination Varchar (20) NOT NULL,
airlines_travel_time Varchar (30) NOT NULL,
airlines_total_distance int (20) NOT NULL);

/* Create few records of airlines */
INSERT INTO airlines VALUES(1,'Airindia','23AZ','Toronto','Hyderabad','23 hour'
    ,'12855 km');
INSERT INTO airlines VALUES(2,'Indigo','131AF','Delhi','Texas','33 hour',
    '13253 km' );
INSERT INTO airlines VALUES(3,'Qatar Airways','19CF','Delhi','hyderabad',' 3
    hour', '1585 km');
INSERT INTO airlines VALUES(4,'Emirates','12A','Chennai','Punjab','6 hour',
    '2000 km');

/* Total hours that Airindia went */
Select airlines_name,airlines_travel_time from airlines
Where airlines_name= 'Airindia';
```

- Total number of airlines belongs to air India




```
CREATE TABLE airlines (
airlines_Id int (10) NOT NULL,
airlines_name Varchar (50) NOT NULL,
airlines_no Int (10) NOT NULL,
airlines_source Varchar (50) NOT NULL,
airlines_destination Varchar (20) NOT NULL,
airlines_travel_time Varchar (30) NOT NULL,
airlines_total_distance int (20) NOT NULL);

/* Create few records of airlines */
INSERT INTO airlines VALUES(1,'Airindia','23AZ','Toronto','Hyderabad','23 hour'
    ,'12855 km');
INSERT INTO airlines VALUES(2,'Indigo','131AF','Delhi','Texas','33 hour',
    '13253 km' );
INSERT INTO airlines VALUES(3,'Qatar Airways','19CF','Delhi','hyderabad',' 3
    hour', '1585 km');
INSERT INTO airlines VALUES(4,'Airindia','12A','Chennai','Punjab','6 hour',
    '2000 km');

/* Total Total number of airlines belongs to Air india*/
Select Count(*) from airlines
Where airlines_name= 'Airindia'
```

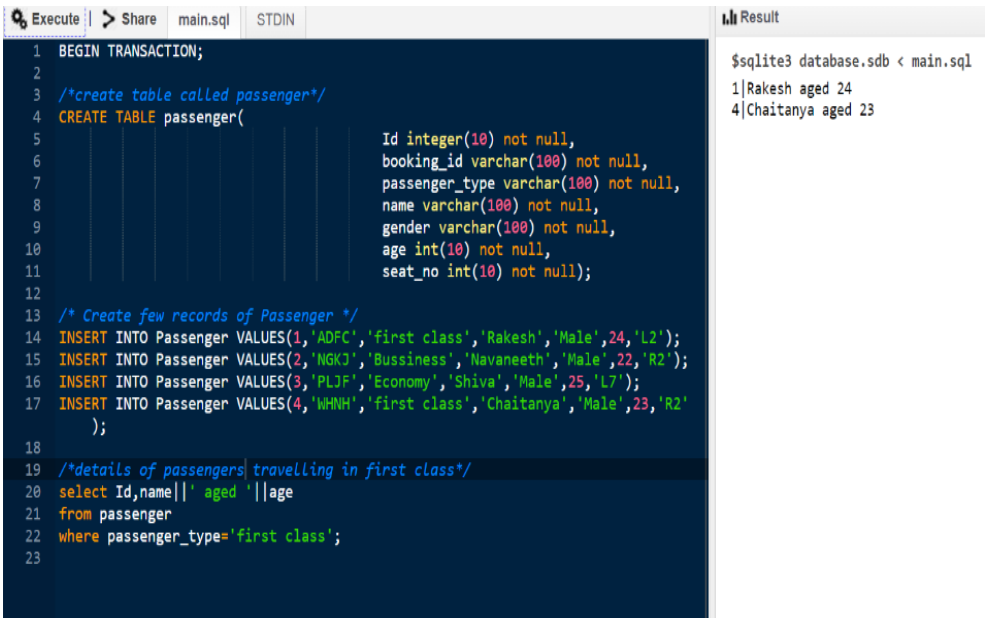
- By creating few records of routes and finding the travel history of specific passenger




```
CREATE TABLE passenger(Id integer(10),booking_id varchar(100),passenger_type
    varchar(100),name varchar(100),gender varchar(100),age int(10),seat_no
    int(10));

/* Create few records of Passenger */
INSERT INTO Passenger VALUES(1,'ADFC','first class','Rakesh','Male',24,'L2');
INSERT INTO Passenger VALUES(2,'NGKJ','Bussiness','Navaneeth','Male',22,'R2'
    );
INSERT INTO Passenger VALUES(3,'PLJF','Economy','Shiva','Male',25,'L7');
INSERT INTO Passenger VALUES(4,'WHNH','first class','Chaitanya','Male',23
    ,'R2');

CREATE TABLE route(Id integer(10),airline_id varchar(100),arrival varchar(100
    ),departure varchar(100),economy_fare varchar(100),bussiness_fare varchar
    (100),first_fare varchar(100));

/* Create few records of route */
INSERT INTO route VALUES(1,'TH','Toronto','Hyderabad','500CAD','700CAD'
    ,'600CAD');
INSERT INTO route VALUES(2,'DT','Delhi','Texas','600CAD','800CAD','700CAD');
INSERT INTO route VALUES(3,'DH','Delhi','Hyderabad','60CAD','90CAD','80CAD');
INSERT INTO route VALUES(4,'CP','Chennai','Punjab','55CAD','80CAD','70CAD');

/*Travel history of a specific passenger*/
select name ||" with booking id "||booking_id||" has travelled from "||
    arrival || " to " ||departure || " in " ||passenger_type
from passenger inner join route
where booking_id='WHNH'
and airline_id= 'TH';
```

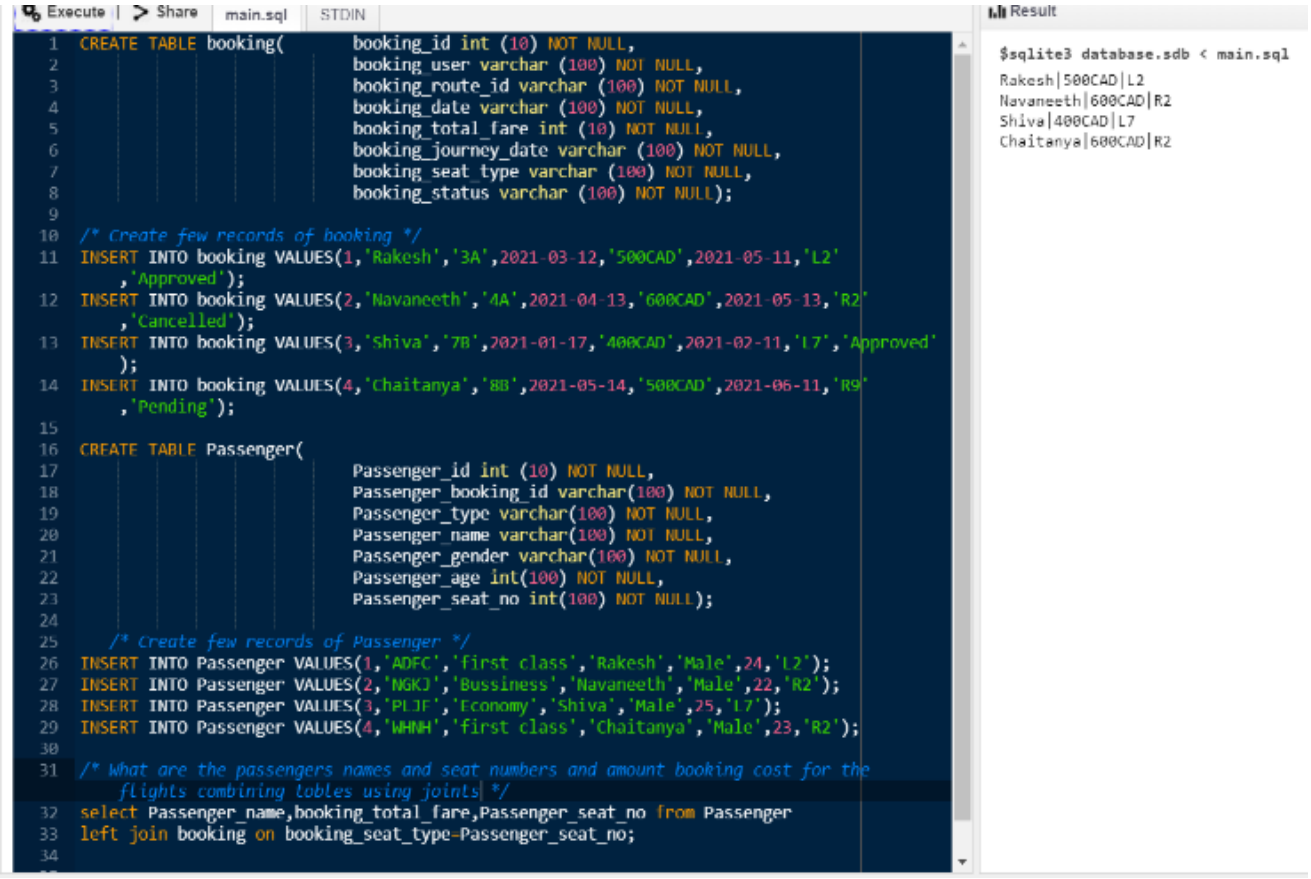
- Creating few records of route and finding what are the route airline id’s which are from Delhi

- Details of the passengers travelling in the first class



- Showing fare price for business class in different cities

- Operation using the joints




```
CREATE TABLE booking(       booking_id int (10) NOT NULL,
                            booking_user varchar (100) NOT NULL,
                            booking_route_id varchar (100) NOT NULL,
                            booking_date varchar (100) NOT NULL,
                            booking_total_fare int (10) NOT NULL,
                            booking_journey_date varchar (100) NOT NULL,
                            booking_seat type varchar (100) NOT NULL,
                            booking_status varchar (100) NOT NULL);

/* Create few records of booking */
INSERT INTO booking VALUES(1,'Rakesh','3A',2021-03-12,'500CAD',2021-05-11,'L2'
    ,'Approved');
INSERT INTO booking VALUES(2,'Navaneeth','4A',2021-04-13,'600CAD',2021-05-13,'R2'
    ,'Cancelled');
INSERT INTO booking VALUES(3,'Shiva','7B',2021-01-17,'400CAD',2021-02-11,'L7','Approved'
    );
INSERT INTO booking VALUES(4,'Chaitanya','8B',2021-05-14,'500CAD',2021-06-11,'R9'
    ,'Pending');

CREATE TABLE Passenger(
                            Passenger_id int (10) NOT NULL,
                            Passenger_booking_id varchar(100) NOT NULL,
                            Passenger_type varchar(100) NOT NULL,
                            Passenger_name varchar(100) NOT NULL,
                            Passenger_gender varchar(100) NOT NULL,
                            Passenger_age int(100) NOT NULL,
                            Passenger_seat_no int(100) NOT NULL);

   /* Create few records of Passenger */
INSERT INTO Passenger VALUES(1,'ADFC','first class','Rakesh','Male',24,'L2');
INSERT INTO Passenger VALUES(2,'NGKJ','Bussiness','Navaneeth','Male',22,'R2');
INSERT INTO Passenger VALUES(3,'PLJF','Economy','Shiva','Male',25,'L7');
INSERT INTO Passenger VALUES(4,'WHNH','first class','Chaitanya','Male',23,'R2');

/* What are the passengers names and seat numbers and amount booking cost for the
    flights combining tables using joints */
select Passenger_name,booking_total_fare,Passenger_seat_no from Passenger
left join booking on booking_seat_type=Passenger_seat_no;
```

- Displaying the details of the passenger travelling on specific date

CONCLUSION

In Conclusion, we have collected the information from the idea and created an Entity Relationship Diagram. Relational Schema was mapped from the ERD and Normalisation check was done and we found all the tables are already normalised. We have created data description tables and wrote SQL statements for the tables.

REFERENCE

https://www.tutorialspoint.com/execute_sql_online.php – SQL online editor is used as a tool to execute SQL statements.

https://apastyle.apa.org/style-grammar-guidelines/paper-format – American Psychological Association (APA) format is used to prepare this documentation.

https://erdplus.com/standalone – we have used this website to design ER diagram.

https://app.diagrams.net/ – We used the DRAW.io website for mapping the ER diagram into the relation schema.