S.R.K.R. ENGINEERING COLLEGE (A), BHIMAVARAM

Department of Computer Science and Engineering

2/4 B. Tech. (CSE) 2nd Semester 2020-2021

DBMS Application Development Report on

AIRLINES RESERVATION SYSTEM

Submitted by

Name:Pulaparthi Jhansi

Regd. No.:**19B91A05K7**

Section: D

Abstract

The main aim of this project named ‘Airlines reservation System’ is a computerized system used to store and retrieve information and conduct transactions related to air travel. The aim the project is to expose the relevance and importance of Airline Reservation Systems.

The system allows the airline passenger to search for flights that are available between the two travel cities, namely the “Departure city” and “Arrival city” for a particular departure date. The system is designed such that flights are available on all days. The system displays all the flight’s details such as flight no, name, price etc.

Then the system checks for the availability of seats on the flight. If the seats are available then the system allows the passenger to book a seat. Otherwise it asks the user to choose another flight. The system asks the customer to enter his details such as name, age, email and contact number to book a flight. The system also allows the customer to cancel his/her reservation, if any problem occurs.

The main purpose of this software is to reduce the manual errors involved in the airline reservation process and make it convenient for the customers to book the flights as and when they require. The software allows customer to make reservations, modify reservations or cancel a particular reservation.

Table of Contents

List of Figures

List of Tables

1. Functional Requirements

The proposed application “Airlines Reservation System” is designed to provide the following major functions.

- The project allows login of two entities, Admin and Customer
- The Admin has a set of functionalities described as:
  - ✦ He can login into the system.
  - ✦ He can log out from his current account.
  - ✦ He can add/delete aircrafts in the database.
  - ✦ He can view all the records of all the bookings that have been made.
- The Customer has a set of functionalities described as:
  - ✦ He can sign up in the system using his credentials.
  - ✦ He can login in to the system using his username and password.
  - ✦ He can check the flights based on the source and destination.
  - ✦ He can book a flight.
  - ✦ He can log out from his current account.
  - ✦ He can view all his bookings.

2. Conceptual DB Design

2.1. Entities identified with attributes and explanation

The proposed application “Airlines Reservation System”involves the following entities. The entities are identified by noun phrases approach.

a) Admin -which stores information about the Airline Administrators who can make changes to the schedule and so on.

b) Customer - which stores all the information about the customer of the airline company.

c) Record - this table contains all the information about all the bookings made by the customers for individual aircrafts.

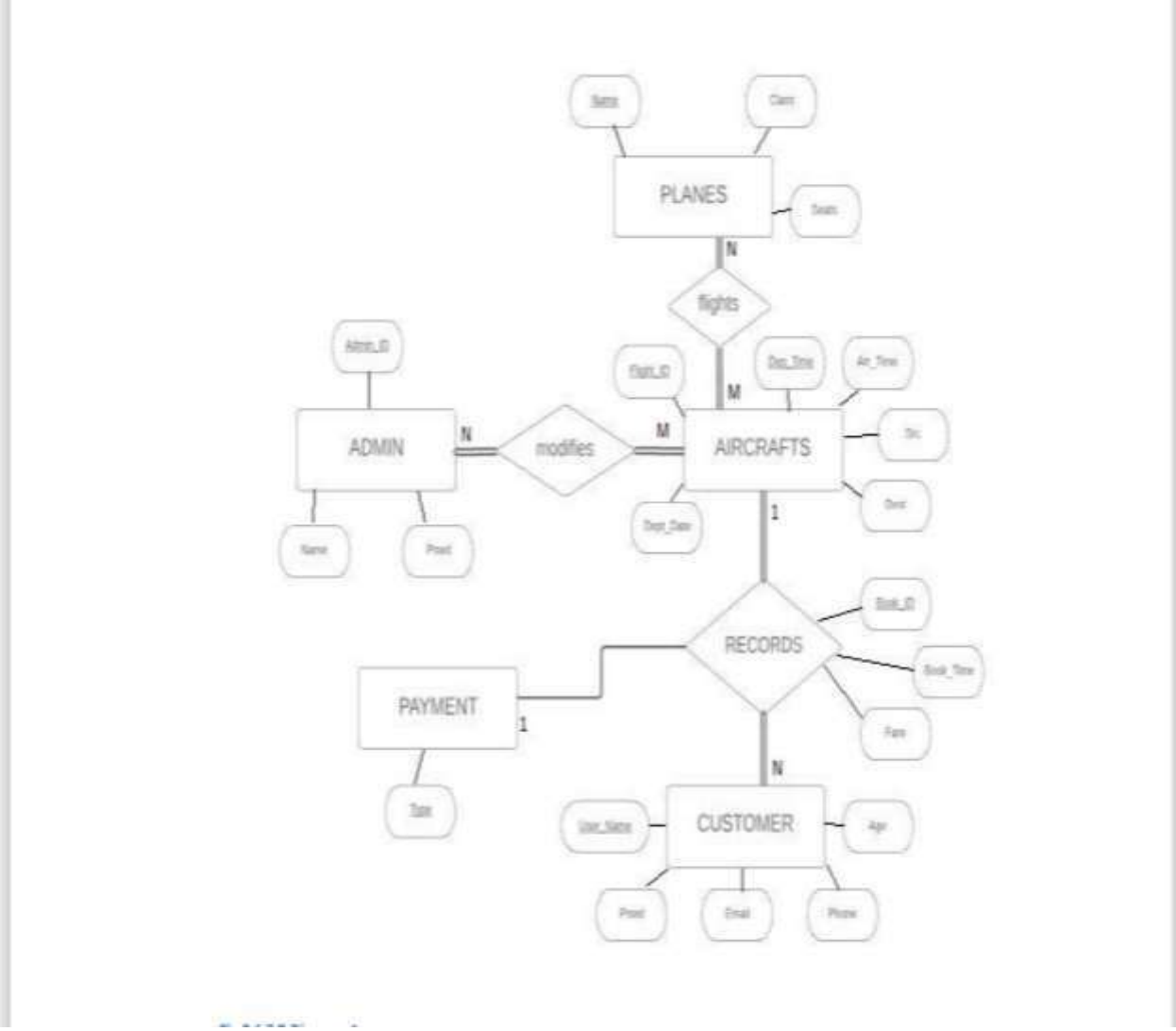
d) Plane - which stores information about all the planes that the airline owns which ply between different cities under different aircraft_id.

e) Aircraft - The planes when fly between one set of cities it's aircraft_id varies on it's to and fro journey.

f) Payment - which store information about the type of payment method used.

2.2. ER diagram with constraints and explanation

The ER diagram with constraints for the proposed application is as follows.



3. Logical DB Design

This Schema tells about how the relation among

3.1 . Relational Schema

This schema tells about how the relation among attributes will be taken. Like it describes which integrity constraint comes under primary key and which constraint comes under Foreign Key and how they should be used in tables. Here are the Relational Schema's taken by considering the ER-Diagram.

3.1.1 Admin

- Admin_id
- Name
- Pswd

3.1.2 Customer

- User_Name
- Pswd
- Email
- Phone
- Age

3.1.3 Aircraft

- Flight_id
- Dep_Time
- Arr_Time
- Plane_Name
- Src
- Dstn
- Fare
- Dep_date

3.1.4 Records

- Book_ID
- Flight_ID
- Dep_Time
- Book_Time

- User_Name
- Payment_Type
- Fare

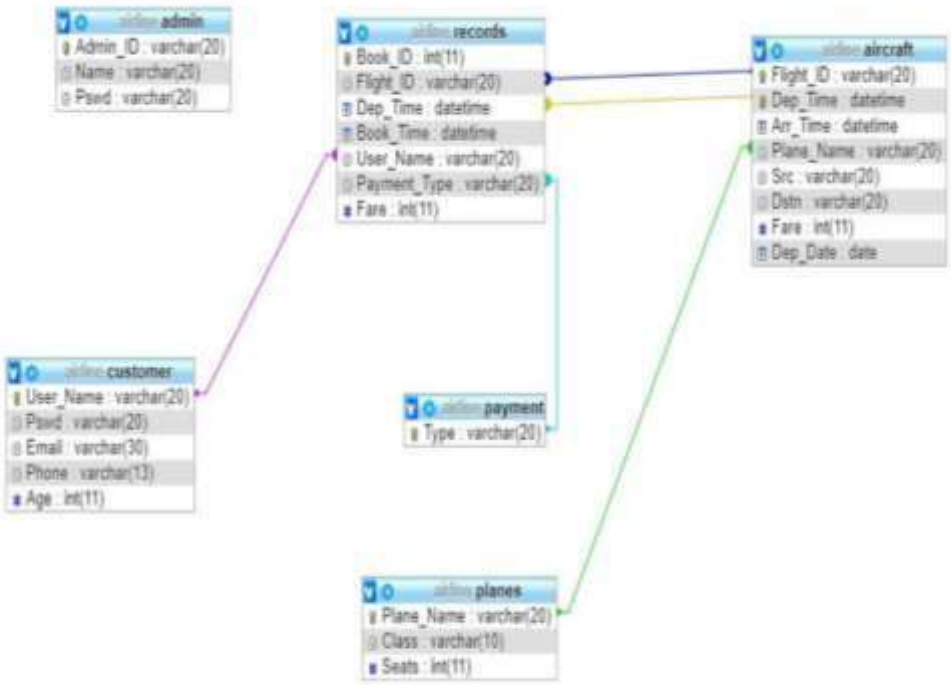
3.1.5 Payment

- Type

3.1.6 Planes

- Plane_Name
- Class
- Seats

3.2 . Relational schema diagram with constraints and explanation



3. Normalization

4.1. Decomposition into BCNF

Admin Table:

Admin_id	Name	pswd

The table has unique columns and atomized cells.

So, it is in 1NF.

The table is in 1NF and does not have partial dependency.

So, it is in 2NF.

The table is in 2NF and does not have transitive dependency.

So, it is in 3NF.

The table is in 3NF and prime attributes are not derived from the non-prime attributes.

So, it is in BCNF.

Customer_Table:

User_Name	Pswd	Email	Phone	Age

The table has unique columns and atomized cells.

So, it is in 1NF.

The table is in 1NF and does not have partial dependency.

So, it is in 2NF.

The table is in 2NF and does not have transitive dependency.

So, it is in 3NF.

The table is in 3NF and prime attributes are not derived from the n

So, it is in BCNF.

Aircraft_Table:

ɔn- rime at ɾibutes.

Flight_ID	Dep_Time	Arr_Time	Plane_Name	Src	Dstn	Fare	Dept_date

The table has unique columns and atomized cells.

So, it is in 1NF.

The table is in 1NF and does not have partial dependency.

So, it is in 2NF.

The table is in 2NF and does not have transitive dependency.

So, it is in 3NF.

The table is in 3NF and prime attributes are not derived from the non-prime attributes.

So, it is in BCNF.

Records_Table:

Book_ID	Flight_ID	Dep_Time	Book_Time	User_Name	Payment_Type	Fare

The table has unique columns and atomized cells.

So, it is in 1NF.

The table is in 1NF and does not have partial dependency.

So, it is in 2NF.

The table is in 2NF and does not have transitive dependency.

So, it is in 3NF.

The table is in 3NF and prime attributes are not derived from the non-prime attributes.

So, it is in BCNF.

Payment_Table:

Type

The table has unique columns and atomized cells.

So, it is in 1NF.

The table is in 1NF and does not have partial dependency.

So, it is in 2NF.

The table is in 2NF and does not have transitive dependency.

So, it is in 3NF.

The table is in 3NF and prime attributes are not derived from the non-prime attributes.

So, it is in BCNF.

Planes_Table:

Plane_Name	Class	Seats

The table has unique columns and atomized cells.

So, it is in 1NF.

The table is in 1NF and does not have partial dependency.

So, it is in 2NF.

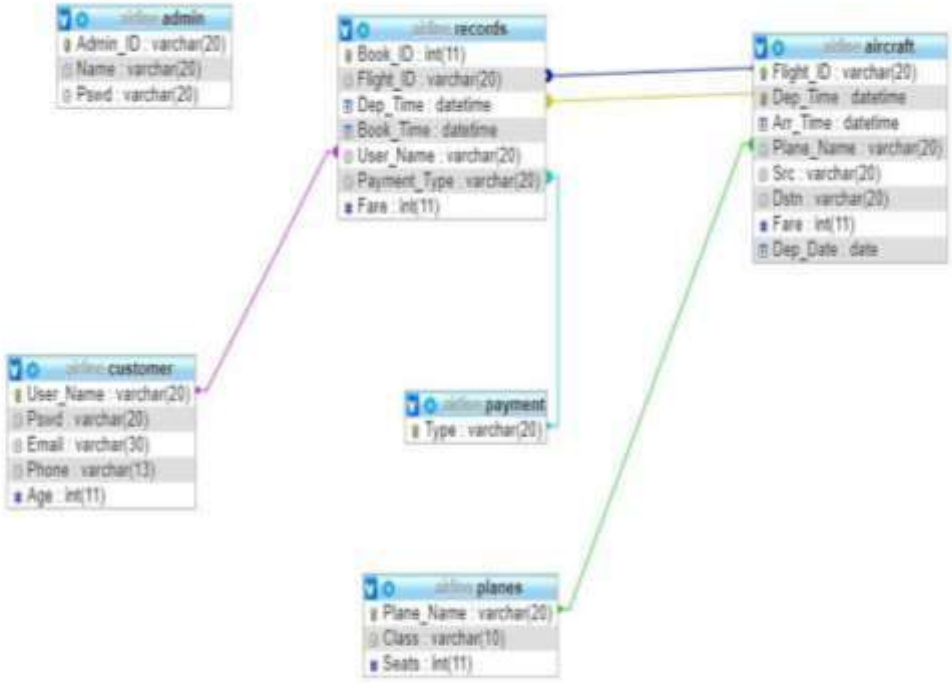
The table is in 2NF and does not have transitive dependency.

So, it is in 3NF.

The table is in 3NF and prime attributes are not derived from the non-prime attributes.

So, it is in BCNF.

4.2. Relational schema diagram after refinement:



4. Database Definition and Construction

5.1. Database definition with constraints and insert commands:

5.1.1. Admin table create

database Airline;

create table Admin(Admin_ID varchar(20) primary key,

Name varchar(20),

Pswd varchar(20));

5.1.2. Payment table create table Payment(Type

varchar(20) primary key);

5.1.3. Planes table create table Planes(Plane_Name

varchar(20) primary key, Class varchar(10), Seats int);

5.1.4. Aircraft table create table Aircraft(

Flight_ID varchar(20),

Dep_TimeDateTime,

Arr_TimeDateTime, Plane_Name varchar(20), foreign key(Plane_Name)

references Planes(Plane_Name) on update cascade on delete cascade, Src

varchar(20), Dstn varchar(20), Fare int, Primary key (Flight_ID, Dep_Time));

5.1.5. Customer table create table Customer(User_Name

varchar(20) primary key,

Pswd varchar(20),

Email varchar(30),

Phone varchar(13), Age int);

5.1.6. Records table create table Records(Book_ID

varchar(20) primary key, Flight_ID varchar(20),

foreign key(Flight_ID) references Aircraft(Flight_ID) on update cascade,

Book_TimeDateTime,

User_Name varchar(20), foreign key(User_Name) references

Customer(User_Name) on delete cascade on update cascade,

Payment_Type varchar(20), foreign key(Payment_Type) references

Payment(Type) on update cascade);

5.2. User creation with authentication and authorization

Trigger used:

```
DELIMITER $$
CREATE TRIGGER Before_Insert_User
BEFORE INSERT ON customer
FOR EACH ROW
BEGIN
IF (EXISTS(SELECT 1 FROM customer WHERE User_Name =
NEW.User_Name)) THEN
SIGNAL SQLSTATE VALUE '45000' SET
MESSAGE_TEXT = 'INSERT failed due to duplicate  mobile
number';
END IF;
END$$
DELIMITER ;
```

The purpose of the trigger is to prevent a existing customer from creating new duplicate account by checking their mobile numbers entered while sign up.

Procedure used:

```
DELIMITER //
CREATE PROCEDURE insert_cust (IN name varchar(20),IN pass
varchar(20),IN email varchar(30) ,IN phone varchar(13), ,IN age int)  BEGIN
insert into Customer(User_Name,Pswd,Email,Phone,Age)  values
(name,pass,email,phone,age);
END//
DELIMITER;
```

Appendix A: Tables with data

- **Admin**

Admin_id	Name	Pswd
5k7	Jhansi	abcdef
5k8	Hema	123456
5n1	Sudha	246800

- **Customer**

User_Name	Pswd	Email	Phone	Age
Rocky	Hjhdfh	rocky123@gmail.com	6789134556	25
Jimmy	Mfkjdu	jimmy456@gmail.com	7896467644	30
Blacky	nmdhj	blacky789@gmail.com	7837838755	19

- **Planes**

Plane_Name	Class	Seats
Indigo	Economy	200
Fighting Falcon	Economy	150
The Voyager	Business	5

- **Aircrafts**

Flight_id	Dep_Time	Arr_Time	Plane_name	Src	Dstn	Fare	Dep_Date
BXCY05	16:00:00	17:00:00	Indigo	VJA	HYD	2000	06-09-21
CZKFH9	18:30:00	20:45:00	Fighting Falcon	HYD	DEL	3000	07-09-21
DKJIR6	09:00:00	11:30:00	The Voyager	DEL	HYD	5000	08-09-21

- **Payment**

Type
Debit card
Credit card
Banking

- **Records**

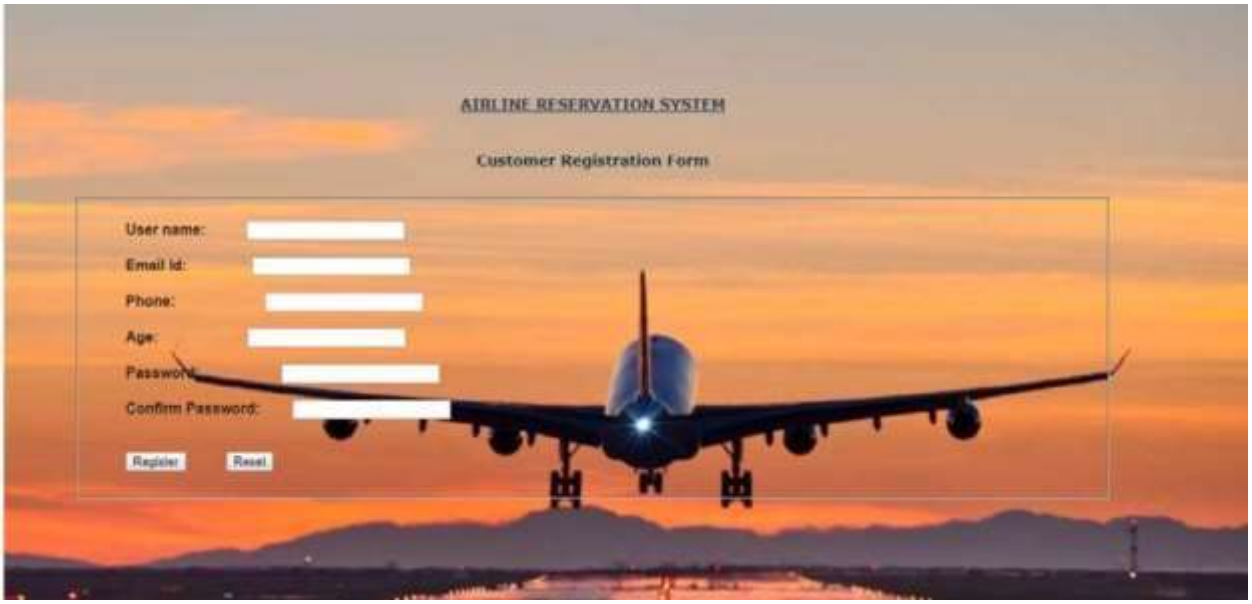
Book_ID	Flight_ID	Dep_Time	Book_Time	User_Name	Payment_Type	Fare
23	BXCY05	16:00:00	12:30:00	Rocky	Debit card	2000
24	CZKFH9	18:30:00	02:46:00	Jimmy	Credit card	3000
25	DKJIR6	09:00:00	04:28:00	Blacky	Banking	5000

Appendix B: Front End Screens

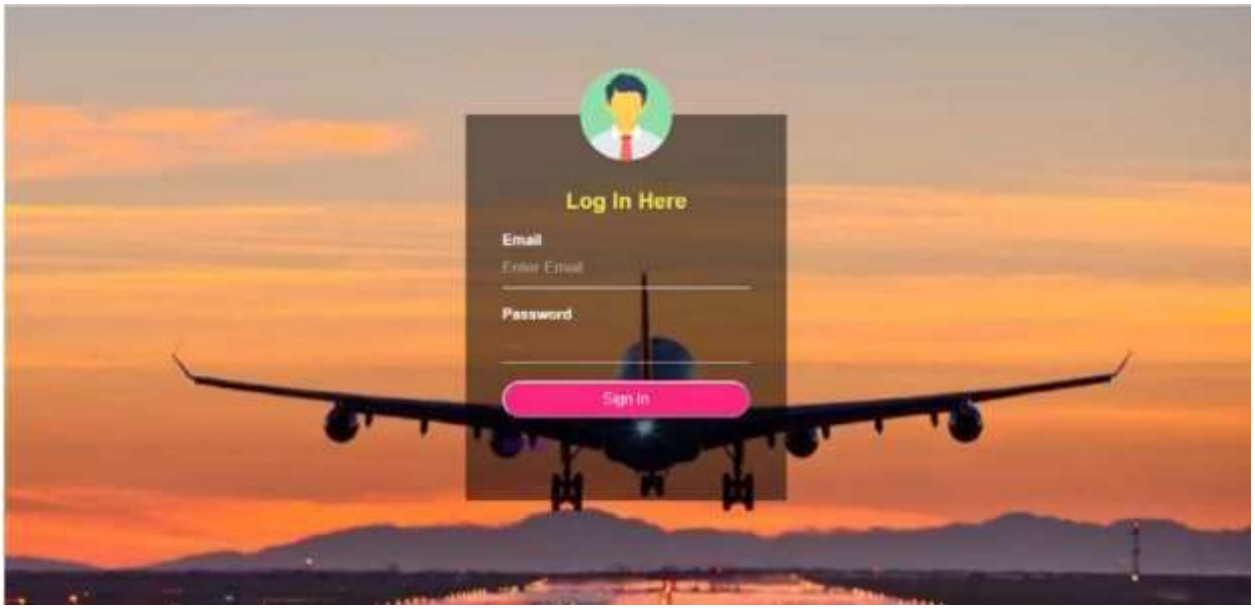
Admin Login Page

Admin Add Flights Page

Admin View Records Page



Customer Sign up Page



Customer Login Page

Customer Choose Source and Destination

Customer Choose Flight Page

Customer Booking Details Page