ABSTRACT

This project on Airline Management System is the automation of registration process of airlines system. The system provides information like passenger's information, flight information, list of all passengers, it allows storing and retrieving data related to the airline industry and make transactions related to air travel etc. The system also allows us to add records when a passenger reserves a ticket. For data storage and retrieval we use MySQL Database. It enables us to add any number of records in our database. The project "Airline Management System" comprises of a large number of flights which belong to a particular airline. The system we have implemented manages different objects viz.

- Airline
- Airline Employees
- Customers/Traveller

Each of these accesses a database schema which has corresponding tables.

Language Used - Java Core
Concept Used - Swing
IDE Used - NetBeans
Database Used - MySQL

CONTENTS

CHAPTER 1

INTRODUCTION

Airline Management System is the administration of airports and airlines. It includes the activities of setting the strategy of airports to gather and provide information on airline commercial and operational priorities. It covers a broad overview of the airline management. It is also studied as a branch of study that teaches management of airport and airlines. This provides a broad overview of the airline industry and creates awareness of the underlying marketing, financial, operational, and other factors influencing airline management. This study provides information on airline commercial and operational priorities, along with teaching the key characteristics of aircraft selection and the impact of airport decision making. It provides some amount of automation in airlines management and helps airline system in making their business more efficient. An added attraction for their potential customers. It will also show the attitude of the management that they are aware to the newly introduced technology and ready to adopt them.

1.1Problem Definition

This project on Flight Management System is the automation of registration process of airline system. The system is able to provide much information like passenger's details, flight details and the booking details. The system allows us to add records when a passenger reserves a ticket. It also allows to delete and update the records based on passenger's requirements. For data storage and retrieval we use the MySQL database. It enables us to add any number of records in our database from the frontend which is Java core. Any changes made in the frontend will be reflected at the backend.

1.2 Need

Electronically handling of flight's record to enhance the accuracy, flexibility, reliability and to remove the human's error. An airline provides air transport services for passengers, generally

with a recognize operating. To provide accurate information about the addition, deletion and modified record. To provide, efficient, accurate, reliable, fast, and robust structure that can handle any number of records. The global airline industry continues to grow rapidly, but consistent and robust profitability is elusive. Measured by revenue, the industry has doubled over the past decade, from US$369 billion in 2004 to a projected $746 billion in 2014, according to the International Air Transport Association(IATA).Much of that growth has been driven by low-cost carriers(LCCs), which now control some 25 percent of the worldwide market and which have been expanding rapidly in emerging markets; growth also came from continued gains by carriers in developed markets, the IATA reported. Yet profit margins are still low, less than 3 percent overall. In the commercial aviation sector, just about every group in the aviation industry chain—airports, airplane manufacturers, jet engine makers, travel agents, and service companies, to name a few—turns a profit. It is seemingly ironic that the airline companies that actually move passengers from one place to another, the most crucial link in the chain, struggle to make a profit.

A few factors that directs us to develop a new system are given below -:

1) Faster System
2) Accuracy
3) Reliability
4) Informative
5) Reservations and cancellations from any where to any place

CHAPTER 2

REQUIREMENTS

2.1 Software Requirement Specifications

Operating System Front End Back End Server Documentation : Windows 10

Frontend Software: Java NetBeans 8.2 : JDK 8

Backend Software: MySQL

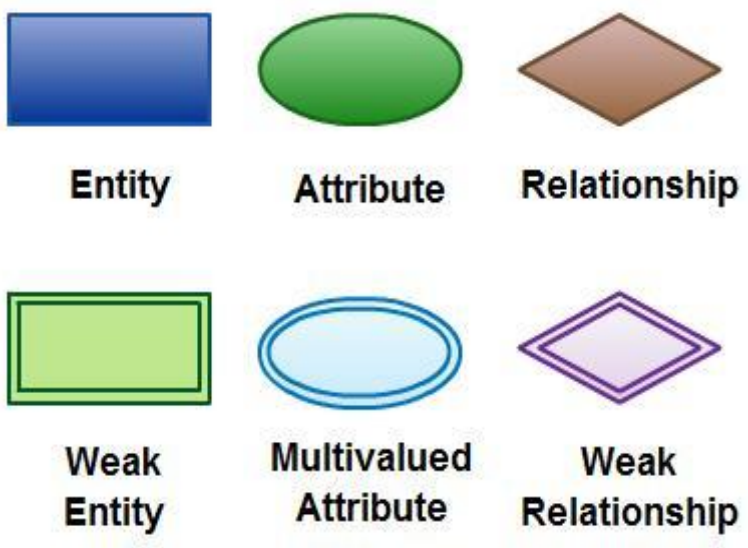
2.2 Hardware Requirement Specifications

Computer Processor Core i3 Processor Speed 2.3 GHz Processor Hard Disk 400 GB or more RAM Min 2GB

CHAPTER 3

ENTITY RELATIONSHIP DIAGRAM

An entity-relationship (ER) diagram is a specialized graphic that illustrates the interrelationships between entities in a database. ER diagrams often use symbols to represent three different types of information. Boxes are commonly used to represent entities. Diamonds are normally used to represent relationships and ovals are used to represent attributes. If the application is primarily a database application, the entity-relationship approach can be used effectively for modeling some parts of the problem. The main focus in ER modeling is the Data Items in the system and the relationship between them. It aims to create conceptual scheme for the Data from the user's perspective. The model thus created is independent of any database model. The ER models are frequently represented as ER diagram. Here we present the ER diagram of the above mentioned project.



CHAPTER 4

SCHEMA DIAGRAM

4.1 SCHEMA DIAGRAM

A database schema is the skeleton structure that represents the logical view of the entire database. A database schema defines its entities and the relationship among them. It contains a descriptive detail of the database, which can be depicted by means of schema diagrams. It defines how the data is organized and how the relations among them are associated. It formulates all the constraints that are to be applied on the data.

A database schema defines its entities and the relationship among them. It contains a descriptive detail of the database, which can be depicted by means of schema diagrams. It's the database designers who design the schema to help programmers understand the database and make it useful.

A database schema can be divided broadly into two categories −

- Physical Database Schema − This schema pertains to the actual storage of data and its form of storage like files, indices, etc. It defines how the data will be stored in a secondary storage.
- Logical Database Schema − This schema defines all the logical constraints that need to be applied on the data stored. It defines tables, views, and integrity constraints.

CHAPTER 5

IMPLEMENTATION

5.1 Backend Implementation

MYSQL

MySQL is an open-source relational database management system (RDBMS). A relational database organizes data into one or more data tables in which data types may be related to each other; these relations help structure the data. SQL is a language programmers use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like MySQL works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups.

Table cancellation:

create table cancellation(pnr_no varchar(10), cancellation_no varchar(10), cancellation_date DATE, fli_code varchar(15));

Table flight:

create table flight(f_code varchar(10), f_name varchar(20), src varchar(30), dst varchar(30));

Table login:

create table login(username varchar(20), password varchar(20));

Table passenger:

create table passenger(pnr_no varchar(10), address varchar(30), nationality varchar(15), name varchar(20), gender varchar(10), ph_no varchar(15), passport_no varchar(20), fl_code varchar(10));

Table payment:

create table payment(pnr_no varchar(10), ph_no varchar(15), cheque_no varchar(15), card_no varchar(20), paid_amt varchar(10), pay_date DATE);

Table reservation:

create table reservation(pnr_no varchar(10), ticket_id varchar(10), f_code varchar(10), jny_date DATE, jny_time varchar(10), src varchar(20), dst varchar(20));

Table sector:

create table sector(flight_code varchar(20), capacity varchar(10), class_code varchar(5), class_name varchar(20));

5.2 Frontend Implementation

Java Core

Core Java is the part of Java programming language that is used for creating or developing a general-purpose application. It uses only one tier architecture that is why it is called as ‘stand alone’ application.Core java programming covers the swings, socket, awt, thread concept, collection object and classess.

Swings

Swing is a GUI widget toolkit for Java. It is part of Oracle's Java Foundation Classes (JFC) – an API for providing a graphical user interface (GUI) for Java programs.

Swing provides a look and feel that emulates the look and feel of several platforms, and also supports a pluggable look and feel that allows applications to have a look and feel unrelated to the underlying platform. It has more powerful and flexible components than AWT. In addition to familiar components such as buttons, check boxes and labels, Swing provides several advanced components such as tabbed panel, scroll panes, trees, tables, and lists.

5.3 Creating mainframe class

```
package airline.management.system;
import net.proteanit.sql.DbUtils;
import java.awt.*;
import javax.swing.*;
import java.awt.event.*;
public class Mainframe extends JFrame{
    public static void main(String[] args) {
        new Mainframe().setVisible(true);
    }
        public Mainframe() {
        super("AIRLINE RESERVATION MANAGEMENT SYSTEM");
        initialize();
  private void initialize() {
        setForeground(Color.CYAN);
        setLayout(null);
            JLabel NewLabel = new JLabel("");
            NewLabel.setIcon(new
ImageIcon(ClassLoader.getSystemResource("airline/management/system/icon/front.jpg")));
            NewLabel.setBounds(0, 0, 1920, 990);
            add(NewLabel);
       JLabel AirlineManagementSystem = new JLabel("AIR INDIA WELCOMES YOU");
            AirlineManagementSystem.setForeground(Color.BLUE);
```

```
        AirlineManagementSystem.setFont(new Font("Tahoma", Font.PLAIN, 36));



AirlineManagementSystem.setBounds(700, 60, 1000, 55);
            NewLabel.add(AirlineManagementSystem);
            JMenuBar menuBar = new JMenuBar();
            setJMenuBar(menuBar);
            JMenu AirlineSystem = new JMenu("AIRLINE SYSTEM");
            AirlineSystem.setForeground(Color.BLUE);
            menuBar.add(AirlineSystem);
             JMenuItem FlightDetails = new JMenuItem("FLIGHT_INFO");
            AirlineSystem.add(FlightDetails);
            JMenuItem ReservationDetails = new JMenuItem("ADD_CUSTOMER_DETAILS");
            AirlineSystem.add(ReservationDetails);
            JMenuItem PassengerDetails = new JMenuItem("JOURNEY_DETAILS");
            AirlineSystem.add(PassengerDetails);
            JMenuItem SectorDetails_1 = new JMenuItem("PAYMENT_DETAILS");
            AirlineSystem.add(SectorDetails_1);
            JMenuItem Cancellation = new JMenuItem("CANCELLATION");
            AirlineSystem.add(Cancellation);
            JMenu Ticket = new JMenu("TICKET");
        Ticket.setForeground(Color.RED);
            menuBar.add(Ticket);
            JMenu List = new JMenu("LIST");
        List.setForeground(Color.BLUE);
            menuBar.add(List);
```

```
JMenu Misc = new JMenu("MISC");
        Misc.setForeground(Color.RED);
           menuBar.add(Misc);
           FlightDetails.addActionListener(new ActionListener(){
           public void actionPerformed(ActionEvent ae){
              new Flight_Info();
           }
           });
          ReservationDetails.addActionListener(new ActionListener(){
           public void actionPerformed(ActionEvent ae){
              try {
                 new Add_Customer();
                    } catch (Exception e) {
                 e.printStackTrace();
                    }
           }
           });
            PassengerDetails.addActionListener(new ActionListener(){
           public void actionPerformed(ActionEvent ae){
              try {
                 new Journey_Details();
              } catch (Exception e) {
                 e.printStackTrace();
```

```
                }



}
        });
        SectorDetails_1.addActionListener(new ActionListener(){
        public void actionPerformed(ActionEvent ae){
           try {
               new Payment_Details();
                  } catch (Exception e) {
               e.printStackTrace();
                  }
        }
        });
        Cancellation.addActionListener(new ActionListener(){
        public void actionPerformed(ActionEvent ae){
           new Cancel();
        }
        });
        setSize(1950,1090);
        setVisible(true);
    }
}
```

CHAPTER 6

SNAPSHOTS

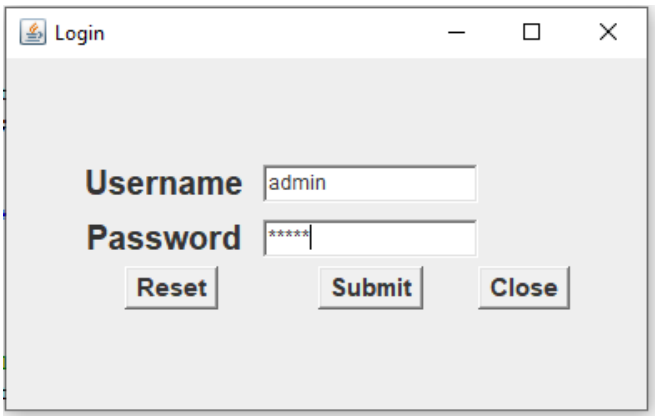


Fig 6.1: Login Operation

Fig 6.2: Mainframe

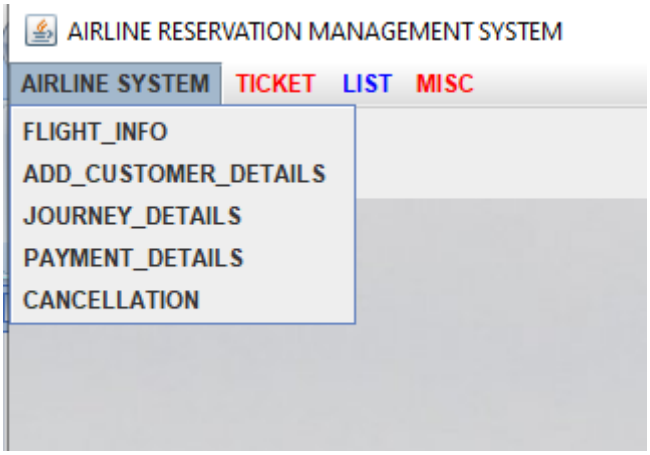


Fig 6.3: Drop-down Menu

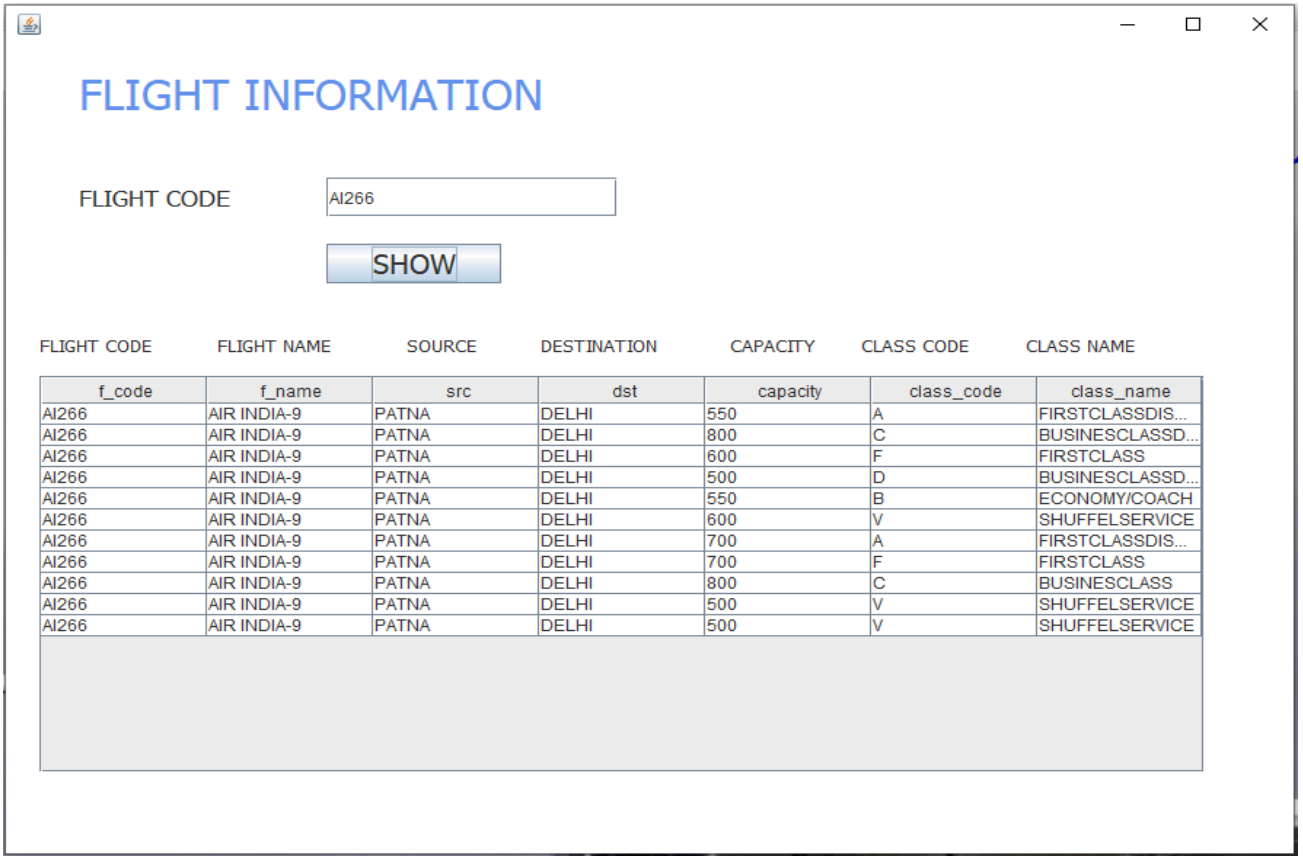


Fig 6.4: Flight Information

Fig 6.5: Adding Customer

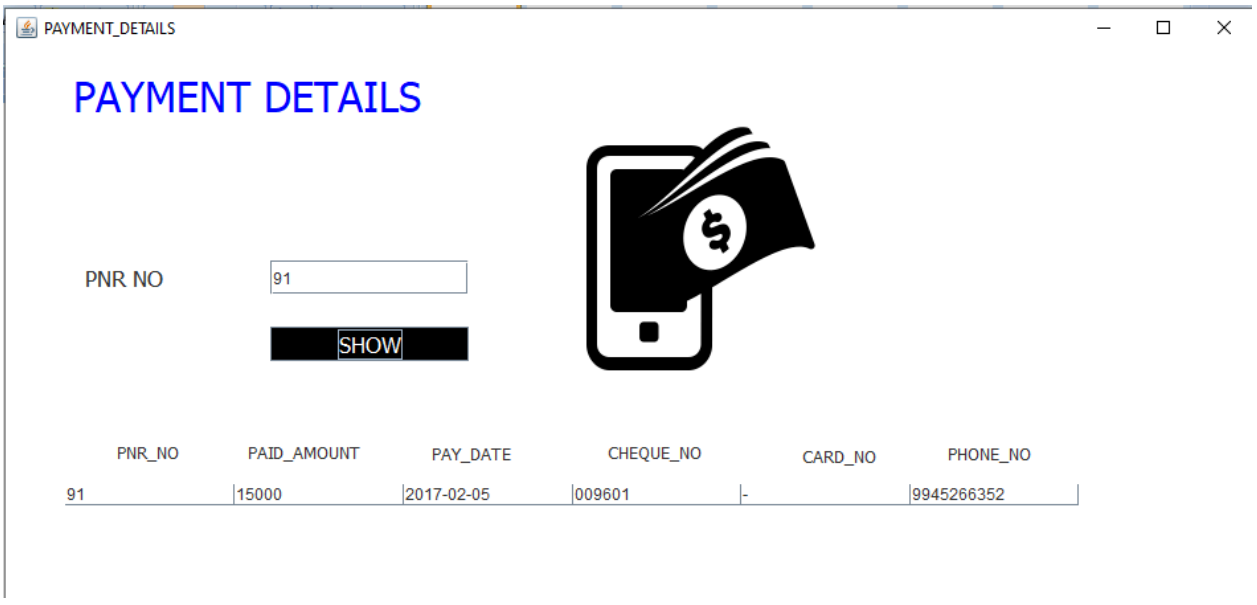


Fig 6.6: Payment Details

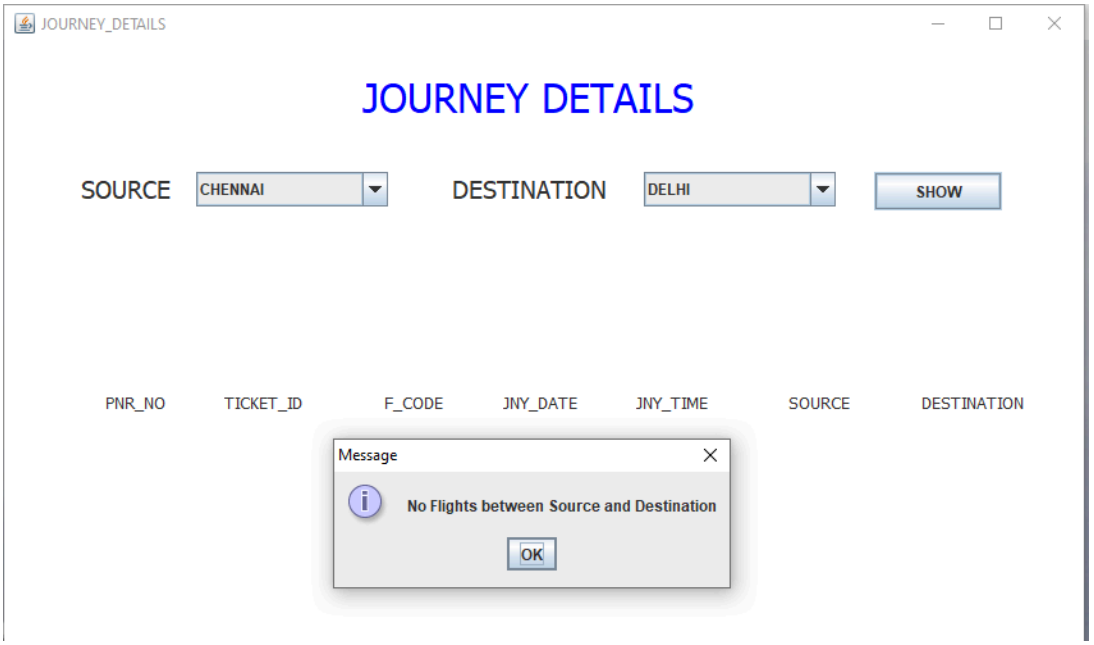


Fig 6.7: Journey Details

Fig 6.8: Cancellation

Fig 6.9: Backend Records

```
mysql> select * from flight;
+--------+--------------+-----------+-----------+
| f_code | f_name       | src       | dst       |
+--------+--------------+-----------+-----------+
| AI266  | AIR INDIA-9  | PATNA     | DELHI     |
| AI274  | AIR INDIA-3  | HYDERABAD | CHENNAI   |
| AI358  | AIR INDIA-7  | DELHI     | PATNA     |
| AI359  | AIR INDIA-6  | CHENNAI   | PATNA     |
| AI913  | AIR INDIA-10 | MUMBAI    | HYDERABAD |
| AI933  | AIR INDIA-8  | HYDERABAD | BANGALORE |
| AI951  | AIR INDIA-4  | BANGALORE | PATNA     |
| AI970  | AIR INDIA-5  | MUMBAI    | CHENNAI   |
| AI9730 | AIR INDIA-1  | BANGALORE | MUMBAI    |
| AI9731 | AIR INDIA-2  | DELHI     | CHENNAI   |
+--------+--------------+-----------+-----------+
10 rows in set (0.05 sec)

mysql> select * from passenger;
+--------+------------------+-------------+---------+--------+------------+-------------+---------+
| pnr_no | address          | nationality | name    | gender | ph_no      | passport_no | fl_code |
+--------+------------------+-------------+---------+--------+------------+-------------+---------+
| 91     | JP NAGAR         | INDIA       | AJAY    | MALE   | 9945266352 | J836982     | AI358   |
| 120    | BSK              | INDIA       | PRIYA   | FEMALE | 9945266353 | J836983     | AI359   |
| 200    | KORAMANGALA      | INDIA       | UDAY    | MALE   | 9945266354 | J836984     | AI266   |
| 400    | KS LAYOUT        | IRAN        | SALEEM  | MALE   | 9945266355 | J836985     | AI913   |
| 331    | JAYNAGAR         | INDIA       | SATHISH | MALE   | 9945266356 | J836986     | AI933   |
| 81     | KENGERI          | AFGHANISTAN | IRFAN   | MALE   | 9945266357 | J836987     | AI951   |
| 49     | BTM LAYOUT       | CANADA      | RICHA   | FEMALE | 9945266347 | J836988     | AI970   |
| 82     | HSR LAYOUT       | INDIA       | RANJITH | MALE   | 9945266358 | J836989     | AI359   |
| 56     | BHAGAT SINGH ROAD | INDIA       | RANJI   | MALE   | 7894561230 | J837003     | AI9730  |
| 232    | jp nagar         | india       | teju    | female | 23467899   | 763023      | 123     |
+--------+------------------+-------------+---------+--------+------------+-------------+---------+
10 rows in set (0.06 sec)
```

```
mysql> select * from login;
+----------+----------+
| username | password |
+----------+----------+
| admin    | admin    |
+----------+----------+
1 row in set (0.49 sec)

mysql> select * from payment;
+--------+------------+------------+-----------------+----------+------------+
| pnr_no | ph_no      | cheque_no  | card_no         | paid_amt | pay_date   |
+--------+------------+------------+-----------------+----------+------------+
| 56     | 9945266350 | 0046234    | -               | 15000    | 2017-01-01 |
| 80     | 9945266351 | 0086001    | -               | 20000    | 2017-01-03 |
| 91     | 9945266352 | 009601     | -               | 15000    | 2017-02-05 |
| 120    | 9945266353 | 0015020    | -               | 20000    | 2017-02-07 |
| 200    | 9945266354 | -          | 5195501955019   | 25000    | 2017-01-09 |
| 400    | 9945266355 | 0805010    | -               | 25000    | 2017-02-11 |
| 331    | 9945266356 | 0915420    | -               | 25000    | 2017-03-14 |
| 81     | 9945266357 | 0315020    | -               | 15000    | 2017-03-18 |
| 49     | 9945266358 | 0815121    | -               | 20000    | 2017-01-16 |
| 82     | 9945266359 | 0025020    | -               | 15000    | 2017-02-20 |
| 401    | 9945266360 | 0515821    | -               | 15000    | 2017-03-24 |
| 409    | 9945266361 | 0235121    | -               | 15000    | 2017-03-26 |
| 500    | 9945266362 | 0345830    | -               | 25000    | 2017-01-28 |
| 300    | 9945266363 | 0345760    | -               | 25000    | 2017-01-07 |
| 320    | 9945266364 | 0565431    | -               | 20000    | 2017-02-03 |
| 151    | 9945266366 | -          | 5090062055019   | 15000    | 2017-03-04 |
| 200    | 9945266365 | -          | 5090051055019   | 15000    | 2017-01-11 |
| 349    | 9945266367 | 0419321    | -               | 15000    | 2017-03-18 |
| 461    | 9945266368 | 0419321    | -               | 25000    | 2017-02-19 |
| 441    | 9945266369 | 0319321    | -               | 20000    | 2017-02-11 |
| 411    | 9945266370 | 0328972    | -               | 20000    | 2017-01-17 |
+--------+------------+------------+-----------------+----------+------------+
21 rows in set (0.05 sec)
```

Fig :6.10 : Back End

```
mysql> select * from sector;
+-------------+----------+------------+----------------------+
| flight_code | capacity | class_code | class_name           |
+-------------+----------+------------+----------------------+
| AI9730      | 550      | A          | FIRSTCLASSDISCOUNT   |
| AI9731      | 800      | C          | BUSINESCLASSDISCOUNT |
| AI274       | 600      | F          | FIRSTCLASS           |
| AI951       | 500      | D          | BUSINESCLASSDISCOUNT |
| AI970       | 550      | B          | ECONOMY/COACH        |
| AI359       | 600      | V          | SHUFFELSERVICE       |
| AI358       | 700      | A          | FIRSTCLASSDISCOUNT   |
| AI933       | 700      | F          | FIRSTCLASS           |
| AI266       | 800      | C          | BUSINESCLASS         |
| AI266       | 500      | V          | SHUFFELSERVICE       |
| AI913       | 500      | V          | SHUFFELSERVICE       |
+-------------+----------+------------+----------------------+
11 rows in set (0.04 sec)
```

CONCLUSION

This project on Airline Management System is the automation of registration process of airline system. The system is able to provide much information like passenger's details, flight details and the booking details. The system allows us to add records when a passenger reserves a ticket. It also allows to delete and update the records based on passenger's requirements. This project has guided our path through various aspects of computer science where developing online application plays a major role.

REFERENCES

[1] https://developers.openshift.com/database/mysql.html

[2] Web References- https://youtu.be/UbIIFLsEeiM