

Airlines Booking System

Team Members

| SN | Student Name | Section Number |
|-----------|--------------------------------|-----------------------|
| 1 | Zyad ashraf mahmoud | 2 |
| 2 | Menna ashraf hanafy | 4 |
| 3 | Hoda mohamed abdullatif | 4 |

Table of Contents

| | |
|-------------------------------------|----------|
| Team Members | 1 |
| Table of Contents..... | 2 |
| 1. Project Objective..... | 3 |
| 2. Project Diagrams | 4 |
| 2.1. Entity Relation Diagram..... | 4 |
| 2.2. Relational Model Diagram | 4 |
| 3.SQL Queries | 5 |
| 4. Project Screenshots | 6 |

1. Project Objective

- The main objective of Airlines Booking System is to manage the details of Airlines Ticket, Flights, Customers, Booking, create user-friendly efficient and secure platform that allows the user to book flights, manage their reservations, and reduce the manual work where the user can create account and have the accessibility to select as many flights as he wanted at the preferred time.
- The system provides a simple and intuitive interface that enables users to search for flights based on destination.
- The system should allow the user to view flight schedules, seat availability and choose their preferred seats.
- Additionally, the system should be able to manage user information, including personal details and travel history.

Programming languages used:

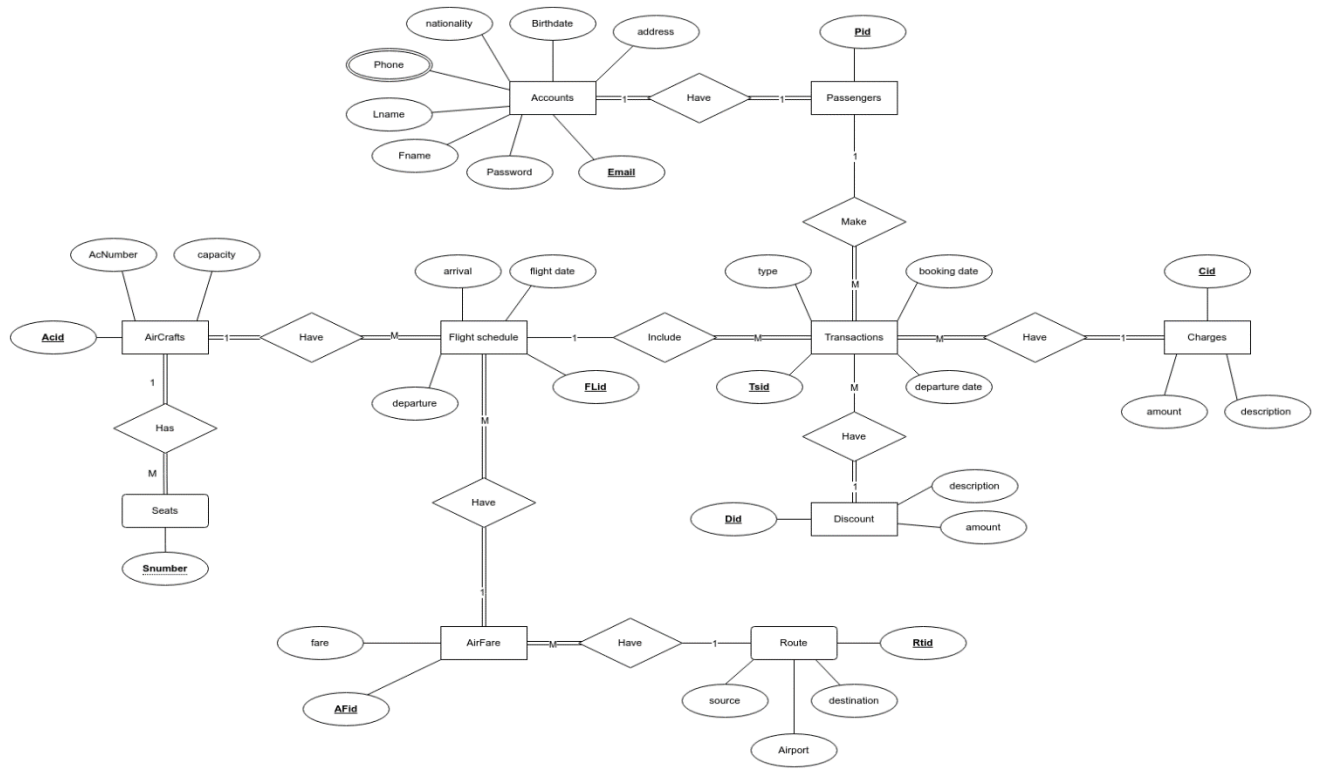
- JAVA
- SQL

Used Tools:

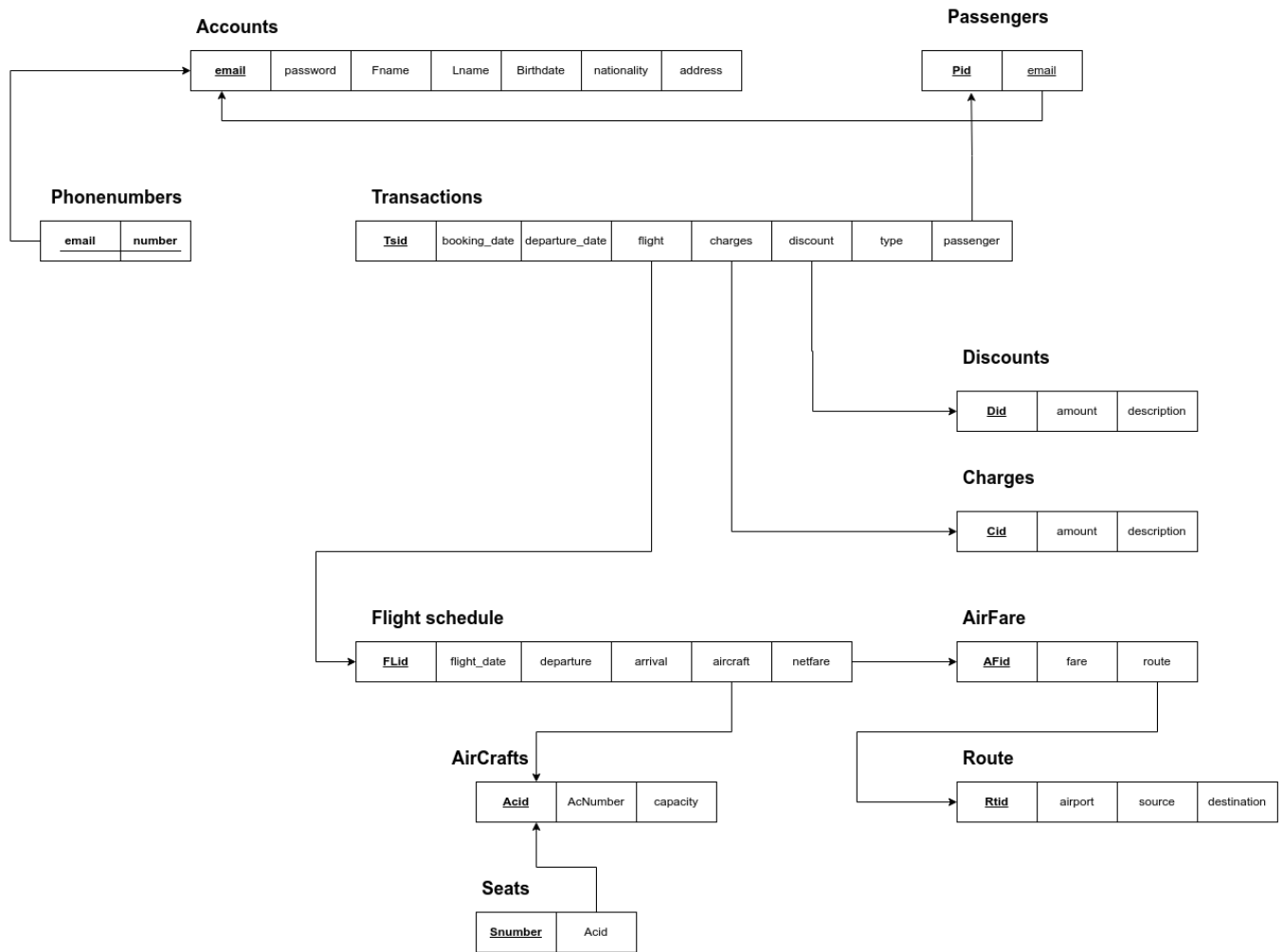
- GUI

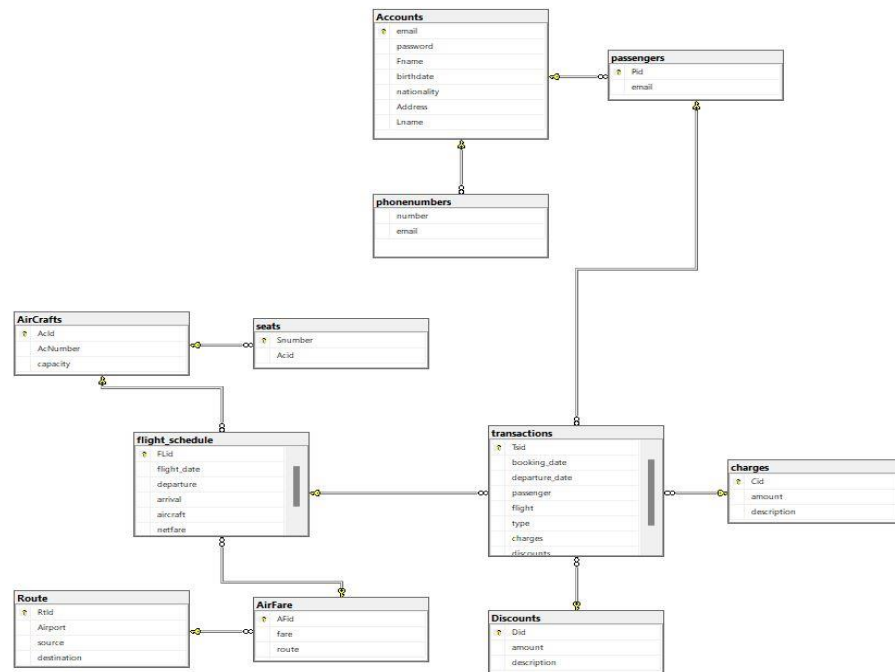
2. Project Diagrams

2.1. Entity Relation Diagram



2.2 Relational Model Diagram





3. SQL Queries

Main (connet to database)

```
public static void main(String[] args) throws SQLException {
20
21     try {
22         Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
23         String dburl = "jdbc:sqlserver://ZEYADELBANNA:1433;DatabaseName=Airlines Booking App;integratedSecurity=true;en
24         Connection connection = DriverManager.getConnection(dburl);
25         Statement stat = connection.createStatement();
26         String query = "select * from Accounts";
27         ResultSet rs = stat.executeQuery(query);
28         while(rs.next())
29         {
30             System.out.println(rs.getString(1) + " " + rs.getString(2));
31         }
32     } catch (ClassNotFoundException ex) {
33         Logger.getLogger(FinalProject.class.getName()).log(Level.SEVERE, null, ex);
34     }
35
36
37
38 }
```

libraries

```
package finalproject;
import java.io.*;
import java.sql.DriverManager;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.logging.Level;
import java.util.logging.Logger;
public class FinalProject {
```

Login button

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    String fn = t1.getText();
    try {
        Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
        String url="jdbc:sqlserver://ZEYADELBANNA:1433;DatabaseName=Airlines Booking App;integratedSecurity=true;enc
        Connection con = DriverManager.getConnection(url);
        String sql = "select * from flight_schedule as f, AirPare as a , Route as r , transactions as t, Passengers
        PreparedStatement pst = con.prepareStatement(sql);
        pst.setString(1,fn );
        ResultSet rs = pst.executeQuery();
        if(rs.next())
        {
            do
            {
                String flightn , aname ,bdate,ddate ,tprice ;
                flightn = rs.getString(1);
                aname = rs.getString(11);
                ddate = rs.getString(16);
                bdate = rs.getString(15);
                tprice = rs.getString(8);
                String DBdata[] =( flightn , aname , ddate ,bdate ,tprice);
                DefaultTableModel tb = (DefaultTableModel)t2.getModel();

```

```

                ddate = rs.getString(16);
                bdate = rs.getString(15);
                tprice = rs.getString(8);
                String DBdata[] =( flightn , aname , ddate ,bdate ,tprice);
                DefaultTableModel tb = (DefaultTableModel)t2.getModel();
                tb.addRow(DBdata);
            } while(rs.next());
        }
        con.close();
    }
    catch (Exception e )
    {
        JOptionPane.showMessageDialog(null, e);
    }
}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    dispose();
    Home f2 = new Home();
    f2.setVisible(true);
}

```

Register button


```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
239
240     try {
241         Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
242         String url="jdbc:sqlserver://ZEYADELBANNA:1433;DatabaseName=Airlines Booking App;integratedSecurity=true;enc
243         Connection con = DriverManager.getConnection(url);
244
245         String fname , lname , nat , phone1,phone2 , adr , email , password , d , m , y ;
246         fname = Fname.getText();
247         lname = Lname.getText();
248         nat = (String) nation.getSelectedItemAt();
249         phone1 = phone.getText();
250         phone2=p2.getText();
251         adr = Address.getText();
252         email = e.getText();
253         password = pass.getText();
254         d = (String) day.getSelectedItemAt();
255         m= (String) month.getSelectedItemAt();
256         y = (String) year.getSelectedItemAt();
257         String date = d + "-" +m+"-"+y;
258         String sql = "insert into Accounts values ('"+email+"', '"+password+"', '"+fname+"', '"+date+"', '"+nat+"', '"+ad
259         PreparedStatement pst = con.prepareStatement(sql);
260         pst.executeQuery();
261     }
}

catch (Exception e )
263
264     {
265         JOptionPane.showMessageDialog(null, "Your Email Created Successuflly");
266         dispose();
267         Login f2 = new Login();
268         f2.setVisible(true);
269     }
}

```

Confirm button in ticket info

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
136
137     String fn = f1.getText();
138
139     try {
140         Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
141         String url="jdbc:sqlserver://ZEYADELBANNA:1433;DatabaseName=Airlines Booking App;integratedSecurity=true;enc
142         Connection con = DriverManager.getConnection(url);
143         String sql = "Select f.departure , f.flight_date , r.source , r.destination , a.fare , f.FLid from flight_sc
144         PreparedStatement pst = con.prepareStatement(sql);
145         ResultSet rs = pst.executeQuery();
146         int flage =1 ;
147
148         if(rs.next())
149         {
150             do
151             {
152                 String s1 = rs.getString(1);
153                 String s2 = rs.getString(2);
154                 String s3 = rs.getString(3);
155                 String s4 = rs.getString(4);
156                 String s5 = rs.getString(5);
157                 String s6 = rs.getString(6);
158                 if(s6.equals(fn))
159                 {
160                     flage =0 ;
161                 }
162             }
163         }
164     }
165 }

```

```

158         flage =0 ;
159         t2.setText(s1);
160         t3.setText(s2);
161         t4.setText(s3);
162         t5.setText(s4);
163         t6.setText(s5);
164         JOptionPane.showMessageDialog(null, "Your Flight booked Successufly");
165     }
166     } while(rs.next());
167
168
169     }
170     if(flage ==1)
171     {
172
173     }
174     con.close();
175
176 }
177 catch (Exception e )
178 {
179     JOptionPane.showMessageDialog(null, e);
180 }
181 }
182

```

Confirm button in flight list

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
227
228     try {
229         Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
230         String url="jdbc:sqlserver://ZEYADELBANNA:1433;DatabaseName=Airlines Booking App;integratedSecurity=true;enc
231         Connection con = DriverManager.getConnection(url);
232         int tid = (int) (Math.random()*100)+5;
233         int d = (int) (Math.random()*25)+5;
234         int m = (int) (Math.random()*9)+2;
235         int y = 2023;
236         String ddate = Integer.toString(d);
237         String mdate = Integer.toString(m);
238         String ydate = Integer.toString(y);
239         String book = ddate+"-"+mdate+"-"+ydate;
240         String flno = (String) fno.getSelectedItemAt();
241         String t = (String) type.getSelectedItemAt();
242         String dis = "154";
243         String chrg = "17";
244         String dep=" ";
245         DefaultTableModel tb = (DefaultTableModel)t1.getModel();
246         for(int i = 0 ; i< tb.getRowCount(); i++)
247         {
248             if(((String)tb.getValueAt(i, 0)).equals(flno))
249             {
250                 dep = (String)tb.getValueAt(i, 3);

```

```

246         for(int i = 0 ; i< tb.getRowCount(); i++)
247         {
248             if(((String)tb.getValueAt(i, 0)).equals(flno))
249             {
250                 dep = (String)tb.getValueAt(i, 3);
251                 break;
252             }
253         }
254         int pid = (int) (Math.random()*150)+5;
255         String pasid = Integer.toString(pid);
256         String email = username.getText();
257         String sql = "insert into passengers values ('"+pasid+"','"+email+"');insert into transactions values ('"+ti
258         PreparedStatement pst = con.prepareStatement(sql);
259         pst.executeQuery();
260
261     }
262     catch (Exception e )
263     {
264         dispose();
265         TicketInfo f2 = new TicketInfo();
266         f2.setVisible(true);
267     }
268 }
269
270

```

Search button in flight list

```

180 private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
181     try {
182         Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
183         String url="jdbc:sqlserver://ZEYADELBANNA:1433;DatabaseName=Airlines Booking App;integratedSecurity=true;enc
184         Connection con = DriverManager.getConnection(url);
185         String source = s.getText();
186         String destination = d.getText();
187         String sql = "select * from flight_schedule as f, AirFare as a , Route as r where r.Rtid = a.route and f.net
188         PreparedStatement pst = con.prepareStatement(sql);
189         pst.setString(1,source );
190         pst.setString(2,destination );
191         ResultSet rs = pst.executeQuery();
192
193         if(rs.next())
194         {
195             do
196             {
197                 String flightn , s , d , date , time , aname , tprice ;
198                 flightn = rs.getString(1);
199                 s = rs.getString(12);
200                 d = rs.getString(13);
201                 date = rs.getString(2);
202                 time = rs.getString(3);
203                 aname = rs.getString(11);

```

```

203         aname = rs.getString(11);
204         tprice = rs.getString(8);
205         String DBdata[] = { flightn , s , d ,date , time , aname , tprice};
206         DefaultTableModel tb = (DefaultTableModel)t1.getModel();
207         tb.addRow(DBdata);
208     } while(rs.next());
209     }
210     con.close();
211
212 }
213
214 catch (Exception e )
215 {
216     JOptionPane.showMessageDialog(null, e);
217 }
218

```

Book List Search

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
117     String fn = t1.getText();
118     try {
119         Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
120         String url="jdbc:sqlserver://ZEYADELBANNA:1433;DatabaseName=Airlines Booking App;integratedSecurity=true;enc
121         Connection con = DriverManager.getConnection(url);
122         String sql = "select * from flight_schedule as f, AirPare as a , Route as r , transactions as t, Passengers
123         PreparedStatement pst = con.prepareStatement(sql);
124         pst.setString(1,fn );
125         ResultSet rs = pst.executeQuery();
126
127         if(rs.next())
128         {
129             do
130             {
131                 String flightn , aname ,bdate,ddate ,tprice ;
132                 flightn = rs.getString(1);
133                 aname = rs.getString(11);
134                 ddate = rs.getString(16);
135                 bdate = rs.getString(15);
136                 tprice = rs.getString(8);
137                 String DBdata[]={ flightn , aname , ddate ,bdate ,tprice};
138                 DefaultTableModel tb = (DefaultTableModel)t2.getModel();
139
135                 ddate = rs.getString(16);
136                 bdate = rs.getString(15);
137                 tprice = rs.getString(8);
138                 String DBdata[]={ flightn , aname , ddate ,bdate ,tprice};
139                 DefaultTableModel tb = (DefaultTableModel)t2.getModel();
140                 tb.addRow(DBdata);
141             } while(rs.next());
142         }
143         con.close();
144     }
145     catch (Exception e )
146     {
147         JOptionPane.showMessageDialog(null, e);
148     }
149 }
150
151 private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
153     dispose();
154     Home f2 = new Home();
155     f2.setVisible(true);
156 }
157

```


4. Project Screenshots

4.1. Login Page

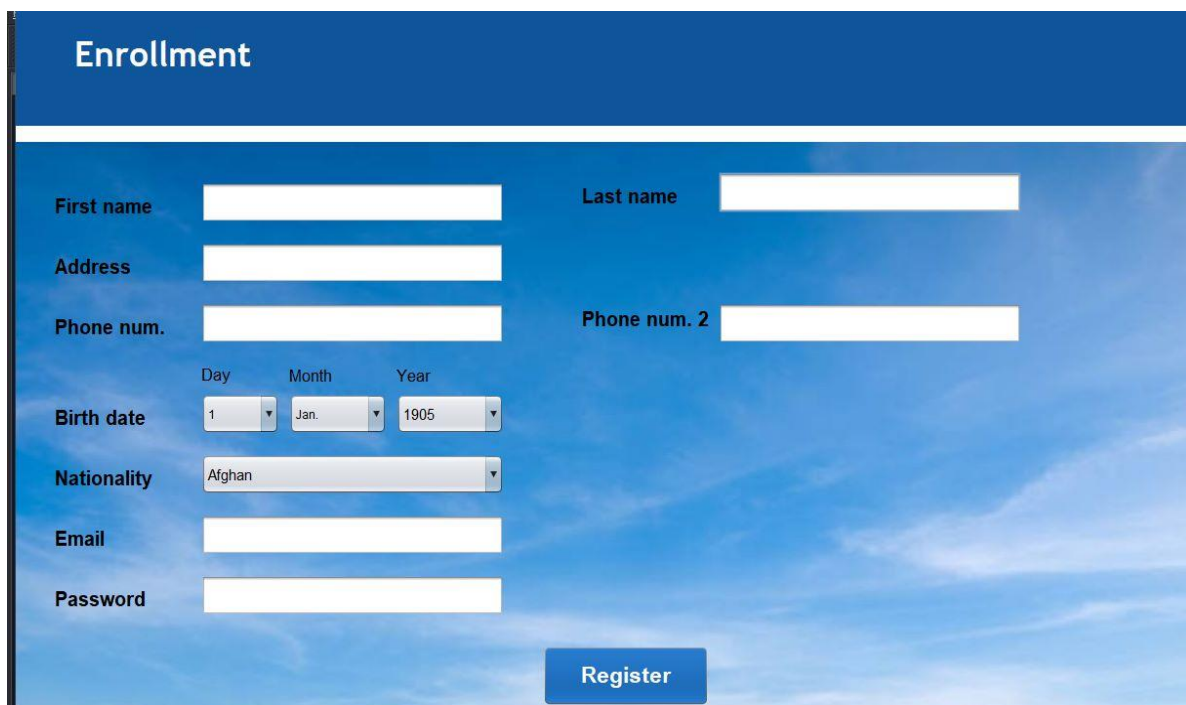


Welcome ! x

[Login To Your Account](#)

Username :


Password :

Enrollment

First name

Last name

Address

Phone num.

Phone num. 2

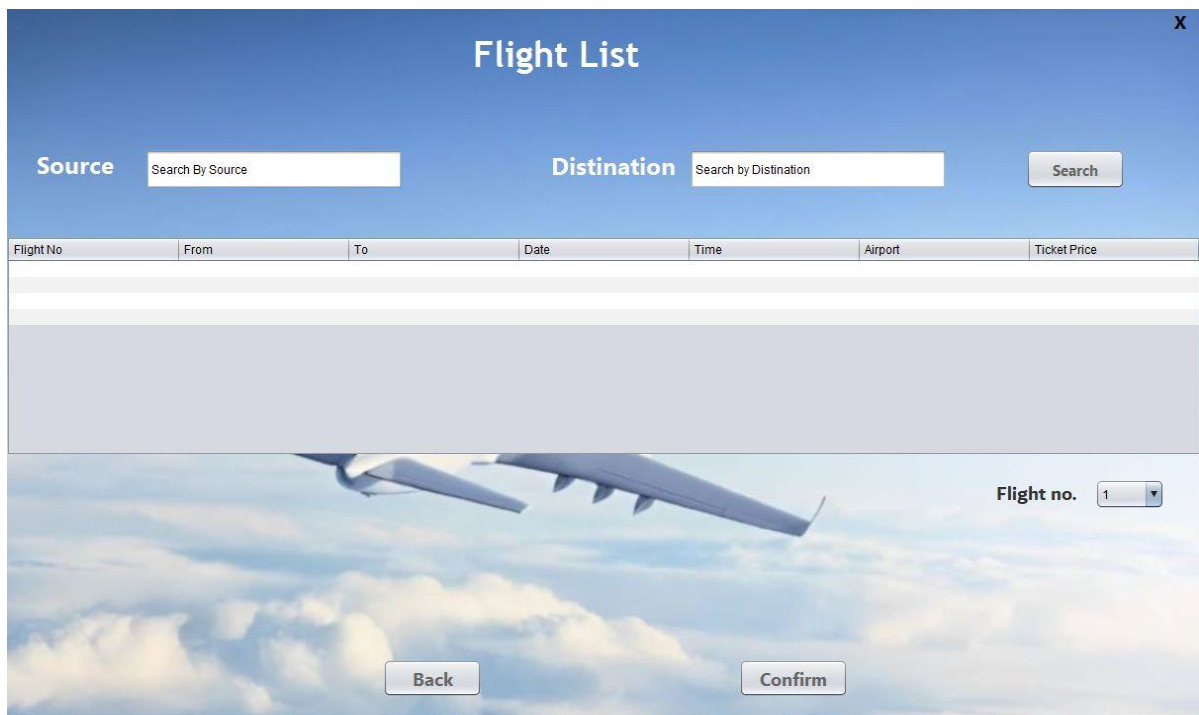
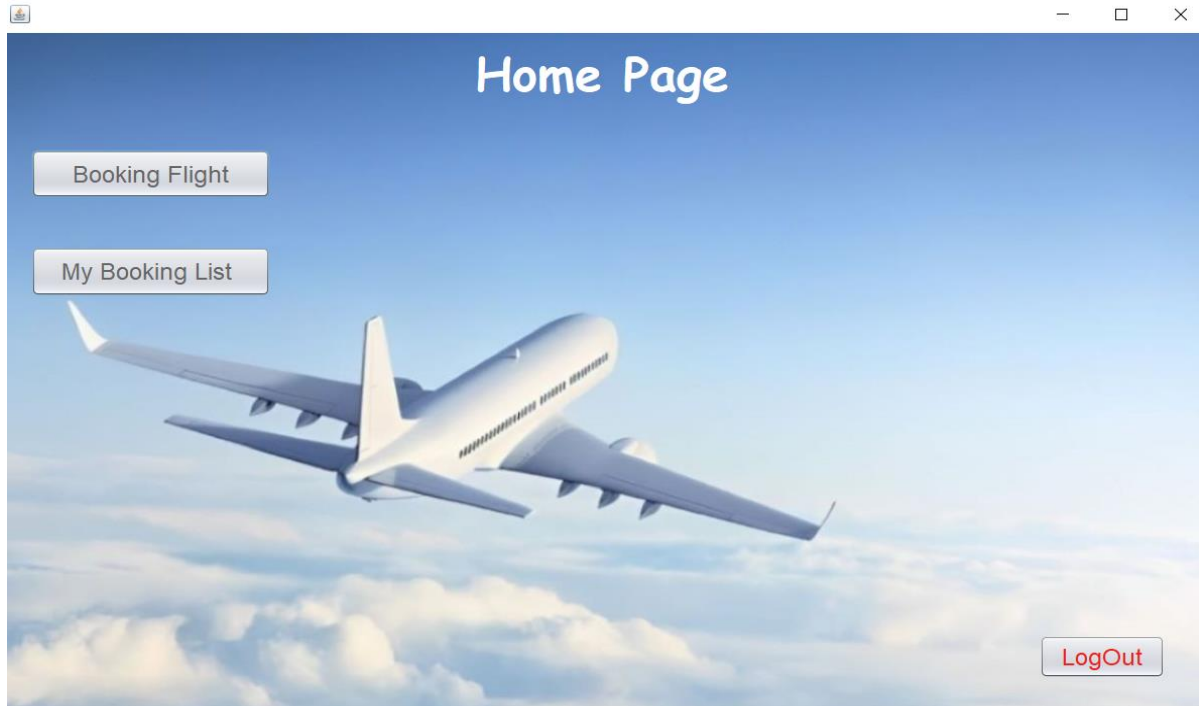
Birth date

| Day | Month | Year |
|--------------------------------|-----------------------------------|-----------------------------------|
| <input type="text" value="1"/> | <input type="text" value="Jan."/> | <input type="text" value="1905"/> |

Nationality

Email

Password



Flight Informations :

Flight number :

Time of Flight :

Date of Flight :

From :

To :

Duration of Flight :

Price of Ticket :

Back

Confirm

Book Ticket List

Flight No

Search By Name

Search

| Flight No | User Name | Mobile No | Book Date | Email Id | No of Person | Ticket Price | Final Price |
|-----------|-----------|-----------|-----------|----------|--------------|--------------|-------------|
|-----------|-----------|-----------|-----------|----------|--------------|--------------|-------------|

Back

Data base tables

Accounts

| | email | password | Fname | birthdate | nationality | Address | Lname |
|----|-------------------|----------|---------|--------------|-------------|-----------|---------|
| 1 | ahmed@gmail.com | 78546 | ahmed | 4-Jun. -1908 | Bangladeshi | alex | ashraf |
| 2 | ali@gmail.com | 78965 | ali | 2-Jun. -1910 | Argentine | cairo | ali |
| 3 | asad@gmail.com | 475896 | asad | 6-Jul. -1911 | Cameroonian | aswan | ali |
| 4 | ashraf@gmail.com | 451214 | ashraf | 4-Jul. -1909 | Austrian | cairo | elbanna |
| 5 | elbanna@gmail.com | 74589 | zeyad | 4-Sep. -1908 | Austrian | cairo | elbanna |
| 6 | hoda@gmail.com | 456 | hoda | 1912-05-04 | Argentine | alex | ahmed |
| 7 | mahmoud@gmail.com | 452154 | mahmoud | 4-Aug. -1911 | Belarusian | cairo | elbanna |
| 8 | mazen@gmail.com | 254 | mazen | 2005-05-10 | american | 115745658 | ashraf |
| 9 | mena@gmail.com | 789 | mena | 1907-05-03 | Angolan | gouna | ashraf |
| 10 | menna@gamil.com | 963 | menna | 1910-05-13 | Austrian | mayo | ashraf |
| 11 | mona@gmail.com | 7895 | mona | 5-Jun. -1923 | Australian | loxur | ibrahim |
| 12 | ofa@gmail.com | 7452 | ofa | 3-Aug. -1907 | Argentine | cairo | ashraf |
| 13 | rrtrrt | | | 1-Jan. -1905 | Afghan | | |
| 14 | saeed@gmail.com | 4785 | saeed | 5-Jul. -1928 | Argentine | giza | ali |
| 15 | salah@gmail.com | 47587 | salah | 6-May -1914 | Belarusian | giza | ali |
| 16 | zeyad@gmail.com | 78965 | zeyad | 1-Sep. -1911 | Austrian | helwan | ashraf |

Aircraft

| Results | | Messages | |
|---------|------|----------|-------|
| | AFid | fare | route |
| 1 | 1 | 2500 | 12 |
| 2 | 2 | 3500 | 22 |
| 3 | 3 | 5000 | 45 |

Airfare

| Results | | Messages | |
|---------|------|----------|-------|
| | AFid | fare | route |
| 1 | 1 | 2500 | 12 |
| 2 | 2 | 3500 | 22 |
| 3 | 3 | 5000 | 45 |

Charges

| Results Messages | | | |
|------------------|-----|--------|---------------|
| | Cid | amount | description |
| 1 | 8 | 500 | extra meal |
| 2 | 17 | 1500 | extra servies |
| 3 | 24 | 2500 | extra bags |

Discounts

| Results Messages | | | |
|------------------|-----|--------|----------------|
| | Did | amount | description |
| 1 | 154 | 500 | winter travels |
| 2 | 478 | 300 | summer travels |

Flight Schedule

| Results Messages | | | | | | |
|------------------|------|-------------|-------------------|------------|----------|---------|
| | FLid | flight_date | departure | arrival | aircraft | netfare |
| 1 | 1 | 2022-01-01 | 20:30:00.00000000 | 2022-01-01 | 1 | 1 |
| 2 | 2 | 2023-05-05 | 18:20:00.00000000 | 2023-05-05 | 3 | 2 |
| 3 | 3 | 2023-09-12 | 15:15:00.00000000 | 2023-09-12 | 2 | 3 |

Passengers

| Results Messages | | |
|------------------|-----|------------------|
| | Pid | email |
| 1 | 10 | zeyad@gmail.com |
| 2 | 22 | hoda@gmail.com |
| 3 | 32 | mena@gmail.com |
| 4 | 34 | mazen@gmail.com |
| 5 | 42 | ashraf@gmail.com |
| 6 | 79 | ali@gmail.com |
| 7 | 97 | ofa@gmail.com |

Route

| Results Messages | | | | |
|------------------|------|----------|--------|-------------|
| | Rtld | Airport | source | destination |
| 1 | 12 | cairo | egypt | italy |
| 2 | 22 | roma | italy | egypt |
| 3 | 45 | new york | usa | france |

Seats

| Results Messages | | |
|------------------|---------|------|
| | Snumber | Acid |
| 1 | 1 | 1 |
| 2 | 19 | 1 |
| 3 | 22 | 1 |
| 4 | 34 | 2 |
| 5 | 36 | 3 |
| 6 | 45 | 2 |
| 7 | 70 | 1 |
| 8 | 78 | 2 |
| 9 | 79 | 3 |
| 10 | 98 | 3 |

Transactions

| Results Messages | | | | | | | | |
|------------------|------|--------------|------------------|-----------|--------|----------|---------|-----------|
| | Tsid | booking_date | departure_date | passenger | flight | type | charges | discounts |
| 1 | 1 | 2023-02-02 | 2023-06-20 | 10 | 1 | economic | 8 | 154 |
| 2 | 2 | 2022-03-03 | 2023-06-11 | 22 | 2 | bussness | 17 | 478 |
| 3 | 32 | 11-5-1441 | 2022-01-01 | 97 | 1 | First | 17 | 154 |
| 4 | 100 | 6-6-3159 | 20:30:00.0000000 | 79 | 1 | First | 17 | 154 |

Phone numbers

| | number | email |
|---|-----------|----------------|
| 1 | 01441215 | asad@gmail.com |
| 2 | 011223365 | asad@gmail.com |