**Mini Project Report on**

**“Airline Management System”**

Submitted by

PA34 Astitwa Mishra 1032201456

PA35 Divyansh Sharma 1032201461

PA40 Saksham Nargotra 1032201618

Under the Guidance of

**Prof. Pratvina Talele**

At



**School of Computer Engineering and Technology**

**ABSTRACT**

Today’s basic requirement under airline management system are passenger service system and revenue management system. We are presenting to you a desktop application with which we can:

1. Add all the available flights leaving for the exact time and destination, to the database.

2. Register all the passengers first to get the details of them and make it hassle free to afterwards book their tickets as many people only want to know about flight details but don't want to book them so we record only those who actually need to book tickets.

3. Book the tickets of intrested passengers who already have registered with their details to the airline's database.

4. There are always some unpredictable situations where we have to cancel plans, whether from passenger's side or from Airline's side. To make it simple we have also made a Cancellation tab, from where passengers can cancel their booked tickets accordingly.

We have created this Airline management system with the help of many technical tools including — Java-Swing, MySQL, JDBC and many more.

**CONTENTS**

1. Introduction:

a. Description of the project (what will it do, what problem will it solve).

2. Problem definition and Objectives

3. Tools and Technologies used

4. Database Design and schema

5. System Architecture

a. Program Modules including their functionality and interfaces, input output modules, Handling Program Errors etc.

6. System Design

a. GUI Design, Exception Handling, Sample codes etc.

7. Conclusion

8. References in IEEE format

**INTRODUCTION:**

We have created an airline management system with the help of Java swing. You can book or cancel flights from your registered account on our platform. You can check the upcoming flights and view their details.

**PROBLEM STATEMENT:** Create a Desktop application for Airline Management System using MySQL(Backend) for database design, Java/Java-Swing(Frontend) for User-Interface design and JDBC for connectivity of both the ends and Implementation[1]. The application should be user-friendly and database should contain atleast 3-4 tables.

**OBJECTIVES:**

1. To study Java/Java-Swing for User-Interface.
2. To study MySQL for Database design.
3. To study JDBC for the connectivity of both frontend and backend.
4. To Implement all of the above.

**TOOLS AND TECHNOLOGIES USED**

1. **Java-Swing:** Swing is a set of program component/s for Java programmers that provide the ability to create graphical user interface (GUI) components, such as buttons and scroll bars, that are independent of the windowing system for specific operating system. Swing components are used with the Java Foundation Classes (JFC). The Java Foundation Classes (JFC) are a set of GUI components which simplify the development of desktop applications. Swing is built on the top of AWT (Abstract Windowing Toolkit) API and entirely written in java. Unlike AWT, Java Swing provides platform-independent and lightweight components.The javax.swing package provides classes for java swing API such as JButton , JTextField , JTextArea , JRadioButton , JCheckbox , JMenu , JColorChooser etc.[1]

The hierarchy of java swing API is given below:



1. **MySQL:** MySQL is a relational database management system based on the Structured Query Language, which is currently the most popular database management system software used for managing the relational database. It is open-source database software, which is supported by Oracle Company. It is fast, scalable, and easy to use database management system in comparison with Microsoft SQL Server and Oracle Database. MySQL is a [Relational Database Management System](https://www.javatpoint.com/what-is-rdbms) (RDBMS) software that provides many things, which are as follows:
   1. It allows us to implement database operations on tables, rows, columns, and indexes.
   2. It defines the database relationship in the form of tables (collection of rows and columns), also known as relations.
   3. It provides the Referential Integrity between rows or columns of various tables.
   4. It allows us to updates the table indexes automatically.
   5. It uses many SQL queries and combines useful information from multiple tables for the end-users.[2]
2. **Apache Server:** Apache is a popular open-source, cross-platform web server that is, by the numbers, the most popular web server in existence. It’s actively maintained by the Apache Software Foundation. Some high-profile companies using Apache include Cisco, IBM, Salesforce, General Electric, Adobe, VMware, Xerox, LinkedIn, Facebook, Hewlett-Packard, AT&T, Siemens, eBay, and many more. [3]

The Apache web server has modules which add more functions to its software, such as MPM (for handling multi-processing modes) or mod\_ssl for enabling SSL v3 and TLS support (suggested reading: TLS vs SSL). Some common features seen in Apache include:

* .htaccess
* IPv6
* FTP
* HTTP/2
* Perl, Lua, and PHP
* Bandwidth throttling
* WebDAV
* Load balancing
* URL rewriting
* Session tracking
* Geolocation based on IP address

1. **JDBC:** JDBC stands for Java Database Connectivity. JDBC is a Java API to connect and execute the query with the database. It is a part of JavaSE (Java Standard Edition). JDBC API uses JDBC drivers to connect with the database.[4] There are four types of JDBC drivers:

* JDBC-ODBC Bridge Driver,
* Native Driver,
* Network Protocol Driver, and
* Thin Driver

We can use JDBC API to access tabular data stored in any relational database. By the help of JDBC API, we can save, update, delete and fetch data from the database. It is like Open Database Connectivity (ODBC) provided by Microsoft



1. **XAMPP:** XAMPP is an abbreviation where X stands for Cross-Platform, A stands for Apache, M stands for MYSQL, and the Ps stand for PHP and Perl, respectively. It is an open-source package of web solutions that includes Apache distribution for many servers and command-line executables along with modules such as Apache server, MariaDB, PHP, and Perl. [5]

XAMPP helps a local host or server to test its website and clients via computers and laptops before releasing it to the main server. It is a platform that furnishes a suitable environment to test and verify the working of projects based on Apache, Perl, MySQL database, and PHP through the system of the host itself. Among these technologies, Perl is a programming anguage used for web development, PHP is a backend scripting language, and MariaDB is the most vividly used database developed by MySQL.

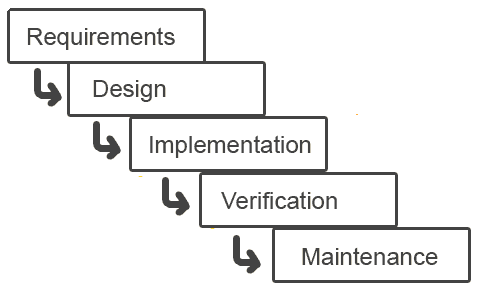
1. **Netbeans IDE:** NetBeans is a popular Integrated Development Environment (IDE), written in Java. It is primarily used for developing applications in Java, but it supports other languages too, such as PHP, C++, and HTML5. NetBeans is an open-source IDE that can be run on multiple platforms such as Windows, Linux, and Mac OS.[6]

Programming languages used: Java

Type: Integrated development environment

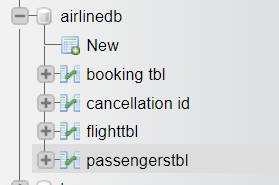
**SOFTWARE ENGINEERING APPROACH:**

The model I have used is Waterfall Model or Classic Life Cycle. In this model first of all the existed system is observed. Then customer requirements are taken in consideration then planning, modeling, construction and finally deployment.

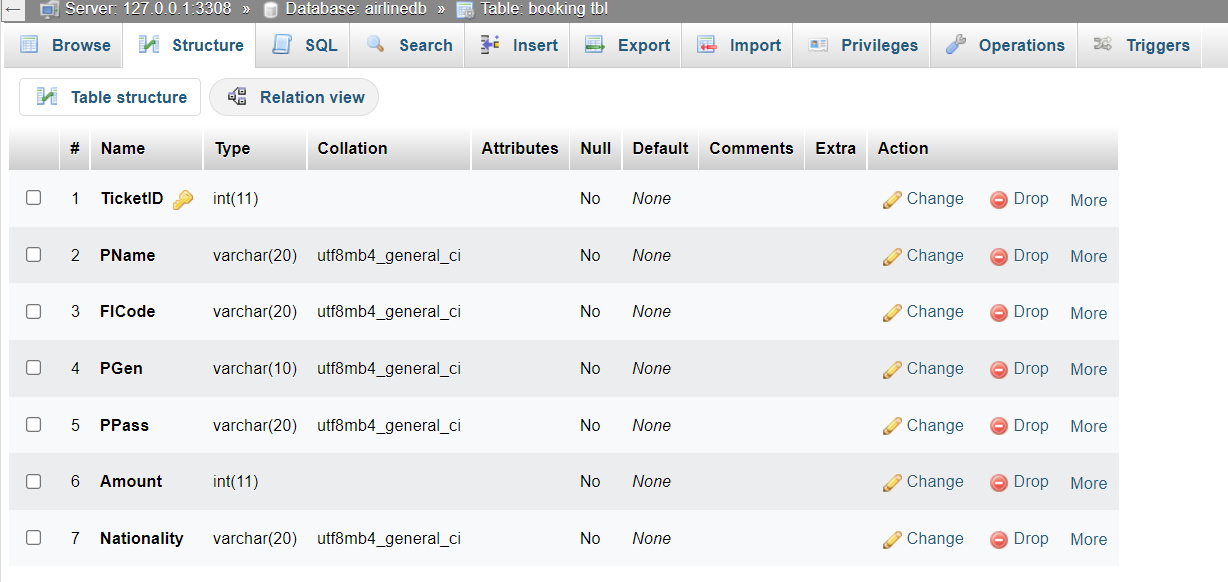
****

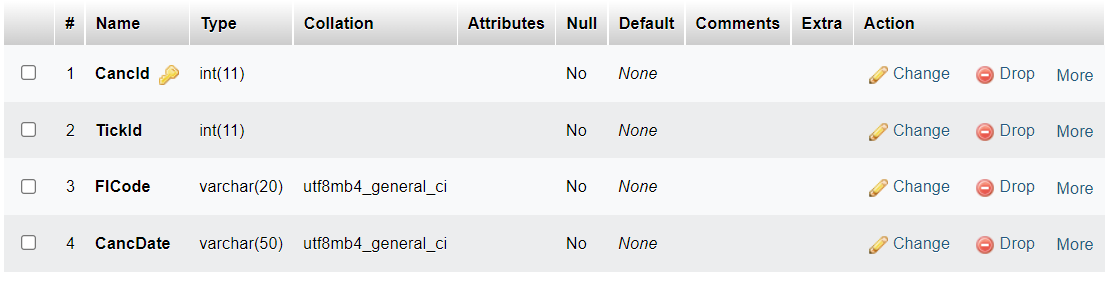
**DATABASE DESIGN**

Structure of the database

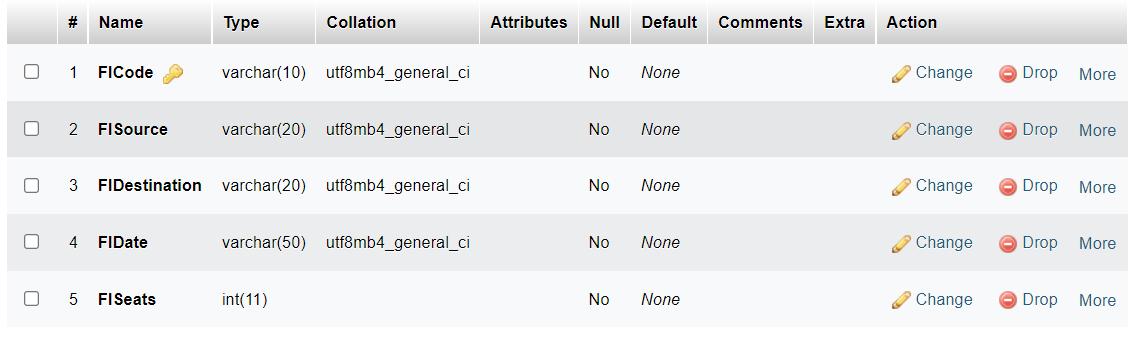


These were the following tables we made using MySQL for the database

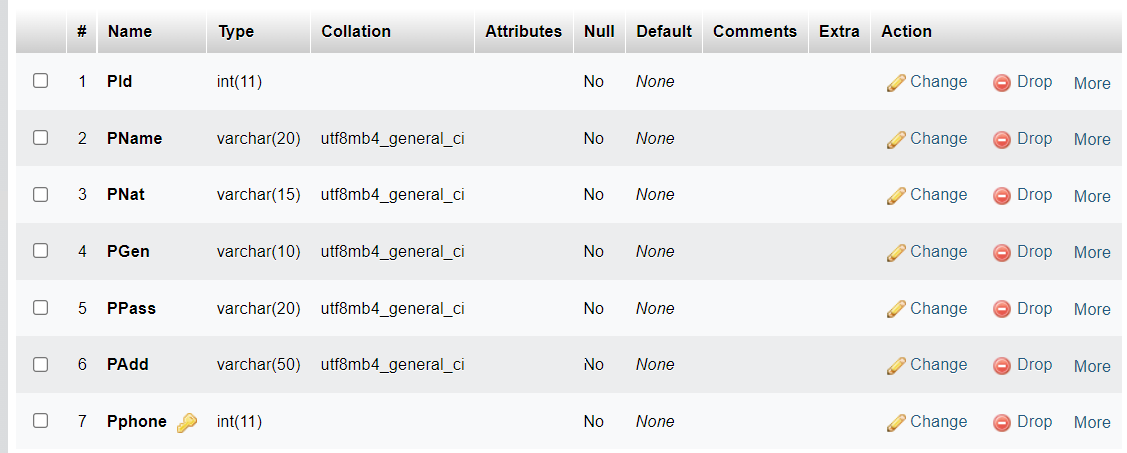
1. Booking Table: named as `booking tbl`
2. Cancellation Table: named as `cancellation id`



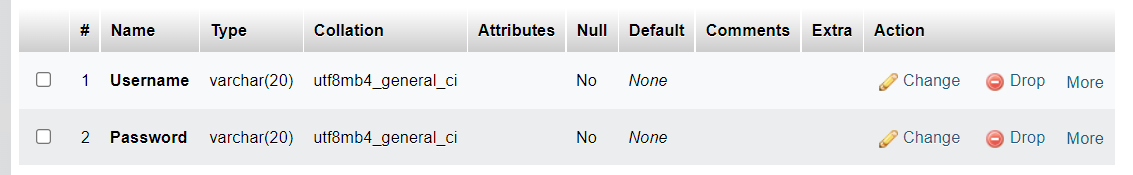
1. Flights Table: named as flighttbl



1. Passengers Table: named as passengerstbl

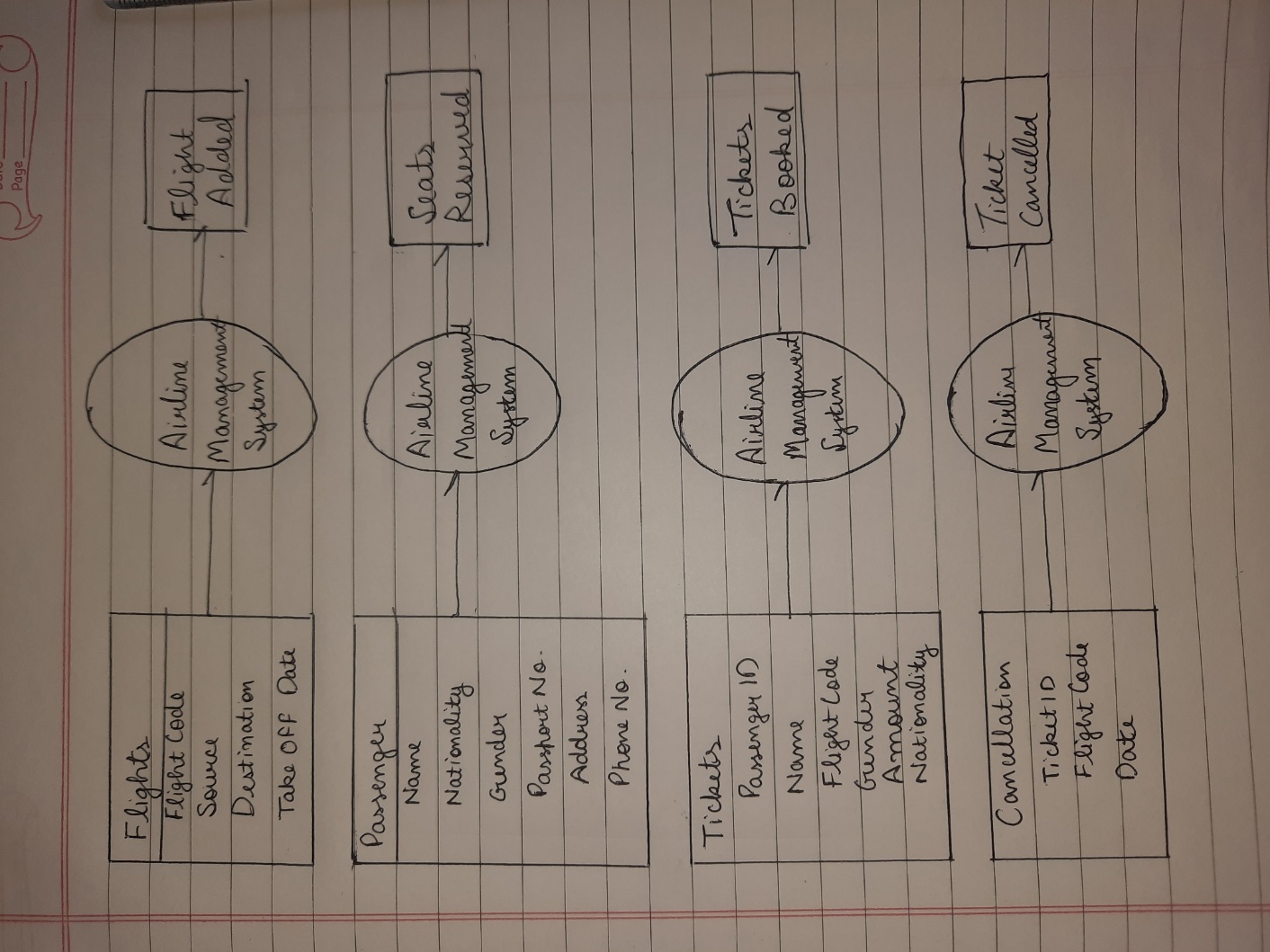


1. Admin: a table to store admin login information named as adminlogin

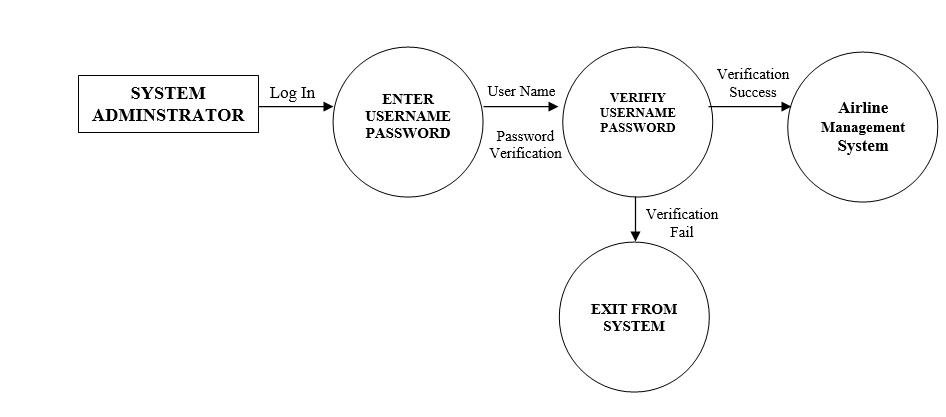


**DATA FLOW DIAGRAMS:**

Data Flow Diagram for Airline Management System



First Level of Data Flow Diagram for System Login

****

**SYSTEM DESIGN:**

**Introduction**

Analysis collects a great deal of unstructured data through interviews, questionnaires, on-site observations, and procedural manuals and like. It is required to organize and convert the data through system flowcharts, data flow diagrams, structured English, decision tables and the like which support future development of the system.

The Data flow diagrams and various processing logic techniques show how, where, and when data are used or changed in an information system, but these techniques do not show the definition, structure and relationships within the data.

It is a way to focus on functions rather than the physical implementation. This is analogous to the architect’s blueprint as a starting point for system design. The design is a solution, a “how to” approach, compared to analysis, a “what is” orientation.

System design is a highly creative process. This system design process is also referred as data modeling. The most common formatted used the E-R notation explains the characteristics and structure of data independent of how the data may be stored in computer memories.

The process of system design can be divided into three stages. They are:

* Structure design
* Database design (already discussed above)
* Interface design

As we know that system design is a solution to “How to approach to the creation of new system”. It provides the understudying and procedural details necessary for implementing the system. The steps involved during system design were as follow: -

**LOGICAL AND PHYSICAL DESIGN**

The current physical system was thoroughly reviewed from point of view how the data flow, what are file contents, its volumes and frequency etc.

After this input, output specifications security & control specification were prepared. It was also decided that how physical information will flow through the system and a physical design walkthrough.

**OUTPUT DESIGN**

The format of outputs is designed in such a way that it is simple to read and interpret In the present output we have clearly labeled all the fields.

**INPUT DESIGN**

.Input should be as simple as possible. It is design to reduce possibility of incorrect data being enter and the need of system user are considered with this view of mind several human factor is evaluated.

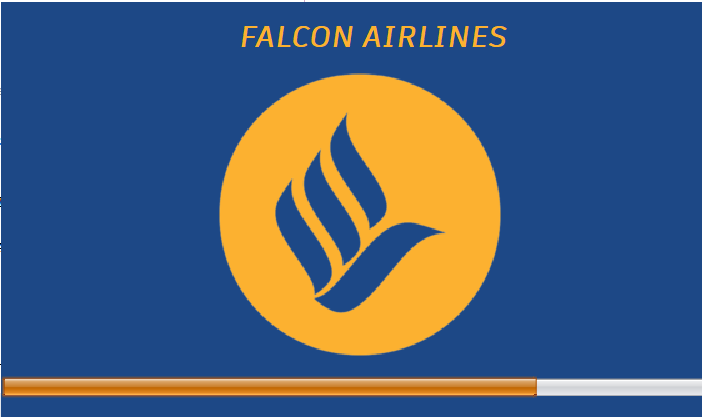
**SCREEN DESIGN**

The screen design for inputting the inputs were also panned as the format of inputs.

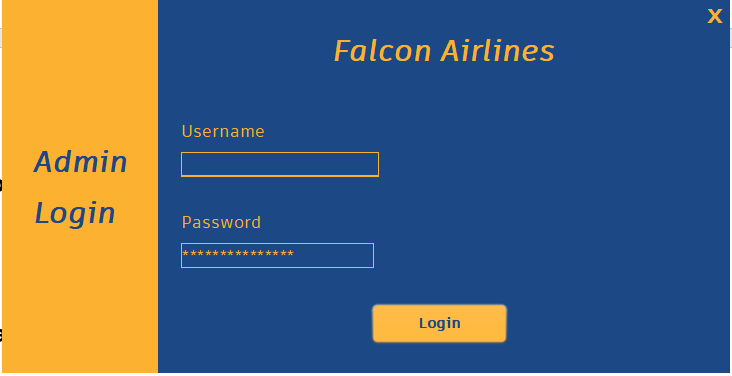
**GUI DESIGN AND PROGRAM CODE**

**GUI DESIGN**

**1. The loading screen (or splash screen) (Splash Module)**

****

**Login Page (Login Module)**

****

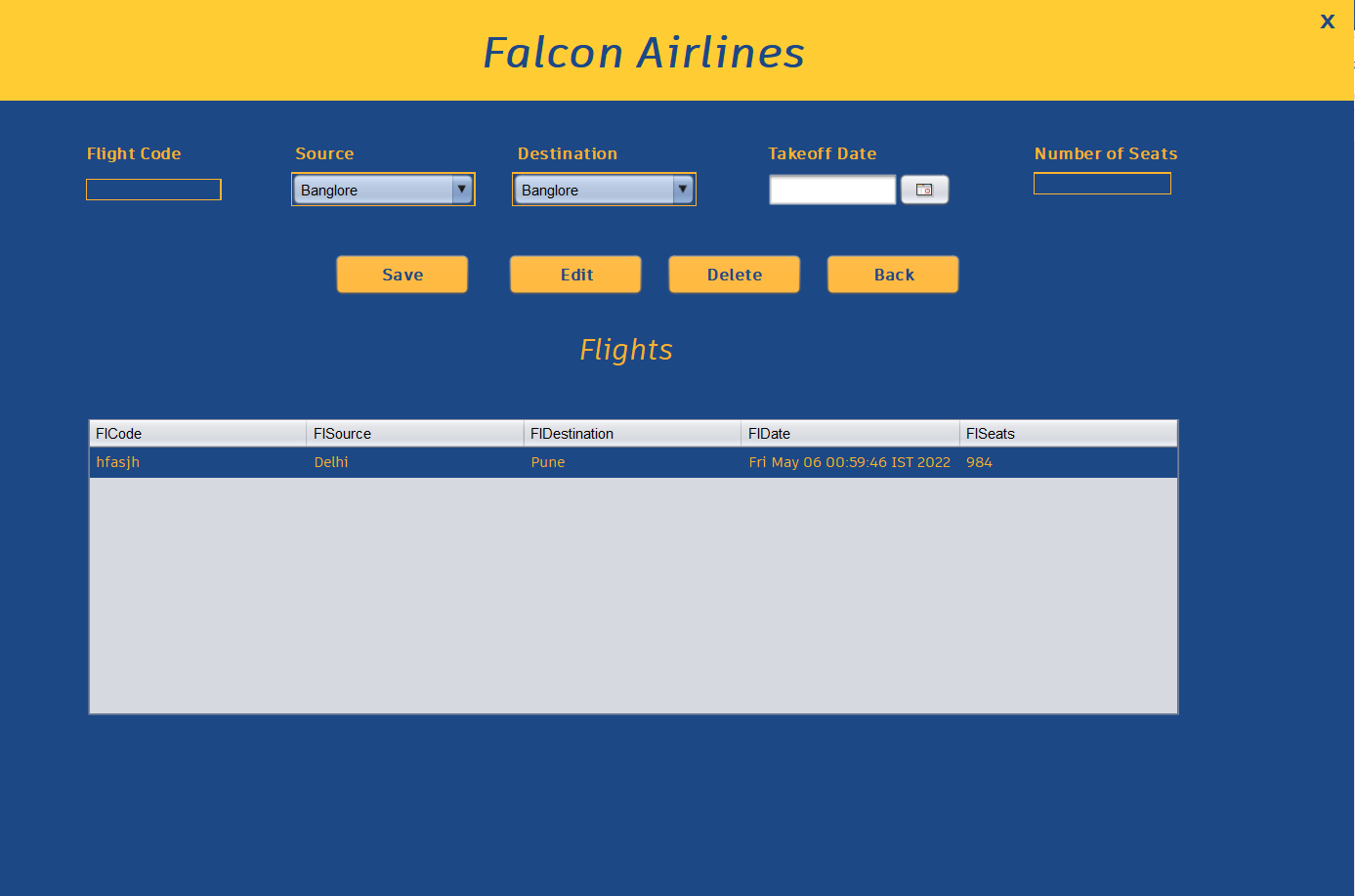
**MainForm Module**

****

**Passengers Module**



**Flights Module**

****

**Ticket Booking Module**

****

**Ticket Cancellation Module**

****

**Exception Handling**:

Exception handling in java is one of the powerful mechanisms to handle runtime errors caused by exceptions. Exception handling plays an important role in software development. This article helps you understand java exception, exception in java, java exception handling, java exception hierarchy, types of exception in java, and many more.

In this project we have used exception handling for general exceptions and for SQLExceptions also , we have used exception handling in these following module

* Passengers
* Flight
* Ticket Booking
* Ticket Cancellation

Example

Here is an example from the passenger’s module, where we used exception handling during saving the passenger’s information Into the database



Here we have used try-catch for exception handling

**Sample Codes:**

All the codes have been shown and submitted along with the report

Below is the code of passengers module for sample

**Passenger Module code:**

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to edit this template

\*/

package airlinemanagement;

import java.sql.\*;

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

import net.proteanit.sql.DbUtils;

/\*\*

\*

\* @author divya

\*/

public class Passengers extends javax.swing.JFrame {

/\*\*

\* Creates new form Passengers

\*/

public Passengers() {

initComponents();

DisplayPassengers();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel5 = new javax.swing.JPanel();

jPanel6 = new javax.swing.JPanel();

jLabel3 = new javax.swing.JLabel();

jLabel4 = new javax.swing.JLabel();

PassengerName = new javax.swing.JLabel();

jLabel8 = new javax.swing.JLabel();

jLabel9 = new javax.swing.JLabel();

jLabel10 = new javax.swing.JLabel();

PassNumTb = new javax.swing.JTextField();

jLabel11 = new javax.swing.JLabel();

NatCb = new javax.swing.JComboBox<>();

GenCb = new javax.swing.JComboBox<>();

PAddressTb = new javax.swing.JTextField();

DeleteBtn = new javax.swing.JButton();

BackBtn = new javax.swing.JButton();

EditBtn = new javax.swing.JButton();

SaveBtn = new javax.swing.JButton();

jLabel12 = new javax.swing.JLabel();

jScrollPane1 = new javax.swing.JScrollPane();

PassengersTable = new javax.swing.JTable();

PNameTb = new javax.swing.JTextField();

jLabel13 = new javax.swing.JLabel();

PPhoneTb = new javax.swing.JTextField();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

setUndecorated(true);

jPanel5.setBackground(new java.awt.Color(29, 72, 134));

jPanel6.setBackground(new java.awt.Color(255, 204, 51));

jLabel3.setFont(new java.awt.Font("Artifakt Element Book", 3, 36)); // NOI18N

jLabel3.setForeground(new java.awt.Color(29, 72, 134));

jLabel3.setText("Falcon Airlines");

jLabel4.setFont(new java.awt.Font("Corbel", 1, 24)); // NOI18N

jLabel4.setForeground(new java.awt.Color(29, 72, 134));

jLabel4.setText("x");

jLabel4.addMouseListener(new java.awt.event.MouseAdapter() {

public void mouseClicked(java.awt.event.MouseEvent evt) {

jLabel4MouseClicked(evt);

}

});

javax.swing.GroupLayout jPanel6Layout = new javax.swing.GroupLayout(jPanel6);

jPanel6.setLayout(jPanel6Layout);

jPanel6Layout.setHorizontalGroup(

jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel6Layout.createSequentialGroup()

.addGap(418, 418, 418)

.addComponent(jLabel3)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jLabel4)

.addGap(15, 15, 15))

);

jPanel6Layout.setVerticalGroup(

jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel6Layout.createSequentialGroup()

.addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel6Layout.createSequentialGroup()

.addGap(17, 17, 17)

.addComponent(jLabel3))

.addGroup(jPanel6Layout.createSequentialGroup()

.addContainerGap()

.addComponent(jLabel4)))

.addContainerGap(16, Short.MAX\_VALUE))

);

PassengerName.setFont(new java.awt.Font("Artifakt Element Book", 1, 14)); // NOI18N

PassengerName.setForeground(new java.awt.Color(252, 177, 48));

PassengerName.setText("Passenger Name");

jLabel8.setFont(new java.awt.Font("Artifakt Element Book", 1, 14)); // NOI18N

jLabel8.setForeground(new java.awt.Color(252, 177, 48));

jLabel8.setText("Gender");

jLabel9.setFont(new java.awt.Font("Artifakt Element Book", 1, 14)); // NOI18N

jLabel9.setForeground(new java.awt.Color(252, 177, 48));

jLabel9.setText("Passport Number");

jLabel10.setFont(new java.awt.Font("Artifakt Element Book", 1, 14)); // NOI18N

jLabel10.setForeground(new java.awt.Color(252, 177, 48));

jLabel10.setText("Address");

PassNumTb.setBackground(new java.awt.Color(29, 72, 134));

PassNumTb.setFont(new java.awt.Font("Artifakt Element Book", 0, 12)); // NOI18N

PassNumTb.setForeground(new java.awt.Color(252, 177, 48));

PassNumTb.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(252, 177, 48)));

PassNumTb.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

PassNumTbActionPerformed(evt);

}

});

jLabel11.setFont(new java.awt.Font("Artifakt Element Book", 1, 14)); // NOI18N

jLabel11.setForeground(new java.awt.Color(252, 177, 48));

jLabel11.setText("Nationality");

NatCb.setBackground(new java.awt.Color(29, 72, 134));

NatCb.setFont(new java.awt.Font("Artifakt Element Book", 0, 12)); // NOI18N

NatCb.setForeground(new java.awt.Color(252, 177, 48));

NatCb.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] { "India", "Usa", "UK" }));

NatCb.setBorder(new javax.swing.border.LineBorder(new java.awt.Color(252, 177, 48), 1, true));

GenCb.setBackground(new java.awt.Color(29, 72, 134));

GenCb.setFont(new java.awt.Font("Artifakt Element Book", 0, 12)); // NOI18N

GenCb.setForeground(new java.awt.Color(252, 177, 48));

GenCb.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] { "Male", "Female", " " }));

GenCb.setBorder(new javax.swing.border.LineBorder(new java.awt.Color(252, 177, 48), 1, true));

PAddressTb.setBackground(new java.awt.Color(29, 72, 134));

PAddressTb.setFont(new java.awt.Font("Artifakt Element Book", 0, 12)); // NOI18N

PAddressTb.setForeground(new java.awt.Color(252, 177, 48));

PAddressTb.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(252, 177, 48)));

DeleteBtn.setBackground(new java.awt.Color(252, 177, 48));

DeleteBtn.setFont(new java.awt.Font("Artifakt Element Book", 0, 14)); // NOI18N

DeleteBtn.setForeground(new java.awt.Color(29, 72, 134));

DeleteBtn.setText("Delete");

DeleteBtn.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 1, 1));

DeleteBtn.addMouseListener(new java.awt.event.MouseAdapter() {

public void mouseClicked(java.awt.event.MouseEvent evt) {

DeleteBtnMouseClicked(evt);

}

});

DeleteBtn.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

DeleteBtnActionPerformed(evt);

}

});

BackBtn.setBackground(new java.awt.Color(252, 177, 48));

BackBtn.setFont(new java.awt.Font("Artifakt Element Book", 0, 14)); // NOI18N

BackBtn.setForeground(new java.awt.Color(29, 72, 134));

BackBtn.setText("Back");

BackBtn.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 1, 1));

BackBtn.addMouseListener(new java.awt.event.MouseAdapter() {

public void mouseClicked(java.awt.event.MouseEvent evt) {

BackBtnMouseClicked(evt);

}

});

BackBtn.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

BackBtnActionPerformed(evt);

}

});

EditBtn.setBackground(new java.awt.Color(252, 177, 48));

EditBtn.setFont(new java.awt.Font("Artifakt Element Book", 0, 14)); // NOI18N

EditBtn.setForeground(new java.awt.Color(29, 72, 134));

EditBtn.setText("Edit");

EditBtn.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 1, 1));

EditBtn.addMouseListener(new java.awt.event.MouseAdapter() {

public void mouseClicked(java.awt.event.MouseEvent evt) {

EditBtnMouseClicked(evt);

}

});

EditBtn.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

EditBtnActionPerformed(evt);

}

});

SaveBtn.setBackground(new java.awt.Color(252, 177, 48));

SaveBtn.setFont(new java.awt.Font("Artifakt Element Book", 0, 14)); // NOI18N

SaveBtn.setForeground(new java.awt.Color(29, 72, 134));

SaveBtn.setText("Save");

SaveBtn.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 1, 1));

SaveBtn.addMouseListener(new java.awt.event.MouseAdapter() {

public void mouseClicked(java.awt.event.MouseEvent evt) {

SaveBtnMouseClicked(evt);

}

});

SaveBtn.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

SaveBtnActionPerformed(evt);

}

});

jLabel12.setFont(new java.awt.Font("Artifakt Element Book", 2, 20)); // NOI18N

jLabel12.setForeground(new java.awt.Color(252, 177, 48));

jLabel12.setText("Passengers List");

PassengersTable.setBackground(new java.awt.Color(29, 72, 134));

PassengersTable.setFont(new java.awt.Font("Artifakt Element Book", 1, 12)); // NOI18N

PassengersTable.setForeground(new java.awt.Color(252, 177, 48));

PassengersTable.setModel(new javax.swing.table.DefaultTableModel(

new Object [][] {

{null, null, null, null},

{null, null, null, null},

{null, null, null, null},

{null, null, null, null}

},

new String [] {

"Title 1", "Title 2", "Title 3", "Title 4"

}

));

PassengersTable.setRowHeight(25);

PassengersTable.setSelectionBackground(new java.awt.Color(252, 177, 48));

PassengersTable.addMouseListener(new java.awt.event.MouseAdapter() {

public void mouseClicked(java.awt.event.MouseEvent evt) {

PassengersTableMouseClicked(evt);

}

});

jScrollPane1.setViewportView(PassengersTable);

PNameTb.setBackground(new java.awt.Color(29, 72, 134));

PNameTb.setForeground(new java.awt.Color(252, 177, 48));

PNameTb.setBorder(new javax.swing.border.LineBorder(new java.awt.Color(252, 177, 48), 1, true));

jLabel13.setFont(new java.awt.Font("Artifakt Element Book", 1, 14)); // NOI18N

jLabel13.setForeground(new java.awt.Color(252, 177, 48));

jLabel13.setText("Phone Number");

PPhoneTb.setBackground(new java.awt.Color(29, 72, 134));

PPhoneTb.setFont(new java.awt.Font("Artifakt Element Book", 0, 12)); // NOI18N

PPhoneTb.setForeground(new java.awt.Color(252, 177, 48));

PPhoneTb.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(252, 177, 48)));

javax.swing.GroupLayout jPanel5Layout = new javax.swing.GroupLayout(jPanel5);

jPanel5.setLayout(jPanel5Layout);

jPanel5Layout.setHorizontalGroup(

jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel6, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel5Layout.createSequentialGroup()

.addGap(0, 122, Short.MAX\_VALUE)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel5Layout.createSequentialGroup()

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(PNameTb, javax.swing.GroupLayout.PREFERRED\_SIZE, 132, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(PassengerName, javax.swing.GroupLayout.PREFERRED\_SIZE, 159, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel11)

.addComponent(NatCb, javax.swing.GroupLayout.PREFERRED\_SIZE, 83, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(32, 32, 32)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel8)

.addComponent(GenCb, javax.swing.GroupLayout.PREFERRED\_SIZE, 83, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(36, 36, 36)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel9, javax.swing.GroupLayout.PREFERRED\_SIZE, 159, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(PassNumTb, javax.swing.GroupLayout.PREFERRED\_SIZE, 126, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(PAddressTb, javax.swing.GroupLayout.PREFERRED\_SIZE, 118, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel10, javax.swing.GroupLayout.PREFERRED\_SIZE, 159, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(PPhoneTb, javax.swing.GroupLayout.PREFERRED\_SIZE, 121, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel13))

.addGap(151, 151, 151))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel5Layout.createSequentialGroup()

.addComponent(SaveBtn, javax.swing.GroupLayout.PREFERRED\_SIZE, 112, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(30, 30, 30)

.addComponent(EditBtn, javax.swing.GroupLayout.PREFERRED\_SIZE, 112, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(18, 18, 18)

.addComponent(DeleteBtn, javax.swing.GroupLayout.PREFERRED\_SIZE, 112, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(18, 18, 18)

.addComponent(BackBtn, javax.swing.GroupLayout.PREFERRED\_SIZE, 112, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(291, 291, 291))))

.addGroup(jPanel5Layout.createSequentialGroup()

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel5Layout.createSequentialGroup()

.addGap(139, 139, 139)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 837, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(jPanel5Layout.createSequentialGroup()

.addGap(451, 451, 451)

.addComponent(jLabel12, javax.swing.GroupLayout.PREFERRED\_SIZE, 180, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

);

jPanel5Layout.setVerticalGroup(

jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel5Layout.createSequentialGroup()

.addComponent(jPanel6, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(44, 44, 44)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel5Layout.createSequentialGroup()

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel9)

.addComponent(jLabel10)

.addComponent(jLabel13))

.addGap(75, 75, 75)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(DeleteBtn, javax.swing.GroupLayout.PREFERRED\_SIZE, 35, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(EditBtn, javax.swing.GroupLayout.PREFERRED\_SIZE, 35, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(BackBtn, javax.swing.GroupLayout.PREFERRED\_SIZE, 35, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(SaveBtn, javax.swing.GroupLayout.PREFERRED\_SIZE, 35, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addGroup(jPanel5Layout.createSequentialGroup()

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(PassengerName)

.addComponent(jLabel8)

.addComponent(jLabel11))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(NatCb, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(GenCb, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(PNameTb, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(PassNumTb, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(PAddressTb, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(PPhoneTb, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))))

.addGap(26, 26, 26)

.addComponent(jLabel12)

.addGap(18, 18, 18)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 246, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(51, Short.MAX\_VALUE))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel5, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel5, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

pack();

setLocationRelativeTo(null);

}// </editor-fold>

private void DeleteBtnActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

private void BackBtnActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

private void EditBtnActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

private void SaveBtnActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

Connection Con = null;

PreparedStatement pst = null;

ResultSet Rs = null, Rs1=null;

Statement St = null, St1=null;

private void DisplayPassengers(){

try{

Con = DriverManager.getConnection("jdbc:mysql://localhost:3308/airlinedb","root","");

St = Con.createStatement();

Rs = St.executeQuery("select \* from PassengersTbl");

PassengersTable.setModel(DbUtils.resultSetToTableModel(Rs));

} catch(Exception e){

}

}

int PassId = 0;

private void CountPassengers()

{

try{

St1 = Con.createStatement();

Rs1 = St1.executeQuery("select Max(PId) from PassengersTbl");

Rs1.next();

PassId = Rs1.getInt(1)+1;

}catch(Exception e){

}

}

private void Clear(){

PNameTb.setText("");

PassNumTb.setText("");

PAddressTb.setText("");

PPhoneTb.setText("");

}

private void SaveBtnMouseClicked(java.awt.event.MouseEvent evt) {

if(PNameTb.getText().isEmpty() || PassNumTb.getText().isEmpty() || PAddressTb.getText().isEmpty() || PPhoneTb.getText().isEmpty() ){

JOptionPane.showMessageDialog(this, "Missing Information");

}else{

try {

CountPassengers();

Con = DriverManager.getConnection("jdbc:mysql://localhost:3308/airlinedb","root","");

PreparedStatement Add = Con.prepareStatement("insert into PassengersTbl values(?,?,?,?,?,?,?)");

Add.setInt(1,PassId);

Add.setString(2, PNameTb.getText());

Add.setString(3, NatCb.getSelectedItem().toString());

Add.setString(4, GenCb.getSelectedItem().toString());

Add.setString(5, PassNumTb.getText());

Add.setString(6, PAddressTb.getText());

Add.setString(7, PPhoneTb.getText());

int row = Add.executeUpdate();

JOptionPane.showMessageDialog(this, "Passenger Added");

Con.close();

DisplayPassengers();

Clear();

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e);

}

}

}

private void DeleteBtnMouseClicked(java.awt.event.MouseEvent evt) {

if(Key==0){

JOptionPane.showMessageDialog(this, "Select a Passenger");

}else{

try{

Con = DriverManager.getConnection("jdbc:mysql://localhost:3308/airlinedb","root","");

String Query = "Delete from PassengersTbl where PId="+Key;

Statement Del = Con.createStatement();

Del.executeUpdate(Query);

JOptionPane.showMessageDialog(this, "Passenger Deleted");

DisplayPassengers();

}catch(Exception e){

JOptionPane.showMessageDialog(this, e);

}

}

}

int Key=0;

private void PassengersTableMouseClicked(java.awt.event.MouseEvent evt) {

DefaultTableModel model = (DefaultTableModel)PassengersTable.getModel();

int MyIndex = PassengersTable.getSelectedRow();

Key = Integer.valueOf(model.getValueAt(MyIndex, 0).toString());

PNameTb.setText(model.getValueAt(MyIndex,1).toString());

NatCb.setSelectedItem(model.getValueAt(MyIndex,2).toString());

GenCb.setSelectedItem(model.getValueAt(MyIndex,3).toString());

PassNumTb.setText(model.getValueAt(MyIndex,4).toString());

PAddressTb.setText(model.getValueAt(MyIndex, 5).toString());

PPhoneTb.setText(model.getValueAt(MyIndex, 6).toString());

}

private void BackBtnMouseClicked(java.awt.event.MouseEvent evt) {

new MainForm().setVisible(true);

this.dispose();

}

private void EditBtnMouseClicked(java.awt.event.MouseEvent evt) {

if(Key == 0){

JOptionPane.showMessageDialog(this, "Select a Passenger");

}else{

try {

// CountPassengers();

Con = DriverManager.getConnection("jdbc:mysql://localhost:3308/airlinedb","root","");

String Query = "Update PassengersTbl set PName=?, PNat=?, PGen=?, Ppass=?, PAdd=?, Pphone=? where PId=?";

PreparedStatement Add = Con.prepareStatement(Query);

Add.setInt(7,Key);

Add.setString(1, PNameTb.getText());

Add.setString(2, NatCb.getSelectedItem().toString());

Add.setString(3, GenCb.getSelectedItem().toString());

Add.setString(4, PassNumTb.getText());

Add.setString(5, PAddressTb.getText());

Add.setString(6, PPhoneTb.getText());

int row = Add.executeUpdate();

JOptionPane.showMessageDialog(this, "Passenger Updated");

Con.close();

DisplayPassengers();

Clear();

} catch (Exception e) {

JOptionPane.showMessageDialog(this, e);

}

}

}

private void PassNumTbActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

private void jLabel4MouseClicked(java.awt.event.MouseEvent evt) {

System.exit(0);

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Passengers.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Passengers.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(Passengers.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(Passengers.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new Passengers().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton BackBtn;

private javax.swing.JButton DeleteBtn;

private javax.swing.JButton EditBtn;

private javax.swing.JComboBox<String> GenCb;

private javax.swing.JComboBox<String> NatCb;

private javax.swing.JTextField PAddressTb;

private javax.swing.JTextField PNameTb;

private javax.swing.JTextField PPhoneTb;

private javax.swing.JTextField PassNumTb;

private javax.swing.JLabel PassengerName;

private javax.swing.JTable PassengersTable;

private javax.swing.JButton SaveBtn;

private javax.swing.JLabel jLabel10;

private javax.swing.JLabel jLabel11;

private javax.swing.JLabel jLabel12;

private javax.swing.JLabel jLabel13;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;

private javax.swing.JLabel jLabel8;

private javax.swing.JLabel jLabel9;

private javax.swing.JPanel jPanel1;

private javax.swing.JPanel jPanel2;

private javax.swing.JPanel jPanel3;

private javax.swing.JPanel jPanel4;

private javax.swing.JPanel jPanel5;

private javax.swing.JPanel jPanel6;

private javax.swing.JScrollPane jScrollPane1;

// End of variables declaration

}

**Conclusion:**

Thus we have successfully implemented Airline Management System using Java Swing, MySQL and JDBC.

**References :**

**[1]** [**https://www.javatpoint.com/java-swing**](https://www.javatpoint.com/java-swing)

[**https://www.techtarget.com/whatis/definition/Swing**](https://www.techtarget.com/whatis/definition/Swing)

**[2]** [**https://www.javatpoint.com/mysql-tutorial**](https://www.javatpoint.com/mysql-tutorial)

**[3]** [**https://kinsta.com/knowledgebase/what-is-apache/**](https://kinsta.com/knowledgebase/what-is-apache/)

**[4]** [**https://www.javatpoint.com/java-jdbc**](https://www.javatpoint.com/java-jdbc)

**[5]** [**https://www.javatpoint.com/xampp**](https://www.javatpoint.com/xampp)

**[6]** [**https://www.tutorialspoint.com/netbeans\_online\_training/index.asp#:~:text=NetBeans%20is%20a%20popular%20Integrated,%2C%20Linux%2C%20and%20Mac%20OS**](https://www.tutorialspoint.com/netbeans_online_training/index.asp#:~:text=NetBeans%20is%20a%20popular%20Integrated,%2C%20Linux%2C%20and%20Mac%20OS)**.**