

**CMR ENGINEERING COLLEGE**

(Accredited by NBA, Approved by AICTE, NEW DELHI, Affiliated to JNTU, Hyderabad)

Kandlakoya, Medchal Road, R.R. Dist. Hyderabad-501 401)

**2021-2022**

**A PBL Report**

**On**

**AIRWAYS RESERVATION SYSTEM**

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING-CYBER SECURITY**

Submitted By

**J.Madhan(208R1A6228)**

**G.Pavan (208R1A6226)**

**D.Sharath (208R1A6215)**

**G.Saketh (208R1A6221)**

**S.Sri Ram (208R1A6253)**

Under the Guidance of

**Mrs. Amrutha**

Assistant Professor, Department of CSE-CS

** CMR ENGINEERING COLLEGE**

*(Accredited by NBA, Approved by AICTE NEW DELHI, Affiliated to JNTU, Hyderabad)*

*Kandlakoya, Medchal Road, Hyderabad-501 401*

**Department of Computer Science & Engineering-cyber security**

**CERTIFICATE**

This is to certify that the project entitled **“AIRWAYS RESERVATION SYSTEM”** is a bonafide work carried out

**J.Madhan(208R1A6228)**

**G.Pavan (208R1A6226)**

**D.Sharath (208R1A6215)**

**G.Saketh (208R1A6221)**

**S.Sri Ram (208R1A6253)**

In partial fulfilment of the requirement for the award of the degree of **BACHELOR OF TECHNOLOGY** in **CSE-CYBER SECURITY**  from CMR Engineering College, affiliated to JNTU, Hyderabad, under our guidance and supervision.

The results presented in this project have been verified and are found to be satisfactory. The results embodied in this project-based Learning has not been submitted to any other university for the award of any other degree or diploma.

Internal Guide Head of the Department

**Mrs.Amrutha Dr. Sathiya Suntharam** Assistant Professor Professor& HOD

Department of CSE-CS, Department of CSE-CS,

CMREC, Hyderabad CMREC, Hyderabad

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| SR.NO | TITLE | PAGE NO |
| 1. | **ABSTRACT** | **4** |
| 2. | **MODULES & REQUIREMENTS** | **5** |
| 3. | **INTRODUCTION** | **6-7** |
| 4. | **CYCLE OF AIRWAYS RESERVATION SYSTEM** | **8** |
| 5. | **SOURCE CODE** | **9-28** |
| 6. | **OUTPUTS** | **29-31** |

**Abstract**

**Objective:**

This software was build to provide assistance for the purpose of reserving tickets, or cancelling a reserved ticket, as well as rescheduling tickets. Since the information will be centralised so it can be easily updated in case of any changes to flight schedule. This minimises the need of help lines and customer support even at any locations. Thus it can help in minimizing the need for manual work.

This system has been developed to help passengers in reserving their tickets online, access information about reserved passengers and passengers on waiting, cancel a booked ticket, and even reschedule a ticket. The main disadvantage of a manual system is eradicated in this software, i.e., the introduction of an error due to manual negligence.

**Existing System:**

Manual systems are more prone to error. Hence many a times they are unreliable. In cases of any changes in the flight schedule or cancellation of any flight, if the changes are not reciprocated to the passengers on time it causes a lot of inconvenience and this affects the reliability of the airlines as well. Any discrepancy may cause a lot of chaos and trouble to the airlines as well as passengers.

**Proposed System:**

The airways reservation system provides passengers with an online means of booking a flight ticket as well as cancellation of a previously booked ticket. A passenger can also reschedule a ticket as per convenience and look up if there are seats available in the questioned flight or not. The passengers can stay updated with the flight information. This provides a lot of ease to the passengers as well as the airlines.

**Modules:**

1. Booking module:- This has two submodules namely: Reservation and Cancellation.
   1. Reservation: This modules lets a passenger to create a new passenger profile and book a ticket on an available flight.
2. Cancellation: This module allows the user to cancel a previously booked ticket
3. Reports module:- This module has two sub-modules:
   1. Confirmed Passenger List: This is a list of all the passengers, whose tickets are confirmed, i.e., the ticket has been successfully booked.
   2. Waiting List: This is the list of the passengers whose tickets are yet to be confirmed for booking.

Software Requirements

* Java Virtual Machine

Technology Used

* Java

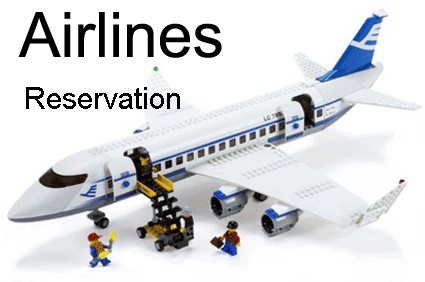
Hardware Requirements

* Hard Disk – 8 GB.
* RAM required – 512 GB (minimum)
* Mouse.
* Keyboard.
* Monitor.

**Introduction**

[Airways reservation systems](https://www.provab.com/airlinereservationsystemsoftware.html) include *passenger reservations, airline schedules, and ticket records*. An airline’s direct distribution works within their reservation system, as well as pushing out information to the GDS. The second type of direct distribution channel are consumers who use the internet or mobile applications to make their reservations.

Check-in is an important and the first procedure for an airport passenger, as passengers are required by airline regulations to check-in at certain moments before a flight departs. The*airline check-ins are one of the main functions* for the airline, it is also important to check and screen the luggage that goes into the aircraft for the passenger as well as the security of the airport.



By keying on a website on the browser of the [mobile phone](https://www.apple.com/in/iphone/)or downloading a specialized application, you can access the check-in function. Then the method is comparable to what one would expect when using a personal computer to check-in.

Some airlines give a *mobile boarding pass* to a mobile device that can be checked at the airport during safety checks and boarding at the end of the portable check-in cycle. Others, however, send a barcode electronic confirmation that can be submitted to the employees at check-in or checked at the kiosks to the check-in process.

Traveling is supposed to be a time when you get to let your hair down and relax. It is meant to be de-stressing and joyful. However, often the task of [managing a vacation](https://www.tripadvisor.in/) or creating its

arrangements becomes daunting. To make things easier, convenient and tension-free, travel portals were developed. Travel portals are internet booking management services that enable you to create reservations on a single platform for hotels, vehicles, buses, airlines, etc.

These [online reservation management systems](https://www.provab.com/onlinereservationsystem_onlinebookingsoftware.html) are created with the latest, cutting edge technology in the software arena. Countless companies are making such software, however, for creating a reliable and popular travel portal, you can only rely on the best. The latest travel technology software created by our company for making online reservation management systems has been designed by some of the brightest minds in our software development team. They are immensely qualified, with great expertise and experience in this arena. They have carried out research to assess the travel tools and technology clients look for or expect to see in an ideal travel portal. Based on this research we have come up with a travel technology software that is best in its class. This latest travel technology software is also very simple to use, high on privacy and security and comes with a user-friendly interface.

Our latest [travel technology software](https://www.provab.com/traveltechnologyfortravelsoftware.html) comes with some of the most amazing and unique features that will allow you to manage multiple sales channels such as B2C, B2B travel portal, B2B2C, and B2B2B. It will also enable them to offer multiple services such as Flights, Hotels, Package, Cars, Insurance, Sightseeing and Transfers through their online portal. The travel technology software we provide can be integrated with GDSs and third party suppliers for hotels, sightseeing, etc. It is an all in one suite to automate your travel agency operations to scale up with agility, flexibility, and elasticity.

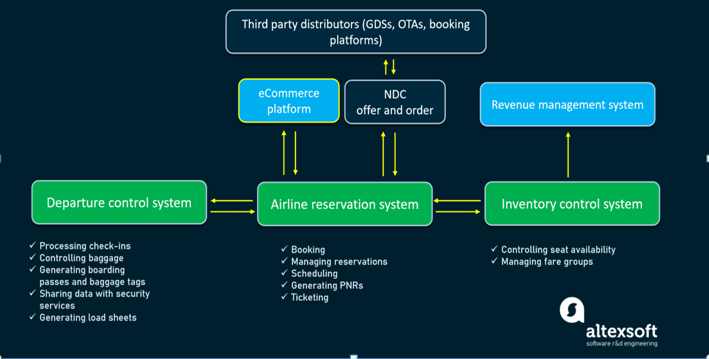
[The flight reservation system](https://www.provab.com/airlinereservationsystem-flightreservationsystem.html) is the way to the modern, scalable and robust architecture of reserving a flight. The most dominant issue that an airline face is the pressure of cost-cutting which slows down the entire growth of the airline industry. The success of the airline depends on two factors, one is the travelers and second is the technology used to book a flight.

As a part of the travel company, you must *hire professionals that can help you build a competitive application for flight.*A credible development firm for a travel software that helps you create a well-organized software for flight booking. A proper FRS (flight reservation system) is the reason for a sustained travel business. This helps customers to reserve flights in a matter of a few minutes hence, it becomes an important aspect of travel.

**Cycle of Airway Reservation system**

A**passenger service system (PSS)**is a suite of software modules, supporting interactions between the carrier and its customers. Its key components are a central or airline reservation system (CRS), an inventory control system (ICS), adeparture control system (DCS.)

The PSS also comprises or integrates with a [revenue management](https://www.altexsoft.com/blog/datascience/dynamic-pricing-explained-use-in-revenue-management-and-pricing-optimization/) tool that analyzes historical data and sets pricing rules and an e-commerce platform (airline website) for flight and ancillary distribution. Some key players among PSSs are [New Skies](https://www.navitaire.com/new-skies-reservation-system) by Navitaire, [Altea](https://amadeus.com/en/portfolio/airlines/altea-reservation-desktop-web-retailing-and-digital) by Amadeus, and [Avantik](https://www.bravo.aero/) by Bravo Aero.

****

**Source Code**

import java.sql.\*;

import java.awt.\*;

import java.awt.event.\*;

public class Cancellation extends Frame implements ActionListener

{

Label l1,l2,l3;

TextField t1;

Button b1,b2;

GridBagLayout gbl;

GridBagConstraints gbc;

Connection con;

PreparedStatement ps;

Statement stmt;

ResultSet rs;

int count;

Font f;

{

setBackground(Color.cyan);

f = new Font("TimesRoman",Font.BOLD,20);

gbl=new GridBagLayout();

gbc=new GridBagConstraints();

setLayout(gbl);

l1 = new Label("PNR No");

l1.setFont(f);

t1 = new TextField(20);

l2 = new Label("");

l3 = new Label("");

b1 = new Button("Submit");

b2 = new Button("Reset");

gbc.gridx=0;

gbc.gridy=0;

gbl.setConstraints(l1,gbc);

add(l1);

gbc.gridx=2;

gbc.gridy=0;

gbl.setConstraints(t1,gbc);

add(t1);

gbc.gridx=0;

gbc.gridy=2;

gbl.setConstraints(l2,gbc);

add(l2);

gbc.gridx=2;

gbc.gridy=2;

gbl.setConstraints(l3,gbc);

add(l3);

gbc.gridx=0;

gbc.gridy=4;

gbl.setConstraints(b1,gbc);

add(b1);

gbc.gridx=2;

gbc.gridy=4;

gbl.setConstraints(b2,gbc);

add(b2);

b1.addActionListener(this);

b2.addActionListener(this);

addWindowListener(new TU());

}

public void actionPerformed(ActionEvent ae)

{

if(ae.getSource()==b2)

{

t1.setText("");

}

if(ae.getSource()==b1)

{

try

{

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

con=DriverManager.getConnection("jdbc:odbc:MyDataSource1");

ps=con.prepareStatement("select FlightNo,TravelDate,Class from Passengers where PNRNo=?");

String pnrno = t1.getText();

ps.setInt(1,Integer.parseInt(pnrno));

rs=ps.executeQuery();

rs.next();

System.out.println(rs.getString(1)+""+rs.getString(2)+" "+rs.getString(3));

ps=con.prepareStatement("delete from Passengers where PNRNo=?");

ps.setInt(1,Integer.parseInt(pnrno));

count = ps.executeUpdate();

con.close();

t1.setText("");

}

catch(Exception e)

{

System.out.println("Error : "+e);

}

}

}

class TU extends WindowAdapter

{

public void windowClosing(WindowEvent e)

{

setVisible(false);

dispose();

}

}

}

import java.awt.\*;

import java.awt.event.\*;

public class Login extends Frame implements ActionListener

{

String username = "iamavnish";

String password = "sachin";

TextField t1,t2;

Label l1,l2,l3,l4,l5,l6;

Button b2,b3,b4;

GridBagLayout gbl;

GridBagConstraints gbc;

Font f1,f2;

public Login()

{

setBackground(Color.cyan);

f1 = new Font("TimesRoman",Font.BOLD,20);

f2 = new Font("TimesRoman",Font.BOLD,15);

gbl=new GridBagLayout();

gbc=new GridBagConstraints();

setLayout(gbl);

l1 = new Label("Username",Label.CENTER);

l1.setFont(f1);

l2 = new Label("Password",Label.CENTER);

l2.setFont(f1);

l3 = new Label("");

l4 = new Label("");

l5 = new Label("");

l6 = new Label("");

t1 = new TextField(15);

t2 = new TextField(15);

t2.setEchoChar('\*');

b2 = new Button("Reset");

b2.setFont(f2);

b3 = new Button("Submit");

b3.setFont(f2);

b4 = new Button("Close");

b4.setFont(f2);

gbc.gridx=0;

gbc.gridy=0;

gbl.setConstraints(l1,gbc);

add(l1);

gbc.gridx=2;

gbc.gridy=0;

gbl.setConstraints(t1,gbc);

add(t1);

gbc.gridx=0;

gbc.gridy=2;

gbl.setConstraints(l2,gbc);

add(l2);

gbc.gridx=2;

gbc.gridy=2;

gbl.setConstraints(t2,gbc);

add(t2);

gbc.gridx=0;

gbc.gridy=4;

gbl.setConstraints(l3,gbc);

add(l3);

gbc.gridx=2;

gbc.gridy=4;

gbl.setConstraints(l4,gbc);

add(l4);

gbc.gridx=0;

gbc.gridy=6;

gbl.setConstraints(b2,gbc);

add(b2);

gbc.gridx=2;

gbc.gridy=6;

gbl.setConstraints(b3,gbc);

add(b3);

gbc.gridx=0;

gbc.gridy=8;

gbl.setConstraints(l4,gbc);

add(l4);

gbc.gridx=2;

gbc.gridy=8;

gbl.setConstraints(l5,gbc);

add(l5);

gbc.gridx=0;

gbc.gridy=10;

gbl.setConstraints(b4,gbc);

add(b4);

b2.addActionListener(this);

b3.addActionListener(this);

b4.addActionListener(this);

}

public void actionPerformed(ActionEvent ae)

{

if(ae.getSource()==b2)

{

t1.setText("");

t2.setText("");

}

if(ae.getSource()==b4)

{

System.exit(0);

}

if(ae.getSource()==b3)

{

if((t1.getText().equals(username))&&(t2.getText().equals(password)))

{

MainMenu m = new MainMenu();

setVisible(false);

m.setSize(400,400);

m.setVisible(true);

m.setTitle("Main Menu");

}

else

{

MessageBox mb = new MessageBox(this);

mb.setLocation(200,200);

mb.setVisible(true);

}

}

}

}

import java.awt.\*;

import java.awt.event.\*;

public class MainMenu extends Frame implements ActionListener

{

MenuBar mbar;

Menu m1,m2,m3;

MenuItem m1\_1,m1\_2,m2\_1,m2\_2,m2\_3,m3\_1;

public MainMenu(){

mbar = new MenuBar();

setMenuBar(mbar);

m1=new Menu("Bookings");

mbar.add(m1);

m1\_1 = new MenuItem("Reservation");

m1.add(m1\_1);

m1\_2 = new MenuItem("Cancellation");

m1.add(m1\_2);

m2=new Menu("Reports");

mbar.add(m2);

m2\_1 = new MenuItem("Confirmed Passengers");

m2.add(m2\_1);

m2\_2 = new MenuItem("Waiting");

m2.add(m2\_2);

m2\_3 = new MenuItem("Daily Collection Report");

m2.add(m2\_3);

m3=new Menu("Close");

mbar.add(m3);

m3\_1 = new MenuItem("Close");

m3.add(m3\_1);

m1\_1.addActionListener(this);

m1\_2.addActionListener(this);

m2\_1.addActionListener(this);

m2\_2.addActionListener(this);

m2\_3.addActionListener(this);

m3\_1.addActionListener(this);

addWindowListener(new M());

}

public void actionPerformed(ActionEvent ae)

{

if(ae.getSource()==m1\_1)

{

Reservation r = new Reservation();

r.setSize(400,400);

r.setVisible(true);

r.setTitle("Reservation Screen");

}

if(ae.getSource()==m1\_2)

{

Cancellation c = new Cancellation();

c.setSize(400,400);

c.setVisible(true);

c.setTitle("Cancellation Screen");

}

if(ae.getSource()==m2\_1)

{

Confirmed cr = new Confirmed();

cr.setSize(400,400);

cr.setVisible(true);

cr.setTitle("Confirmed Passengers List");

}

if(ae.getSource()==m2\_2)

{

Waiting wr = new Waiting();

wr.setSize(400,400);

wr.setVisible(true);

wr.setTitle("Waiting Lis”);

}

if(ae.getSource()==m2\_3)

{

Collection dcr = new Collection();

dcr.setSize(400,400);

dcr.setVisible(true);

dcr.setTitle("Daily Collection Report")

}

if(ae.getSource()==m3\_1)

{

System.exit(0);

}

}

{

MainMenu m = new MainMenu();

m.setTitle("Main Menu");

m.setSize(400,400);

m.setVisible(true);

class M extends WindowAdapter

{

public void windowClosing(WindowEvent e)

{

setVisible(false);

dispose();

}

}

import java.sql.\*;

import java.awt.\*;

import java.awt.event.\*;

public class Ticket extends Frame implements ActionListener

{

TextField t1;

Label l1;

Button b1;

GridBagLayout gbl;

GridBagConstraints gbc;

Connection con;

PreparedStatement ps;

Statement stmt;

ResultSet rs;

int count;

Font f;

Ticket();

setBackground(Color.cyan);

t1 = new TextField(20);

l1 = new Label("PNR NO ");

l1.setFont(f);

gbc.gridx=0;

gbc.gridy=0;

gbl.setConstraints(l1,gbc);

add(l1);

gbc.gridx=0;

gbc.gridy=2;

gbl.setConstraints(t1,gbc);

add(t1);

addWindowListener(new W());

}

public void actionPerformed(ActionEvent ae)

{

}

class W extends WindowAdapter

{

public void windowClosing(WindowEvent e)

{

setVisible(false);

dispose();

}

}

{

Ticket t = new Ticket();

t.setSize(400,500);

t.setVisible(true);

t.setTitle("Ticket Screen");

}

}import java.awt.\*;

public class Project extends Frame

{

public static void main(String args[])

{

Login L = new Login();

L.setLocation(200,100);

L.setSize(300,300);

L.setVisible(true);

L.setTitle("Login Screen");

}

}import java.awt.\*;

import java.awt.event.\*;

public class Warning extends Frame

{

GridLayout g;

Button b1;

Label l;

Warning()

{

g = new GridLayout(2,1,10,40);

setLayout(g);

l = new Label("Incorrect username or password",Label.CENTER);

b1 = new Button("Ok");

add(l);

add(b1);

b1.addActionListener(new Y());

addWindowListener(new X());

}

class Y implements ActionListener

{

public void actionPerformed(ActionEvent ae)

{

if(ae.getSource()==b1)

{

//dispose();

System.exit(0);

}

}

}

class X extends WindowAdapter

{

public void windowClosing(WindowEvent e)

{

setVisible(false);

dispose();

}

}

public Insets getInsets()

{

return new Insets(40,40,40,40);

}

import java.awt.\*;

import java.awt.event.\*;

public class Reservation extends Frame implements ActionListener

{

Button b1,b2,b3;

Label l1,l2;

GridBagLayout gbl;

GridBagConstraints gbc;

Font f;

Reservation()

{

setBackground(Color.cyan);

f = new Font("TimesRoman",Font.BOLD,20);

gbl=new GridBagLayout();

gbc=new GridBagConstraints();

setLayout(gbl);

b1=new Button("Check Availability");

b1.setFont(f);

b2=new Button(" Create Passenger ");

b2.setFont(f);

l1= new Label("");

l2= new Label("");

gbc.gridx=0;

gbc.gridy=0;

gbl.setConstraints(b1,gbc);

add(b1);

gbc.gridx=0;

gbc.gridy=4;

gbl.setConstraints(l1,gbc);

add(l1);

gbc.gridx=0;

gbc.gridy=8;

gbl.setConstraints(b2,gbc);

add(b2);

b1.addActionListener(this);

b2.addActionListener(this);

addWindowListener(new W());

}

public void actionPerformed(ActionEvent ae)

{

if(ae.getSource()==b1)

{

Check m = new Check();

//setVisible(false);

m.setSize(400,400);

m.setVisible(true);

m.setTitle("Check Availability Screen");

}

if(ae.getSource()==b2)

{

Create v = new Create();

v.setSize(400,500);

v.setVisible(true);

v.setTitle("Create Passenger Screen");

}

}

class W extends WindowAdapter

{

public void windowClosing(WindowEvent e)

{

setVisible(false);

dispose();

}

}

}

}import java.awt.\*;

import java.awt.event.\*;

public class Waiting extends Frame

{

Waiting()

{

addWindowListener(new W());

}

class W extends WindowAdapter

{

public void windowClosing(WindowEvent e)

{

setVisible(false);

System.exit(0);

import java.awt.\*;

import java.awt.event.\*;

public class Warning extends Frame

{

GridLayout g;

Button b1;

Label l;

Warning()

{

g = new GridLayout(2,1,10,40);

setLayout(g);

l = new Label("Incorrect username or password",Label.CENTER);

b1 = new Button("Ok");

add(l);

add(b1);

b1.addActionListener(new Y());

addWindowListener(new X());

}

class Y implements ActionListener

{

public void actionPerformed(ActionEvent ae)

{

if(ae.getSource()==b1)

{

System.exit(0);

}

}

}

class X extends WindowAdapter

{

public void windowClosing(WindowEvent e)

{

setVisible(false);

dispose();

}

}

public Insets getInsets()

{

return new Insets(40,40,40,40);

}

{

Warning m = new Warning();

m.setTitle("Message Box");

m.setSize(300,200);

m.setVisible(true);

}

}

}

}

**Out Puts:**

