

# Online Airline Management System



## **BHARATI VIDYAPEETH'S COLLEGE OF ENGINEERING FOR WOMEN**

KATRAJ-DHANKAWADI, PUNE-43.

**INFORMATION TECHNOLOGY DEPARTMENT**

**A MINI PROJECT REPORT**

**ON**

## **“Online Airline Management System”**

Project Guided by

**Prof. A.V. Kanade**

Submitted By

**Ms. Anam Bagwan (Roll No: 3305)**

**Ms. Samiksha Bandgar (Roll No: 3307)**

**Ms. Nikita Bankar (Roll No: 3308)**

**Ms. Rutika Bhosale (Roll No: 3311)**

**Class: TE IT**

**Academic Year:2022-23**

# Online Airline Management System



## **BHARATI VIDYAPEETH'S COLLEGE OF ENGINEERING FOR WOMEN**

KATRAJ-DHANKAWADI, PUNE-43.

### **INFORMATION TECHNOLOGY DEPARTMENT**

#### **CERTIFICATE**

*This is to certify that the Project titled*

### **“Online Airline Management System”**

was prepared and presented by

**Ms. Anam Bagwan (Roll No: 3305)**

**Ms. Samiksha Bandgar (Roll No: 3307)**

**Ms. Nikita Bankar (Roll No: 3308)**

**Ms. Rutika Bhosale (Roll No: 3311)**

#### **Class: TE IT**

Under the guidance of

**Prof. A.V.Kanade**

Towards partial fulfillment of

TE IT course under

Savitribai Phule Pune University

(Year 2022-23)

**Prof. A.V. Kanade**

**Subject Teacher**

**Prof. Dr. D.A.Godse**

**HOD IT**

# Online Airline Management System

## ACKNOWLEDGEMENT

To complete any type of project work is team work. It involves all the technical/non-technical expertise from various sources. The contribution from the experts in the form of knows-how and other technical supports is of its vital importance. We are indebted to our inspiring guide **Prof. A.V. Kanade** who has extended all valuable guideline, help and constant encouragement through the various difficult stages for the onslaught of the project. We are also very thankful to our **HOD Prof. Dr. D.A. Godse** for their valuable guidance.

We have great pleasure in offering our honorable Principal **Prof. Dr. S. R. Patil** Last but not least, we would like to thanks all the direct and indirect help provided by the staff and our entire classmate for successful completion of this project.

**Date:**

**Ms. Anam Bagwan (Roll No: 3305)**

**Ms. Samiksha Bandgar (Roll No: 3307)**

**Ms. Nikita Bankar (Roll No: 3308)**

**Ms. Rutika Bhosale (Roll No: 3311)**

**T.E.I.T**

# Online Airline Management System

---

## **ABSTRACT**

Online Airline Management System contains the details about flight schedules and passenger reservations and ticket records. An airline inventory contains all flights with their available seats. The inventory of an airline service is generally divided into three category of classes along with prices and booking conditions. The airport database admin can update the information stored in the database. The passengers can check for the availability of the flight in the online application.

# Online Airline Management System

---

## INDEX

SR.NO	NAME	PAGE NO.
1.	Introduction	1
2.	Requirement Analysis	2
3.	Source code	3
4.	Graphical User Interface	6
5.	Conclusion	9
6.	References	10

# Online Airline Management System

## **INTRODUCTION**

The project specifies the working of an airline. It is an online airline management system. The project contains the details of passengers, flights, their bags, tickets, payments. The airport database admin can update the information stored in the database. The passengers can check for the availability of the flight in the online application.

The passenger can book the flight, update it, cancel it, search for his ticket incase of lost and also, we can see all the passenger information all together who has booked the flight. The passenger information is consistent in the database.

# Online Airline Management System

## REQUIREMENT ANALYSIS

### 3.1 HARDWARE REQUIREMENTS:

- Processor : Intel(R) core(TM) i3-5005 U CPU @ 2.00 GHZ 2.00GHZ
- RAM : 4GB
- Hard disk : 1TB

### SOFTWARE REQUIREMENTS:

- Operating System : Ubuntu, Windows 10/11
- Back end : MongoDB

### Resources required

Sr. No.	Name of Resource	Specification	Quantity
1.	Computer System/Laptop	Intel (R) Core(TM) i3/i5-6100 CPU@ 3.70GHz, 8.00 GB RAM, 1TB Hard-disk	1
2.	Software	MongoDB	-

## SOURCE CODE

# Online Airline Management System

## Home page source code:

```
import java.awt.*;
import javax.swing.*;
import javax.swing.border.Border;
import java.awt.event.*;
import com.mongodb.*;

public class Index extends JFrame implements ActionListener{
    JLabel nameofcom,slogan;
    Font ftop = new Font("Comic Sans MS",Font.BOLD,35);
    Font fbut = new Font("Comic Sans MS",Font.BOLD,25);
    JButton bpaxinsert,bdisplay,bflightinsert,bsearch,bupdate,bdelete;

    public Index(){
        super("OneWorld Airline");
        setSize(700,700);
        setVisible(true);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        p1 = new JPanel();
        p1.setBackground(Color.white);
        p2 = new JPanel();
        p2.setBackground(Color.cyan);
        p3 = new JPanel();
        p3.setBackground(Color.cyan);
        p4 = new JPanel();
        p4.setBackground(Color.white);
        p5 = new JPanel();
        p5.setBackground(Color.cyan);

        GridLayout g = new GridLayout(2,1,5,5);
        p1.setLayout(g);
        bpaxinsert = new JButton("Book a Flight");
        bdisplay = new JButton("Display Booked Flights");
        bflightinsert = new JButton("Add a new Flight");
        bsearch = new JButton("Search for a booking");
        bupdate = new JButton("Update/Change Booking");
        bdelete = new JButton("Cancel a Flight");

        bpaxinsert.setFont(fbut);
        bdisplay.setFont(fbut);
        bflightinsert.setFont(fbut);
        bsearch.setFont(fbut);
        bupdate.setFont(fbut);
    }
}
```



## Online Airline Management System

```
bdelete.setFont(fbut);

bpaxinsert.setForeground(Color.BLUE);
bdisplay.setForeground(Color.BLUE);
bflightinsert.setForeground(Color.BLUE);
bsearch.setForeground(Color.BLUE);
bupdate.setForeground(Color.BLUE);
bdelete.setForeground(Color.BLUE);

GridLayout gbu = new GridLayout(3,3,10,10);
p4.setLayout(gbu);
p4.add(bpaxinsert);
p4.add(bflightinsert);
p4.add(bdisplay);
p4.add(bsearch);
p4.add(bupdate);
p4.add(bdelete);
bpaxinsert.addActionListener(this);
bdisplay.addActionListener(this);
bflightinsert.addActionListener(this);
bsearch.addActionListener(this);
bupdate.addActionListener(this);
bdelete.addActionListener(this);

}

public void actionPerformed(ActionEvent e){
    if(e.getSource() == bflightinsert){
        setVisible(false);
        new InsertFlight();
    }
    if(e.getSource() == bpaxinsert){
        setVisible(false);
        new InsertPassangers();
    }
    if(e.getSource() == bupdate){
        setVisible(false);
        new UpdatePax();
    }
    if(e.getActionCommand().equals("Cancel a Flight")){
        setVisible(false);
        new DeletePax();
    }
    if(e.getSource() == bdisplay){
        setVisible(false);
        new DisplayAll();
    }
}
```

## Online Airline Management System

```
    }
    if(e.getSource() == bsearch){
        setVisible(false);
        new SearchPax();
    }
}
public static void main(String args[]){
    new Index();
}
}
```

- **Insert page source code:**

```
import java.awt.*;
import javax.swing.*;
import javax.swing.border.Border;
import java.awt.event.*;
import com.mongodb.*;
public class InsertPassangers extends JFrame implements ItemListener,
ActionListener{
    ArrayList<String> flightnoarray = null;
    JPanel p1,p2,p3,p4,p5;
    JTextArea add;
    String n1,n2,n3,n4,n5,d1[],dy[];
    int i;
    JTextField t[];
    static String[] info=new String[10];
    JComboBox<String> flightno,d,m,y;
    CheckboxGroup c=new CheckboxGroup();
    Checkbox c1=new Checkbox("BUSINESS",c,false);
    Checkbox c2=new Checkbox("ECONOMY",c,false);
    public InsertPassangers(){
        super("BOOK YOUR FLIGHT");
        setSize(950,670);
        setVisible(true);
        BorderLayout b=new BorderLayout();
        setLayout(b);
        add(BorderLayout.NORTH,p1);
        add(BorderLayout.CENTER,p2);
        add(BorderLayout.SOUTH,p3);
        add(BorderLayout.EAST,p4);
        add(BorderLayout.WEST,p5);
        p5.setBackground(Color.CYAN);
        p4.setBackground(Color.CYAN);

        //*****CENTER LAYOUT*****
    }
}
```

## Online Airline Management System

```
p2.setLayout(null);
p2.setLayout(gl);

lform=new JLabel[11];
for(i=0;i<lform.length;i++){
    if(i<3){
        lform[i]=new JLabel(labelstr[i],JLabel.CENTER);
    }
    else{
        lform[i]=new JLabel(labelstr[i]);
    }
    if(i>2){
        lform[i].setFont(fh1);
    }
}
MongoClient mongoClient = null;
DBCursor cursor = null;
try{
    mongoClient = new MongoClient( "localhost" , 27017 );
    DB db = mongoClient.getDB( "airline" );
    DBCollection coll = db.getCollection("flight");
    flightnoarray = new ArrayList<String>();
    cursor = coll.find();
    while(cursor.hasNext()) {
       DBObject obj = cursor.next();

        flightnoarray.add((String)obj.get("flight_no"));
    }
    cursor.close();
    mongoClient.close();
}
catch(Exception ex){
}

public void actionPerformed(ActionEvent e) {
    if(e.getSource()==register){
        int res;
        res=JOptionPane.showConfirmDialog(InsertPassangers.this,
        "Registered Successfully!! Continue?");
        System.out.println(res);
        switch(res){
            case 0:
                info[0]=t[0].getText();
                info[1]=t[1].getText();
                info[2]=t[2].getText();
                info[3]=add.getText();
                info[4]=(String) flightno.getSelectedItemAt();
```

## Online Airline Management System

```
if(c1.getState()){
    info[5] = "Business";
}
else{
    info[5] = "Economy";
}
info[6]=(String) d.getSelectedItem();
info[7]=(String) m.getSelectedItem();
info[8]=(String) y.getSelectedItem();

info[9]=t[3].getText();
for(i=0;i<10;i++){
    System.out.println(info[i]);
}
try {
    MongoClient mongoClient = null;
    mongoClient = new MongoClient(
        "localhost",27017);
    DB db = mongoClient.getDB("airline");

    DBCollection coll = db.getCollection("pax");
    System.out.println("Collection Accessed");
    BasicDBObject doc = new BasicDBObject("cust_id",
        info[0]).append("name",info[1]).append("mob_no",
        info[2]).append("address",info[3]).append("fligh
        t_no",info[4]).append("status",info[5]).append("
        flight_date",info[6]+" "+info[7]+"
        "+info[8]).append("amount",info[9]);
    if(t[1].getText().equals("")){
        JOptionPane.showMessageDialog(this,"PLEASE
        ENTER CUSTOMER ID","ERROR",
        JOptionPane.ERROR_MESSAGE);
    }

    coll.insert(doc);
    JOptionPane.showMessageDialog(this,"RECORD
    INSERTED SUCCESSFULLY!!!");
    this.dispose();
    new Index();
}
catch(Exception ex){
    JOptionPane.showMessageDialog(this,"INSERTION
    FAILED!!!","MESSAGE",JOptionPane.ERROR_MESSAGE);
}
this.dispose();
new Index();
break;
```

## Online Airline Management System

```
    }
}
if(e.getSource()==reset){
    for(i=0;i<4;i++){
        t[i].setText("");
    }
    add.setText("");
}
if(e.getSource()==home){
    this.dispose();
    new Index();
}
}
public void itemStateChanged(ItemEvent e){
    if (e.getSource() == flightno){
        try {
            MongoClient mongoClient = null;
            mongoClient = new MongoClient( "localhost",27017);
            DB db = mongoClient.getDB("airline");

            DBCollection coll = db.getCollection("flight");
            System.out.println("Collection established");
            BasicDBObject search11 = new BasicDBObject();
            search11.put("flight_no",flightno.getSelectedItem());
            System.out.println(flightno.getSelectedItem());
            DBCursor cursor11 = coll.find(search11);
            if(cursor11.hasNext())
            {
                System.out.println("wdymm");
                DBObject search111=cursor11.next();
            }
        }
        catch(Exception e1){

        }
    }
}
}
```

- **Update page source code:**

```
import java.awt.*;
import javax.swing.*;
import javax.swing.border.Border;
import java.awt.event.*;
import com.mongodb.*;
public class UpdatePax extends JFrame implements ActionListener{
```

## Online Airline Management System

```
JPanel p1,p2,p3,p4,p5;
int i;
JTextField t[];
static String[] info=new String[8];
static String[] lans = new String[8];
JLabel lform[],con;
JButton Search,update,home;
Font fl=new Font("Arial",Font.BOLD,18);
Font fh=new Font("Century Schoolbook",Font.BOLD,24);
Font fh1=new Font("Century Schoolbook",Font.BOLD,20);
GridLayout gl=new GridLayout(8,2,8,8);
GridLayout gh=new GridLayout(4,1);

public UpdatePax(){
    super("BOOK YOUR FLIGHT");
    setSize(950,670);
    setVisible(true);
    setLayout(b);
    p2.setLayout(null);
    p2.setLayout(gl);
    lform=new JLabel[11];
    for(i=0;i<lform.length;i++){
        if(i<3){
            lform[i]=new JLabel(labelstr[i],JLabel.CENTER);
        }
        else{
            lform[i]=new JLabel(labelstr[i]);
        }
        if(i>2){
            lform[i].setFont(fh1);
        }
    }
    if(e.getSource()==update){
        info[0] = t[0].getText();
        info[1] = t[1].getText();
        info[2] = t[2].getText();
        info[3] = t[3].getText();
        info[4] = t[4].getText();
        info[7] = t[7].getText();
        System.out.println(info[0]+info[1]);
        if(t[0].getText().equals("") || t[1].getText().equals("")){
            JOptionPane.showMessageDialog(this,"PLEASE ENTER ALL
            DETAILS","ERROR",JOptionPane.ERROR_MESSAGE);
        }
        else{
            MongoClient mongoClient = null;
```

## Online Airline Management System

```
mongoClient = new MongoClient( "localhost",27017);
@SuppressWarnings("deprecation")
DB db = mongoClient.getDB("airline");
DBCollection coll = db.getCollection("pax");
BasicDBObject document = new BasicDBObject();
document.put("cust_id",info[0]);
DBCursor cursor = coll.find(document);
if(cursor.hasNext()){
    DBObject obj = cursor.next();
    lans[0]= (String)obj.get("cust_id");
    lans[1]= (String)obj.get("name");
    lans[2]= (String)obj.get("mob_no");
    lans[3]= (String)obj.get("address");
    lans[4]= (String)obj.get("flight_no");
    lans[5]= (String)obj.get("status");
    lans[6]= (String)obj.get("flight_date");
    lans[7]= (String)obj.get("amount");
    coll.remove(document);
    lans[1] = t[1].getText();
    lans[2] = t[2].getText();
    lans[3] = t[3].getText();
    lans[6] = t[7].getText();
    BasicDBObject doc = new BasicDBObject("cust_id");
    coll.insert(doc);
    JOptionPane.showMessageDialog(this,"DETAILS
    UPDATED SUCCESSFULLY!!");
    this.dispose();
    new Index();
}
else {
    for(i=0;i<8;i++){
        t[i].setText("");
    }
    JOptionPane.showMessageDialog(this,"SOMETHING");
}
}
}
if(e.getSource()==home){
    this.dispose();
    new Index();
}
}
```

- **Delete Page source code:**

## Online Airline Management System

```
import java.awt.*;
import javax.swing.*;
import javax.swing.border.Border;
import java.awt.event.*;
import com.mongodb.*;

public class DeletePax extends JFrame implements ActionListener{
    public DeletePax(){
        super("BOOK YOUR FLIGHT");
        setSize(950,670);
        setVisible(true);

        BorderLayout b=new BorderLayout();

        p1=new JPanel();
        p1.setBackground(Color.red);

        p2=new JPanel();
        p2.setBackground(Color.white);

        p4=new JPanel();
        p4.setBackground(Color.white);

        p5=new JPanel();
        p5.setBackground(Color.white);

        p3=new JPanel();
        p3.setBackground(Color.yellow);

        setLayout(b);
        add(BorderLayout.NORTH,p1);
        add(BorderLayout.CENTER,p2);
        add(BorderLayout.SOUTH,p3);
        add(BorderLayout.EAST,p4);
        add(BorderLayout.WEST,p5);
        p5.setBackground(Color.CYAN);
        p4.setBackground(Color.CYAN);

        //BUTTON
        delete=new JButton("Cancel Booking");
        home=new JButton("Home");
        delete.setFont(fh1);
        home.setFont(fh1);
        JPanel rb=new JPanel();
        rb.setBackground(Color.white);
        GridLayout gb=new GridLayout(1,2,20,20);
```



## Online Airline Management System

```
rb.setLayout(gb);
rb.add(home);
rb.add(delete);
public void actionPerformed(ActionEvent e) {
if(e.getSource()==delete){
    info[0]=t[0].getText();
    info[1] = t[1].getText();
    info[2] = t[2].getText();
    System.out.println(info[0]);
    MongoClient mongoClient = null;
    mongoClient = new MongoClient( "localhost",27017);
    @SuppressWarnings("deprecation")
    DB db = mongoClient.getDB("airline");
    DBCollection coll = db.getCollection("pax");

    BasicDBObject document = new BasicDBObject();
    document.put("cust_id",info[0]);
    document.put("flight_no", info[1]);
    document.put("mob_no", info[2]);
    DBCursor cursor = coll.find(document);
    if(cursor.hasNext()){
        coll.remove(document);
        this.dispose();
        new Index();
    }
}
if(e.getSource()==home){
    this.dispose();
    new Index();
}
}
```

### Search Page source code:

```
import java.awt.*;
import javax.swing.*;
import javax.swing.border.Border;
import java.awt.event.*;
import com.mongodb.*;

public class SearchPax extends JFrame implements ActionListener{

    JPanel p1,p2,p3,p4,p5;
    int i;
    JTextField t[];
```

## Online Airline Management System

```
static String[] info=new String[1];
JLabel lform[],lans[],con;
JButton Search,reset,home;
Font fl=new Font("Arial",Font.BOLD,18);
Font fh=new Font("Century Schoolbook",Font.BOLD,24);
Font fh1=new Font("Century Schoolbook",Font.BOLD,20);
GridLayout gl=new GridLayout(11,2,8,8);
GridLayout gh=new GridLayout(4,1);

public SearchPax(){
    super("BOOK YOUR FLIGHT");
    setSize(950,670);
    setVisible(true);

    BorderLayout b=new BorderLayout();

    p1=new JPanel();
    p1.setBackground(Color.red);

    p2=new JPanel();
    p2.setBackground(Color.white);

    p4=new JPanel();
    p4.setBackground(Color.white);

    p5=new JPanel();
    p5.setBackground(Color.white);

    p3=new JPanel();
    p3.setBackground(Color.yellow);

    setLayout(b);
    add(BorderLayout.NORTH,p1);
    add(BorderLayout.CENTER,p2);
    add(BorderLayout.SOUTH,p3);
    add(BorderLayout.EAST,p4);
    add(BorderLayout.WEST,p5);
    p5.setBackground(Color.CYAN);
    p4.setBackground(Color.CYAN);
    //labels of forms
    for(i=0;i<lform.length;i++){
        if(i<3){
            lform[i]=new JLabel(labelstr[i],JLabel.CENTER);
        }
        else{
            lform[i]=new JLabel(labelstr[i]);
        }
    }
}
```

## Online Airline Management System

```
    }
    if(i>2){
        lform[i].setFont(fh1);
    }
}

//BUTTON
Search=new JButton("Search");
home=new JButton("Home");
reset=new JButton("Reset");
Search.setFont(fh1);
home.setFont(fh1);
reset.setFont(fh1);
JPanel rb=new JPanel();
rb.setBackground(Color.white);
GridLayout gb=new GridLayout(1,2,20,20);
rb.setLayout(gb);
rb.add(home);
rb.add(reset);

GridLayout gdob=new GridLayout(1,3,5,5);
gdob.setVgap(30);
rdob.setLayout(gdob);
JLabel temp = new JLabel("ENTER CUSTOMER ID: ");
temp.setFont(new Font("Century Schoolbook",Font.BOLD,20));
rdob.add(temp);
rdob.add(t[0]);
rdob.add(Search);
rdob.setBorder(BorderFactory.createMatteBorder(0, 10, 0, 0,
Color.CYAN));

Border border = BorderFactory.createLineBorder(Color.black, 1);
for(i=0;i<1;i++){
    t[i].setBorder(border);
}

p2.add(lform[3]);
p2.add(lans[0]);
p2.add(lform[4]);
p2.add(lans[1]);
p2.add(lform[5]);
p2.add(lans[2]);
p2.add(lform[6]);
p2.add(lans[3]);
p2.add(lform[7]);
p2.add(lans[4]);
```

## Online Airline Management System

```
p2.add(lform[8]);
p2.add(lans[5]);
p2.add(lform[9]);
p2.add(lans[6]);
p2.add(lform[10]);
p2.add(lans[7]);
p2.add(lform[11]);
p2.add(lans[8]);
p2.add(lform[12]);
p2.add(lans[9]);

p2.add(home);
p2.add(reset);
//          p2.setBorder(BorderFactory.createMatteBorder(3, 0, 0,
0, Color.BLACK));
//          p2.add(rb1);

//*****TOP LAYOUT*****
p1.setLayout(gh);
p1.add(lform[0]);
lform[0].setFont(fh);
lform[0].setForeground(Color.yellow);
lform[1].setForeground(Color.yellow);
lform[2].setForeground(Color.WHITE);
lform[1].setFont(f1);
p1.add(lform[1]);
lform[2].setFont(f1);
p1.add(lform[2]);
p1.add(rdob);

//Bottom layout
p3.setLayout(null);
FlowLayout fo=new FlowLayout();
p3.setLayout(fo);
con=new JLabel("For any queries Contact: oneworld@airline.com");
con.setFont(f1);
p3.add(con);
reset.addActionListener(this);
home.addActionListener(this);
Search.addActionListener(this);

}

@Override
public void actionPerformed(ActionEvent e) {
    if(e.getSource()==Search){
```

## Online Airline Management System

```
info[0]=t[0].getText();
System.out.println(info[0]);
if(t[0].getText().equals("")){
}
else{
    MongoClient mongoClient = null;
    DBCursor cursor = null;
    DBCursor cursor2 = null;
    DBCursor cursor3 = null;
    try {
        mongoClient=new MongoClient( "localhost",27017 );
        @SuppressWarnings("deprecation")
        DB db = mongoClient.getDB( "airline" );
        DBCollection coll = db.getCollection("pax");
        DBCollection coll3 = db.getCollection("pax_information");
        DBCollection coll2 = db.getCollection("flight");
        // cursor = coll.find();
        BasicDBObject search = new BasicDBObject();
        BasicDBObject search2 = new BasicDBObject();
        BasicDBObject search3 = new BasicDBObject();
        search.put("cust_id",info[0]);
        search3.put("cust_id",info[0]);
        cursor = coll.find(search);
        cursor3 = coll3.find(search3);
        if(cursor3.hasNext()){
            System.out.println("In cs3");
            DBObject obj3 = cursor3.next();

            System.out.println((String)obj3.get("cust_id"));

            lans[0].setText((String)obj3.get("cust_id"));
            lans[1].setText((String)obj3.get("name"));

            lans[2].setText((String)obj3.get("mob_no"));

            lans[3].setText((String)obj3.get("address"));
            if((String)obj3.get("flight_date") ==
null){
                lans[7].setText("NaN");
                lans[8].setText("NaN");
                lans[9].setText("NaN");
                //String
            flightno=(String)obj.get("flight_no");
                lans[4].setText("NaN");
            }
        }
    }
}
```

## Online Airline Management System

```
        }
    }

    search2.put("flight_no", flightno);
    cursor2 = coll2.find(search2);
    System.out.println(flightno+" in seatch");
    System.out.println(cursor2.hasNext());
    if(cursor2.hasNext()) {
        DBObject search11=cursor2.next();
    }
}
else {
    for(i=0;i<1;i++){
        t[i].setText("");
    }
    JOptionPane.showMessageDialog(this,"Booking");
}
cursor.close();
cursor2.close();
mongoClient.close();
}

}

if(e.getSource()==reset){
    for(i=0;i<1;i++){
        t[i].setText("");
    }
    for(i=0;i<lans.length;i++){
        lans[i].setText("NaN");
    }
}

if(e.getSource()==home){
    this.dispose();
    new Index();
}

}

public static void main(String str[]) {
    new SearchPax();
}

}
```

## GRAPHIC INTERFACE DESIGN

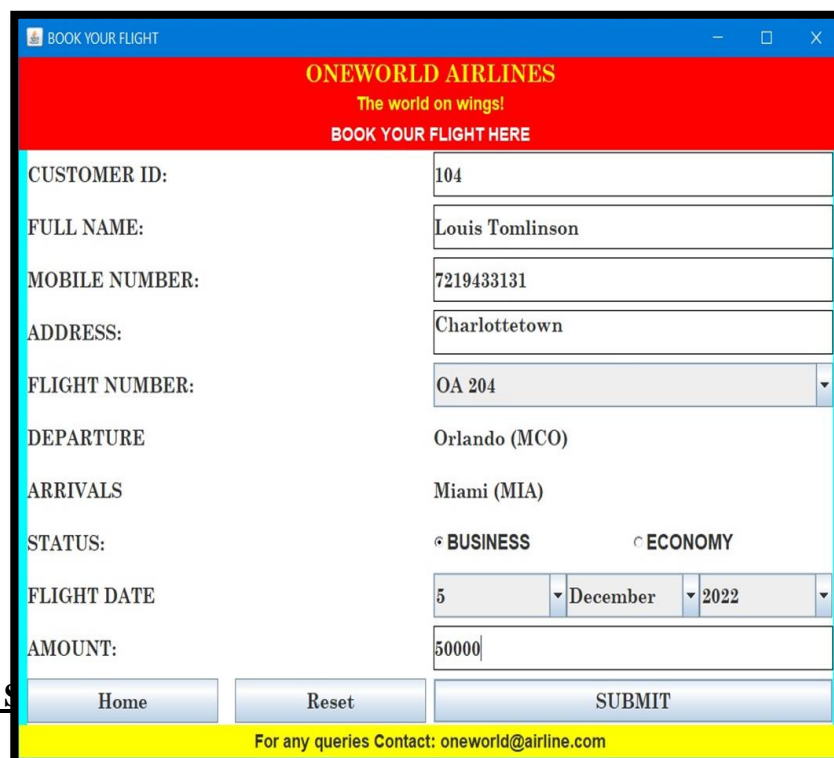
# Online Airline Management System

- **Output1: Home Page**



The screenshot shows a web browser window titled "OneWorld Airline". The page features the airline's logo, which includes a globe with a red star and the text "ONEWORLD AIRLINES" and "The world on wings." Below the logo, there is a grid of six buttons: "Book a Flight", "Add a new Flight", "Display Booked Flights", "Search for a booking", "Update/Change Booking", and "Cancel a Flight".

- **Output 2: Booking page:**



The screenshot shows a web browser window titled "BOOK YOUR FLIGHT". The page has a red header with the text "ONEWORLD AIRLINES", "The world on wings!", and "BOOK YOUR FLIGHT HERE". Below the header, there is a form with the following fields and values:

CUSTOMER ID:	104
FULL NAME:	Louis Tomlinson
MOBILE NUMBER:	7219433131
ADDRESS:	Charlottetown
FLIGHT NUMBER:	OA 204
DEPARTURE	Orlando (MCO)
ARRIVALS	Miami (MIA)
STATUS:	<input checked="" type="radio"/> BUSINESS <input type="radio"/> ECONOMY
FLIGHT DATE	5 December 2022
AMOUNT:	50000

At the bottom of the form, there are three buttons: "Home", "Reset", and "SUBMIT". Below the buttons, there is a yellow footer with the text "For any queries Contact: oneworld@airline.com".

- **Output 3: \$**

# Online Airline Management System

**BOOK YOUR FLIGHT**

**ONEWORLD AIRLINES**  
The world on wings!  
SEARCH YOUR FLIGHT HERE

ENTER CUSTOMER ID: 104

CUSTOMER ID: 104

FULL NAME: Louis Tomlinson

MOBILE NUMBER: 7219433131

ADDRESS: Charlottetown

FLIGHT NUMBER: OA 204

DEPARTURE: Orlando (MCO)

ARRIVALS: Miami (MIA)

FLIGHT DATE: 5 December 2022

CLASS: Business

AMOUNT: 50000

For any queries Contact: oneworld@airline.com

- Output 4: Update Page

**BOOK YOUR FLIGHT**

**ONEWORLD AIRLINES**  
The world on wings!  
UPDATE YOUR INFORMATION HERE

ENTER CUSTOMER ID: 104

FULL NAME: Louis William Tomlinson

MOBILE NUMBER: 7219433131

ADDRESS: Charlottetown, USA

FLIGHT NUMBER:

DEPARTURE: Orlando (MCO)

ARRIVALS: Miami (MIA)

FLIGHT DATE: 5 December 2022

For any queries Contact: oneworld@airline.com

**Message**  
DETAILS UPDATED SUCCESSFULLY!!



# Online Airline Management System

- **Output 5: Delete Page**

BOOK YOUR FLIGHT

**ONEWORLD AIRLINES**  
The world on wings!  
CANCEL YOUR TICKET HERE

CUSTOMER ID: 104

FLIGHT NUMBER: OA 204

MOBILE NUMBER: 7219433131

Home Cancel Booking

For any queries Contact: oneworld@airline.com

## **CONCLUSION**

The entire project has been developed and deployed as per the requirements stated by user, it is found to be bug free as per the testing standard that are implemented.

Computer does maximum work with in minimum time. Because it is used in every field so that it provides comfort and suitability to everyone. Providing maximum facilities and comfort to customers to customers is main goal of the firm. To achieve this goal, other modern facilities relating to computer should have to be provided.

# Online Airline Management System

## REFERENCES

- ❖ Ivan Bayross, “SQL, PL/SQL: The Programming Language of Oracle”, BPB Publication
- ❖ Silberschatz A., Korth H., Sudarshan S. “Database System Concepts”, 6th edition, Tata McGraw Hill Publishers
- ❖ G. K. Gupta “Database Management Systems” , Tata McGraw Hill
- ❖ <https://dev.mysql.com/doc/refman/8.0/en/>
- ❖ <https://ieeexplore.ieee.org/document/5347569>
- ❖ <https://www.cleartrip.com/flights>