NEW HORIZON
COLLEGE OF ENGINEERING
Autonomous College Permanently Affiliated to VTU, Approved by AICTE & UGC
Accredited by NAAC with 'A' Grade.

# THE INDIAN TRIPPERS

**A MINI PROJECT**

**REPORT**

***Submitted by***

**B. LAKSHMI KEERTHI**

***In partial fulfillment for the award of***

***the degree of***

**BACHELOR OF ENGINEERING**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

NEW HORIZON
COLLEGE OF ENGINEERING
Autonomous College Permanently Affiliated to VTU, Approved by AICTE & UGC
Accredited by NAAC with 'A' Grade.

# *Certificate*

*This is to certify that the mini project work titled*

## THE INDIAN TRIPPERS

*Submitted in partial fulfillment of the degree of Bachelor of Engineering in Computer Science and Engineering*

*Submitted by*

B. LAKSHMI KEERTHI

USN:1NH18CS038

*DURING*

*EVEN SEMESTER 20192020*

*For*

*COURSE CODE:19CSE48*

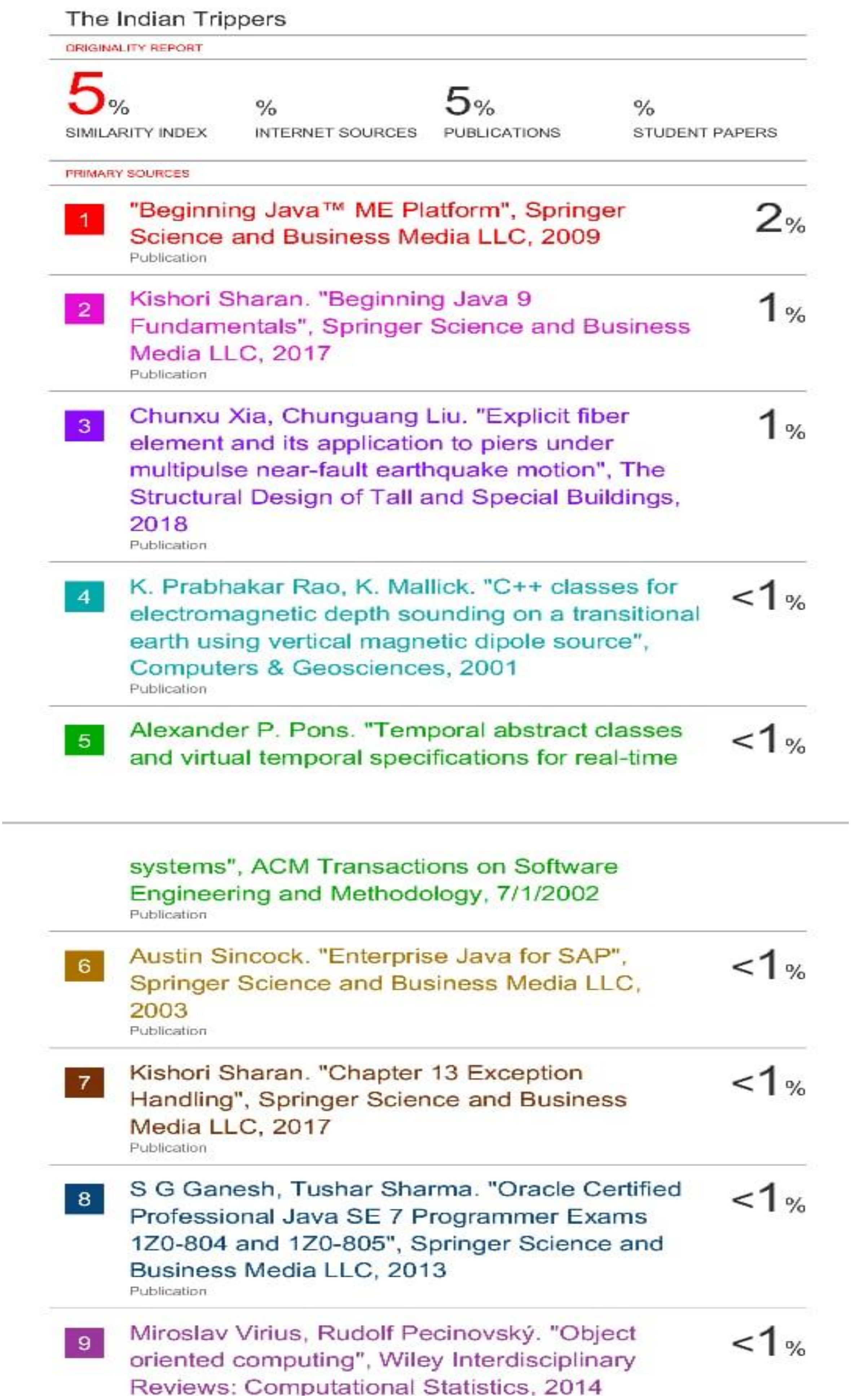
Signature of Reviewer Signature of HOD

## SEMESTER END EXAMINATION

*Name of the Examiner* *Signature with date*

1.__________________________ __________________________

2.__________________________ __________________________



The Indian Trippers

ORIGINALITY REPORT

| 5% | % | 5% | % |
|---|---|---|---|
| SIMILARITY INDEX | INTERNET SOURCES | PUBLICATIONS | STUDENT PAPERS |

PRIMARY SOURCES

| | | |
|---|---|---|
| 1 | "Beginning Java™ ME Platform", Springer Science and Business Media LLC, 2009<br>Publication | 2% |
| 2 | Kishori Sharan. "Beginning Java 9 Fundamentals", Springer Science and Business Media LLC, 2017<br>Publication | 1% |
| 3 | Chunxu Xia, Chunguang Liu. "Explicit fiber element and its application to piers under multipulse near-fault earthquake motion", The Structural Design of Tall and Special Buildings, 2018<br>Publication | 1% |
| 4 | K. Prabhakar Rao, K. Mallick. "C++ classes for electromagnetic depth sounding on a transitional earth using vertical magnetic dipole source", Computers & Geosciences, 2001<br>Publication | <1% |
| 5 | Alexander P. Pons. "Temporal abstract classes and virtual temporal specifications for real-time systems", ACM Transactions on Software Engineering and Methodology, 7/1/2002<br>Publication | <1% |
| 6 | Austin Sincock. "Enterprise Java for SAP", Springer Science and Business Media LLC, 2003<br>Publication | <1% |
| 7 | Kishori Sharan. "Chapter 13 Exception Handling", Springer Science and Business Media LLC, 2017<br>Publication | <1% |
| 8 | S G Ganesh, Tushar Sharma. "Oracle Certified Professional Java SE 7 Programmer Exams 1Z0-804 and 1Z0-805", Springer Science and Business Media LLC, 2013<br>Publication | <1% |
| 9 | Miroslav Virius, Rudolf Pecinovský. "Object oriented computing", Wiley Interdisciplinary Reviews: Computational Statistics, 2014 | <1% |

# ABSTRACT

In the world there are many places to visit. India is one among them. In the world many people are very fond of visiting India for its uniqueness. It was formed around 55,000 years ago. Each and every land scape has its own beauty. With thousands of years of rich history behind its India has some rather spectacular building and structures. Things that will blow your mind like beauty of Taj mahal etc. Everywhere you go you will find something new fascinating.

My project will help the person to explore all the things and during the peak times also. Any quires can contact the allocating members. This enables a person to view their data if it is confidential and they can login to select their more preferred choices. This project will also help the user to find out about the bus bookings, best hotels to stay and best restaurants to visit and moreover it displays the recommended items to try for the user. In this mini project the user easily accesses the traveling page even if he/she don't have any idea how to operate it.

According to their choice's user can select the places that is being displayed, if the user wants to know more about the city or the nearby places of that particular city this mini project can directly connect him to official page of the GOOGLE TRAVEL of that particular city. So, by this way the user can get the more information about the place in detailed so that he/she prepared what to carry (clothes, shoes etc.).

# ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be impossible without the mention of the people who made it possible, whose constant guidance and encouragement crowned our efforts with success.

I have great pleasure in expressing gratitude to **Dr. Mohan Manghnani**, Chairman of New Horizon Educational Institutions for providing necessary infrastructure and creating good environment.

I take this opportunity to express my profound gratitude to **Dr. Manjunatha,** Principal NHCE, for his constant support and encouragement.

I am grateful to **Dr. Prashanth C.S.R**, Dean Academics, for his unfailing encouragement and suggestions, given to me in the course of my project work.

I would also like to thank **Dr. B. Rajalakshmi**, Professor and Head, Department of Computer Science and Engineering, for her constant support.

I express my gratitude to **Dr.S.Sridevi,** Associate professor, my project guide, for constantly monitoring the development of the project and setting up precise deadlines. Her valuable suggestions were the motivating factors in completing the work.

B LAKSHMI KEERTHI

1NH18CS038.

# CONTENTS

# CHAPTER 1

# INTRODUCTION

## 1.1 PROBLEM STATEMENT:

In the present generation, the number of people visiting the Indian places is increasing largely. Every year many people from all over the world visit India and they find it difficult to find the right travel and accommodation. Though people have a break they find it difficult to book or plan to travel a trip. So, this mini project helps the user to explore India through travel by displaying best hotel and restaurants all under one roof. This project will ease the process of finding the best among the available options to the user. Hence this project will ease the proceedings related to the traveling.

## 1.2 OBJECTIVES:

1. A web page is been created where it displays 2 options that is LOGIN and SIGNUP.
2. If the user is new it provides sign up page to register and if the user is already existing it provides a login page to login again.
3. A booking page for the user based on his preferred city.
4. A general enquiry page for the user to get the basic details about the available options.
5. To provide the user with the best possible available options in a particular city for a specific date range.
6. To provide the user with hassle free experience which allow them to focus more on their experience.

## 1.3 METHODOLOGY TO BE FOLLOWED:

1. Firstly, we need to create 8 divisions: The main introduction form, login, signup form, welcome form to explore, Puna, Bangalore, Hyderabad, Chennai.
2. In signup/login part, the user will register with his basic details (like username, password, email, phone….) and use his username and password to login.
3. The welcome form should allow the user to select the city of their trip and provide best available options based on their preference.
4. In the all the web pages the user can get know the best bus, best hotels to reside and best restaurants.
5. We will provide the user with general information of the particular city of what he/she selected.
6. For visiting our website, a small thanks giving page will be shown.

## 1.4 EXPECTED OUTCOMES:

1. My mini project will help the tourists to explore the cities in India and displays best hotel accommodation and best restaurants accommodations according to user specifications.

2. This user can operate easily.

3. It provides best in the class customized and improved services.

4. This project will reduce the paper work and simplify the transactions.

5. It provides best food in highly recommended restaurants present in that particular city.

6.Any quires we can able to contact the numbers that is been displayed.

## 1.5 SOFTWARE AND HARDWARE REQUIREMENTS:

1.Software Requirements-

a. NetBeans IDE7.2

b. MYSQL Data base.

2.Hardware Requirements-

a. Processor: Intel core i5 generation or more.

b. Windows 10.

# CHAPTER 2

# OBJECT ORIENTED CONCEPTS

As the name suggests, Object-Oriented Programming or the OOPs refers to languages that uses objects in programming. Object-oriented programming aims to implement real-world entities like inheritance, hiding, polymorphism etc. in programming. The main aim of OOP is to bind together the data and the functions that operate on them so that no other part of the code can access this data except that functions.

## 2.1 CLASS

A class is a user define blueprint from which objects are created. It represents all the set of properties or method that are common to all objects of one type. In general, class declarations can include these components, in order:

Modifiers: A class can be public or has default access.

Class name: The name should be public for the default modifiers.

Superclass: The name of the class's parents, if any, preceded by the keyword extends. A class can only extend one parent.

Interface: A common separated list of interfaces implemented by the class, if any, preceded by the keyword implements. Class can implement more than one interface.

Body: The class body surrounded by the braces {}

## 2.2 OBJECT:

It is the basic unit of the Object-Oriented Programming and represents the real-life entities. A typical java program creates many objects, which as you know, interact by invoking methods. An object consists of:

State: It represents the attributes of the objects. It also reflects the properties of an objects.

Behavior: It is represented by the methods of an object.it reflects the response of an objects with the other objects.

Identity: It gives a unique name to an object and enables one object to interact with other objects.

Syntax: Box obj=new Box ();

Class-var = new class name ();

## 2.3 INHERITANCE:

Inheritance is an important pillar of OOPS.it is the mechanism in java by which one class is allow to inherit the features of another class.

Important terminology:

- Super class: the class whose features are inherited is known as super class
- Sub class: The class that inherits the other class is known as subclass. The subclass can add its own fields and methods addition to the superclass fields and methods.
- Reusability: Inheritance supports the concept of “reusability”, i.e. when we want to create a new class and there is already a class that includes some of the code that we want, we derive our new class from the existing class. By doing this, we are reusing the fields and methods of the existing class.

## 2.4 ABSTRACT CLASS:

ABSTRACT: Data Abstraction is the property by virtue of which only the essential details are displayed to the user. The trivial or the non-essentials units are not displayed to the user. EX: A car is viewed as a car rather than its individual components.

Data Abstraction may also be defined as the process of identifying only the required characteristics of an object ignoring the irrelevant details. The properties and behaviors of an object differentiate it from other objects of similar type and also help in classifying/grouping the objects.

Consider a real-life example of a man driving a car. The man only knows that pressing the accelerators will increase the speed of car or applying brakes will stop the car but he does not know about how on pressing the accelerator the speed is actually increasing, he does not know about the inner mechanism of the car or the implementation of accelerator, breaks etc. in the car. This is what abstraction is.

In java, abstraction is achieved by interfaces and abstract classes. We can achieve 100% abstraction using interfaces.

## 2.5 I/O FUNCTION:

The I/O package, is which it provides a set of input streams and a set of output streams which is used to read and write data to the files or other input and output sources. There are three categories of classes in java I/O: input streams, output streams and everything else.

**INPUT STREAMS**

Input streams can read the data from an input source. An input source can be a file, a string, or a memory anything which can contain a data. All input streams inherit from input stream where an abstract class that defines the programming interface for all the input streams.

The input stream class defines a programming interface for different reading bytes or the arrays of bytes, marking locations in the stream, skipping bytes of input, finding out the number of bytes those which are available for reading, and resetting the current position within the stream. An input stream is opened automatically when you create it. You can explicitly close a stream with the close() method, or keeps it be closed implicitly when the object is garbage or differently collected.

**OUTPUT STREAMS**

Output streams can write the data to an output source similar to input sources, an output source can be done anything that can contain data like a file, a string, or memory.

The output stream class is a sibling to input stream and it is used to write data that which can then be read by an input stream. The output stream class defines a programming interface for writing bytes or an array of bytes to the stream and passing it to the stream. Like an input stream, an output stream is opened automatically when you create it. You can explicitly close an output stream with the close () method, and let’s be closed implicitly when the object is collected in the form of garbage.

## 2.6 JAVA PACKAGES:

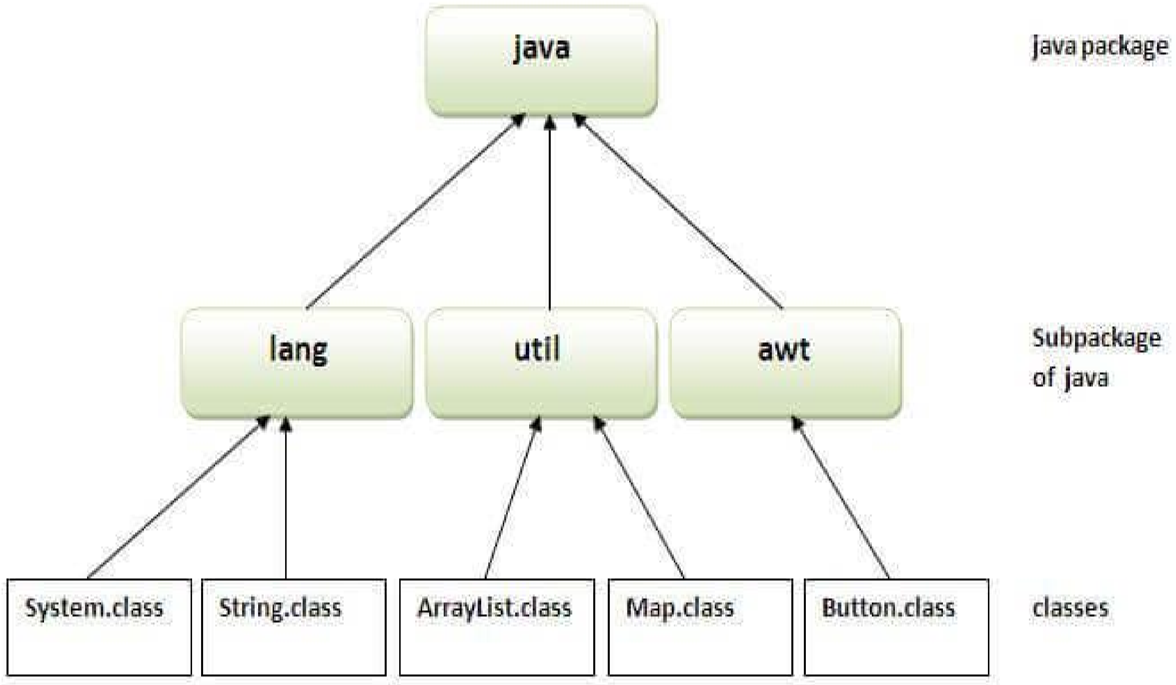
- A java package is a group of similar types of classes, interfaces and sub-packages.
  - Package in java can be categorized in two form, built-in package and user-defined package.
  - There are many built-in packages such as java, lang, awt, javax, swing, net, io, util, sql etc.

**Advantage of Java Package**

1) Java package is used to categorize the classes and interfaces so that they can be easily maintained.

2) Java package provides access protection.

3) Java package removes naming collision.



## 2.7 EXCEPTION HANDLINGS:

- The Exception Handling in Java is one of the powerful mechanisms to handle the runtime errors so that normal flow of the application can be maintained.
- Exception Handling is mechanism to handle runtime errors such as ClassNotFoundException, IOException, SQLException, RemoteException, etc.

**Advantage of Exception Handling**:

The core advantage of exception handling is to maintain the normal flow of the application. An exception normally disrupts the normal flow of the application that is why we use exception handling.

**Types of Java Exception**

There are mainly two types of exceptions: checked and unchecked.

- According to Oracle, there are three types of exceptions:
  - Checked Exception
  - Unchecked Exception
  - Error

- Difference between Checked and Unchecked Exceptions

1. **Checked Exception:**

   The classes which directly inherit Throwable class except RuntimeException and Error are known as checked exceptions e.g. IOException, SQLException etc. Checked exceptions are checked at compile-time.

2. **Unchecked Exception**:

   The classes which inherit RuntimeException are known as unchecked exceptions e.g. ArithmeticException, NullPointerException, ArrayIndexOutOfBoundsException etc. Unchecked exceptions are not checked at compile-time, but they are checked at runtime.

3. **Error:**

   Error is irrecoverable e.g. OutOfMemoryError, VirtualMachineError, AssertionError etc.

- **Java try block**
  - Java try block is used to enclose the code that might throw an exception. It must be used within the method.
  - If an exception occurs at the particular statement of try block, the rest of the block code will not execute. So, it is recommended not to keeping the code in try block that will not throw an exception.

- **Java catch block**
  - Java catch block is used to handle the Exception by declaring the type of exception within the parameter. The declared exception must be the parent class exception (i.e., Exception) or the generated exception type.
  - The catch block must be used after the try block only. You can use multiple catch block with a single try block.

# CHAPTER 3

# DESIGN GOALS

## _3.1 FLOW-CHART

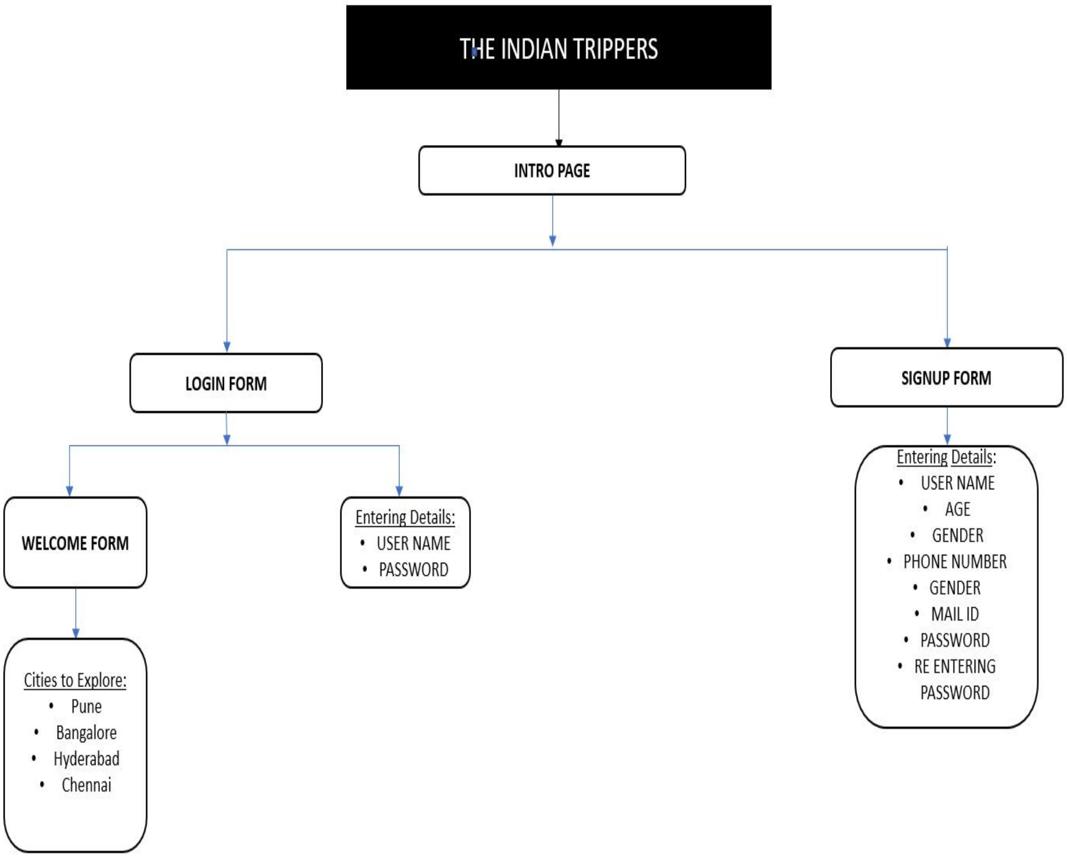


Fig 3.1 The opening of the Indian Trippers

When the user opens the page of Indian trippers then he/she will get a display form in that page the user can be able to see two options one is login and the other is sign in so if the user is quite new to the Indian trippers then he/she should go and sign in where basic questions like name, age, gender, phone number etc. is been asked after this sign in process the data of the user will be stored confidentially and the user will get back to the login page so he/she can be logged in easily.

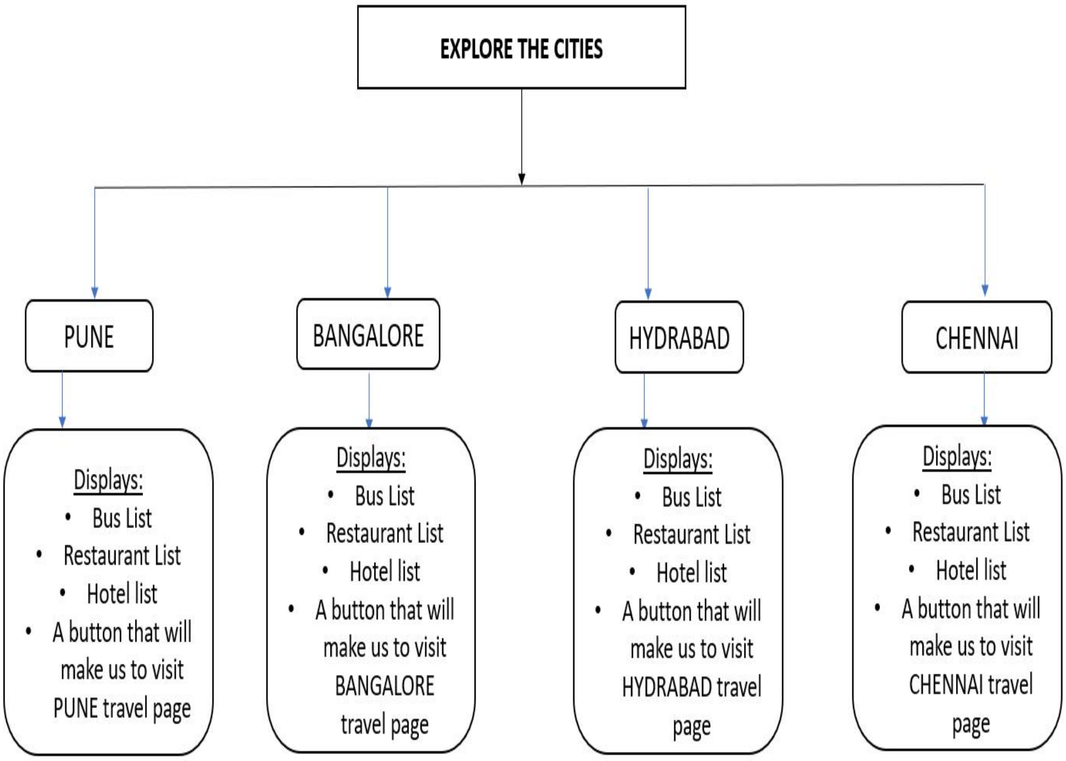


Fig 3.2 The displaying the cities to explore.

After the user logged in to the Indian trippers then a welcome form is been displayed so that he/she can select the place where they want to visit here in this form it displays four places i.e.

1. Pune
2. Bangalore
3. Hyderabad
4. Chennai

So, if the user selects any of the cities then he/she will be able to see the buses availability, hotel availability and restaurants in that particular places. one option is been provided that if user wants to explore more things, he/she can click on that button so that it will connect the user to the official Google travel page of that particular city.

# CHAPTER 4

# IMPLEMENTATION

## 4.1MODULE 1:

The first module is Intro page. In this module the user will be given a brief intro on what Indian Tripper is and also provided with options to either Login for existing or Signup if it is a new user.

```
public class INTRO extends java. swing. Frame {
   public INTRO () {
      initComponents ();
   }


   private void initComponents() {


      jPanel1 = new javax.swing.JPanel();
      jLabel1 = new javax.swing.JLabel();
      jButton1 = new javax.swing.JButton();
      jButton2 = new javax.swing.JButton();
      jLabel2 = new javax.swing.JLabel();
      jLabel4 = new javax.swing.JLabel();



      jPanel1.setLayout(null);


      jLabel1.setFont(new java.awt.Font("Comic Sans MS", 3, 36));
```

```
jLabel1.setText("WELCOME TO THE INDIAN TRIPPERS!!!");
jPanel1.add(jLabel1);
jLabel1.setBounds(60, 110, 740, 54);


jButton1.setFont(new java.awt.Font("Comic Sans MS", 2, 18));
jButton1.setText("LOGIN");
jButton1.addActionListener(new java.awt.event.ActionListener() {
   public void actionPerformed(java.awt.event.ActionEvent evt) {
      jButton1ActionPerformed(evt);
   }
});
jPanel1.add(jButton1);
jButton1.setBounds(40, 455, 130, 30);


jButton2.setFont(new java.awt.Font("Comic Sans MS", 2, 18));
jButton2.setText("SIGNUP");
jButton2.addActionListener(new java.awt.event.ActionListener() {
   public void actionPerformed(java.awt.event.ActionEvent evt) {
      jButton2ActionPerformed(evt);
   }
});
jPanel1.add(jButton2);
jButton2.setBounds(660, 450, 140, 30);
```

```
    jLabel2.setIcon(new javax.swing.ImageIcon(getClass().getResource("/THE INTRO.jpg")));

    jLabel2.setText("jLabel2");

    jPanel1.add(jLabel2);

    jLabel2.setBounds(0, 0, 830, 540);


    jLabel4.setText("jLabel4");

    jPanel1.add(jLabel4);

    jLabel4.setBounds(390, 40, 51, 20);


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

    getContentPane().setLayout(layout);

    layout.setHorizontalGroup(

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

      .addGroup(layout.createSequentialGroup()

        .addGap(27, 27, 27)

        .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE, 836, javax.swing.GroupLayout.PREFERRED_SIZE)

        .addContainerGap(56, Short.MAX_VALUE))

    );

    layout.setVerticalGroup(

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

      .addGroup(layout.createSequentialGroup()

        .addGap(47, 47, 47)
```

```
            .addComponent(jPanel1,      javax.swing.GroupLayout.PREFERRED_SIZE,      542,
javax.swing.GroupLayout.PREFERRED_SIZE)

            .addContainerGap(31, Short.MAX_VALUE))

      );


      pack();

   }


   private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

      FormLogin obj=new FormLogin ();

      obj.setVisible(true);

      dispose();

   }


   private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {

      SIGNUPFORM obj=new SIGNUPFORM ();

      obj.setVisible(true);

      dispose();

   }

   public static void main(String args[]) {


      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(INTRO.class.getName()).log(java.util.logging.Level.SEV
ERE, null, ex);
    } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(INTRO.class.getName()).log(java.util.logging.Level.SEV
ERE, null, ex);
    } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(INTRO.class.getName()).log(java.util.logging.Level.SEV
ERE, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(INTRO.class.getName()).log(java.util.logging.Level.SEV
ERE, null, ex);
    }
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new INTRO().setVisible(true);
        }
    });
    }
```

## 4.2 MODULE 2:

This module has two parts

1) Sign Up

2) Login

For new user, the user must sign up, for this they will be asked to provide the basic details like Name, password, email, gender etc. to. Once entered, the details will be stored in a table in MYSQL database.

Once the user register, he/she can use the login page to enter the website. They will be prompted to enter their username and password, once entered it will be validated with the details stored in the backend MYSQL database, on successful validation, the user will be allowed to use the website further. If the validation fails, the user will be prompted with unsuccessful login and they won't be able to book the tickets.

## 4.3 MODULE 3:

After the login then the exploring form is showed been showed in that there will be displaying 4 cities option i.e.

1. Pune
2. Bangalore
3. Hyderabad
4. Chennai

In this page one "back" option is also been provided so that after pressing the button it gets the user back to the Intro page.

```
public class welcomeform extends javax.swing.JFrame {


    public welcomeform() {
        initComponents ();

    }


    private void initComponents() {


        jLabel1 = new javax.swing.JLabel();

        jButton1 = new javax.swing.JButton();

        jLabel3 = new javax.swing.JLabel();

        jLabel6 = new javax.swing.JLabel();

        jButton3 = new javax.swing.JButton();

        jButton4 = new javax.swing.JButton();

        jButton5 = new javax.swing.JButton();

        jButton6 = new javax.swing.JButton();

        jLabel4 = new javax.swing.JLabel();


        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        getContentPane().setLayout(null);

        getContentPane().add(jLabel1);

        jLabel1.setBounds(200, 75, 0, 0);


        jButton1.setFont(new java.awt.Font("Comic Sans MS", 2, 18));

        jButton1.setText("BACK TO INTRO");

        jButton1.addMouseListener(new java.awt.event.MouseAdapter() {

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

```
        jButton1MouseClicked(evt);

      }

    });

    getContentPane().add(jButton1);

    jButton1.setBounds(490, 530, 190, 35);


    jLabel3.setFont(new java.awt.Font("Comic Sans MS", 2, 36));

    jLabel3.setText("EXPLORE NOW!!!!");

    getContentPane().add(jLabel3);

    jLabel3.setBounds(60, 80, 420, 51);


    jLabel6.setFont(new java.awt.Font("Comic Sans MS", 3, 36));

    jLabel6.setForeground(new java.awt.Color(255, 255, 51));

    jLabel6.setText("SELECT A PLACE THAT YOU WANT TO EXPLORE!!!!");

    getContentPane().add(jLabel6);

    jLabel6.setBounds(90, 210, 950, 51);


    jButton3.setFont(new java.awt.Font("Comic Sans MS", 2, 18));

    jButton3.setForeground(new java.awt.Color(51, 51, 51));

    jButton3.setText("PUNE");

    jButton3.addMouseListener(new java.awt.event.MouseAdapter() {

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

        jButton3MouseClicked(evt);

      }
```

```
});

getContentPane().add(jButton3);

jButton3.setBounds(80, 320, 160, 40);


jButton4.setFont(new java.awt.Font("Comic Sans MS", 2, 18));

jButton4.setText("BANGALORE");

jButton4.addMouseListener(new java.awt.event.MouseAdapter() {

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

      jButton4MouseClicked(evt);

   }

});

getContentPane().add(jButton4);

jButton4.setBounds(310, 320, 150, 40);


jButton5.setFont(new java.awt.Font("Comic Sans MS", 2, 18));

jButton5.setText("HYDRABAD");

jButton5.addMouseListener(new java.awt.event.MouseAdapter() {

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

      jButton5MouseClicked(evt);

   }

});

getContentPane().add(jButton5);

jButton5.setBounds(690, 320, 150, 40);
```

```
    jButton6.setFont(new java.awt.Font("Comic Sans MS", 2, 18));

    jButton6.setText("CHENNAI");

    jButton6.addMouseListener(new java.awt.event.MouseAdapter() {

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

        jButton6MouseClicked(evt);

      }

    });

    getContentPane().add(jButton6);

    jButton6.setBounds(900, 320, 140, 40);


    jLabel4.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/welcomepg.jpg")));
jLabel4.setText("jLabel4");

    getContentPane().add(jLabel4);

    jLabel4.setBounds(30, 40, 1070, 580);


    pack();

  }


  private void jButton1MouseClicked(java.awt.event.MouseEvent evt) {

    INTRO obj=new INTRO();

    obj.setVisible(true);

    dispose();

  }
```

```
private void jButton3MouseClicked(java.awt.event.MouseEvent evt) {
         PUNE obj=new PUNE();
          obj.setVisible(true);
           dispose();
}

private void jButton4MouseClicked(java.awt.event.MouseEvent evt) {
          bangalore obj=new bangalore ();
          obj.setVisible(true);
           dispose();
}

private void jButton5MouseClicked(java.awt.event.MouseEvent evt) {
  hydrabad obj=new hydrabad();
    obj.setVisible(true);
    dispose();
}

private void jButton6MouseClicked(java.awt.event.MouseEvent evt) {
   chennai obj =new chennai();
   obj.setVisible(true);
   dispose();
}
public static void main(String args[]) {
    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(welcomeform.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    } catch (InstantiationException ex) {
java.util.logging.Logger.getLogger(welcomeform.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {
java.util.logging.Logger.getLogger(welcomeform.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {
java.util.logging.Logger.getLogger(welcomeform.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    }
      java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {

                new welcomeform().setVisible(true);

            }

        });

    }
```

## 4.4 MODULE 4:

If the user selects the City PUNE then the Pune page will be opened so, the user can be able to see the list of buses that are available in Pune , list of hotels where moreover the location ,cost, contact is also been displayed and finely list of restaurants in addition to displays the recommended item of that particular restaurant, besides that a small button is also given so that if the user wants to know more about Pune he/she can click the button so that it will directly connects the user to the GOOGLE OFFICIAL PUNE's website where it tells best places to visit.

## 4.5 MODULE 5:

If the user selects the City BANGALORE then the Bangalore page will be opened so, the user can be able to see the list of buses that are available in Bangalore , list of hotels where moreover the location ,cost, contact is also been displayed and finely list of restaurants in addition to displays the recommended item of that particular restaurant, besides that a small button is also given so that if the user wants to know more about Pune he/she can click the button so that it will directly connects the user to the GOOGLE OFFICIAL BANGALORE's website where it tells best places to visit.

## 4.6 MODULE 6:

If the user selects the City HYDERABAD then the Hyderabad page will be opened so, the user can be able to see the list of buses that are available in Hyderabad , list of hotels where moreover the location ,cost, contact is also been displayed and finely list of restaurants in addition to displays the recommended item of that particular restaurant, besides that a small button is also given so that if the user wants to know more about

Pune he/she can click the button so that it will directly connects the user to the GOOGLE OFFICIAL HYDERABAD's website where it tells best places to visit.

## 4.7 MODULE 7:

If the user selects the City CHENNAI then the Chennai page will be opened so, the user can be able to see the list of buses that are available in Chennai , list of hotels where moreover the location ,cost, contact is also been displayed and finely list of restaurants in addition to displays the recommended item of that particular restaurant, besides that a small button is also given so that if the user wants to know more about Chennai he/she can click the button so that it will directly connects the user to the GOOGLE OFFICIAL Chennai's website where it tells best places to visit.

# CHAPTER 5

## RESULTS

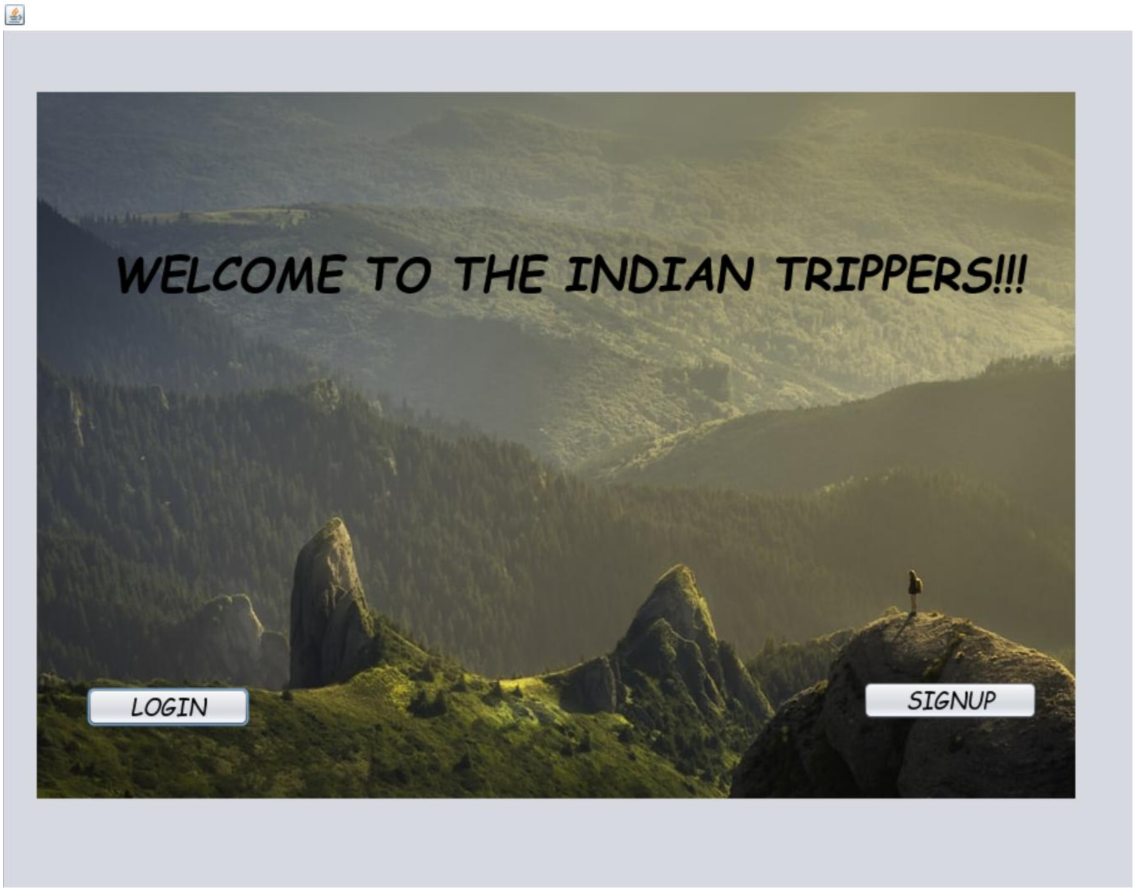
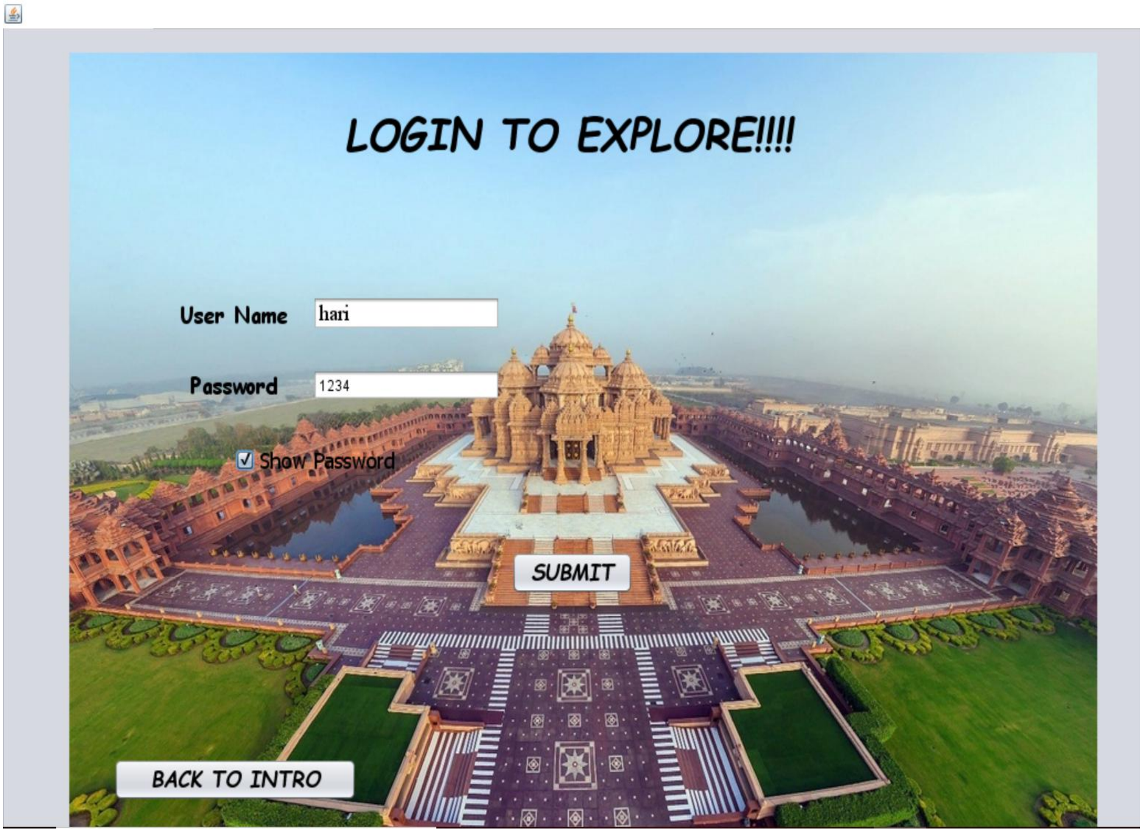


Fig 5.1 The Intro Page.

This intro page displays two options one is login and the other is sign in so if the user is new to the Indian trippers then he will go to the sign in and is the user is already existing then he/she can go to the log in page to login.



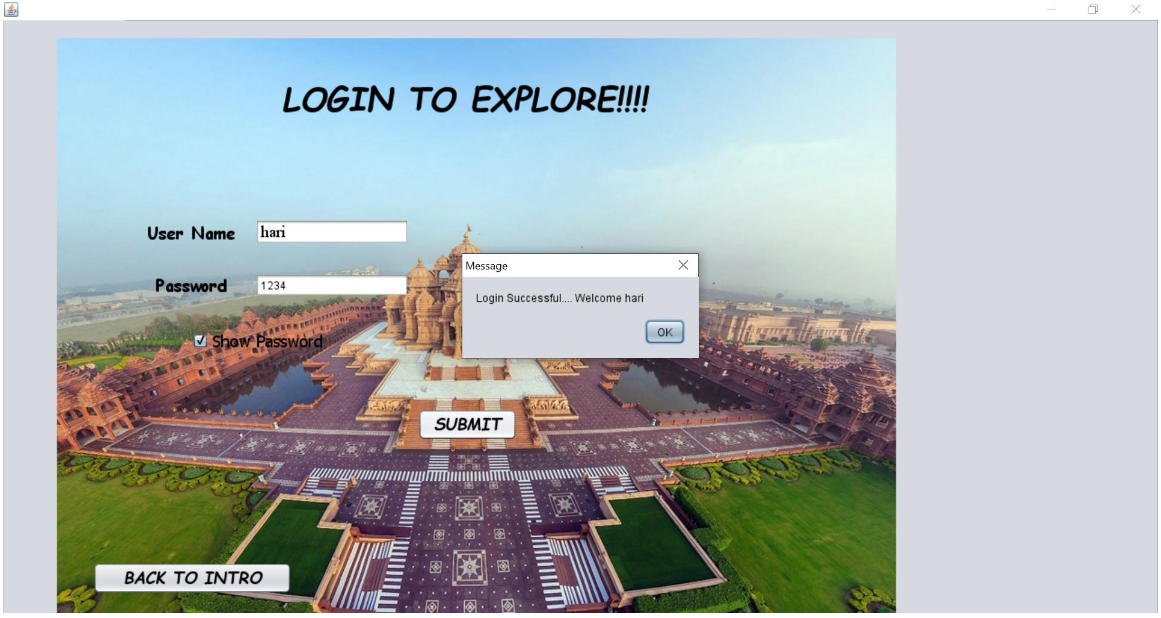
User Name hari

Password 1234

☑ Show Password

Fig 5.2 The Login page.

In the login page user name and password is been asked to move on further into the succeeding page. If want to go back he/she can press the button "back to intro" so that they can get back to the intro page again.



User Name: hari

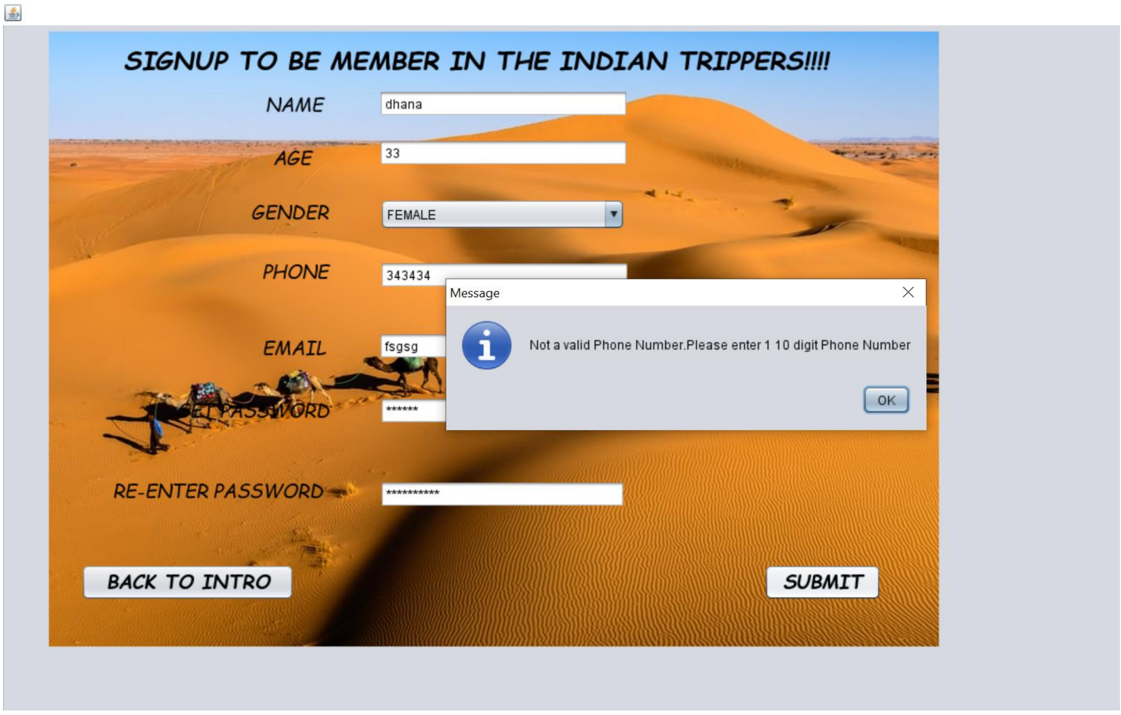
Password: 1234

☑ Show Password

Message

Login Successful.... Welcome hari

Fig 5.3 The login page after successfully logging in.



NAME: dhana

AGE: 33

GENDER: FEMALE

PHONE: 343434

EMAIL: fsgsg

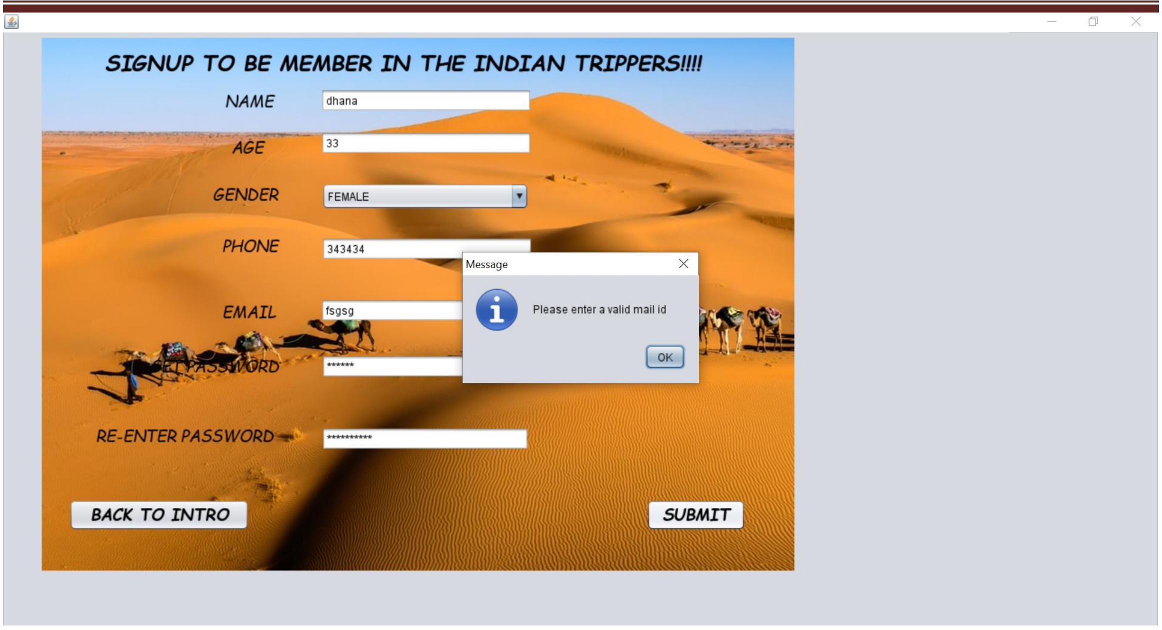
SET PASSWORD: ******

RE-ENTER PASSWORD: **********

Message

Not a valid Phone Number.Please enter 1 10 digit Phone Number

Fig 5.4 In the signup form if the phone number is not valid it asks the user the valid phone number.



SIGNUP TO BE MEMBER IN THE INDIAN TRIPPERS!!!!

NAME dhana

AGE 33

GENDER FEMALE

PHONE 343434

EMAIL fsgsg

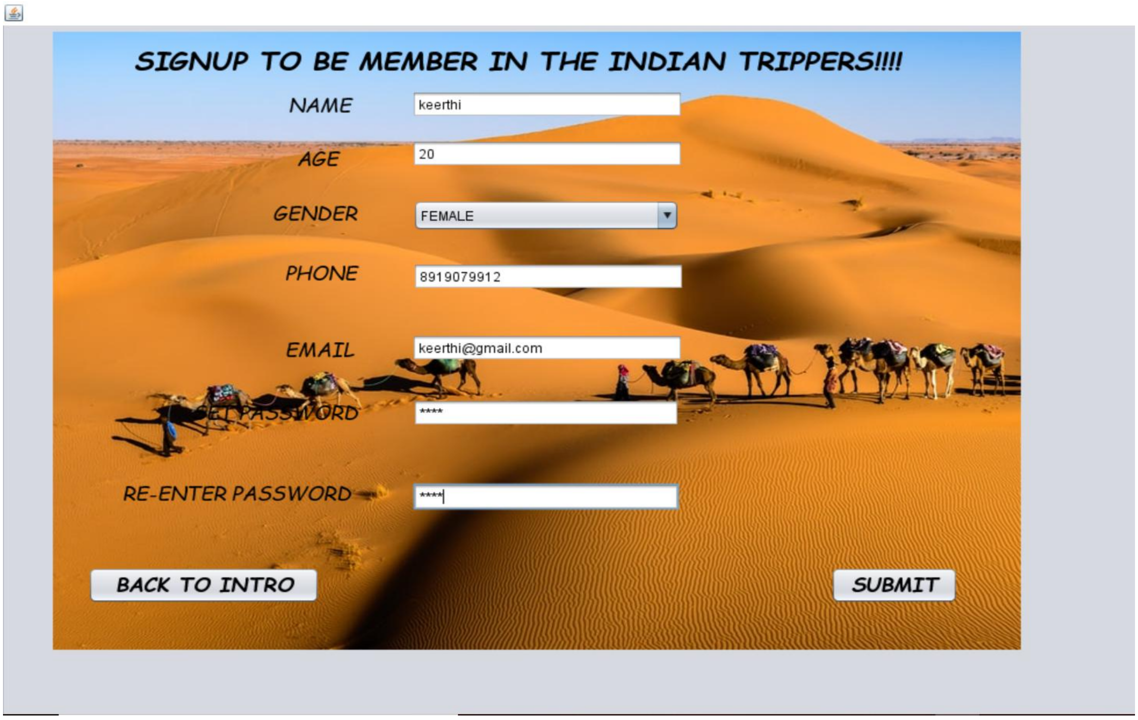
SET PASSWORD ******

RE-ENTER PASSWORD **********

Message

Please enter a valid mail id

Fig 5.5 In the signup form if the mail id is not valid it asks the user the valid mail id.



SIGNUP TO BE MEMBER IN THE INDIAN TRIPPERS!!!!

NAME keerthi

AGE 20

GENDER FEMALE

PHONE 8919079912

EMAIL keerthi@gmail.com

SET PASSWORD ****

RE-ENTER PASSWORD ****

Fig 5.6 The signup page.

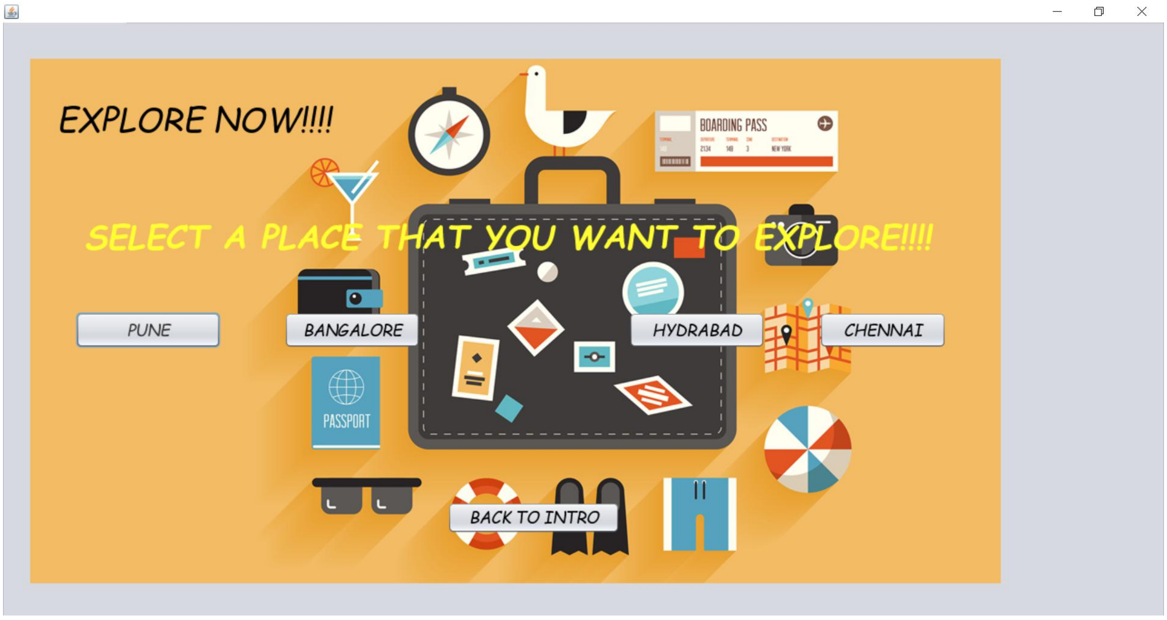
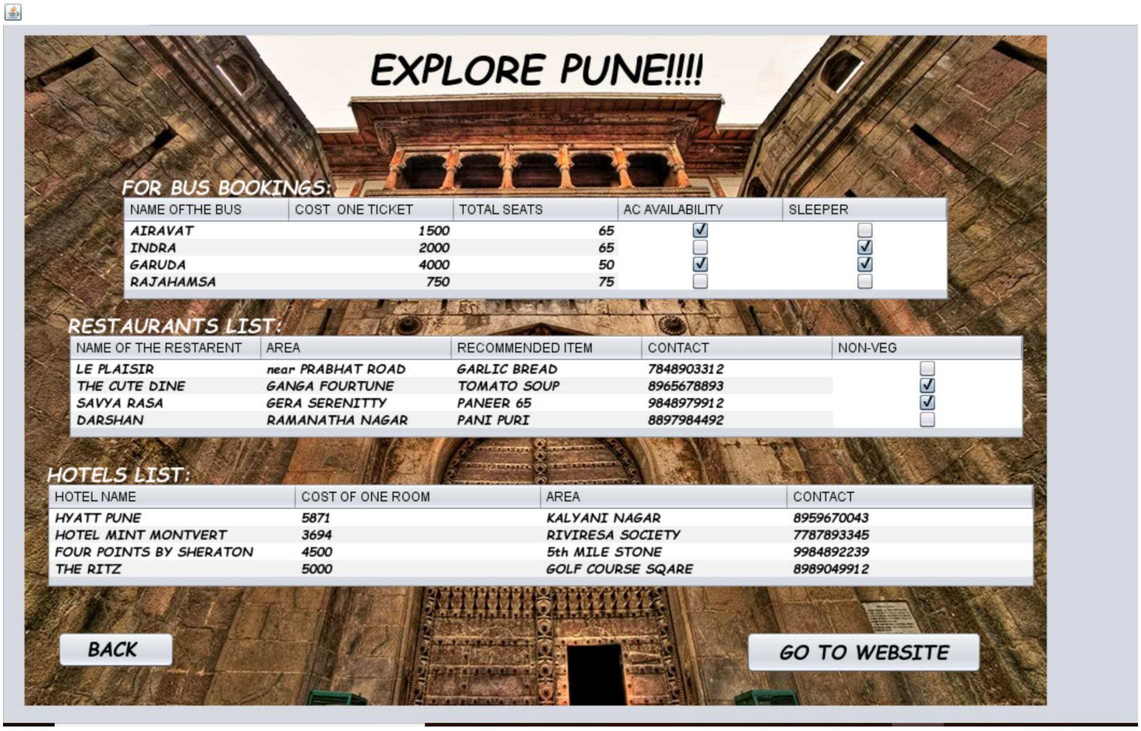


Fig 5.7 The welcome form to select the cities.

After the opening of the this exploring page the user can be able to see the cites, PUNE,BAMGALORE,HYDRABAD,CHENNAI.



EXPLORE PUNE!!!!

FOR BUS BOOKINGS:

| NAME OFTHE BUS | COST ONE TICKET | TOTAL SEATS | AC AVAILABILITY | SLEEPER |
|---|---|---|---|---|
| AIRAVAT | 1500 | 65 | ☑ | □ |
| INDRA | 2000 | 65 | □ | ☑ |
| GARUDA | 4000 | 50 | ☑ | ☑ |
| RAJAHAMSA | 750 | 75 | □ | □ |

RESTAURANTS LIST:

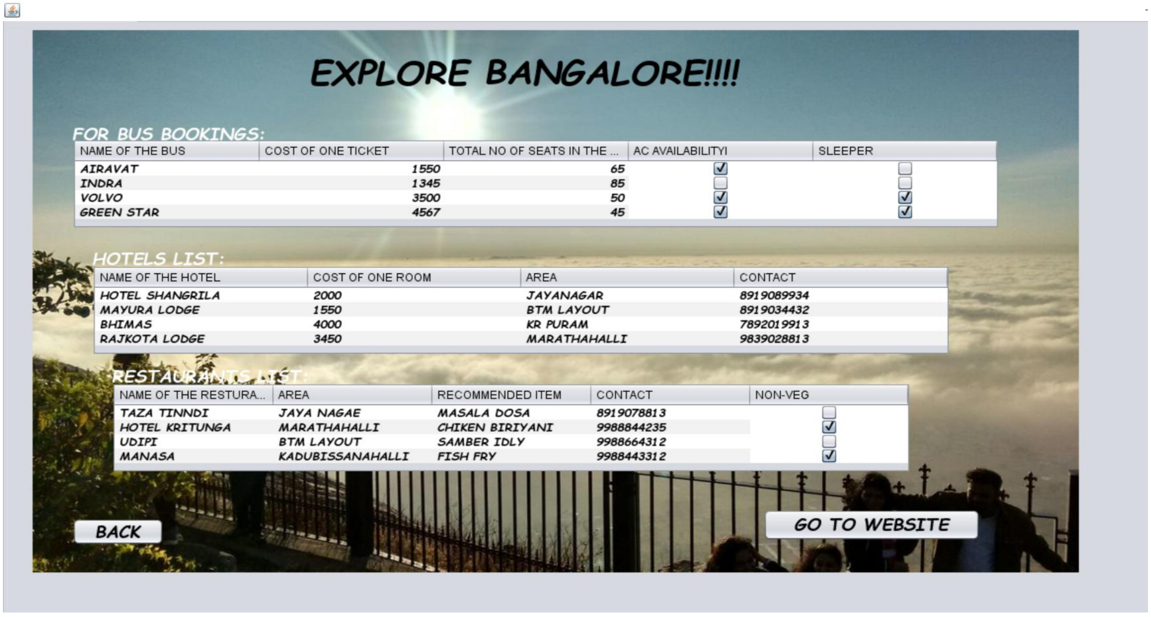
| NAME OF THE RESTARENT | AREA | RECOMMENDED ITEM | CONTACT | NON-VEG |
|---|---|---|---|---|
| LE PLAISIR | near PRABHAT ROAD | GARLIC BREAD | 7848903312 | □ |
| THE CUTE DINE | GANGA FOURTUNE | TOMATO SOUP | 8965678893 | ☑ |
| SAVYA RASA | GERA SERENITTY | PANEER 65 | 9848979912 | ☑ |
| DARSHAN | RAMANATHA NAGAR | PANI PURI | 8897984492 | □ |

HOTELS LIST:

| HOTEL NAME | COST OF ONE ROOM | AREA | CONTACT |
|---|---|---|---|
| HYATT PUNE | 5871 | KALYANI NAGAR | 8959670043 |
| HOTEL MINT MONTVERT | 3694 | RIVIRESA SOCIETY | 7787893345 |
| FOUR POINTS BY SHERATON | 4500 | 5th MILE STONE | 9984892239 |
| THE RITZ | 5000 | GOLF COURSE SQARE | 8989049912 |

Fig 5.8 The PUNE page to explore.

If user selects the Pune page then he/she will get the list of buses, hotel availability, restaurants availability in that place if he/she wants to know more about the places of Pune then they can click on the button “GO TO WEBSITE” so that this will connect the user to the official GOOGLE TRAVEL PAGE of PUNE to explore the best places in Pune .



EXPLORE BANGALORE!!!!

FOR BUS BOOKINGS:

| NAME OF THE BUS | COST OF ONE TICKET | TOTAL NO OF SEATS IN THE ... | AC AVAILABILITYI | SLEEPER |
|---|---|---|---|---|
| AIRAVAT | 1550 | 65 | ☑ | □ |
| INDRA | 1345 | 85 | □ | □ |
| VOLVO | 3500 | 50 | ☑ | ☑ |
| GREEN STAR | 4567 | 45 | ☑ | ☑ |

HOTELS LIST:

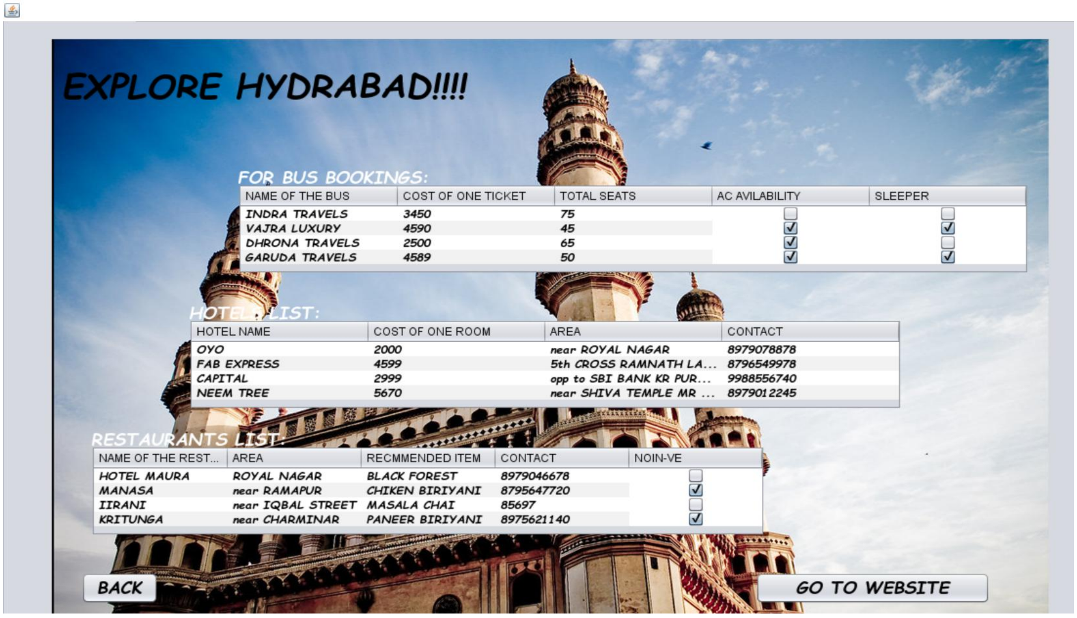
| NAME OF THE HOTEL | COST OF ONE ROOM | AREA | CONTACT |
|---|---|---|---|
| HOTEL SHANGRILA | 2000 | JAYANAGAR | 8919089934 |
| MAYURA LODGE | 1550 | BTM LAYOUT | 8919034432 |
| BHIMAS | 4000 | KR PURAM | 7892019913 |
| RAJKOTA LODGE | 3450 | MARATHAHALLI | 9839028813 |

RESTAURANTS LIST:

| NAME OF THE RESTURA... | AREA | RECOMMENDED ITEM | CONTACT | NON-VEG |
|---|---|---|---|---|
| TAZA TINNDI | JAYA NAGAE | MASALA DOSA | 8919078813 | □ |
| HOTEL KRITUNGA | MARATHAHALLI | CHIKEN BIRIYANI | 9988844235 | ☑ |
| UDIPI | BTM LAYOUT | SAMBER IDLY | 9988664312 | □ |
| MANASA | KADUBISSANAHALLI | FISH FRY | 9988443312 | ☑ |

Fig 5.9 The Bangalore page to explore.

If user selects the BANGALORE page then he/she will get the list of buses, hotel availability, restaurants availability in that place if he/she wants to know more about the places of Bangalore then they can click on the button “GO TO WEBSITE” so that this will connect the user to the official GOOGLE TRAVEL PAGE of BANGALORE to explore the best places in Bangalore.



FOR BUS BOOKINGS:

| NAME OF THE BUS | COST OF ONE TICKET | TOTAL SEATS | AC AVILABILITY | SLEEPER |
|---|---|---|---|---|
| INDRA TRAVELS | 3450 | 75 | □ | □ |
| VAJRA LUXURY | 4590 | 45 | ☑ | ☑ |
| DHRONA TRAVELS | 2500 | 65 | ☑ | □ |
| GARUDA TRAVELS | 4589 | 50 | ☑ | ☑ |

HOTELS LIST:

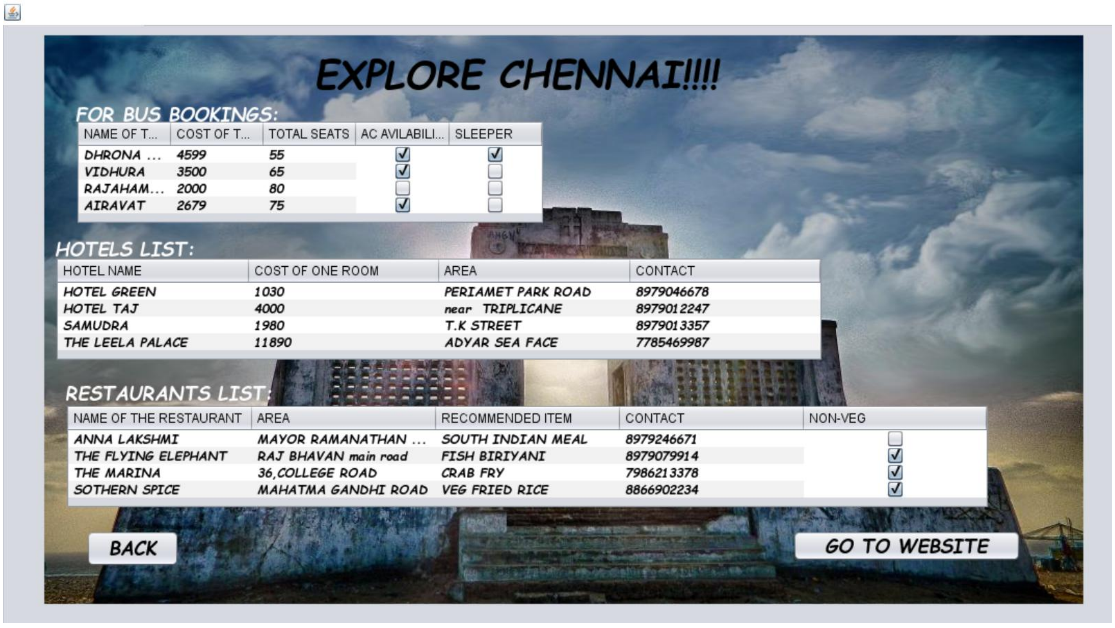
| HOTEL NAME | COST OF ONE ROOM | AREA | CONTACT |
|---|---|---|---|
| OYO | 2000 | near ROYAL NAGAR | 8979078878 |
| FAB EXPRESS | 4599 | 5th CROSS RAMNATH LA... | 8796549978 |
| CAPITAL | 2999 | opp to SBI BANK KR PUR... | 9988556740 |
| NEEM TREE | 5670 | near SHIVA TEMPLE MR ... | 8979012245 |

RESTAURANTS LIST:

| NAME OF THE REST... | AREA | RECMMENDED ITEM | CONTACT | NOIN-VE |
|---|---|---|---|---|
| HOTEL MAURA | ROYAL NAGAR | BLACK FOREST | 8979046678 | □ |
| MANASA | near RAMAPUR | CHIKEN BIRIYANI | 8795647720 | ☑ |
| IIRANI | near IQBAL STREET | MASALA CHAI | 85697 | □ |
| KRITUNGA | near CHARMINAR | PANEER BIRIYANI | 8975621140 | ☑ |

Fig 5.10 The Hyderabad page to explore.

If user selects the Hyderabad page then he/she will get the list of buses, hotel availability, restaurants availability in that  place if he/she wants to know more about the places of Hyderabad then they can click on the button “GO TO WEBSITE” so that this will connect the user to the official GOOGLE TRAVEL PAGE of HYDERBAD to explore the best places in Hyderabad.



FOR BUS BOOKINGS:

| NAME OF T... | COST OF T... | TOTAL SEATS | AC AVILABILI... | SLEEPER |
|---|---|---|---|---|
| DHRONA ... | 4599 | 55 | ☑ | ☑ |
| VIDHURA | 3500 | 65 | ☑ | □ |
| RAJAHAM... | 2000 | 80 | □ | □ |
| AIRAVAT | 2679 | 75 | ☑ | □ |

HOTELS LIST:

| HOTEL NAME | COST OF ONE ROOM | AREA | CONTACT |
|---|---|---|---|
| HOTEL GREEN | 1030 | PERIAMET PARK ROAD | 8979046678 |
| HOTEL TAJ | 4000 | near TRIPLICANE | 8979012247 |
| SAMUDRA | 1980 | T.K STREET | 8979013357 |
| THE LEELA PALACE | 11890 | ADYAR SEA FACE | 7785469987 |

RESTAURANTS LIST:

| NAME OF THE RESTAURANT | AREA | RECOMMENDED ITEM | CONTACT | NON-VEG |
|---|---|---|---|---|
| ANNA LAKSHMI | MAYOR RAMANATHAN ... | SOUTH INDIAN MEAL | 8979246671 | □ |
| THE FLYING ELEPHANT | RAJ BHAVAN main road | FISH BIRIYANI | 8979079914 | ☑ |
| THE MARINA | 36,COLLEGE ROAD | CRAB FRY | 7986213378 | ☑ |
| SOTHERN SPICE | MAHATMA GANDHI ROAD | VEG FRIED RICE | 8866902234 | ☑ |

Fig 5.11 The Chennai page to explore.

If user selects the Chennai page then he/she will get the list of buses, hotel availability, restaurants availability in that  place if he/she wants to know more about the places of Chennai then they can click on the button “GO TO WEBSITE” so that this will connect the user to the official GOOGLE TRAVEL PAGE of Chennai to explore the best places in Chennai.

# CHAPTER 6

# CONCLUSION

In the conclusion, this project will maintain the records of the of all the user details confidentially now a days in our day to day life people are showing more interest in the traveling and exploring new this of any particular place so this project will help the people to explore the new places which they want to visit.

We can give more advance software for traveling including more facilities. We will host the platform on exploring, traveling and booking to make it accessible worldwide.

In this project MY SQL database is used to store the user records in the database. Frontend is used for designing project including the java concepts along with oops. The Software is used as NetBeans 8.1 in this project.

# REFERENCE

1. For learning java programming language "THE PROGRAMMING HUB"
2. JAVA PROGRAMMING BASICS for absolute beginners by NATHAN CLARK.
3. "ALL IN ONE BOOK" 12th class CBSE.
4. Learning the designing part of net bean"www.youtube.com"
5. Oops using java by BalaGuruSwamy.