

**MINI PROJECT REPORT**

**ON**

**Expense Tracker Application**

**BY**

**Name Roll No**

**Rahul Nair 22073**

**Course: MCA I Sem.: II**

**PRATIBHA INSTITUTE OF BUSINESS MANAGEMENT CHINCHWAD, PUNE-411019**

**2022-2023**



# CERTIFICATE

This is to certify that the Project

“Expense Tracker Application”

Submitted by Rahul Nair 22073

is a bonafide work carried out by Students under the supervision of Prof. **Kalyan Singh Patil** and it is submitted towards the partial fulfillment of the requirement of Master of Computer Applications Department.

### Prof.KalyanSingh Patil Prof. Manish Patankar Project Guide Head of Department (MCA)

Internal Examiner External Examiner

# Acknowledgments

It gives us great pleasure in presenting the preliminary project report on

‘**Expense Tracker Application**’.

I would like to take this opportunity to thank my internal guide **Prof. KalyanSingh Patil** for giving me all the help and guidance I needed. I am grateful to them for their kind support. Their valuable suggestions were very helpful.

I am also grateful to **Prof. Manish Patankar**, Head of MCA Department, Pratibha Institute of Business Management for his indispensable support, suggestions.

In the end our special thanks to other Team Members for providing various resources such as laboratory with all needed software platforms, continuous Internet connection, for Our Project.

### Student Name Roll No

**Rahul Nair 22073**

**Index**

|  |  |
| --- | --- |
| **Chapter** | **Page number** |
| **CHAPTER 1 : INTRODUCTION** | |
| 1.1 Existing System |  |
| 1.2 Need for System |  |
| 1.3 Operating Environment Hardware and Software |  |
| **CHAPTER 2 : PROPOSED SYSTEM** | |
| 2.1 Proposed System (Introduction of system) |  |
| 2.2 Module specifications (Scope) |  |
| 2.3 Objectives of System |  |
| **CHAPTER 3 : ANALYSIS & DESIGN** | |
| 3.1 Use Case Diagrams |  |
| 3.2 Activity Diagram |  |
| 3.3 Class Diagram |  |
| 3.3 Module Hierarchy Diagram |  |
| 3.4 Table specifications (Database design) |  |
| 3.5 Data dictionary |  |
| **CHAPTER 4 : USER MANUAL** | |
| 4.1 User Interface Screens (Input) |  |
| 4.2 Output Screens with data |  |
| 4.3 Data Reports |  |
| 4.4 Sample program code |  |
| 4.5 Limitations and Bibliography |  |

#### CHAPTER 1 : INTRODUCTION

* 1. **Existing System**

In the existing system, each task is carried out manually and processing is also a tedious job. In previous system users were maintaining diary manually in pen and paper, which was time taking and costly. The users is not able to achieve its need in time and also the results may not accurate. Because of the manual maintenance there are number of difficulties and drawbacks exist in the system.

Some of them are Drawbacks of the Existing System:

* Increased transaction leads to increased source document and hence maintenance becomes difficult.
* If any admin, user entry is wrongly made then the maintenance becomes very difficult.

#### Need for System

The proposed system is designed to be more efficient than the manual system. It invokes all base tasks that are now carried out manually, such as the users transactions . The proposed System is completely computer-based application. Thousands of records can search and displayed without taking any significant time

Advantages of the Proposed System:

* + - Gives accurate information
    - Simplifies the manual work
    - It minimizes the documentation related work
    - Provides up to date information
    - Friendly Environment by providing warning messages.

#### Operating Environment Hardware and Software

**1.1 At Server Side:**

#### Hardware Configuration (Minimum Requirements)

|  |  |
| --- | --- |
| Name | Details |
| Processor | i3 Generation & above |

|  |  |
| --- | --- |
| RAM | 4GB & above |
| Hard Drive | 500 GB & Above |

**Software Environment (Minimum Requirements)**

|  |  |
| --- | --- |
| Name | Details |
| Operating System | Windows 10 & above |
| Database Sever | MongoDb |
| Web Server | - |
| Browser | Chrome Browser |
| Framework | React-JS & Node-JS |
| IDE | VS Code |

#### CHAPTER 2: PROPOSED SYSTEM

* 1. **Proposed System (Introduction of system)**

Expense Tracker Application is used to add income & expense from anywhere in the world by a single dynamic website which will help the user to know his saving in a month.

Through browser admin can add his income and expenditure for the given month and he can view his saving in graphical format too.

It is a easiest platform for all users which can be easily add records and know the all details. Expense Tracker Application is a dynamic website for users. It is dynamic and responsive web design.

The Expense Tracker Application allows the user of the system access all the details of income and expense. The main purpose is to help users to manage add income and expnese etc.

#### Module specifications (Scope)

1. Income Track: This module is mainly based for user. System will check the total income added for the user
2. Expense Track: This module is mainly based for user. System will check the total income added for the user
3. Graph based details: Budgeting the records for the users in graph based record.
4. Exception Handling: If any wrong record will be added then it will throw a error

#### Objectives of System

System design is the process of defining the elements of a system such as the architecture, modules and components, the different interfaces of those components and the data that goes through that system. It is meant to satisfy specific needs and requirements of a business or organization through the engineering of a coherent and well-running system.

System designing in terms of software engineering has its own value and importance in the system development process. To mention it may though seem as simple as anything or simply the design of systems, but in a broader sense it implies a systematic and rigorous approach to design such a system which fulfils all the practical aspects including flexibility, efficiency, and security.

Before there is any further discussion of system design, it is important that some points be made clear. As it goes without saying that nothing is created that is not affected by the world in which it’s made. So, the systems are not created in a vacuum.

They are created to meet the needs of the users. They are not only intended to solve the existing problems, but they also come up with acceptable solutions to the problems that may arise in the future. The whole process of system development, from blueprint to the actual product, involves considering all the relevant factors and taking the required specifications and creating a useful system based on strong technical, analytical and development skills of the professionals.

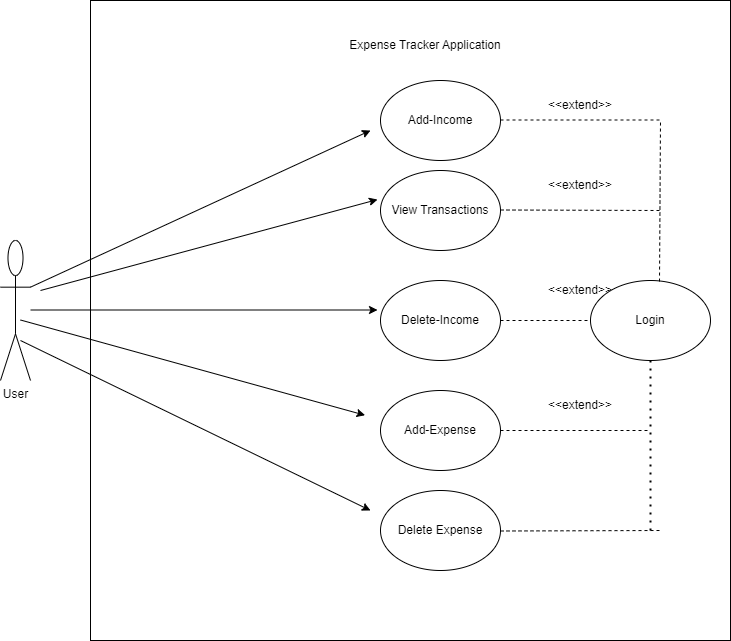
Let’s get back to our discussion about what the system design phase is and the importance of system design in the process of system development. Being another important step in the system development process, system designing phase commences after the system analysis phase is completed. It’s appropriate to mention that the output or the specifications taken through the phase of system analysis become an input in the system design phase which in turn leads to workout based on the user defined estimations.

The importance of this phase may be understood by reason of the fact that it involves identifying data sources, the nature and type of data that is available. For example, to design a

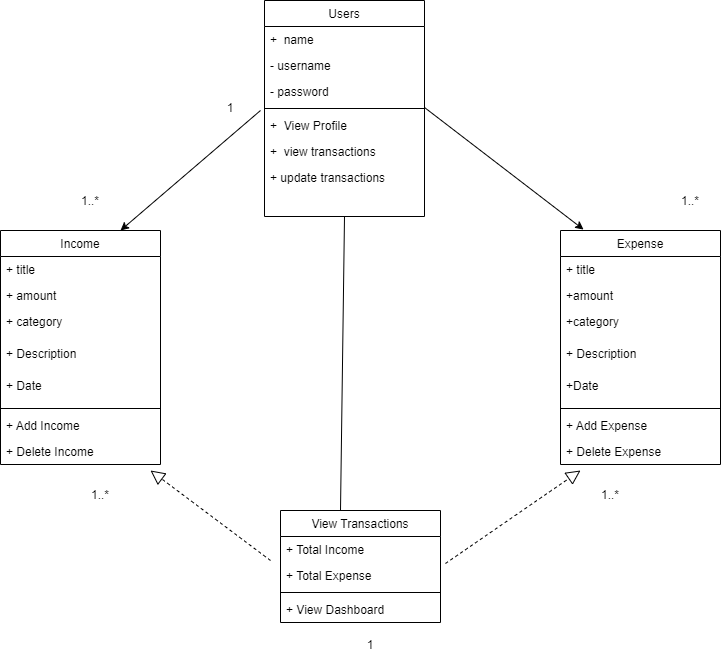
salary system, there is a need for using inputs, such as, attendance, leave details, additions or deductions etc. This facilitates understanding what kind of data is available and by whom it is supplied to the system so that the system may be designed considering all the relevant factors. In addition, system designing leads to ensure that the system is created in such a way that it fulfils the need of the users and keep them at ease being user oriented.

#### CHAPTER 3: ANALYSIS & DESIGN

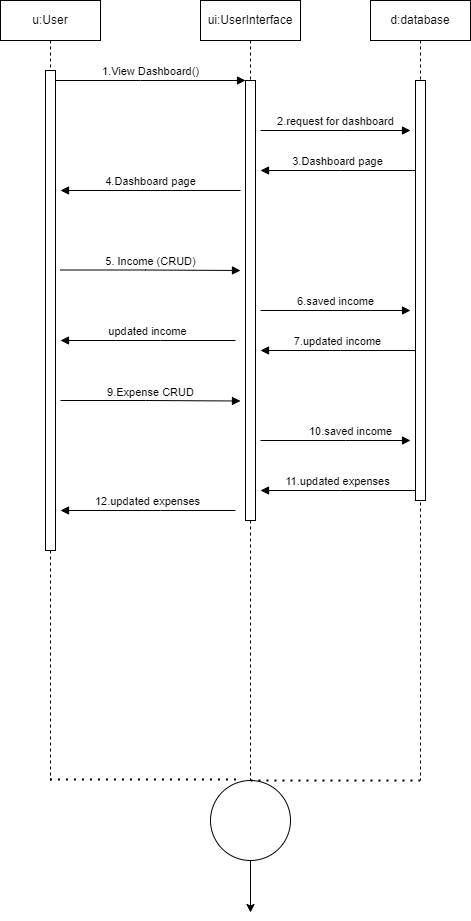
* 1. **Use Case Diagrams**

****

#### Class Diagram

****

* 1. **Sequence Diagram**

****

* 1. **A**Activity Diagram

Select Tour Package

Request Scheme



New Customer

Register Customers

New Tour Information

Make Payment

Reserve Tour

**3.4 Table specifications (Database design)**

**TABLE 3.4.1: Income:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** |
| u | varchar(30) | NO | PRI | NULL |
| Name | varchar(30) | NO | PRI | NULL |
| password | varchar(30) | NO |  | NULL |
| security | varchar(30) | NO |  | NULL |
| Answer | varchar(30) | NO |  | NULL |

**TABLE 3.4.2: CUSTOMER:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** |
| Id | number | NO | PRI | NULL |
| amount | number (30) | NO |  | NULL |
| type | varchar (30) | NO |  | NULL |
| category | varchar(30) | NO |  | NULL |
| description | varchar(30) | NO |  | NULL |
| country | varchar(30) | NO |  | NULL |
| address | varchar(30) | NO |  | NULL |
| phone | varchar(30) | NO |  | NULL |
| email | varchar(30) | NO |  | NULL |

**TABLE 3.4.3: BOOK PACKAGE:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** |
| id | varchar(30) | NO | PRI | NULL |
| amount | num (30) | NO |  | NULL |
| type | varchar (10) | NO |  | NULL |
| date | varchar(30) | NO | MUL | NULL |
| category | varchar(30) | NO |  | NULL |
| description | varchar(30) | NO | PRI | NULL |
| price | number (30) | NO |  | NULL |

**TABLE 3.4.1: Hotel:**

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Null** |
| name | varchar(30) | NO |
| costperday | varchar(30) | NO |
| foodcharges | varchar(30) | NO |
| Ac charges | varchar(30) | NO |
| Answer | varchar(30) | NO |

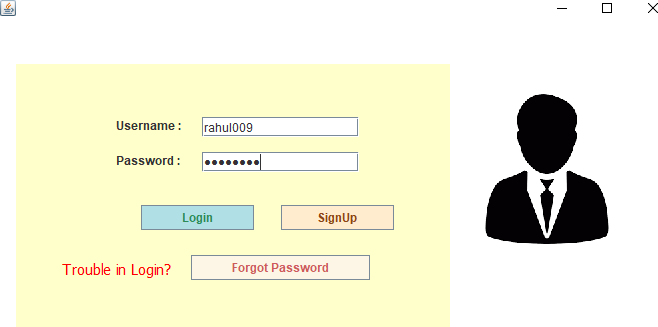
**TABLE 3.4.2:: Book Hotel**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** |
| username | varchar(30) | NO |  |
| Hotel Name | varchar(30) | NO | PRI |
| Total Persons | varchar(30) | NO |  |
| No of days | number(10) | NO |  |
| Ac/ Non Ac | varchar(30) | NO |  |
| Food | varchar(30) | NO |  |
| ID | varchar(30) | NO |  |
| phone | number(10) | NO |  |
| email | varchar(30) | NO |  |

**TABLE 3.4.3:Cash Payment:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** |
| username | varchar(30) | NO |  |
| amount | Number(10) | NO |  |

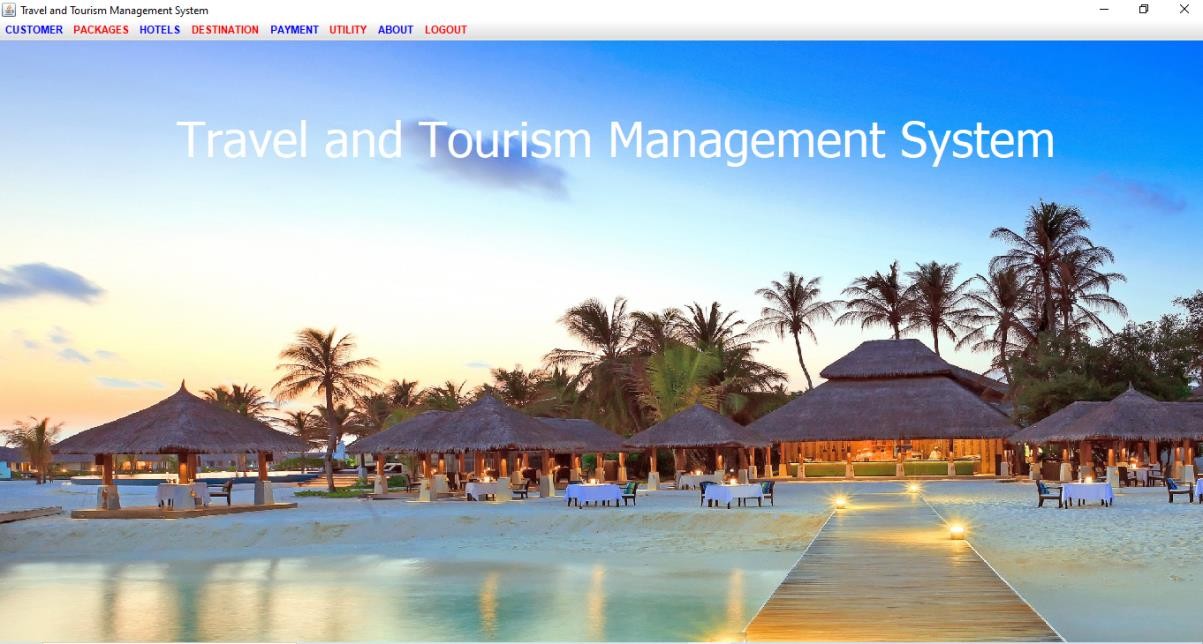
**CHAPTER 4: USER MANUAL**

* 1. **User Interface Screens (Input) Login Page**

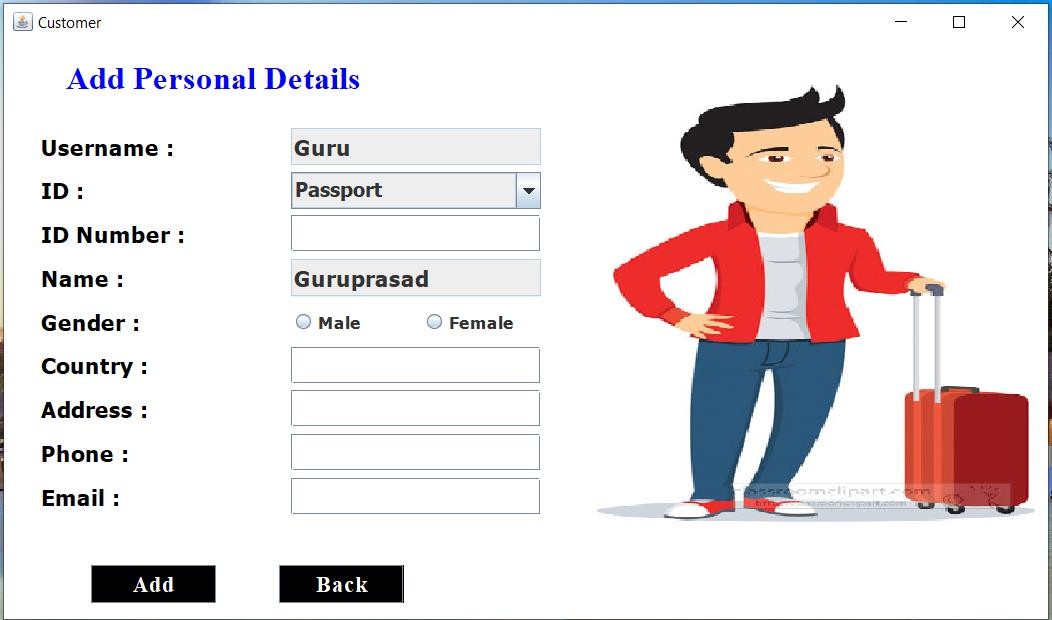
**Loading Page:**



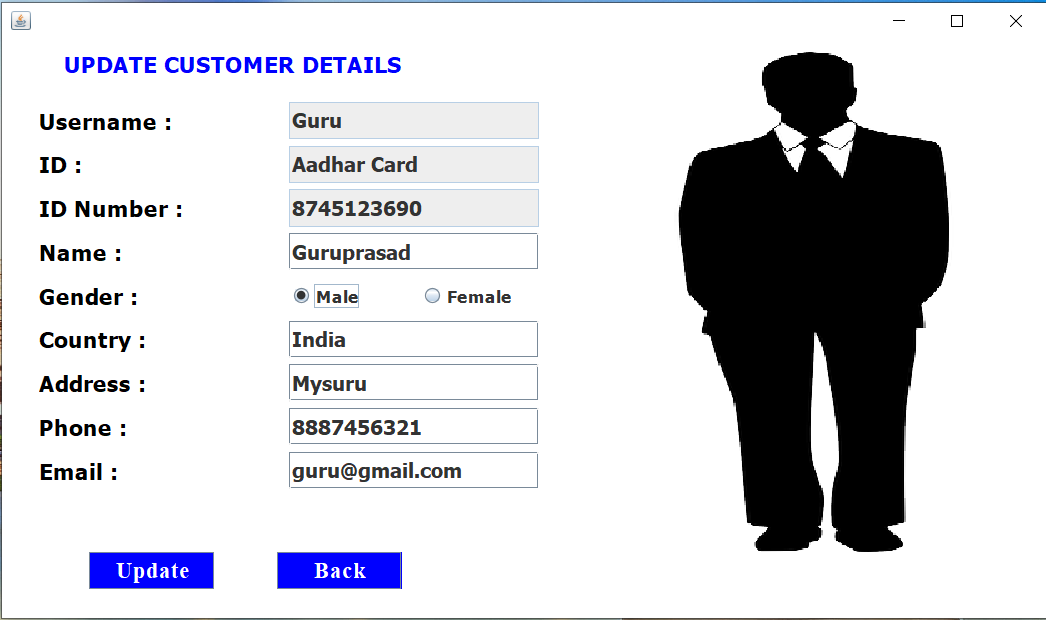
**Main Frame:**

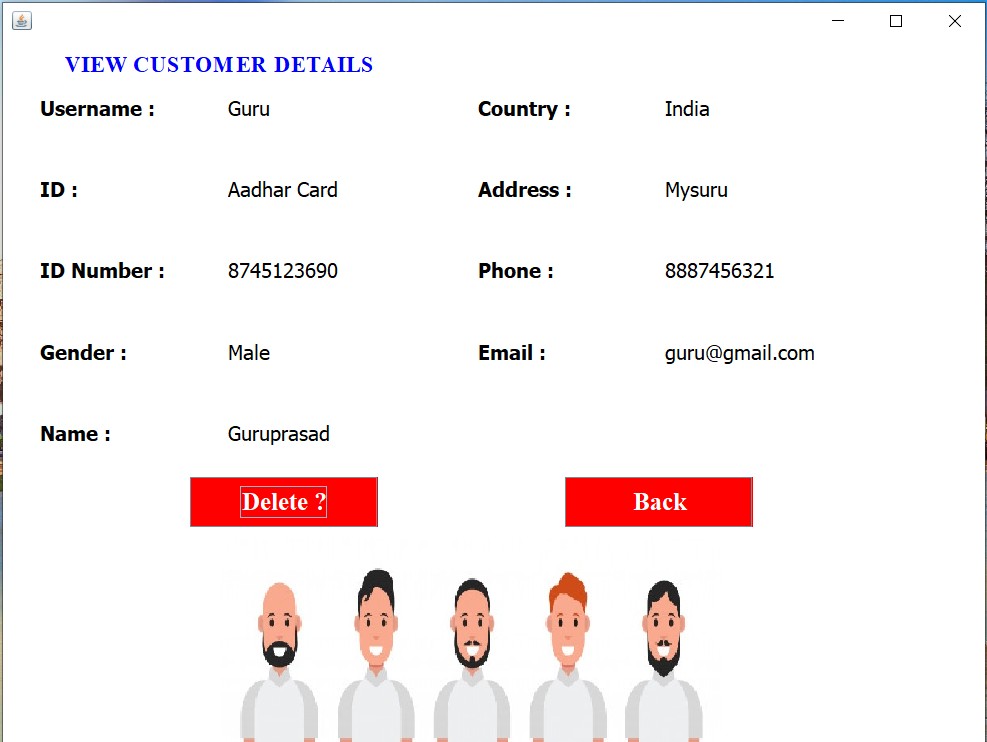


**Personal Details:**

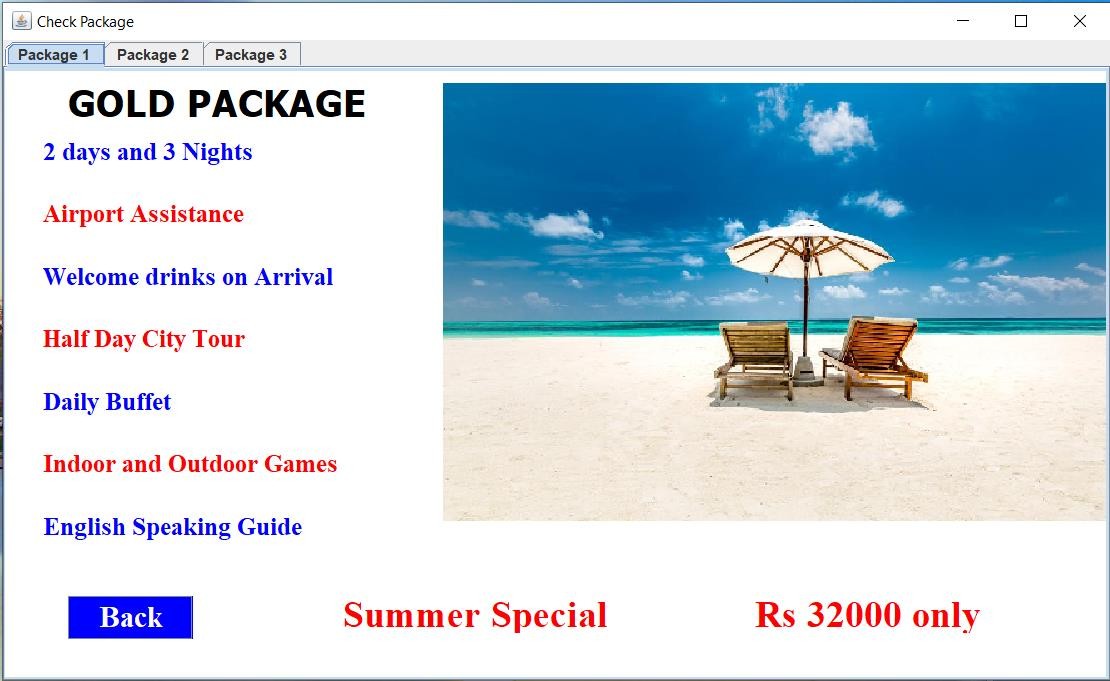


**UPDATE CUSTOMER PAGE:**

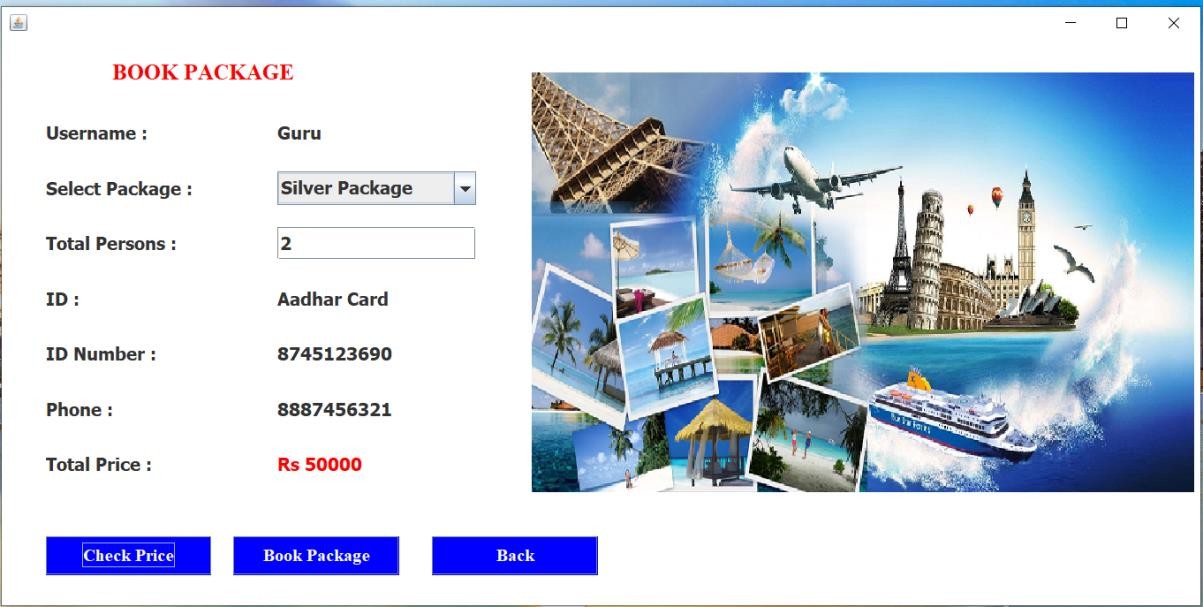


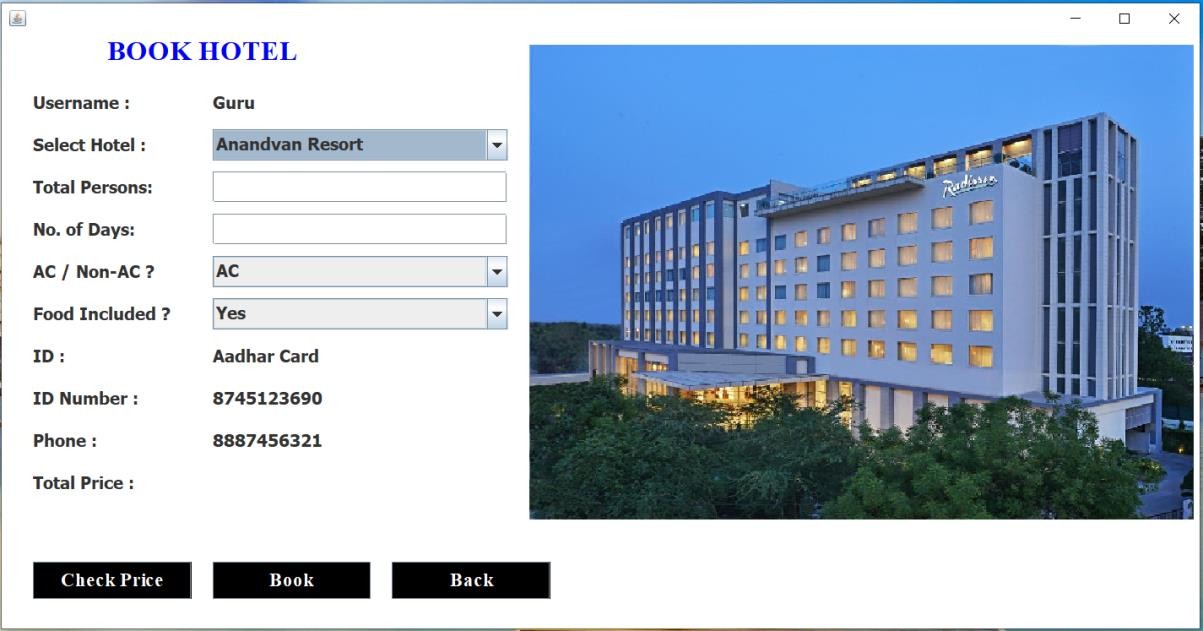
**View Customer Page:**

**Package Page:**

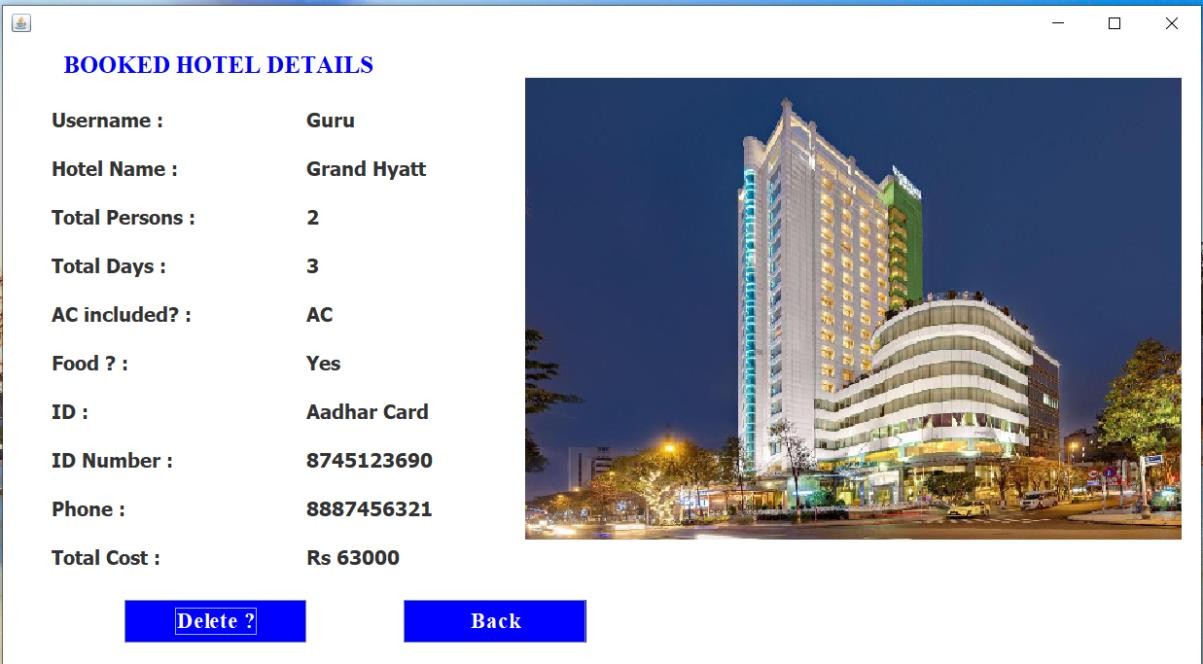


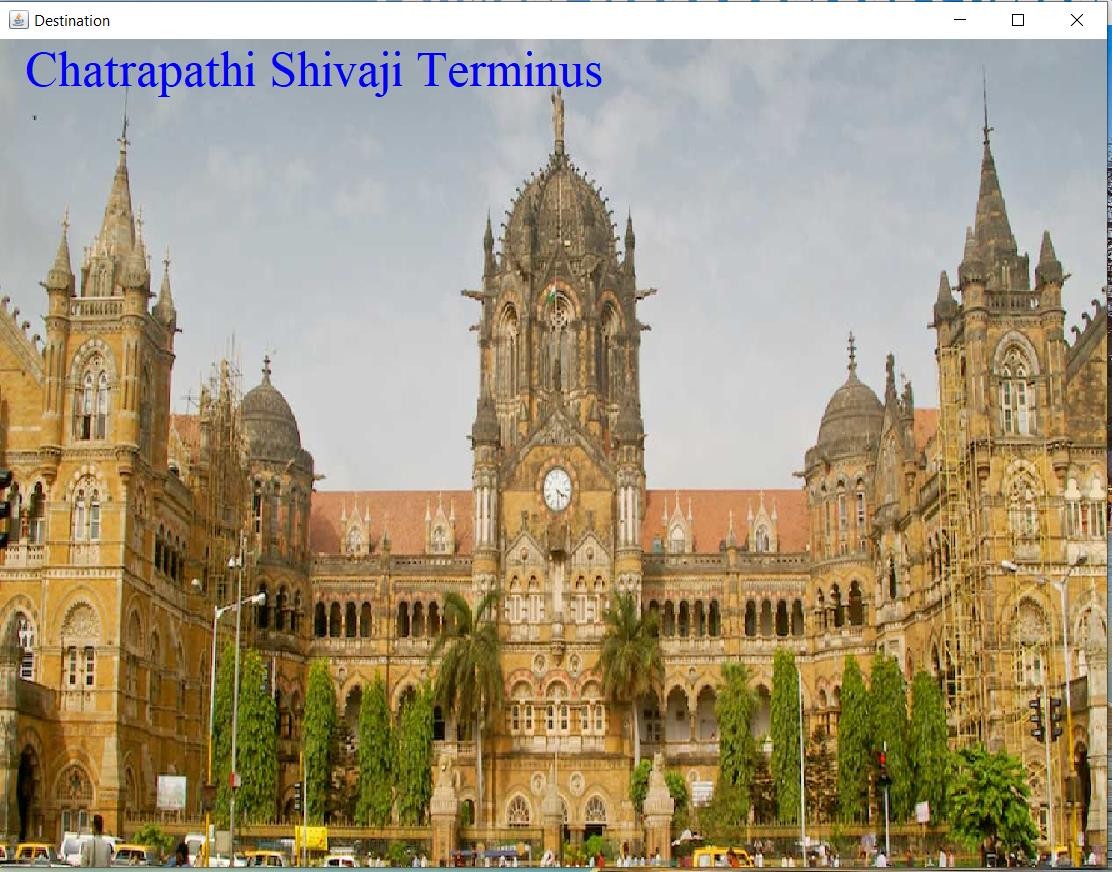
**Book Package:**



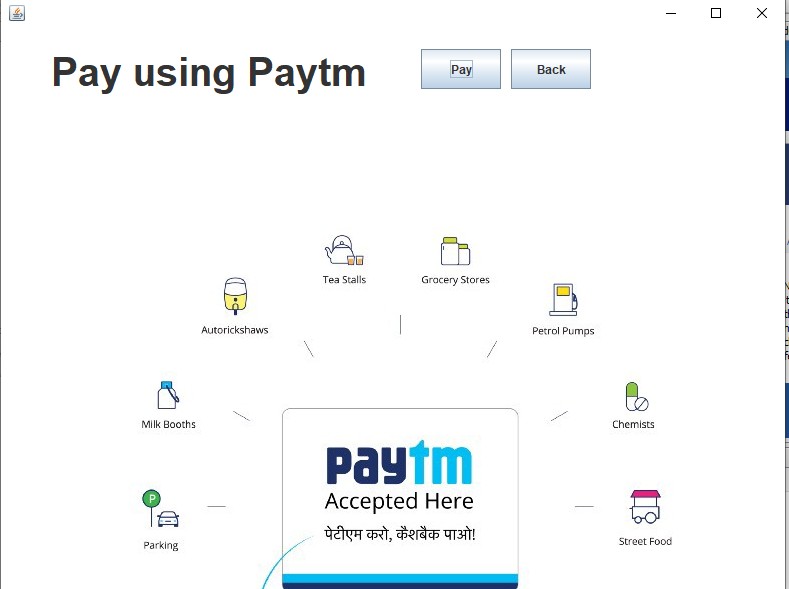
**Book Hotel:**

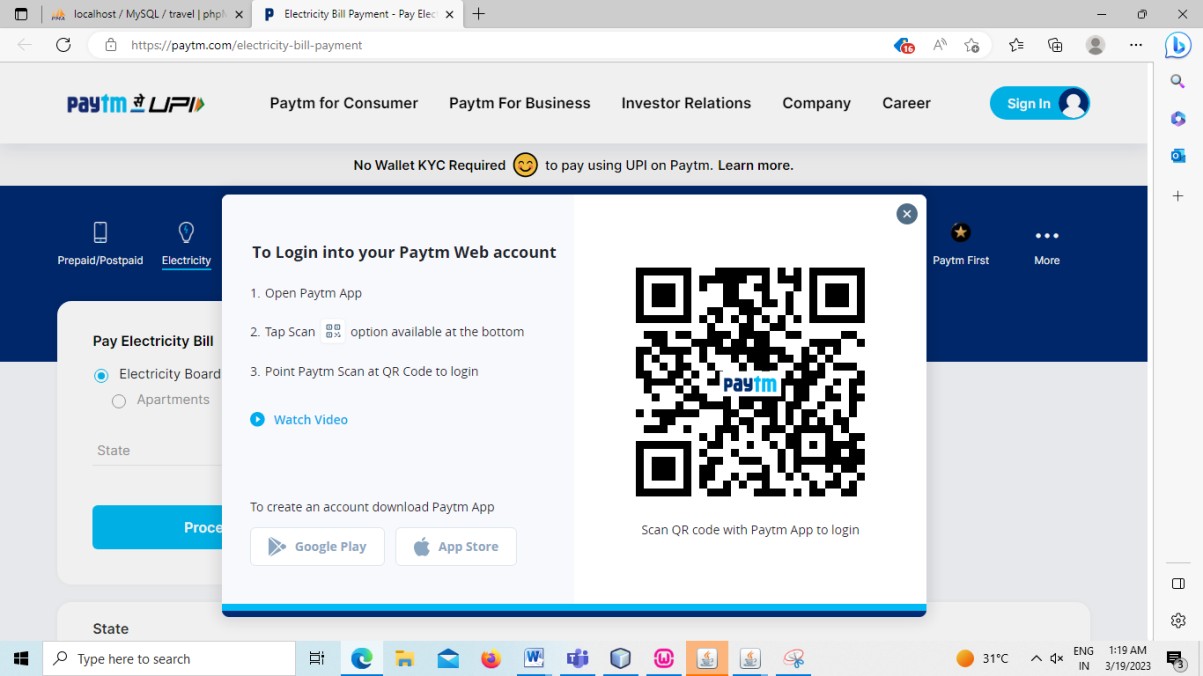
**View Hotel:**



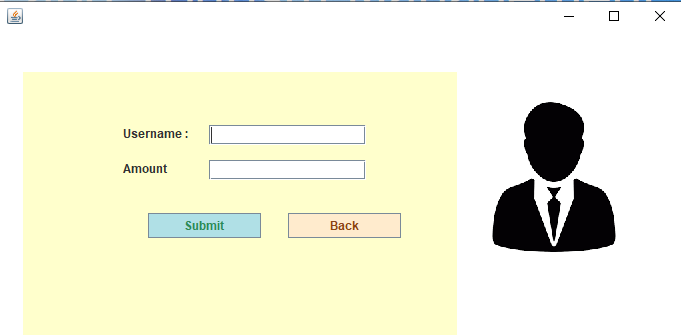
**Destination Page:**

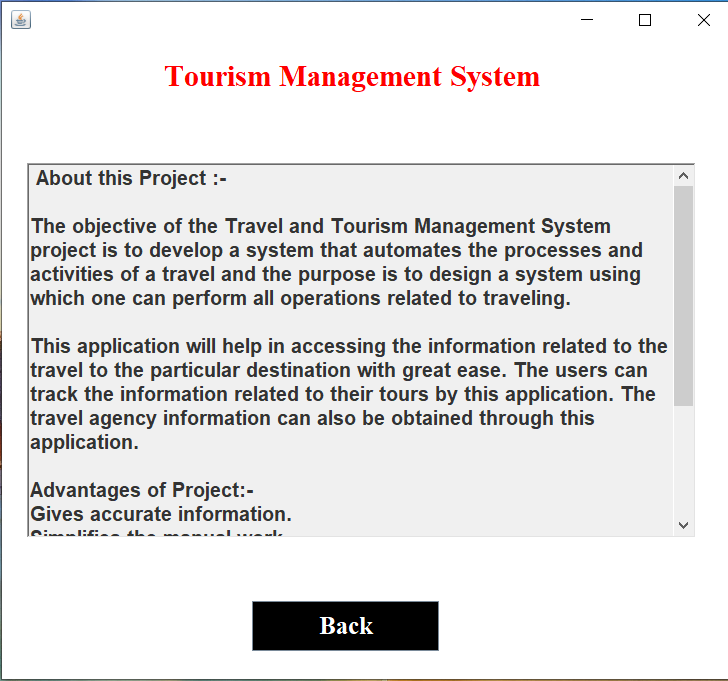
**Payment Page:**





**Cash:**



**About Page:**

* 1. Sample Program Code

/\*

* To change this license header, choose License Headers in Project Properties.
* To change this template file, choose Tools | Templates
* and open the template in the editor.

\*/

package travel.management.system;

import java.awt.\*; import javax.swing.\*; import java.awt.event.\*; import java.sql.\*;

public class Login extends JFrame implements ActionListener{ private JPanel panel;

private JTextField textField;

private JPasswordField passwordField; private JButton b1,b2,b3;

public Login()

setBackground(new Color(255, 255, 204));

setBounds(550, 250, 700, 400);

panel = new JPanel(); panel.setBackground(Color.WHITE); setContentPane(panel); panel.setLayout(null);

JLabel l1 = new JLabel("Username : ");

l1.setBounds(124, 89, 95, 24); panel.add(l1);

JLabel l2 = new JLabel("Password : "); l2.setBounds(124, 124, 95, 24); panel.add(l2);

textField = new JTextField(); textField.setBounds(210, 93, 157, 20); panel.add(textField);

passwordField = new JPasswordField(); passwordField.setBounds(210, 128, 157, 20); panel.add(passwordField);

JLabel l3 = new JLabel(""); l3.setBounds(377, 79, 46, 34); panel.add(l3);

JLabel l4 = new JLabel(""); l4.setBounds(377, 124, 46, 34); panel.add(l3);

ImageIcon c1 = new ImageIcon(ClassLoader.getSystemResource("Travel/Management/System/icons/login.png"))

;

Image i1 = c1.getImage().getScaledInstance(150, 150,Image.SCALE\_DEFAULT); ImageIcon i2 = new ImageIcon(i1);

JLabel l6 = new JLabel(i2); l6.setBounds(480, 70, 150, 150); add(l6);

b1 = new JButton("Login"); b1.addActionListener(this) b1.setForeground(new Color(46, 139, 87));

b1.setBackground(new Color(176, 224, 230));

b1.setBounds(149, 181, 113, 25); panel.add(b1);

b2 = new JButton("SignUp"); b2.addActionListener(this); b2.setForeground(new Color(139, 69, 19));

b2.setBackground(new Color(255, 235, 205));

b2.setBounds(289, 181, 113, 25); panel.add(b2);

b3 = new JButton("Forgot Password"); b3.addActionListener(this);

b3.setForeground(new Color(205, 92, 92));

b3.setBackground(new Color(253, 245, 230));

b3.setBounds(199, 231, 179, 25); panel.add(b3);

JLabel l5 = new JLabel("Trouble in Login?"); l5.setFont(new Font("Tahoma", Font.PLAIN, 15)); l5.setForeground(new Color(255, 0, 0));

l5.setBounds(70, 235, 110, 20); panel.add(l5);

JPanel panel2 = new JPanel(); panel2.setBackground(new Color(255, 255, 204));

panel2.setBounds(24, 40, 434, 263);

panel.add(panel2);

}

public void actionPerformed(ActionEvent ae){ if(ae.getSource() == b1){

Boolean status = false; try {

Conn con = new Conn();

String sql = "select \* from account where username=? and password=?"; PreparedStatement st = con.c.prepareStatement(sql);

st.setString(1, textField.getText()); st.setString(2, passwordField.getText()); ResultSet rs = st.executeQuery();

if (rs.next()) { this.setVisible(false);

new Loading(textField.getText()).setVisible(true);

} else

JOptionPane.showMessageDialog(null, "Invalid Login or Password!")

} catch (Exception e2) { e2.printStackTrace(); }

}

if(ae.getSource() == b2){ setVisible(false);

Signup su = new Signup(); su.setVisible(true);

}

if(ae.getSource() == b3){

setVisible(false);

ForgotPassword forgot = new ForgotPassword(); forgot.setVisible(true);

}

}

public static void main(String[] args) { new Login().setVisible(true);

}

}

#### Limitations and Bibliography

We have taken references from many resources like YouTube and many websites and Books.

Books:

* + - Java and Integrated Approach by Nageswara Rao
    - Java Programming by Nirali Publication Websites:
    - [https://www.stackoverflow.com](https://www.stackoverflow.com/)
    - [https://www.javatpoint.com](https://www.javatpoint.com/)
    - [https://www.codecademy.com](https://www.codecademy.com/) YouTube video links:

1. <https://youtu.be/5vzCjvUwMXg>
2. <https://youtu.be/dwVj_g3TpZ4>
3. <https://youtu.be/L5RpqspNAuc>