**A**

**PROJECT REPORT**

**ON**

# TOURS & TRAVELS MANAGEMENT

# SYSTEM

SUBMITTED BY

**Ms. Sonawane Vaishnavi Navnath**



**SUBMITTED TO**

## **SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE**

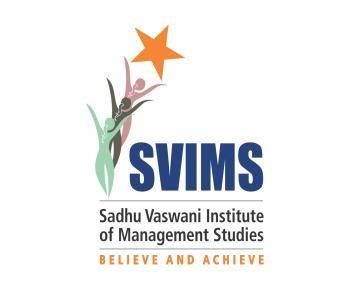
IN FULFILLMENT OF DEGREE

### **MASTER OF COMPUTER APPLICATION (SEM-II)**

UNDER THE GUIDANCE OF

**Dr. Shveti Chandan**

Through,



**Sadhu Vaswani Institute of Management Studies for Girls,**

**Koregaon Park, Pune 411001**

**2024-25**



**DECLARATION BY STUDENT**

To,

The Director,

SVIMS, Koregaon Park, Pune

I, undersigned hereby declare that this project titled, **"Tours & Travel Management System"** written and submitted by me to SPPU, Pune, in partial fulfilment of the requirement of the award of the degree of **MASTER OF COMPUTER APPLICATION (MCA-II)** under the guidance of Dr. Shveti Chandan, is my original work.

I further declare that to the best of my knowledge and belief, this project has not been submitted to this or any other University or Institution for the award of any Degree.

**Place: Pune**

**Date: (Sonawane Vaishnavi Navnath)**

**ACKNOWLEDGMENT**

I extend my sincere gratitude to **Dr. B.H.Nanwani**, **Dr. Neeta** **Raskar** and **Dr. Shveti Chandan** for allowing me to carry out the study and for their constant encouragement, valuable suggestions and guidance during the research work.

I extend my special thanks to **Mrs.Deepali Gavhane** for their kind co-operation and inspiration.

I extend my special gratitude to my dearest family members and friends who encouraged and motivated me to complete the project report.

Place: Pune

Date :

Sonawane Vaishnavi Navnath

Your Name and Signature

**INDEX**

|  |  |  |
| --- | --- | --- |
| **CHAPTER** | **DETAILS** | **PAGE NO** |
| 1 | **CHAPTER 1: INTRODUCTION**   * 1. Client/Organization Profile   2. Need for System   3. Scope & Feasibility of Work   4. Operating Environment – H/w & S/w   5. Architecture of system   6. Detail Description of Technology Used | 3 - 12 |
| 2 | **CHAPTER 2: PROPOSED SYSTEM**   * 1. Proposed System   2. Objectives of System   3. User Requirements | 13 |
| 3 | **CHAPTER 3: ANALYSIS & DESIGN**   * 1. DFD   2. Table specifications (Database)   3. ERD   4. Class Diagram   5. Use Case Diagrams | 14 – 19 |
| 4 | **CHAPTER 4: USER MANUAL**  4.1User Interface Design (Screens etc.)  4.2Limitations  4.3Future enhancement  **BIBLIOGRAPHY**  **ANNEXURE:** Sample program code | 20– 42 |

**CHAPTER 1: INTRODUCTION**

* 1. **client/organization profile**

The “Tours And Travels Management” has been developed to override the problems in the practicing manual system. This software is supported to eliminate and, in some cases, reduce the hardships faced by this existing system. Moreover, this system is designed for the need of the company to carry out operation in a smooth and effective manner.

Today, the management of gym details by computer systems has been popularized in many franchises. However, there are still many problems to develop and configure system transactions. It is required to maintain every Franchise

must develop own software. It takes lesser time these days to build and develop information systems according to user needs.

**1.2Need for System**

1)Booking Management: Efficiently handle booking requests, including reservations, cancellations, and modifications for various tour packages, accommodations, transportation, and activities.

2)Customer Management: Maintain a database of customer information, including preferences, past bookings, and contact details, to provide personalized services and targeted marketing.

3)Inventory Management: Keep track of available resources such as hotel rooms, rental vehicles, tour.guides, and tickets to ensure optimal utilization and prevent overbooking.

4)Due to the system, searching & modification makes easy.

5)To prevent unauthorized access in accounting.

6)It gives fast service to customer.

7)No need of extra things.

8)Communication and Notifications: Implement communication channels such as email, SMS,

or notifications within the application to keep customers informed about booking confirmations .

9)Enhance Security : - With features like access and member authentication

**1.3 Scope & Feasibility of Work**

**Scope:**

The website is developed based on real life. Today's extremely exhausting work environment dedicates that individuals requires some joyful holiday. The website will provides a stress free joyful refreshing holiday with cost comparative and customized packages according to their requirements. As it is web based it can provide service in almost each and every city of India and whole World. It offer tour and travel services

include tickets, booking, holiday tour package, tour package only in one click. It provide the most suitably designed as well as the customized travel packages to the customer. for tour operating companies. The main aim of this project is to help the tourism

Tours and Travels management system is an integrated software developed companies to manage their customers and agents. It makes all operation of the tour company easy and accurate. The standalone platform makes tourism management easy by handling agencies requests and providing servers for the customers located at different parts of the various cities.

**Feasibility of work:**

This system USES VS CODE as its foreground development tool, it has the powerful database development function and the rich table, the graph output function, the actual effect exquisite report form printing function, easy to read with the flexible language, the rapid friendly development interface and so on the characteristic. Using Java to develop database projects can accelerate and efficiently make database management projects. This system background USES the MySQL, which is a new type of interactive relational database management system.

Feasibility Study:

* **Technical Feasibility:**

This includes the study of function, performance & constraint that may affect the ability to achieve an acceptable system. For this feasibility study, we studied complete functionality to be provided in the system, as described in the Systems Requirement Specification (SRS), & checked if everything was possible using different type of frontend & backend platforms.

**Operational Feasibility:**

No doubt the proposed system is fully GUI based that is very user friendly & all inputs to be taken all self-explanatory even to a layman. Besides, a proper training has been conducted to let known the essence of the system to the users so that they feel comfortable with new system. As far our study is concerned the clients are comfortable & happy as the system has cut down their loads & doing.

Economic Feasibility:

This is a very important aspect to be considered while developing a project. We decided the technology based on minimum possible cost factor. All hardware & software cost must be borne by the organization.

Overall, we have estimated that the benefits the organization is going to receive from the proposed system will surely overcome the initial costs & the later running cost for system.

**1.4 Operating Environment – Hardware & software**

**Hardware Configuration:**

Processor : Intel core series , AMD Ryzen Series .

RAM : 8 GB or More.

Hard Disk Drive : SSD or another 500GB or More

**Software Configuration:**

Frontend - Java(via Swing/AWT for GUI),VS Code as IDE

Backend - Java

Database - MySQL(Relational Database)

IDE - VS Code

Documentation – MS Word

Operational System – Windows 11

**1.5 Architecture of system**

The architecture of the Tours and Travels Management System follows a three-tier architecture, ensuring modularity, maintainability, and separation of concerns. Each layer handles specific responsibilities and communicates with the others in a structured manner:

**1)Presentation Layer (Frontend/UI Layer)**

🔹 Technology Used:

* Java Swing/AWT (GUI Components)

🔹 Description:

* This is the user interface layer where both customers and administrators interact with the system.
* It includes screens such as:
  + Login/Sign-up Forms
  + Booking Form
  + Admin Dashboard
  + Customer Profile and Reports
* The GUI components handle user input and display outputs based on the backend logic.

🔹 Role:

* Accepts user commands (e.g., booking, login)
* Shows data retrieved from the database
* Ensures user-friendly interaction

2) **Business Logic Layer (Application Layer)**

**🔹** Technology Used:

* Java

🔹 Description:

* This layer contains the core logic of the application.
* Handles:
  + Data validation (e.g., verifying booking details)
  + Processing user requests (like calculating payable amounts, filtering tours)
  + User authentication
  + Generating bills and reports

🔹 Role:

* Acts as a bridge between the UI and database
* Ensures that business rules are enforced
* Controls data flow and application processes

3) **Data Layer (Database Layer)**

**🔹** Technology Used:

* MySQL

🔹 Description:

* Stores and manages all the persistent data used by the system, such as:
  + Customer details
  + Tour packages
  + Booking records
  + Admin credentials
* MySQL ensures data integrity, indexing, and relational storage with SQL queries.

🔹 Role:

* Responds to queries from the business layer
* Maintains structured data storage
* Ensures security and consistency

**1.6** **Detail Description of Technology Used**

Technology used:

|  |
| --- |
| * Java (via Swing/AWT for GUI), VS Code as IDE |

|  |  |
| --- | --- |
| * **Backend (Logic):** | Java |

|  |  |
| --- | --- |
| * **Database** | * :MySQL (Relational Database) |

**1. Frontend Development: Java (Swing/AWT)**

* The Graphical User Interface (GUI) of the application is built using Java Swing or AWT, providing a window-based environment.
* Features like forms, buttons, tables, and text fields are designed for both admin and user interaction.
* Offers a user-friendly, responsive interface for booking, registration, and data management.

**2. Backend Development: Java**

* The application's business logic and core functionalities are written in Java.
* Java is known for its platform independence, security, and robust object-oriented structure, making it ideal for enterprise-level applications like this.
* It handles operations like booking validation, data retrieval, and user authentication.

**3. Database: MySQL**

* MySQL, a powerful relational database management system (RDBMS), is used to store:
  + Customer information
  + Booking records
  + Tour packages
  + Payment and billing data
* SQL queries are used for data insertion, retrieval, and management.
* Ensures data consistency, integrity, and efficient storage.

4**. IDE: Visual Studio Code (VS Code)**

* VS Code is used as the primary Integrated Development Environment (IDE).
* Offers support for Java, extensions for SQL, and tools for version control (Git).
* Provides a smooth development experience with debugging and syntax support.

**5. Operating System: Windows**

* The system is developed and tested on Windows 11.
* Compatible with most standard desktop environments, making it easy to deploy in offices or small travel agencies.

**6. Documentation Tools: MS Word**

* Microsoft Word is used to prepare system documentation including:
  + Project reports
  + User manuals
  + System requirements
  + Diagrams (ER, Use Case, Class Diagrams)

**CHAPTER 2: PROPOSED SYSTEM**

**2.1 Proposed System**

The proposed Tours and Travels Management System is a desktop-based application designed to automate and streamline the operations of a travel agency. The system replaces the traditional manual process of managing bookings, customer details, tour packages, and payments with a computerized and user-friendly interface.

It allows customers to register and book their travel plans seamlessly, while administrators can manage bookings, view reports, and access customer information securely. The software ensures real-time data access, reduces paperwork, minimizes human error, and provides efficient service delivery. Developed using Java for the frontend, MySQL for the backend database, and operated via VS Code, the system is designed to be scalable and adaptable to the evolving needs of travel businesses.

This centralized application supports booking management, customer tracking, inventory of resources (vehicles, hotels, guides), and automated billing, making it a comprehensive solution for travel companies aiming to modernize their services.

**2.2 Objectives of System:**

1) To Reduce paper Work and save time.

2) To minimize lengthy processes.

3) To manage tours data and details.

4) To Avoid Errors.

5) To handle database details using computer system.

**2.3 User Requirements**

**🔸** *Functional Requirements:*

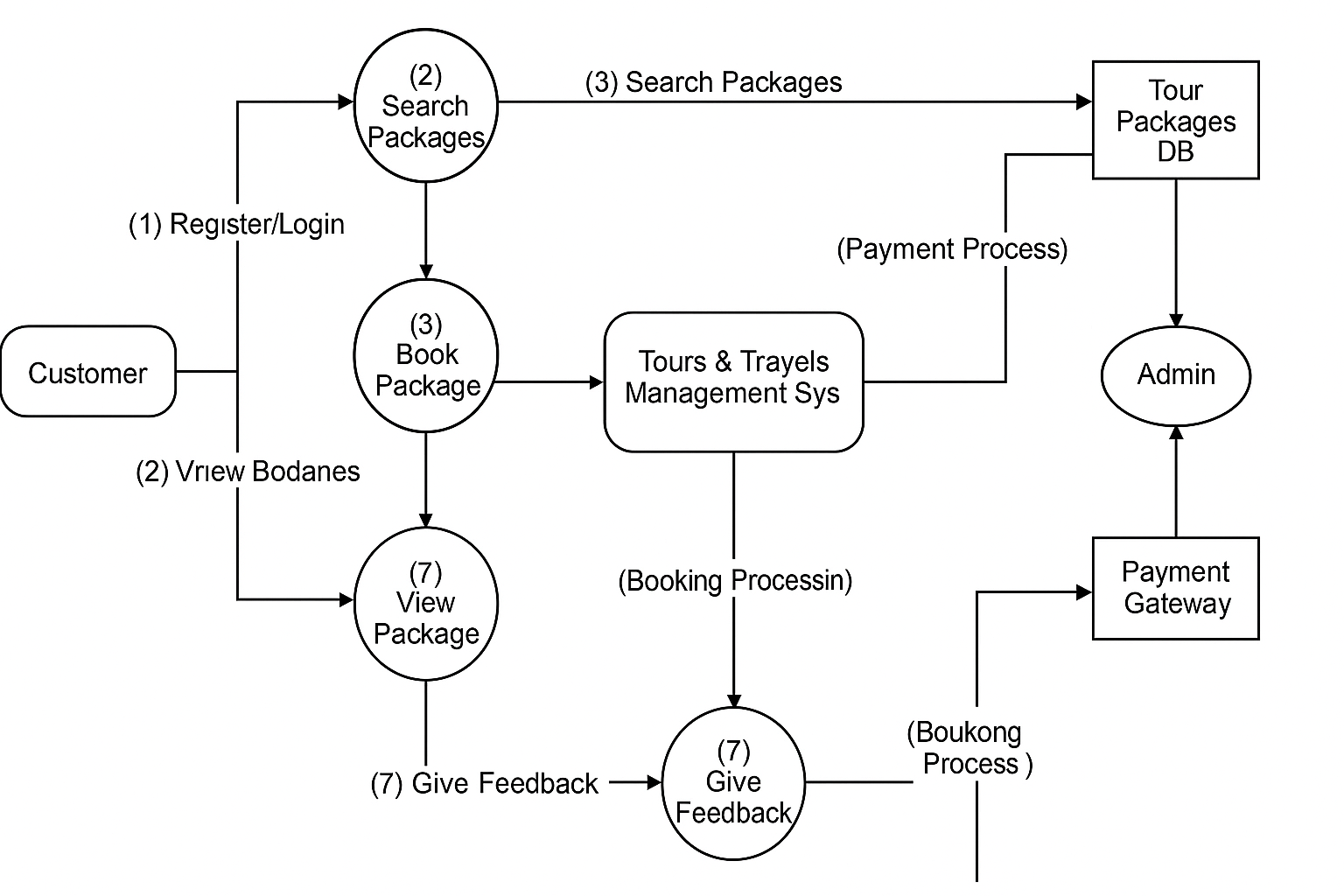
1. Users must be able to register, log in, and log out securely.
2. Users can view and book tours, including entering journey details.
3. Admin can view reports, manage bookings, and access customer data.
4. The system must store customer and booking data in a secure database.
5. Users should receive confirmation notifications after bookings.

🔸 *Non-Functional Requirements:*

1. The system should be user-friendly and GUI-based.
2. Should ensure data security and access control.
3. Must provide fast performance for all actions.
4. Should be easily maintainable and scalable for future enhancements.

**CHAPTER 3: ANALYSIS & DESIGN**

**3.1 DFD**



### **3.2 Table Specification (Database)**

**MYSQL Table Structure**

**Table Name: tbllogin**

+----------------+-------------- +-----+-------+----------+-------------------+

| Field | Type | Null | Key | Default | Extra |

+---------------+--------------+------+-------+-----------+------------------+

| ID | int | NO | PRI | NULL | auto\_increment |

| NAME | varchar(100) | NO | | NULL | |

| MOBILE | varchar(100) | NO | | NULL | |

| ADDRESS | varchar(100) | NO | | NULL | |

| EMAIL | varchar(100) | NO | | NULL | |

| USERNAME | varchar(100) | NO | | NULL | |

| PASSWORD | varchar(100) | NO | | NULL | |

+-------=------ +----------------+------+-----+---------+--------------------+

**Table Name: tblbooking**

**+----------------------------+-----------------+------- +------+----------+----------------------+**

**| Field | Type | Null | Key | Default | Extra |**

**+---------------------------- +-------------- +-------+-------+---------- + --------------------+**

**| ID | int | NO | PRI | NULL | auto\_increment |**

**| NAME | varchar(100) | NO | | NULL | |**

**| SOURCE | varchar(100) | NO | | NULL | |**

**| DESTINATION | varchar(100) | NO | | NULL | |**

**| NO\_OF\_PASSENGERS | varchar(100) | NO | | NULL | |**

**| TIME | varchar(100) | NO | | NULL | |**

**| DATE | varchar(100) | NO | | NULL | |**

**| AMOUNT | varchar(100) | NO | | NULL | |**

**| BOOKING\_STATUS | varchar(100) | NO | | BOOKED | |**

**+------------------------------+------------------+------+-----+--------------+------------------+**

**Table Name: admin**

**+------------------+---------------+---------+-------+---------+----------------------+**

**| Field | Type | Null | Key | Default | Extra |**

**+------------------+--------------+---------+-------+---------+-----------------------+**

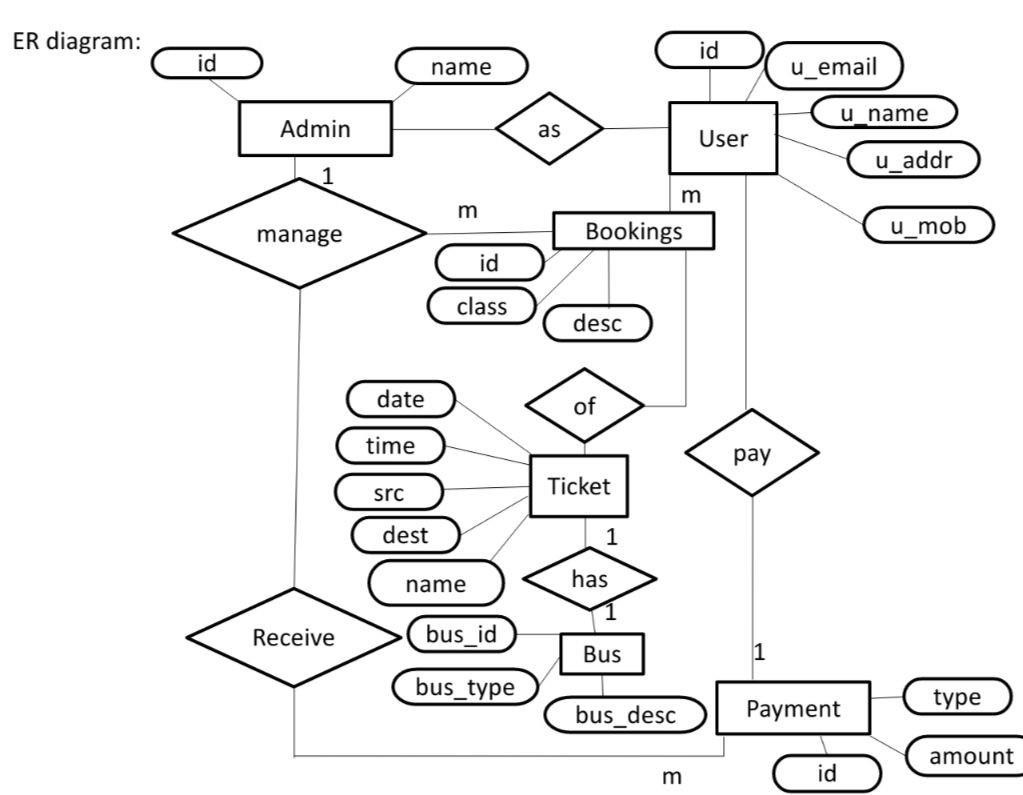
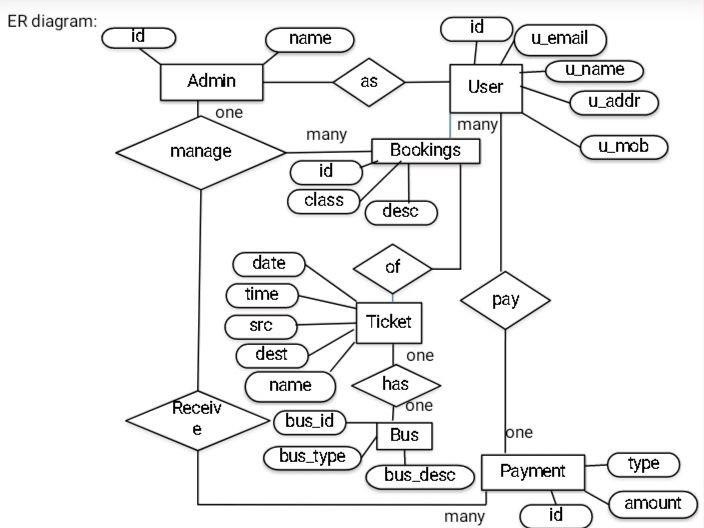
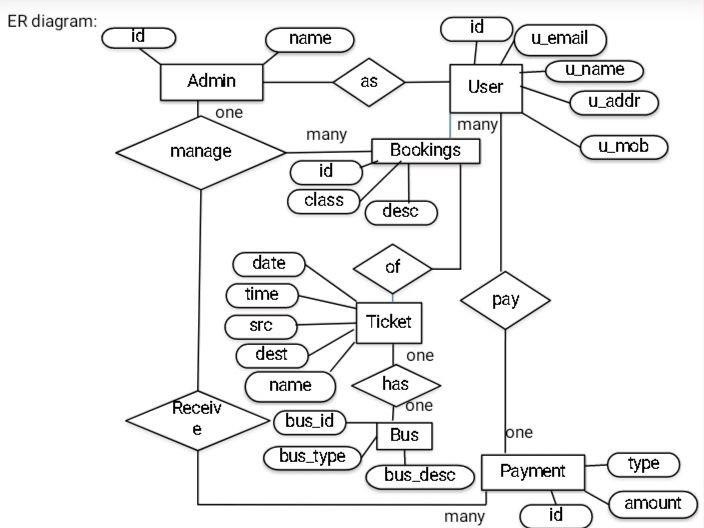
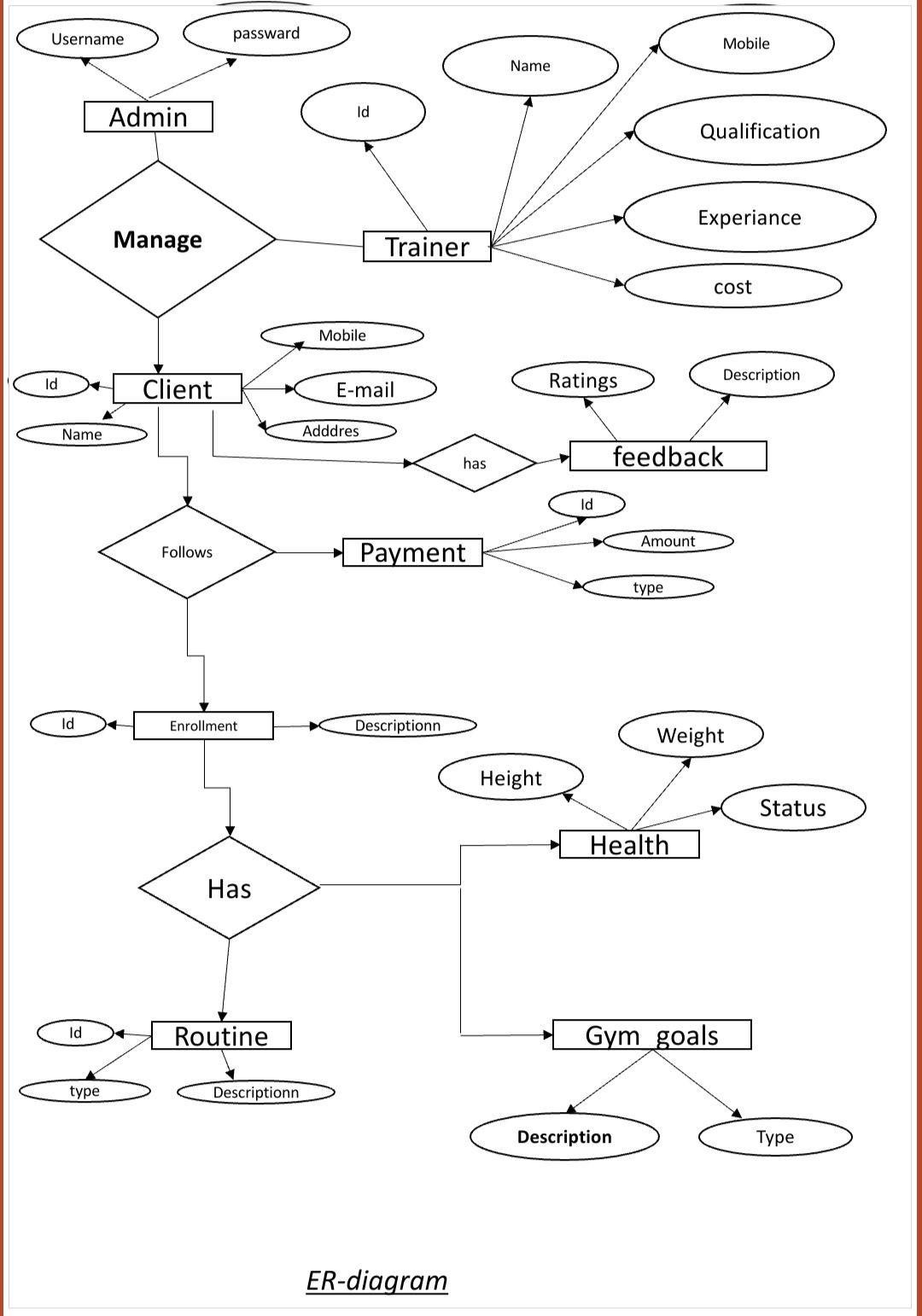
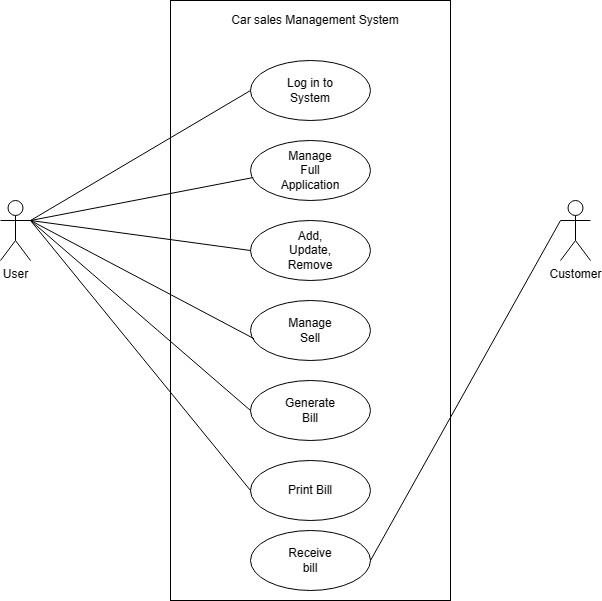
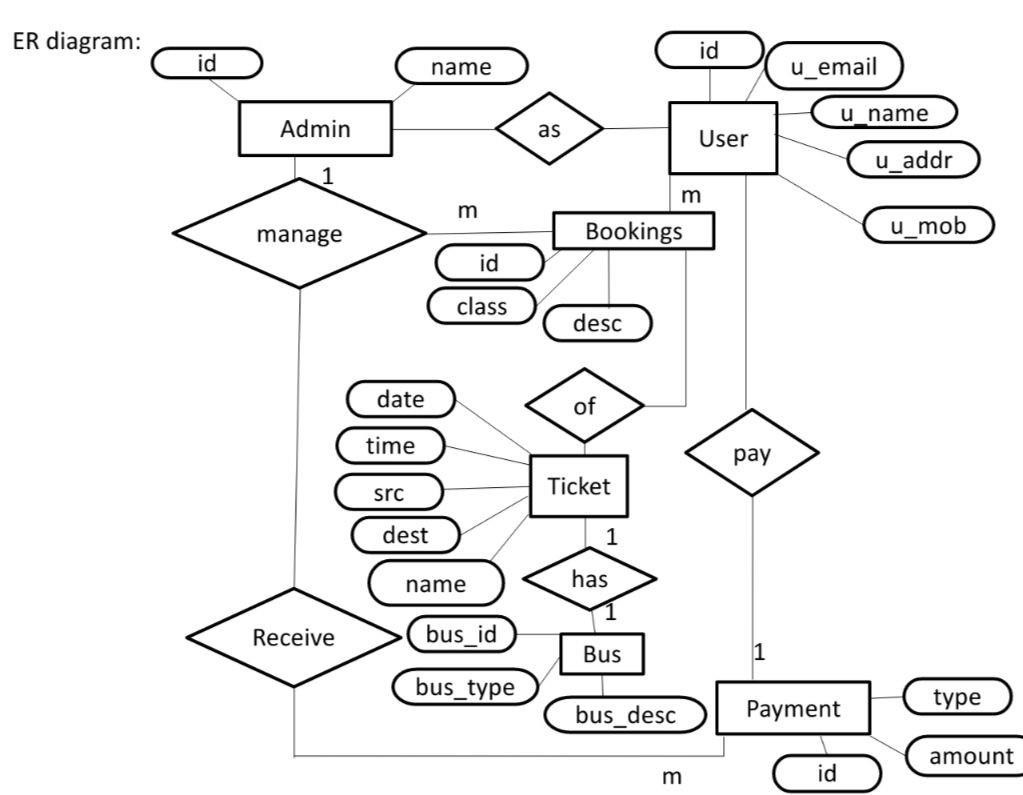
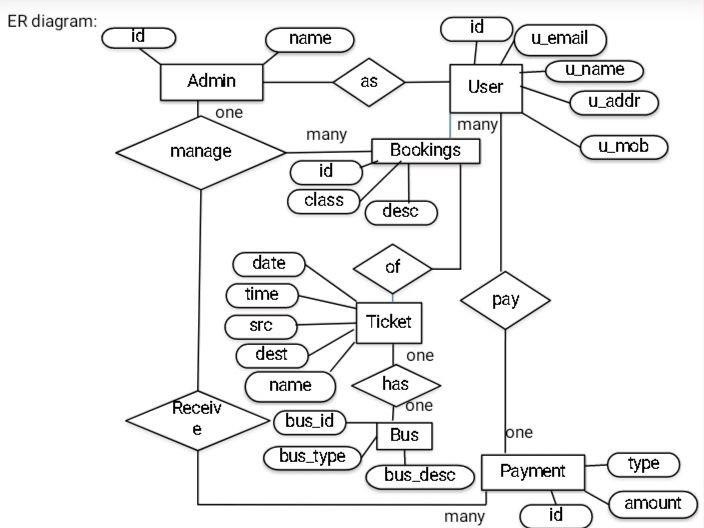
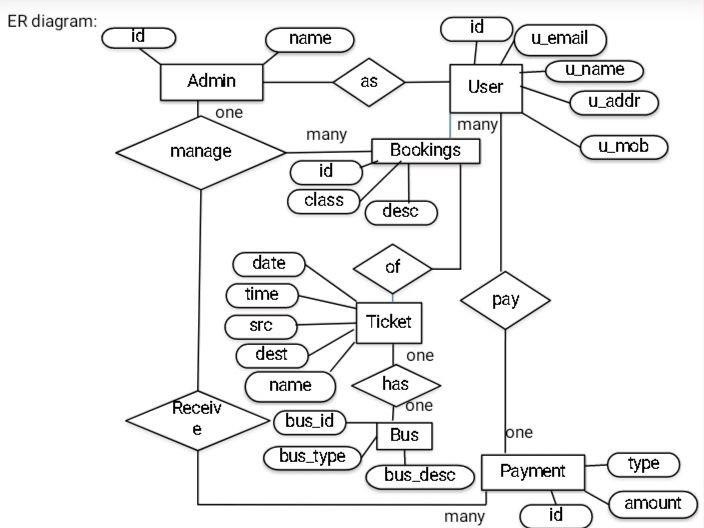
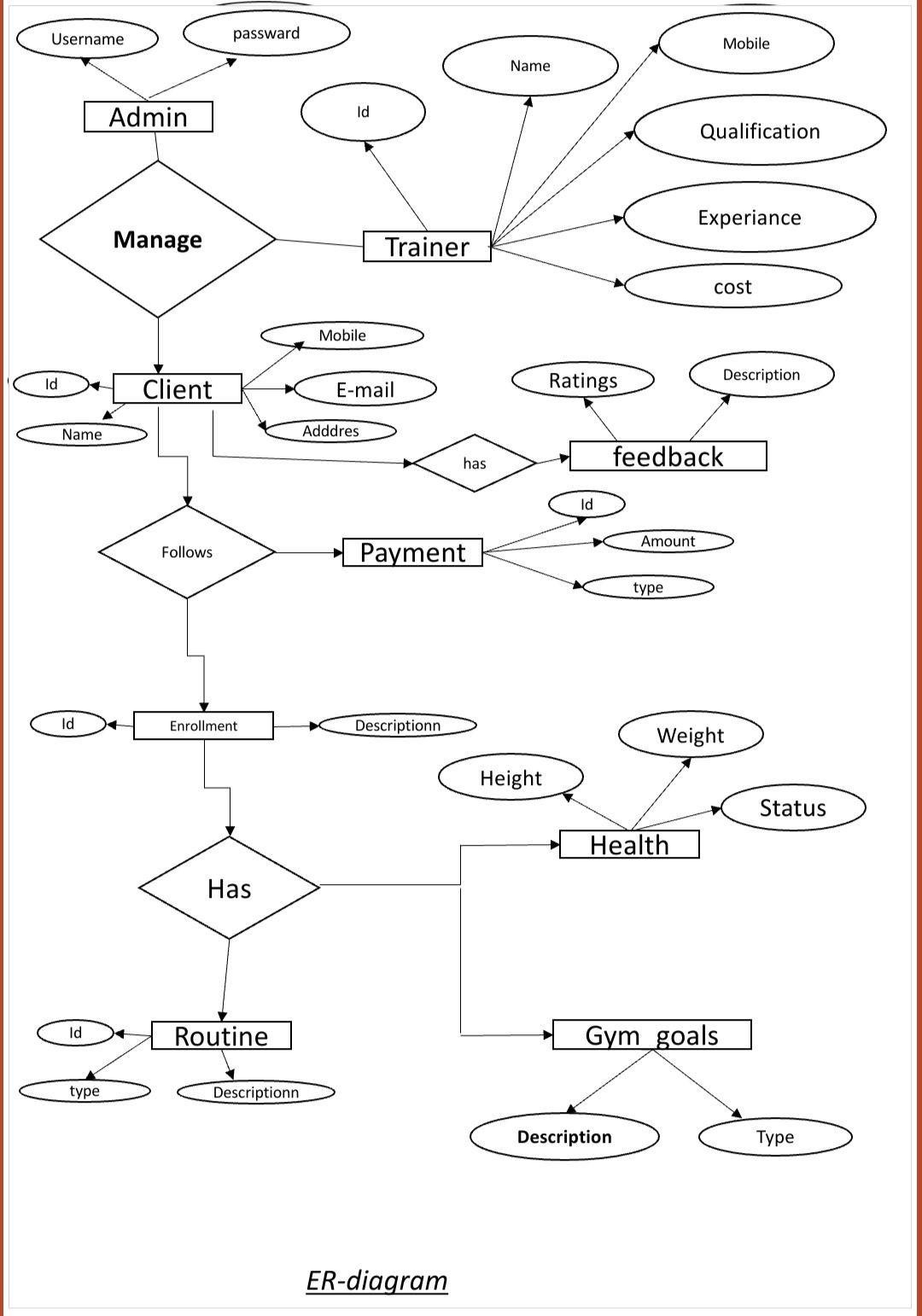
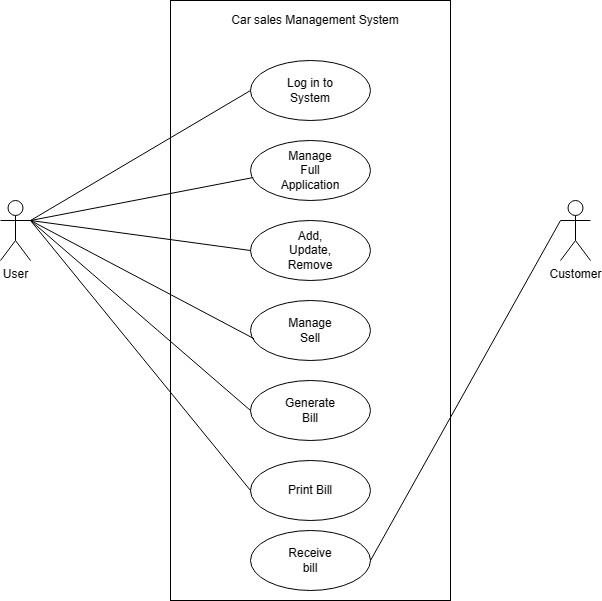
**| ID | int | NO | PRI | NULL | auto\_increment |**

**| USERNAME | varchar(100) | NO | | NULL | |**

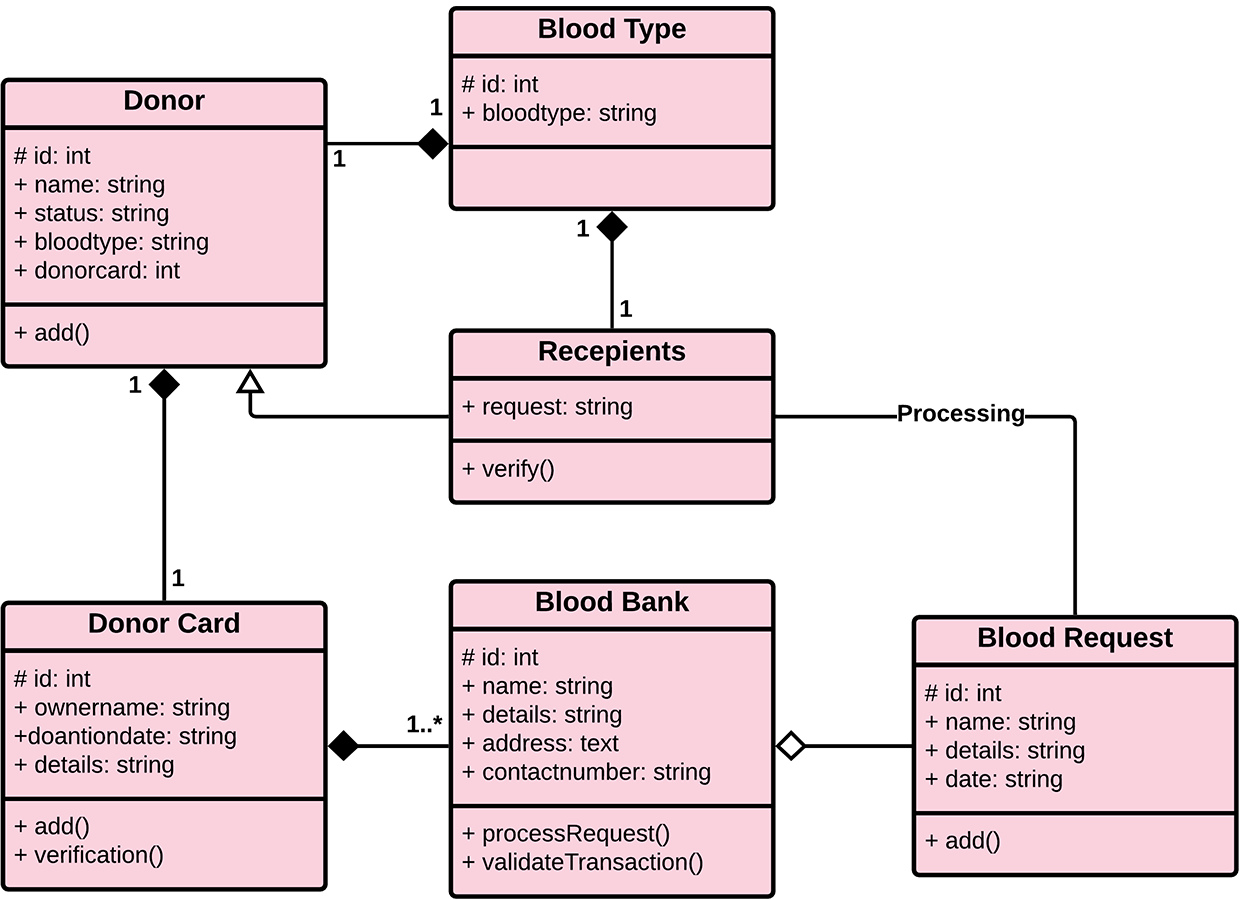
**| PASSWORD | varchar(100) | NO | | NULL | |**

**+-----------------+------==--------+------+-----+--------=------+-------==---------+**

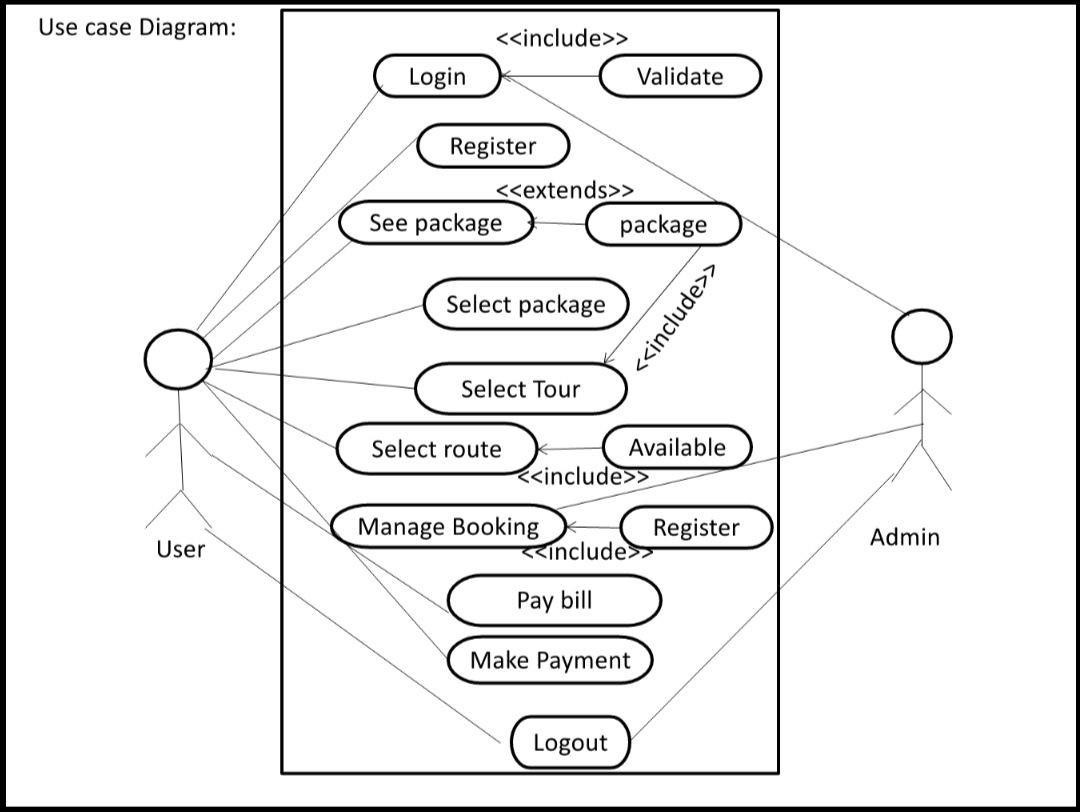
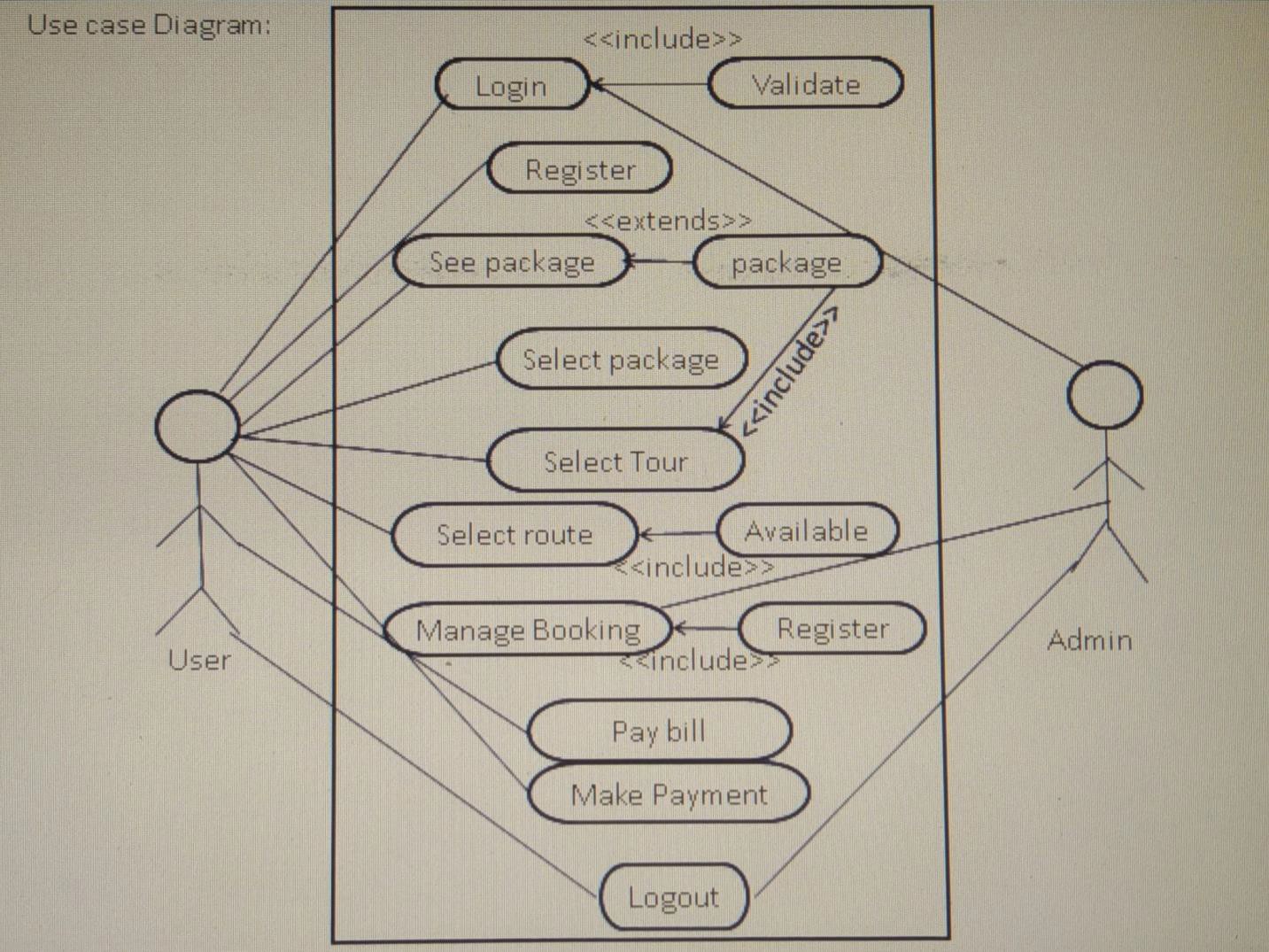
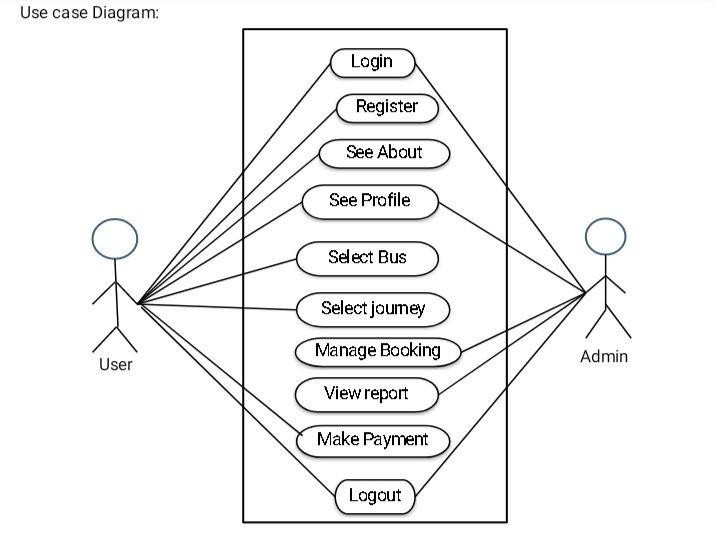
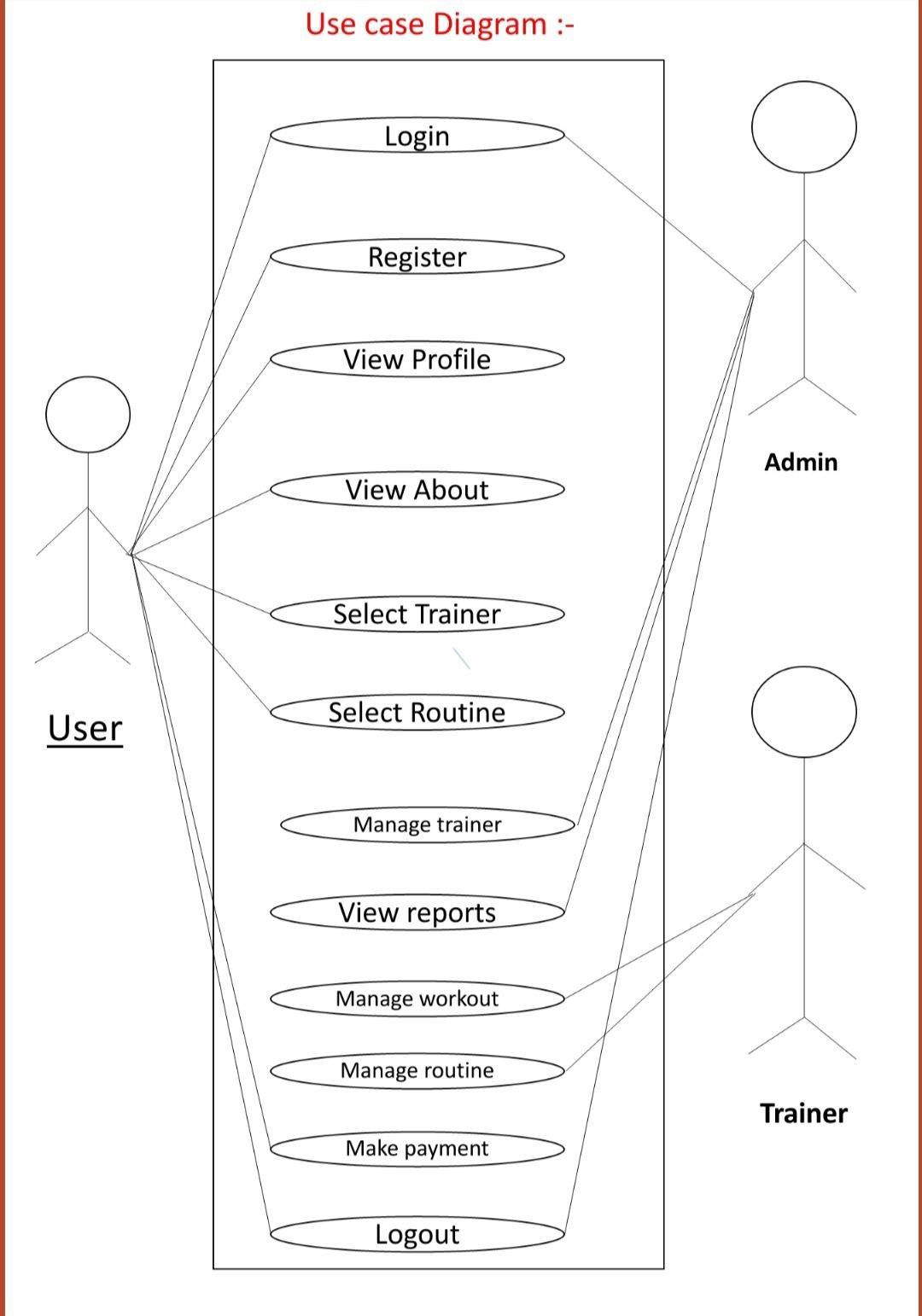
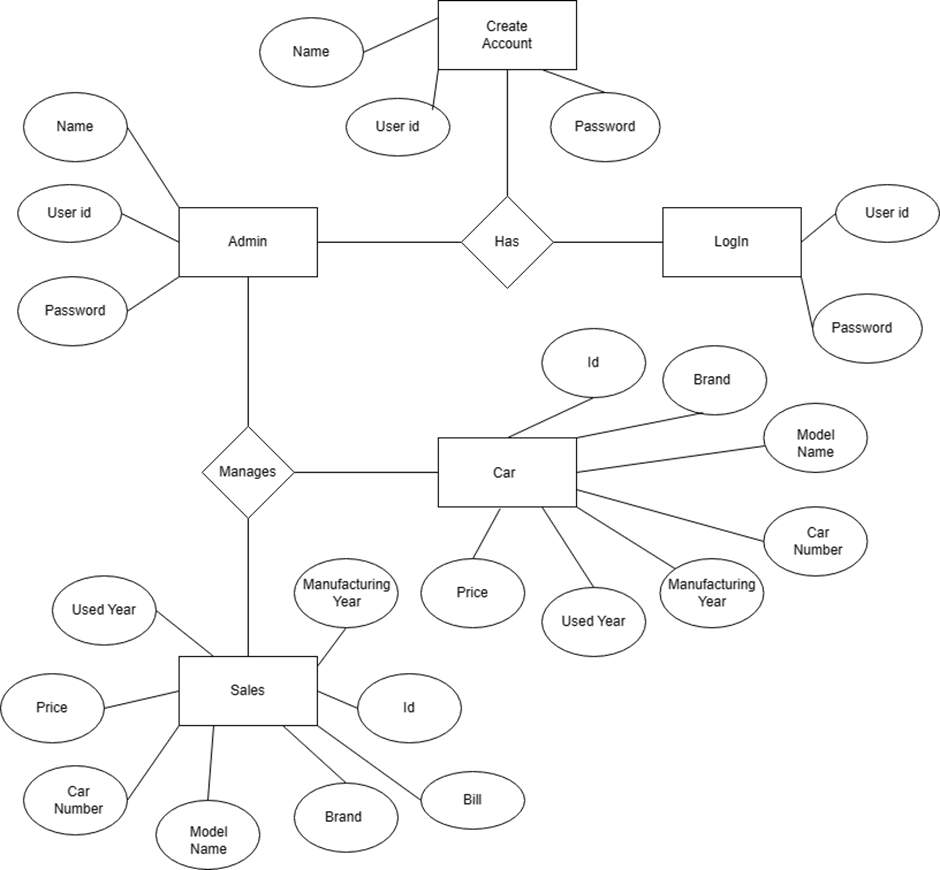
**3.3 ER diagram**



**3.5** **Class Diagram**

****

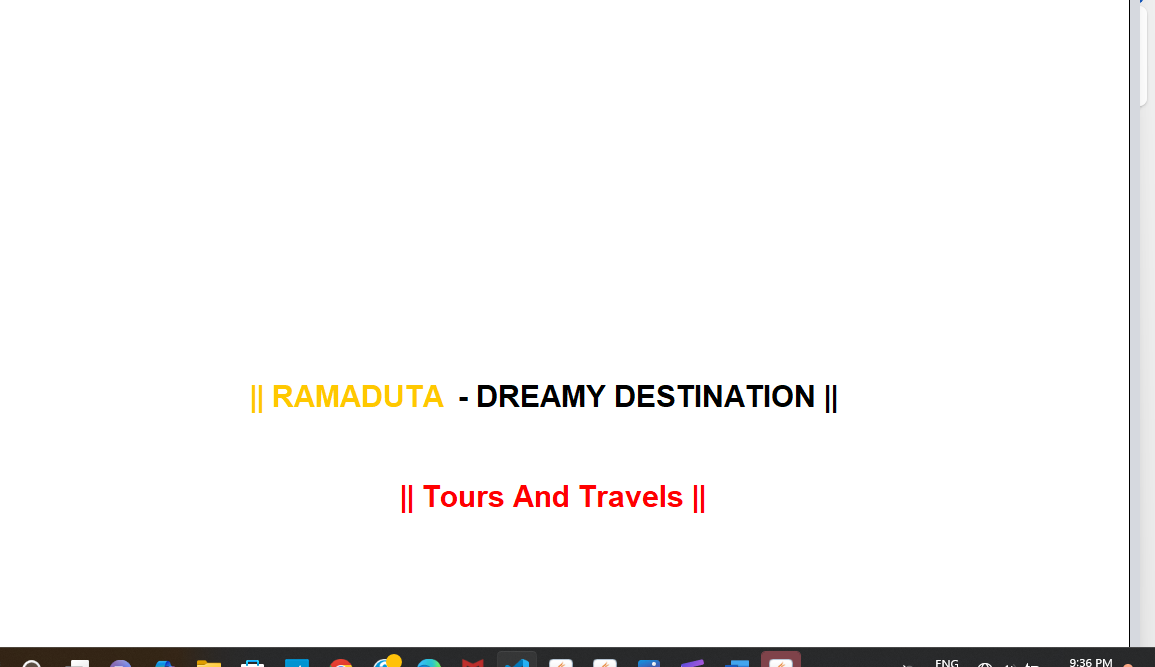
**3.6 Use Case Diagram**



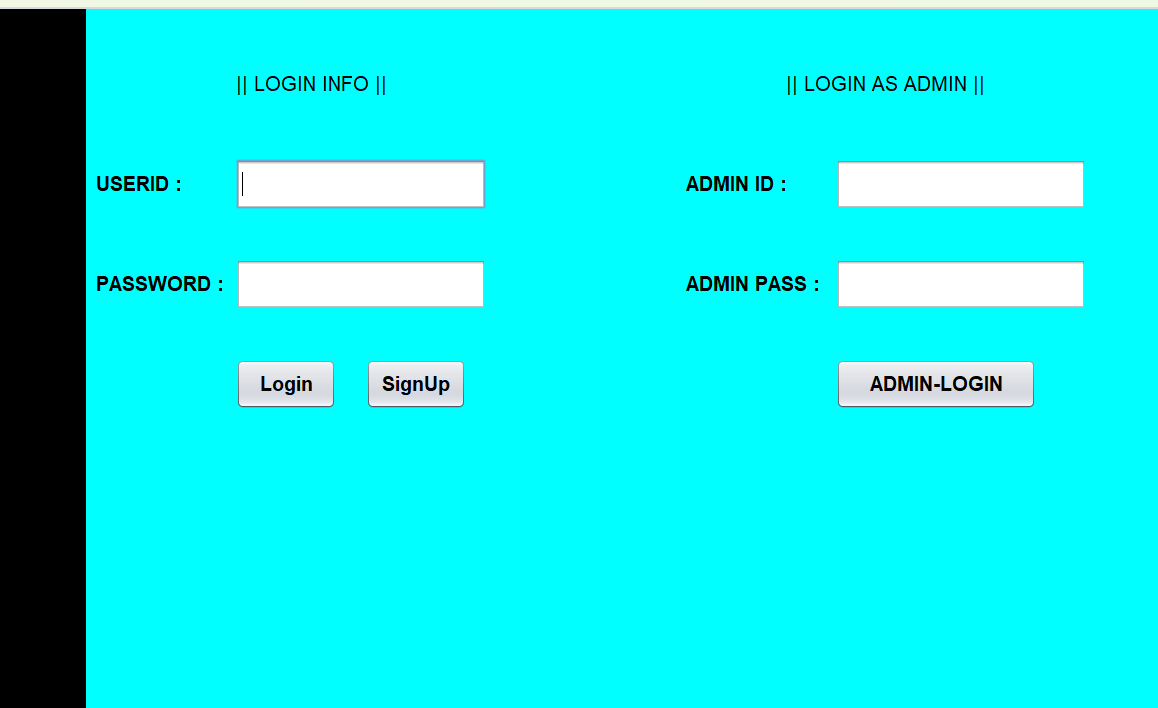
**CHAPTER 4:USER MANUAL**

**4.1 User interface Design**

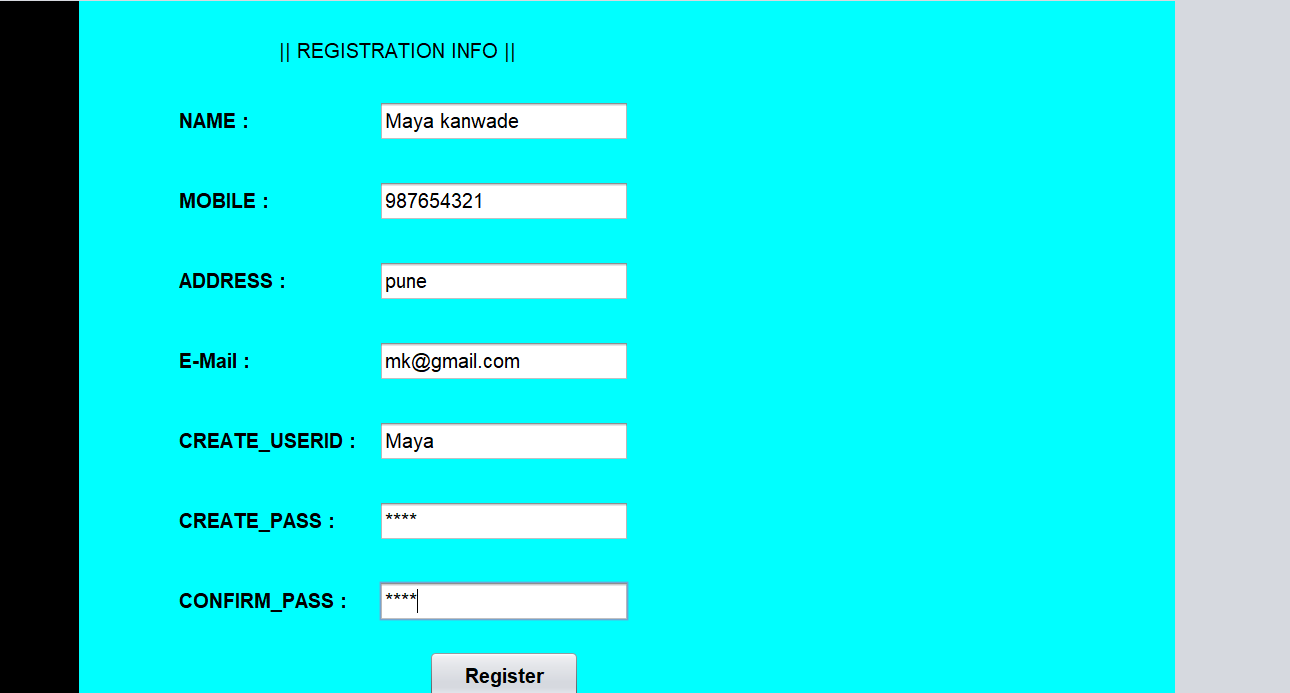
* **Welcome Windows**: It is welcome page for users and admin.

****

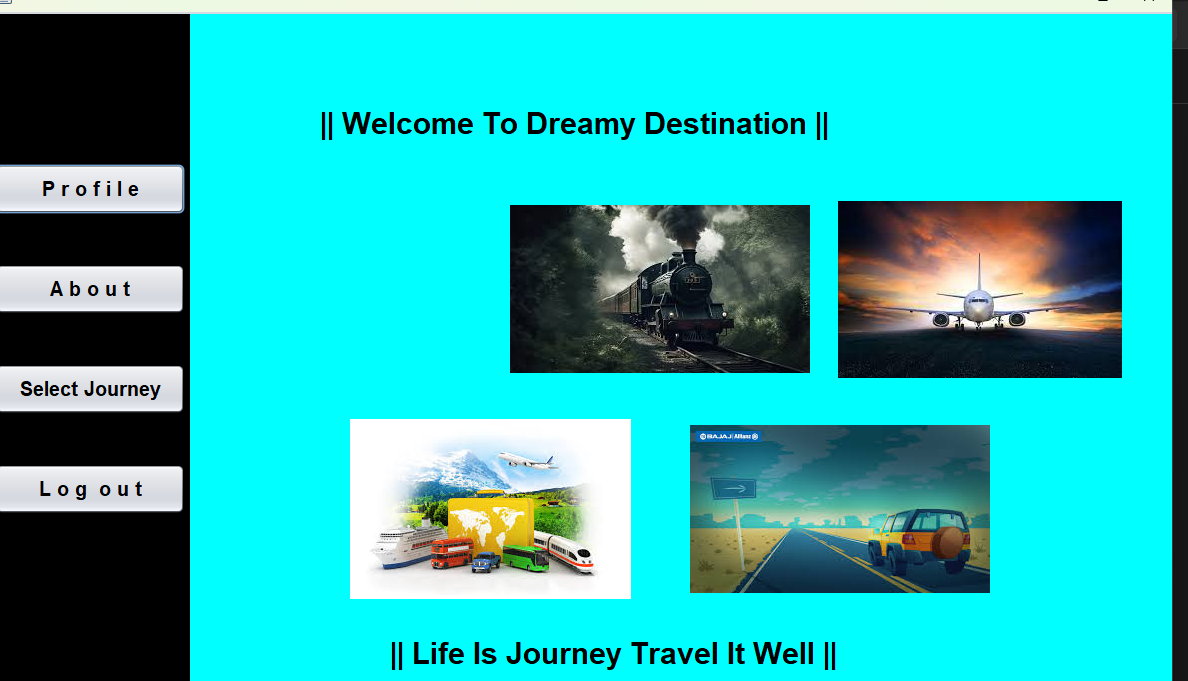
* **Login Page :** Login details refer to the credentials (username and password) required to authenticate and grant access to a system or application.

****

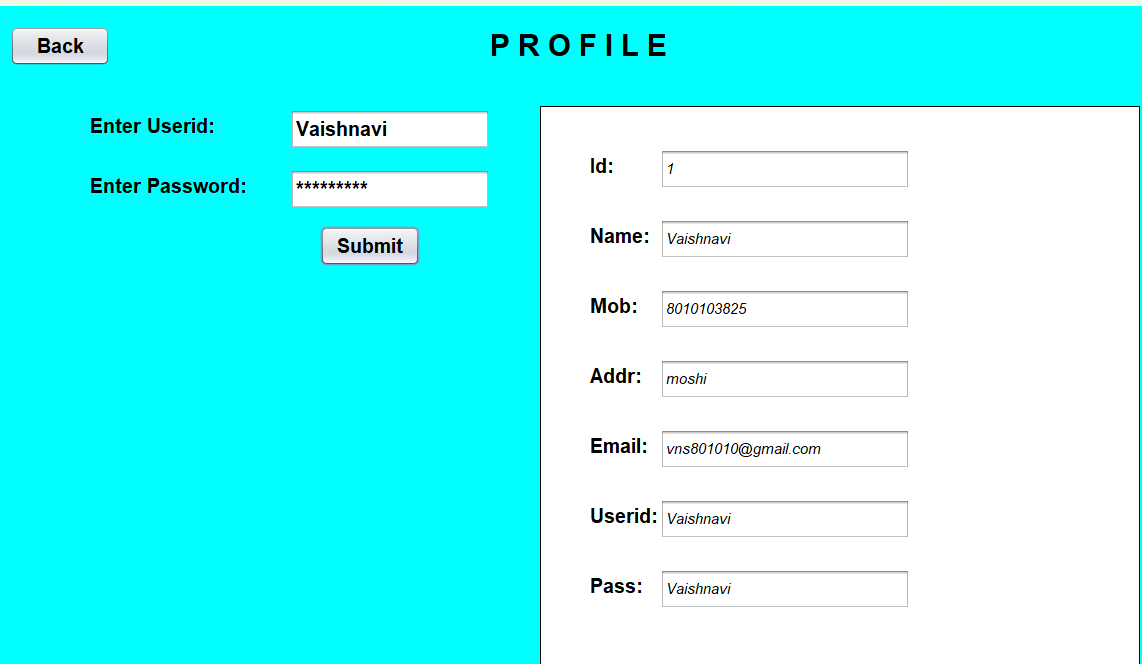
* **Registration Form :-** A registration form collects essential user information to create a new account or sign up for a service.

****

* **Home Page:-** This is the Main Dashboard of the application titled “Welcome To Dreamy Destination”.

****

* **Profile page:-** A profile page displays a user's personal information, preferences, and account details in one centralized location.

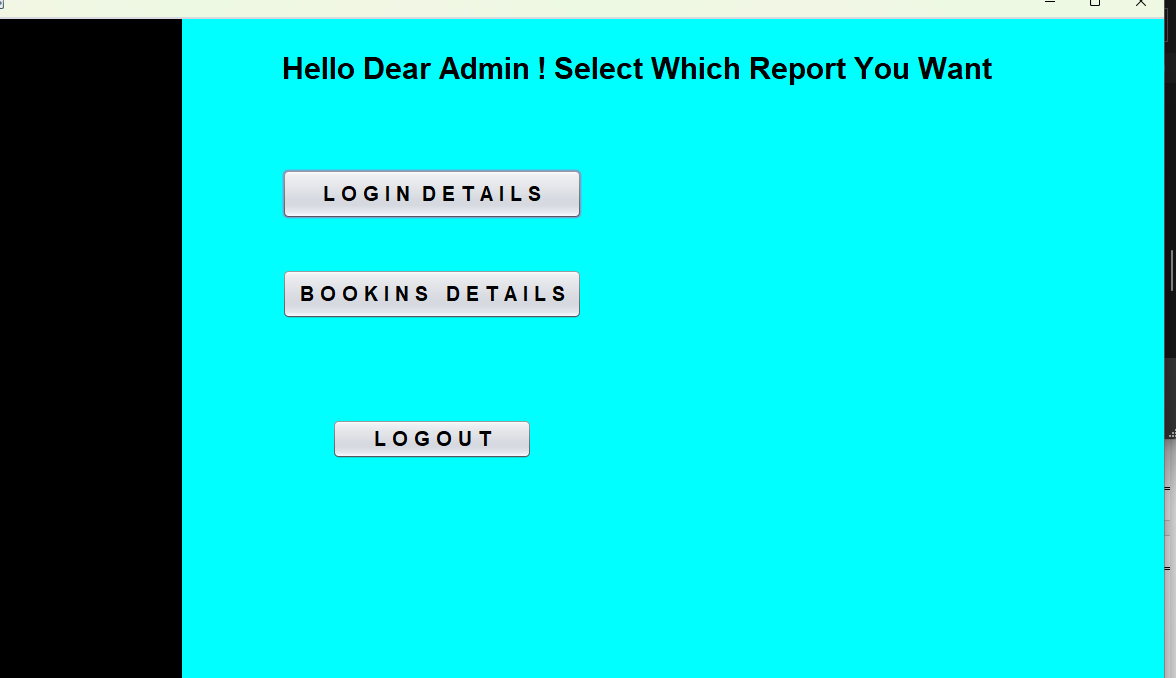
****

* **About Company : -**In this section about of company.



* **Booking Form :-** Form used to collect booking details including source, destination, number of passengers, travel date, and payment amount**.**

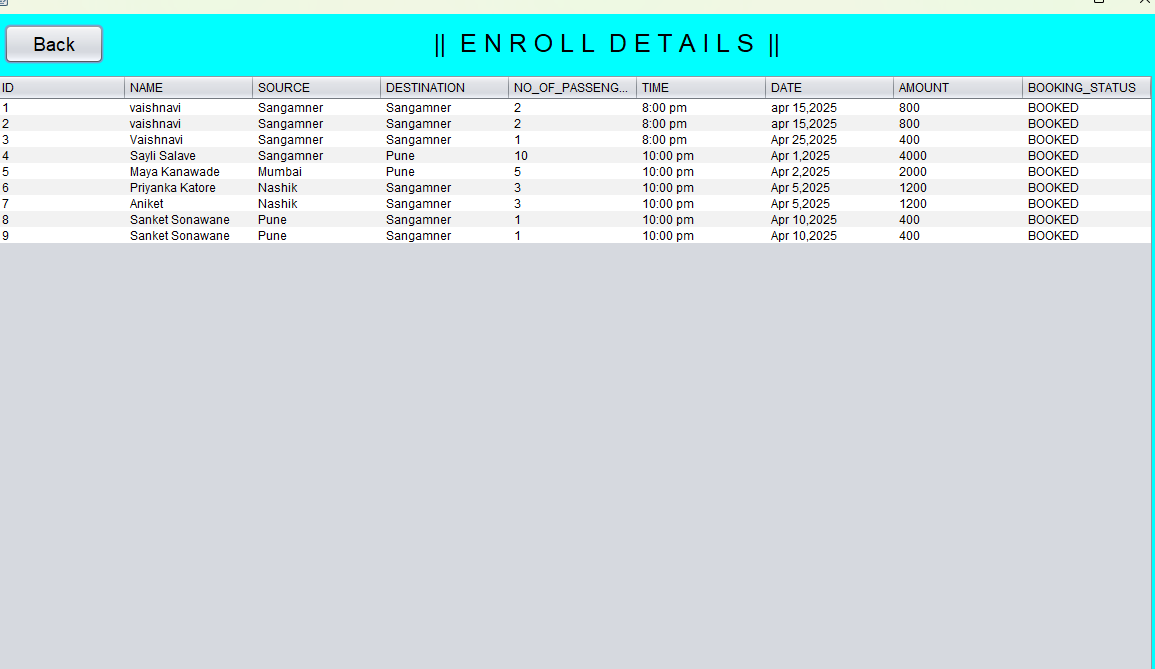
**Home Page For Admin :-** Home page providing the admin with access to core functionalities like employee management, enrollment, and record viewing.



* **Login Details :-** This page is the Login Interface for users to access the system securely



* **Booking Details : -** Section for entering and storing new employee enrollment information including ID, name, department, and salary.

****

* 1. **Limitations**

 **Offline System:** It is a desktop-based application; no online access or cloud synchronization.

 **No Online Payment Integration:** Payment methods are not connected to real-time gateways.

 **Requires Basic Computer Skills:** Not suitable for people unfamiliar with computer systems.

**4.3Future Enhancement**

* **Make the System Web-Based:** Convert the desktop system into a cloud-hosted website accessible via browsers.
* **Mobile App Development:** Create Android/iOS versions for customer and admin use.
* **Online Payment Integration:** Add support for credit card, UPI, and wallet payments.
* **Advanced Billing:** Generate automated tax-inclusive invoices with breakdowns.
* **Analytics Dashboard:** Visualize user behavior, bookings, and revenue through charts.

BIBLIOGRAPHY

 W3Schools.  
Topics on HTML, CSS, and MySQL (for any future enhancement ideas).  
Website: <https://www.w3schools.com>

 TutorialsPoint.  
Reference for MySQL database operations and Java GUI development.  
Website: <https://www.tutorialspoint.com>

 MySQL Documentation.  
Available at: <https://dev.mysql.com/doc/>  
– Used for syntax reference, queries, and database optimization.

 GeeksforGeeks.  
Articles on Java Swing, MySQL connectivity, and application development.  
Website: <https://www.geeksforgeeks.org>

**ANNEXURE: Sample program code**

**About :**

package Travel\_Mgmt;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.UIManager;

import Travel\_Mgmt.home1;

class About1 implements ActionListener{

    JPanel p1;

    JButton b1;

    final JWindow w;

    public About1()

    {

        w=new JWindow();

        w.setBounds(150, 20, 1200, 750);

        p1=new JPanel();

        p1.setLayout(null);

        p1.setBounds(10, 10, 1180, 730);

        p1.setBorder(BorderFactory.createLineBorder(Color.BLACK));

        JLabel l1=new JLabel("||  A B O U T   O U R S E L F  ||");

        l1.setBounds(350, 50, 700, 50);

        Font f1=new Font("arial", Font.BOLD, 30);

        Font f2=new Font("arial", Font.ITALIC,20);

        l1.setFont(f1);

        p1.add(l1);

        JTextArea ta1=new JTextArea();

        ta1.setBounds(50,150,1100, 550);

        ta1.setText("                                                                                           \r\n" +//

                 " Founded in the year 2005 as RAMDUTA.com, an online travel platform, the company boosted as a \r\n" + //

                " private limited in the year 2013 and has emerged as the “Best Upcoming Inbound Tour Operators in India”.\r\n" + //

                " It has been awarded in the category of “Excellence in the Tourism Industry” by World Tourism Brand\r\n" + //

                " Academy.\r\n" + //

                " The SANGAMNER based company, with a strong presence in inbound travel trade and corporate segment,\r\n" + //

                " today has excelled its branches over Sangamner,Pune,Mumbai,Delhi,Mumbai, Agra, Jaipur, Haridwar. \r\n" + //

                " The company with its professionally managed travel engine specializes mainly in organizing Adventure,\r\n" + //

                " Cultural, Religious, hill station & wildlife tours in India through a sprawling network.\r\n" + //

                " It offers 24 X 7 hours services that include travel planning, itinerary design, hotel bookings,\r\n" + //

                " ticket reservations and transport facilities. It also provides holiday packages,\r\n" + //

                " customized as per client’s need and budget.\r\n" + //

                "\r\n" + //

                " Our Motto:\r\n" + //

                " Customer satisfaction: It’s the prime motto of our business, which has helped us to build a good network\r\n" + //

                " with travellers from the farthest corners of the world. The company today holds more than 100,000 \r\n" + //

                " satisfied travellers and is still framing the travel diaries of fresh clients.\r\n" + //

                "\r\n" + //

                " Now you can have a comfortable and hassle free holiday in India where in you leave all the worries\r\n" + //

                " to us. Right from the arrival at the airport to personalized assistance of departure, we take care \r\n" + //

                " of all the needs of the travellers. Our guests just sit back and enjoy their holidays with all the \r\n" + //

                " value for the money they have spent.");

                 ta1.setFont(f2);

         p1.add(ta1);

         w.add(p1);

        b1=new JButton("Back");

        b1.setBounds(30, 50, 100, 40);

        b1.setFont(f2);

        p1.add(b1);

        b1.addActionListener(this);

        w.revalidate();

        w.setVisible(true);

    }

    public void actionPerformed(ActionEvent e)

    {

        if(e.getSource()==b1)

        {

          JOptionPane.showConfirmDialog(w, "Are you sure to go back", "Confirmation",JOptionPane.CLOSED\_OPTION, JOptionPane.INFORMATION\_MESSAGE);

          w.dispose();

          new home1();

        }

    }

}

public class About {

    public static void main(String[] args) {

        try{

             String str="javax.swing.plaf.nimbus.NimbusLookAndFeel";

        UIManager.setLookAndFeel(str);

        }catch(Exception e){}

        new About1();

    }

}

**Login\_home :**

package Travel\_Mgmt;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.sql.\*;

class Admin\_home1 extends JFrame implements ActionListener

{

    Container c;

    JPanel p1,p2;

    JLabel l1,l2,l3;

    JButton b1,b2,b3,b4,b5;

    public Admin\_home1()

    {

        c=getContentPane();

        c.setLayout(null);

        setDefaultCloseOperation(DISPOSE\_ON\_CLOSE);

        setBounds(150, 20,1200, 750);

        p1=new JPanel();

        p1.setLayout(null);

        p1.setBounds(2, 2, 200,746);

        p1.setBackground(Color.black);

        add(p1);

        p2=new JPanel();

        p2.setLayout(null);

        p2.setBounds(202, 2, 996,746);

        p2.setBackground(Color.cyan);

        add(p2);

        l1=new JLabel("Hello Dear Admin ! Select Which Report You Want");

        l1.setBounds(100, 10, 900, 80);

        Font f1=new Font("arial",Font.BOLD,30);

        l1.setFont(f1);

        l1.setForeground(Color.black);

        p2.add(l1);

        Font f2=new Font("Arial", Font.BOLD, 20);

        b1=new JButton("L O G I N  D E T A I L S");

        b1.setBounds(100, 150, 300, 50);

        b1.setFont(f2);

        p2.add(b1);

        b2=new JButton("B O O K I N S   D E T A I L S");

        b2.setBounds(100, 250, 300, 50);

        b2.setFont(f2);

        p2.add(b2);

        b5=new JButton("L O G O U T");

        b5.setBounds(150, 400, 200, 40);

        b5.setFont(f2);

        p2.add(b5);

        b1.addActionListener(this);

        b2.addActionListener(this);

        b5.addActionListener(this);

       setVisible(true);

    }

    public void actionPerformed(ActionEvent e)

    {

      if(e.getSource()==b1)

      {

        new Login\_db1();

      }

      if(e.getSource()==b2)

      {

        new Booking\_db1();

      }

      if(e.getSource()==b5)

      {

        new Login1();

      }

    }

}

public class Admin\_home {

    public static void main(String[] args) {

        try{

            String str="javax.swing.plaf.nimbus.NimbusLookAndFeel";

            UIManager.setLookAndFeel(str);

        }catch(Exception e1){}

        new Admin\_home1();

    }

}

**Booking :**

package Travel\_Mgmt;

import net.sourceforge.jdatepicker.\*;

import net.sourceforge.jdatepicker.JDateComponentFactory;

import javax.swing.\*;

import Travel\_Mgmt.home1;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

class Booking1 extends JFrame implements ActionListener

{

    Container c;

    JLabel l1,l2,l3,l4,l5,l6,l7,l8,l9,l10,one,two,three,four,five,six;

    JTextField jt1,jt2,jt3,jt4;

    JComboBox cb1,cb2,cb3;

    JDatePicker jpc1;

    JPanel p1,p2;

    JButton b1,b2,b3,b4;

    Font f2=new Font("Arial",Font.ROMAN\_BASELINE, 20);

    Font f1=new Font("Arial",Font.ITALIC, 30);

    JTextArea ta1;

 public Booking1()

 {

      c=getContentPane();

      c.setLayout(null);

      setBounds(100, 10,1250, 800); //original

     // setBounds(150, 20,1200, 750);

     c.setBackground(Color.CYAN);

      setDefaultCloseOperation(Login1.DISPOSE\_ON\_CLOSE);

      l1=new JLabel("|| WELCOME TO BUS BOOKING SYSTEM ||");

      l1.setBounds(300, 20, 1000, 50);

      l1.setFont(f1);

      c.add(l1);

      l2=new JLabel("Name : ");

      l2.setBounds(100, 90, 100, 50);

      l2.setFont(f2);

      c.add(l2);

      jt1=new JTextField();

      jt1.setBounds(200, 90, 250, 40);

      jt1.setFont(f2);

      c.add(jt1);

      l3=new JLabel("Source : ");

      l3.setBounds(100, 150, 100, 50);

      l3.setFont(f2);

      c.add(l3);

      Object values[]={"Sangamner","Pune","Mumbai","Nashik"};

      cb1=new JComboBox<>(values);

      cb1.setBounds(200, 150, 150, 40);

      cb1.setFont(f2);

      c.add(cb1);

      l4=new JLabel("Destination : ");

      l4.setBounds(450, 150, 150, 50);

      l4.setFont(f2);

      c.add(l4);

      String values2[]={"Sangamner","Pune","Mumbai","Nashik"};

      cb2=new JComboBox<>(values2);

      cb2.setBounds(580, 150, 150, 40);

      cb2.setFont(f2);

      c.add(cb2);

      l3=new JLabel("No. of Passengers : ");

      l3.setBounds(100, 230, 200, 50);

      l3.setFont(f2);

      c.add(l3);

      jt2=new JTextField();

      jt2.setBounds(300, 230, 100, 40);

      jt2.setFont(f2);

      c.add(jt2);

      l5=new JLabel("Time : ");

      l5.setBounds(480, 230, 100, 50);

      l5.setFont(f2);

      c.add(l5);

      String values3[]={"8:00 pm","9:00 pm","10:00 pm","11:00 pm"};

      cb3=new JComboBox<>(values3);

      cb3.setBounds(580, 230, 150, 40);

      cb3.setFont(f2);

      c.add(cb3);

      l6=new JLabel("Select Date : ");

      l6.setBounds(300, 290, 150, 50);

      l6.setFont(f2);

      c.add(l6);

      p1=new JPanel();

      p1.setBounds(100, 350, 500, 150); //215

      p1.setBackground(Color.white);

      c.add(p1);

      jpc1=JDateComponentFactory.createJDatePicker();

      p1.add((JComponent) jpc1);

      getContentPane().add(p1,BorderLayout.CENTER);

      l10=new JLabel("Re-Enter Date : ");

      l10.setBounds(100, 520, 200, 50);

      l10.setFont(f2);

      c.add(l10);

      jt4=new JTextField();

      jt4.setBounds(260, 520, 200, 40);

      jt4.setFont(f2);

      c.add(jt4);

      b1=new JButton("SUBMIT");

      b1.setBounds(260,590, 150, 50);

      b1.setFont(f2);

      c.add(b1);

      l7=new JLabel("Payble Amount : ");

      l7.setBounds(100, 660, 200, 50);

      l7.setFont(f2);

      c.add(l7);

      jt3=new JTextField();

      jt3.setBounds(260, 660, 200, 40);

      jt3.setFont(f2);

      c.add(jt3);

      ta1=new JTextArea();

      ta1.setBounds(770, 150, 400, 450);

      c.add(ta1);

      b2=new JButton("RESET");

      b2.setBounds(800,630, 150, 50);

      b2.setFont(f2);

      c.add(b2);

      b3=new JButton("BOOK");

      b3.setBounds(980,630, 150, 50);

      b3.setFont(f2);

      c.add(b3);

      l8=new JLabel("T I C K E T  S U M M A R Y");

      l8.setBounds(55, 10, 250, 50);

      l8.setFont(f2);

      ta1.add(l8);

      l9=new JLabel("-------------------------------------------------------------------");

      l9.setBounds(5, 30, 390, 50);

      l9.setFont(f2);

      ta1.add(l9);

      b4=new JButton("Back");

      b4.setBounds(30,30, 100, 40);

      b4.setFont(f2);

      c.add(b4);

      b1.addActionListener(this);

      b2.addActionListener(this);

      b3.addActionListener(this);

      b4.addActionListener(this);

      setVisible(true);

 }

 public void actionPerformed(ActionEvent e)

 {

    String name1=jt1.getText().toString();

    Object src=cb1.getSelectedItem();

    Object dest=cb2.getSelectedItem();

  //  int passenger=Integer.parseInt(jt2.getText());

   String passenger=jt2.getText().toString();

    Object time1=cb3.getSelectedItem();

    String date2=jt4.getText().toString();

   // int amt=400;

   // String amtt=Integer.toString(amt);

  //  String price=jt2.getText();

  //  int price1=Integer.parseInt(price)\*400;

   // Object date1=jpc1.getI18nStrings();

    if(e.getSource()==b1)

    {

        String price=jt2.getText();

       // String price1=price\*amt;

      int price1=Integer.parseInt(price)\*400;

        if(jt1.getText().isEmpty() && jt2.getText().isEmpty() && jt4.getText().isEmpty())

        {

            JOptionPane.showConfirmDialog(c, "Please Fill The Information", "Error",JOptionPane.OK\_CANCEL\_OPTION, JOptionPane.ERROR\_MESSAGE);

        }

        else

        {

           ta1.setText("\n\n\nNAME : "+name1+"\n\n"+"SOURCE : "+src+"\n\n"+"DEST : "+dest+"\n\n"+

           "No.Of PASSENGER : "+passenger+"\n\n"+"TIME : "+time1+"\n\n"+"DATE : "+date2);

           ta1.setFont(f2);

           jt3.setText(Integer.toString(price1));

        }

    }

    if(e.getSource()==b2)

    {

        if(jt1.getText().isEmpty() && jt2.getText().isEmpty() && jt3.getText().isEmpty() && jt4.getText().isEmpty())

        {

            JOptionPane.showConfirmDialog(c, "All Fields Are Already Empty !", "MESSAGE",JOptionPane.OK\_CANCEL\_OPTION, JOptionPane.ERROR\_MESSAGE);

        }

        else

        {

            jt1.setText(null);

            jt2.setText(null);

            jt3.setText(null);

            jt4.setText(null);

            ta1.setText(null);

        }

    }

    if(e.getSource()==b3)

    {

        String price=jt2.getText();

        int price1=Integer.parseInt(price)\*400;

        if(ta1.getText().isEmpty())

        {

            JOptionPane.showConfirmDialog(c, "Please Submit Your Booking Details First", "ERROR",JOptionPane.OK\_CANCEL\_OPTION, JOptionPane.ERROR\_MESSAGE);

        }

        else

        {

            try{

                Class.forName("com.mysql.cj.jdbc.Driver");

                String url="jdbc:mysql://localhost:3306/project";

                String user="root";

                String pass="Vaishnavi@25";

                Connection con=DriverManager.getConnection(url, user, pass);

                String query="insert into booking(name,source,destination,no\_of\_passengers,time,date,amount)values(?,?,?,?,?,?,?)";

                PreparedStatement pstmt=con.prepareStatement(query);

                pstmt.setString(1, name1);

                pstmt.setObject(2, src);

                pstmt.setObject(3, dest);

                pstmt.setString(4, passenger);

                pstmt.setObject(5, time1);

                pstmt.setString(6, date2);

                pstmt.setInt(7, price1);

                JOptionPane.showConfirmDialog(c, "Congratulations ...Your Ticket is Booked", "Confirmation",JOptionPane.OK\_CANCEL\_OPTION, JOptionPane.INFORMATION\_MESSAGE);

                pstmt.executeUpdate();

        }

        catch(Exception e1)

        {

        }

    }

    if(e.getSource()==b4)

    {

        JOptionPane.showConfirmDialog(c, "Are You Sure To Go Back", "Confirmation",JOptionPane.OK\_CANCEL\_OPTION, JOptionPane.INFORMATION\_MESSAGE);

        new home1();

    }

 }

}

}

public class Booking {

    public static void main(String[] args) {

        try{

            String str="javax.swing.plaf.nimbus.NimbusLookAndFeel";

            UIManager.setLookAndFeel(str);

        }catch(Exception e1){}

        new Booking1();

    }

}