**Chapter 1:**

**Introduction**

**1.1 Client /Organization Profile:**

Client Name: - Jay Ladava

Location & Address: - Kothrud, Sastry Nagar, near ram Nager, Pune.

**About Organization:**

Hotel are able to automate the process. It is useful for authorities which keep track of all the users registered. The authority can hotel packages room detail and availability of rooms, online booking and other packages. hotel provides all kind of rooms like single, double. During the past several decades, the records are supposed to be manually handled for all activities. The manual handling of records is time consuming and highly prone to error. To improve the performance of the hotel management system, the computerized system had to be undertaken.

**1.2 Need for System:**

* The changing technology waves have drastically turned every aspect of our lives. And, most travelers these days, use technology to plan their trips, one way or the other. Therefore, it is essential to implement the latest hospitality technologies; for hoteliers to meet their guests’ needs.
* Using a well-crafted, fine-tuned software for hotel business saves your both time and effort in carrying out of your managerial tasks and business services.
* It saves organization resources and expenses.
* This boosts your staff productivity as well as avoids operational mistakes.

**1.3 Scope and feasibility of Work:**

* Today hotel management is not only confined to hotels but has gone deep into tourism, catering, clubs, etc. making it a very paying and an exciting career option.
* With the rapid growth of the hotel industry pushed forward by foreign and domestic tourism and business travels, the demand for well trained and quality personnel too has gone up high. India is one of the preferred tourist and travel destinations.
* In future more features may be added category-wise. It may try to analyze the user behavior and preferences and accordingly suggest.

**1.4 Operating Environment: -**

* **Client side-hardware:**

RAM: 2 GB

CPU Speed: 2.4 GHZ

Hard Disk: 250 GB

* **Server-Side Hardware:**

RAM: 2GB

CPU Speed 2.6 GHz

Hard Disk: 150 GB

* **Client-side Software:**
* **Windows Front end:**

Java Core

AWT

Swing

* **Backend:**

MySQL Command Line Client 5.5

**1.5 Architecture of System:**

I have tried to show how the data/information in hotels is managed. This is just an overview of management in hotels. It manages and maintains the records of customers, rooms, employees and drivers in the hotel. The project is aimed to maintain the day-to-day state of admission/vacation of Residents, List of employees, room details etc. Main objective of this project is to provide a solution for hotels to manage most their work using computerized processes.

This software application will help admin to handle customer information, room allocation details, Payment details etc.

The rooms have different categories like single bed, double bed etc. so their charges and records will be maintained accordingly.

This software has been made in a user-friendly interface, so that anyone can add, delete, update the entries and handle all the transactions easily. As a security I have provided Admin username and Password.

The project, Hotel Management System is a desktop-based application that allows the hotel manager to handle all hotel activities online. Interactive GUI and the ability to manage various rooms, employees, drivers and customers make this system very flexible and convenient. This application gives managers the power and flexibility to manage the entire system from a single online system. Hotel management project provides room booking, staff management and other necessary hotel management features. The system allows the manager to post available rooms in the system.

**1.6 Detail Description of Technology Used:**

We have used java programming language in this project.

java is a general-purpose, class-based, object-oriented programming language designed for having lesser implementation dependencies. It is a computing platform for application development. Java is fast, secure, and reliable, therefore. It is widely used for developing Java applications in laptops, data centers, game consoles, scientific supercomputers, cell phones, etc.

**Swing and AWT**

It is a Java Graphical User Interface (GUI) toolkit. It is an Application Programming Interface (API) for providing a Graphical User Interface (GUI) for Java programs.

It is a part of the JFC (Java Foundation Classes), that is an API for providing a graphical user interface for Java programs.

It is used to create a GUI with Java.

**Features of Swing:**

* All features of AWT
* Provides a rich set of higher-level components like a tree, tabbed panes, list boxes, etc.
* Pluggable look and feel.
* No dependency on peer components.

AWT is used for supporting user interface in [Java Application](https://www.educba.com/applications-of-java/). It provides the high-level abstraction for Java program since it hides underlying GUI details. Since AWT is an API build on an Operating system to provide a [graphical user interface for Java](https://www.educba.com/user-interface-design-principles/). Its component has a dependency on the underlying counterpart (like the look and feel of that OS) to handle its functionality

**AWT Features Includes:**

* A rich set of user interface components.
* The robust event handling model
* Layout Manager for different window layout
* Data transfer classes support cut-paste through the native platform.

**Java Development kit 8.0(JDK):**

JDK is a software development environment used for making applets and Java applications. The full form of JDK is Java Development Kit. Java developers can use it on Windows, macOS, Solaris, and Linux. JDK helps them to code and run Java programs. It is possible to install more than one JDK version on the same computer

**Java Virtual Machine (JVM):**

Java Virtual Machine (JVM) is an engine that provides a runtime environment to drive the Java Code or applications. It converts Java bytecode into machine language. JVM is a part of the Java Run Environment (JRE).

**Chapter 2:**

**Proposed System**

**2.1 Purposed System:**

* Proposed system is the computerized version of the existing system which provides easy and quick access over the data.
* Keeping the records of admission of Resident properly so that facilities provided by hotels are fully utilized in effective and efficient manner.
* In proposed system customer and organization data will be secure with facility of username password also these system gives facility to assigns the roles and rights.
* System will be error free, secure, reliable and fast management system

**Advantages:**

* Less Time consuming
* Maintain accuracy

**2.2 Objective:**

* To provide facility to employee to manage bookings i.e., Check-in and Checkout tasks
* To provide facility to admin to access and manage all hotel bookings and manage users
* To Keeping records of admission of customers.
* To provide Facilities to hotels are fully utilized in an effective and efficient manner.

**2.3 User requirement:**

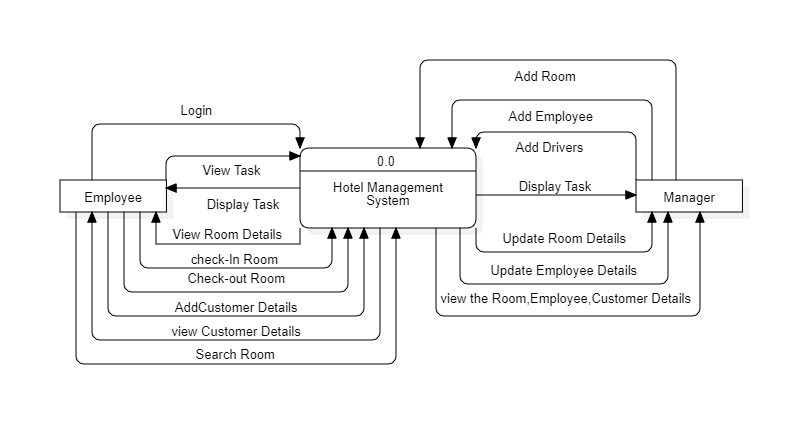
There should be software which allocates rooms automatically and maintains records of customers.

* record the customer’s first name and first name
* record the room number
* record reservations
* record the expected check-in and check-out
* record the customer’s phone number
* Admin can add Room, Employee, Driver
* Staff can add Check room status, check all employees’ details, Check all Customers’ details, Update room status, Update check-in check-out status etc.
* Also, admin can access all information
* To retrieve customer information the name or room number shall be used
* The system shall allow managers to assign user password

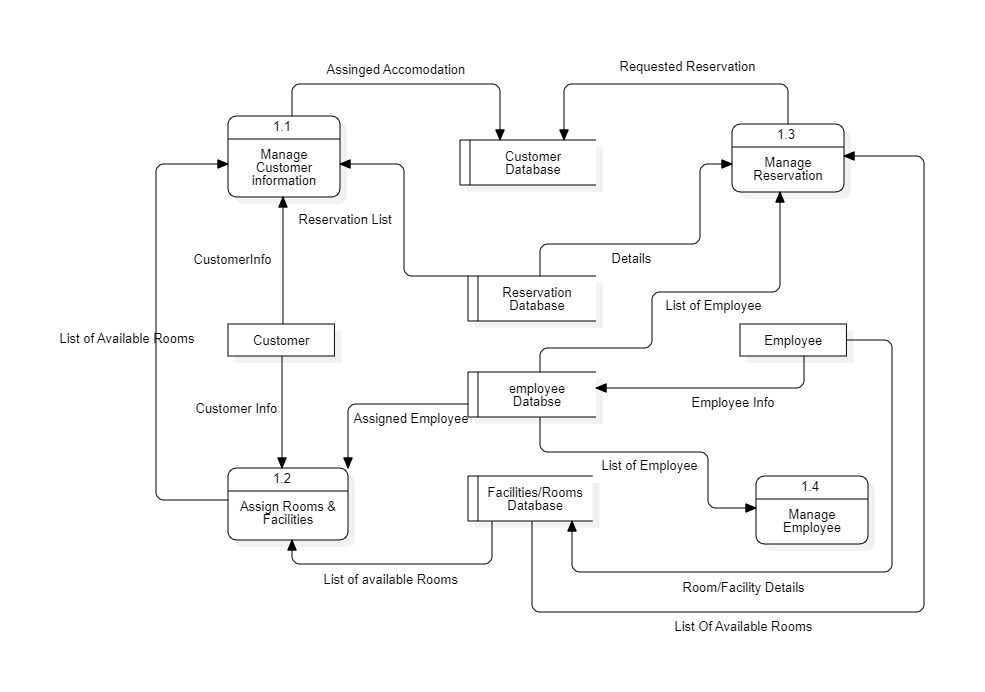
**Chapter 3:**

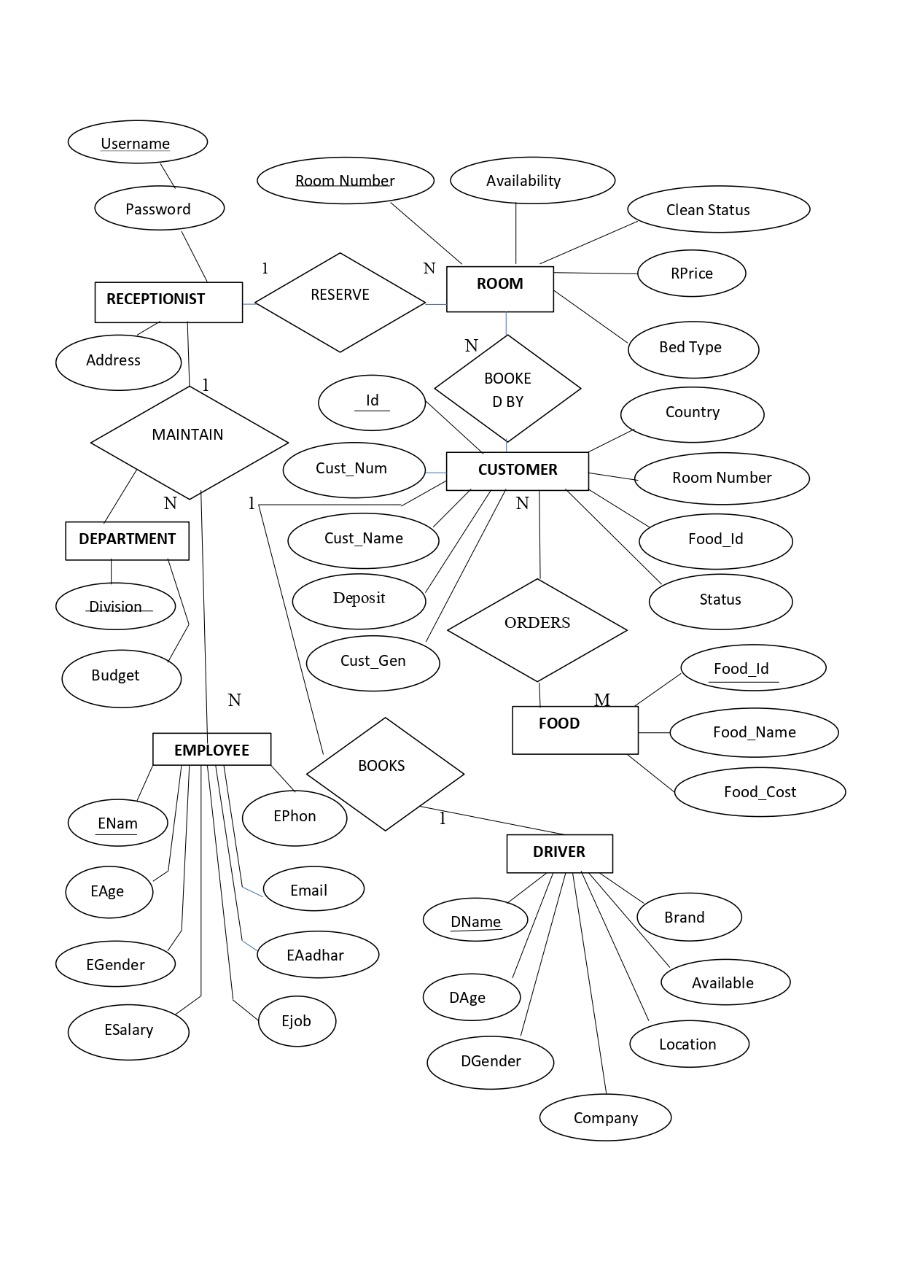
**Analysis & Design**

**3.1 Data Flow Diagram for Hotel management System**

**Context level Diagram:**

**First Level Data Flow Diagram**



**3.2 Entity Relationship Diagram for Hotel management System**

**3.3 SEQUENCE DIAGRAM For Hotel Management System:**

Customer

/Receptionist

/Staff

/Database

Visit Hotel

Ask For Help

Request Accommodation availability

Room Available

Check Room

Get Data

Avail Services

Provide Services

Cancel Booking

Update Database

Retrieve current availability

Ask for keys and Payment

Pay

Provide Bill

Request Information

Retrieve Information and generate bill

**3.4 COMPONENT DIAGRAM For Hotel Management System:**

staff

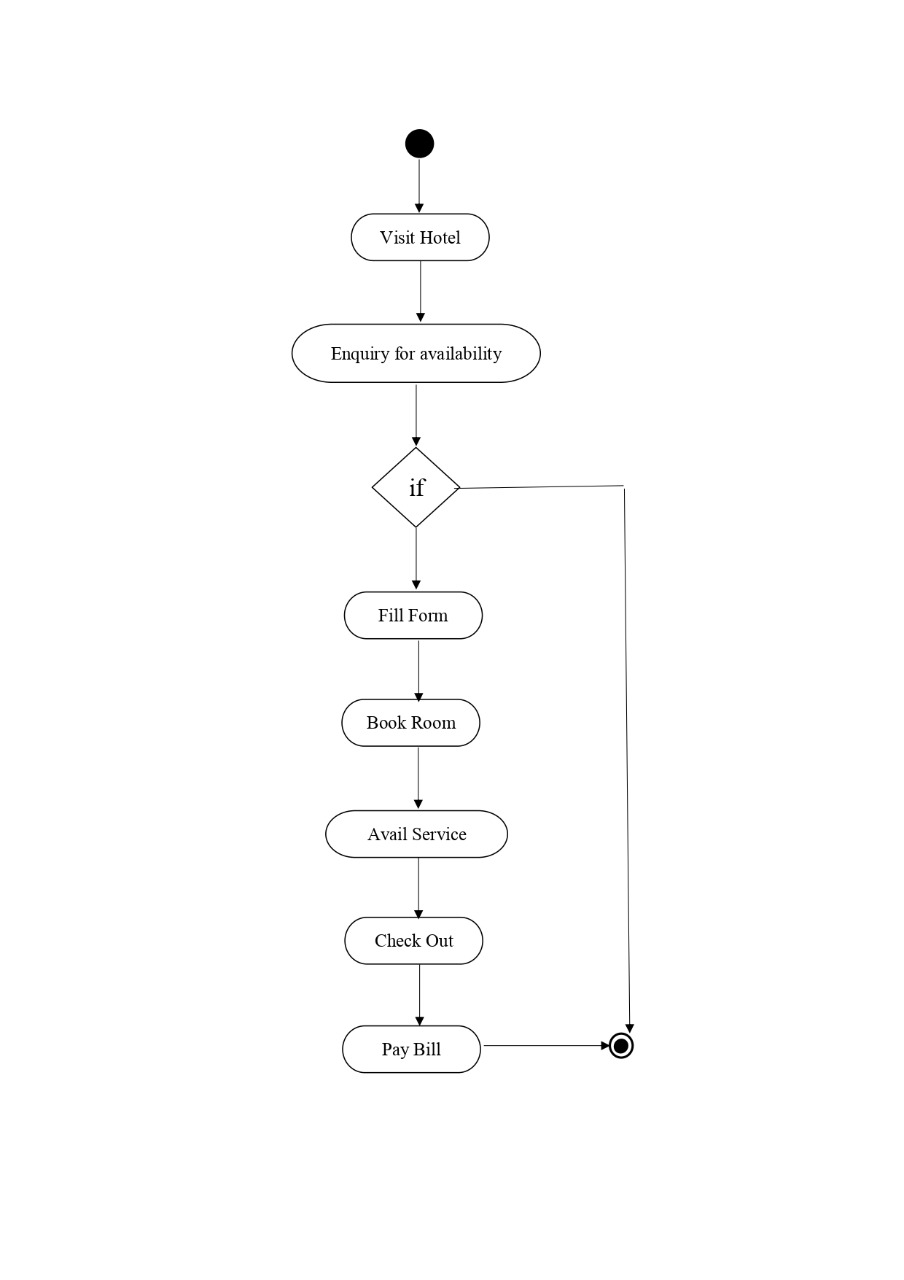
Room

Customr

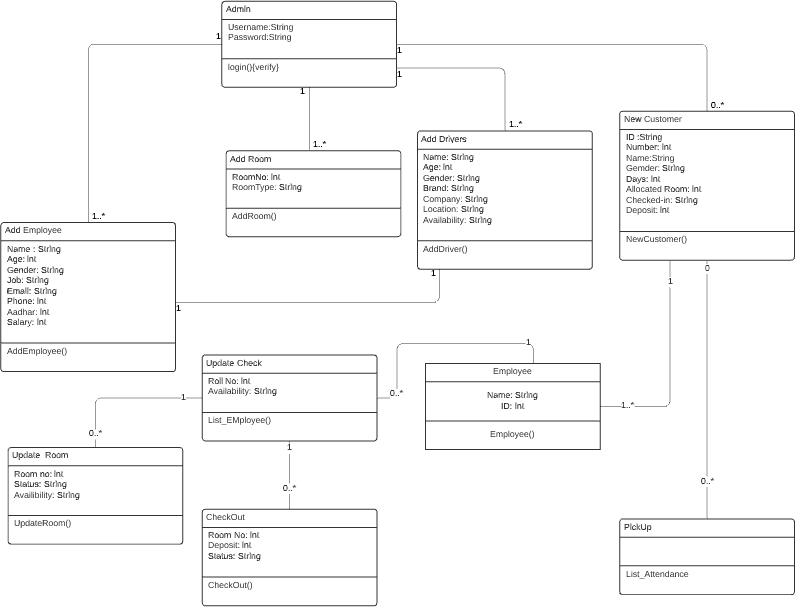
Hotel Management system Database

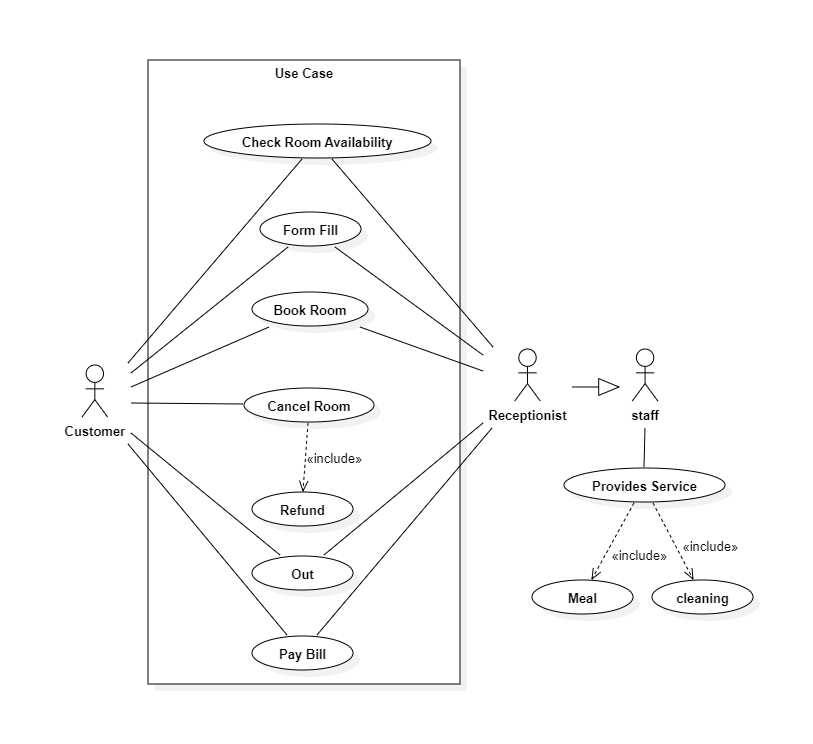
Contains Reception.java

And Servicestaff.java

**3.5 ACTIVITY DIAGRAM:**

**3.6 Class Diagram for Hotel management System:**



3**.7 Use Case Diagram for Hotel management System:**

 **3.8 Use Case Diagram for Hotel management System:**

**Chapter 4**

**Database Table:**

**1.Login**

This table Contains all details of login credential

Primary Key - Username

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Constraint |
| Username | Varchar (30) | Primary Key |
| Password | Varchar (30) | Not Null |

**2.Registration:**

This table Contains all details of login credential

Primary Key - First Name

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Constrain |
| First Name | Varchar (30) | Primary Key |
| Last Name | Varchar (30) | Not Null |
| Username | Varchar (30) | Not Null |
| Password | Varchar (30) | Not Null |

**3.Room:**

This table Contains all details of room available in hotel

Primary Key - Room\_number

|  |  |  |
| --- | --- | --- |
| Field | Datatype | Constraint |
| Room\_number | Numeric (30) | Primary Key |
| availability | Varchar (30) | Not Null |
| Clean status | Varchar (30) | Not Null |

**4.Customer:**

ssThis Table contains all details of Customers stay in hotel

Primary Key - number

|  |  |  |
| --- | --- | --- |
| Field | Datatype | Constraint |
| id | Varchar (30) | Not Null |
| number | Numeric (30) | Primary Key |
| name | Varchar (30) | Not Null |
| gender | Varchar (30) | Not Null |
| country | Varchar (30) | Not Null |
| Room number | Numeric (30) | Not Null |
| Status | Varchar (30) | Not Null |
| Deposit | Numeric (30) | Not Null |

**5.Employee:**

This Table contains all details of Employee in hotel

Primary Key – Aadhar

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Constraint |
| Name | Varchar (30) | Not Null |
| Age | Numeric (30) | Not Null |
| Gender | Varchar (30) | Not Null |
| Job | Varchar (30) | Not Null |
| Salary | Number (30) | Not Null |
| Phone | Number (30) | Not Null |
| Aadhar | Number (30) | Primary Key |
| Email | Varchar (30) | Not Null |

**6 Driver:**

This Table contains all details of Driver in hotel

Primary Key - Name

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Constraint |
| Name | Varchar (30) | Primary Key |
| Age | Numeric (30) | Not Null |
| Gender | Varchar (30) | Not Null |
| Company | Varchar (30) | Not Null |
| Brand | Varchar (30) | Not Null |
| Available | Varchar (30) | Not Null |
| Location | Varchar (30) | Not Null |

**7.Department:**

This Table contains all details of Driver in hotel

Primary Key - Department

|  |  |  |
| --- | --- | --- |
| Field | Data Type | Constraint |
| Department | Varchar (30) | Primary Key |
| Budget | Numeric (30) | Not Null |

**8 DATA DICTIONARY:**

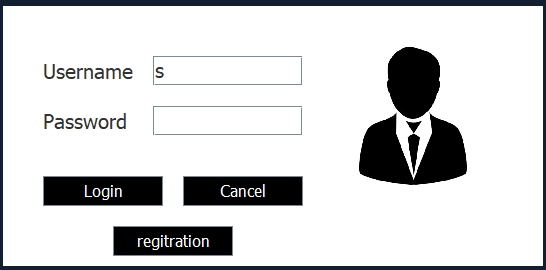
|  |  |  |
| --- | --- | --- |
| Field Name | Data Type | Description |
| Aadhar | Number (30) | Unique Identification of user |
| Age | Numeric (30) | Age of customer |
| Availability | Varchar (30) | Check room available or not |
| Brand | Varchar (30) | Vehicle brand name |
| Budget | Numeric (30) | Budget of hotel |
| Clean status | Varchar (30) | Check Room is clean or dirty |
| Company | Varchar (30) | Vehicle Company name |
| country | Varchar (30) | Country of the customer |
| Department | Varchar (30) | Store the Employee which department work |
| Deposit | Numeric (30) | Customer pay the advance |
| First Name | Varchar (30) | First name of customer |
| gender | Varchar (30) | Gender of customer |
| id | Varchar (30) | Unique Identification of customer |
| Job | Varchar (30) | Employee work in which department |
| Last Name | Varchar (30) | Last name of customer |
| Location | Varchar (30) | Location of driver |
| Name | Varchar (30) | Name of employee |
| number | Numeric (30) | Unique Identification of employee |
| Password | Varchar (30) | Password of employee |
| Phone | Number (30) | Phone number of customer |
| Room number | Numeric (30) | Unique Identification of room |
| Salary | Number (30) | Salary of employee |
| Status | Varchar (30) | Check Room Status available or not |
| Username | Varchar (30) | Username of employee |

**Chapter 5:**

**User Manual**

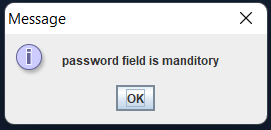
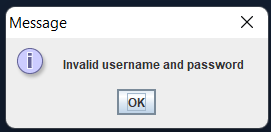
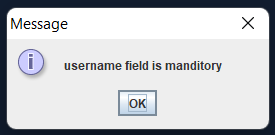
**4.1 User Interface Design (Screen etc.)**

**Start Screen**

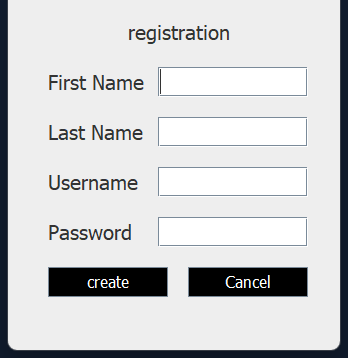
**login Screen**

This is Login Page it validates User Name and Password

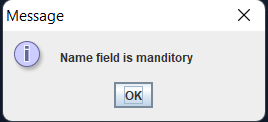
**Validation**

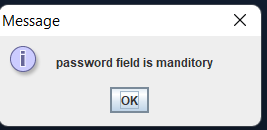


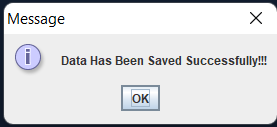
**Registration**

****

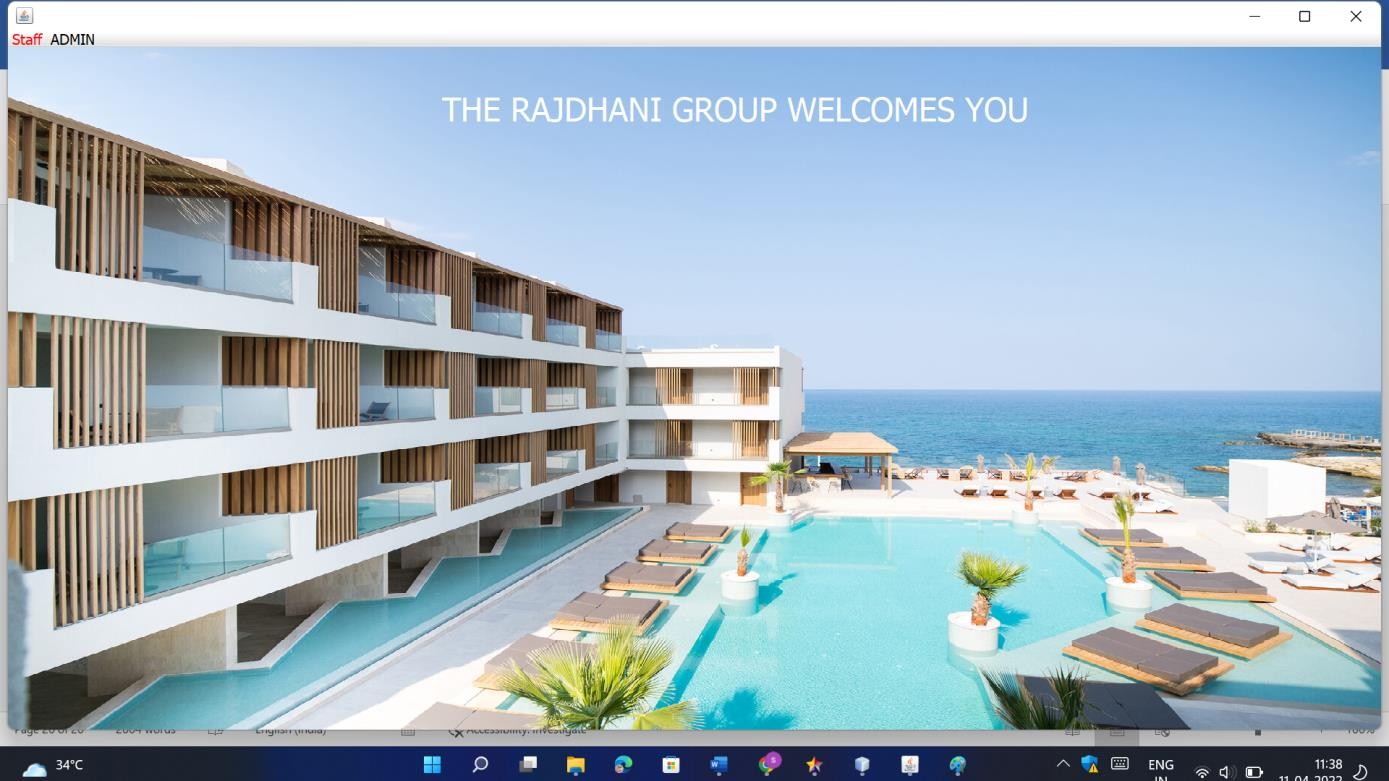
04



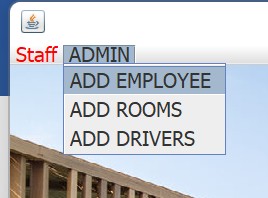


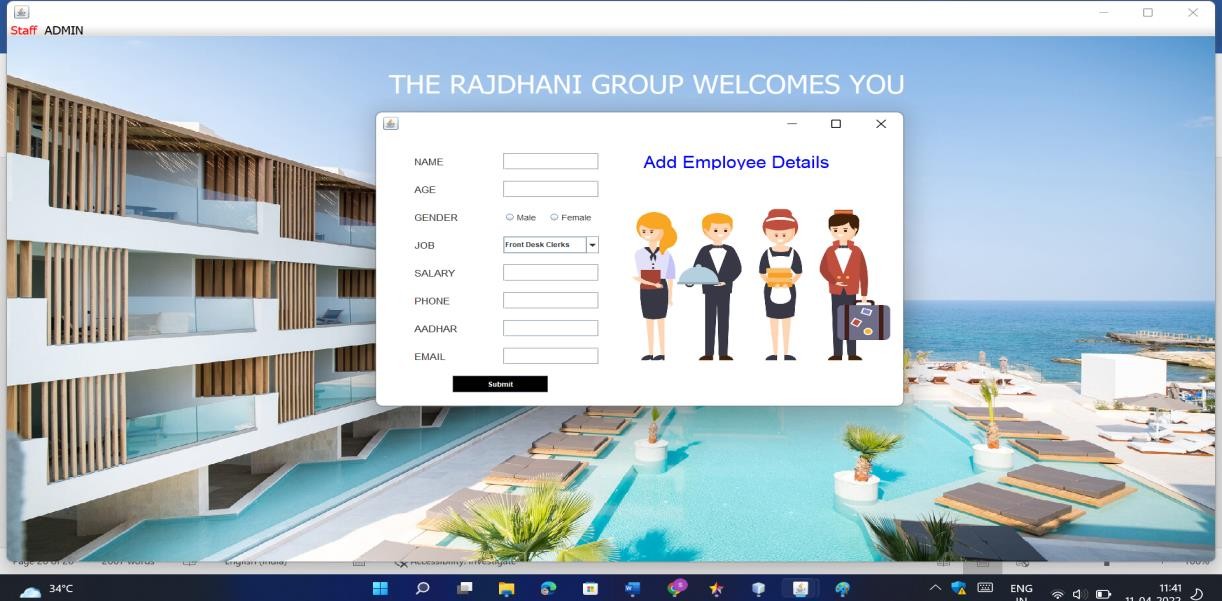


**Dash Board Page**



**Add Employee**

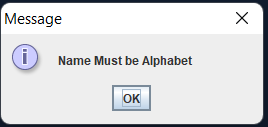
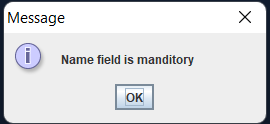




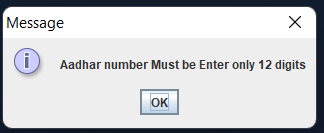
`

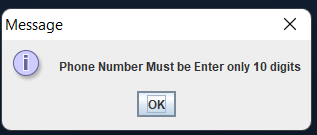
**Validation**

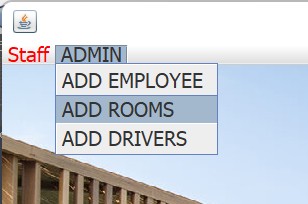


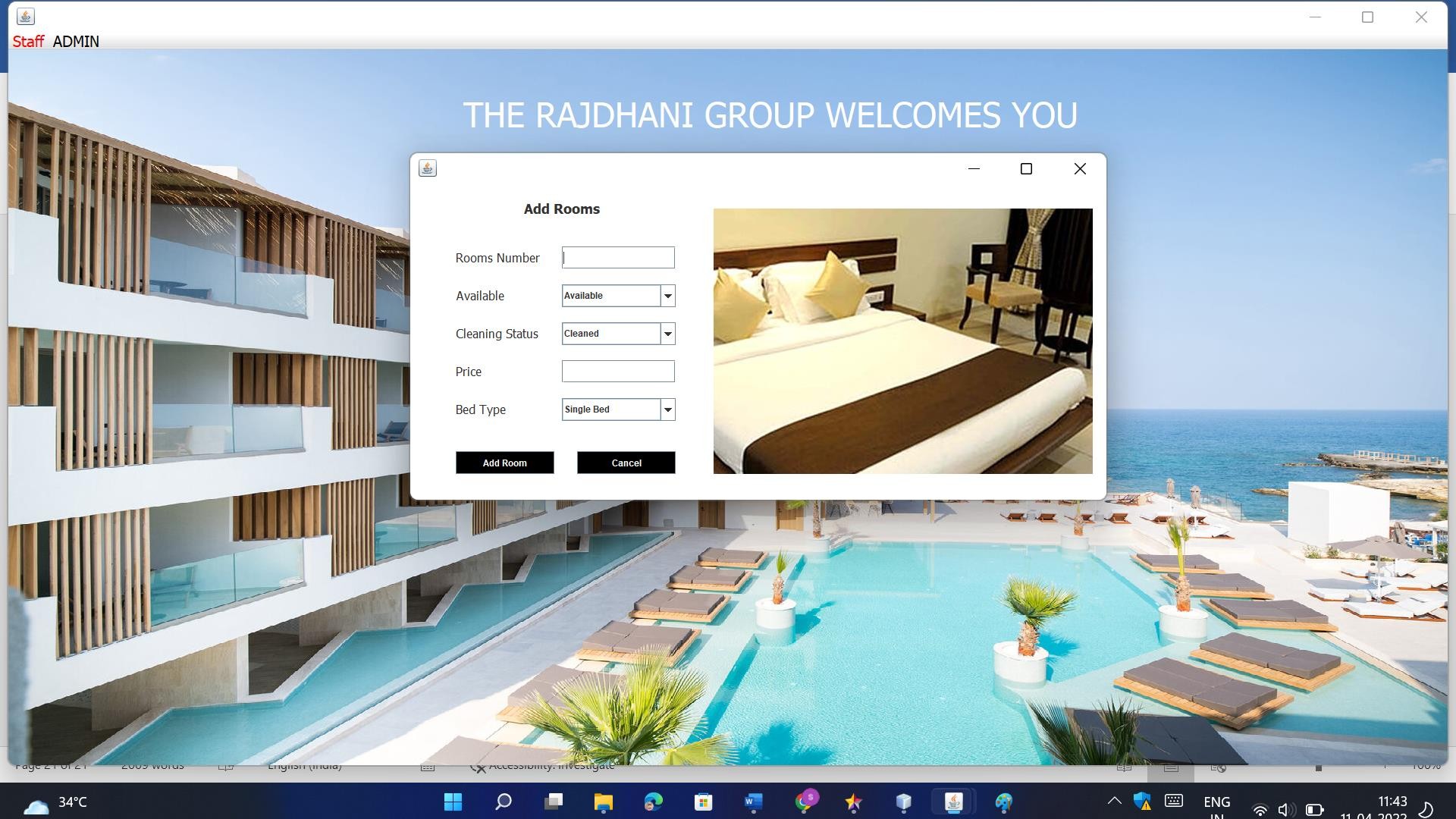






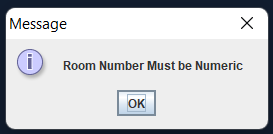
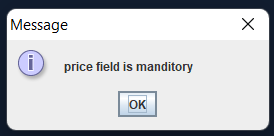
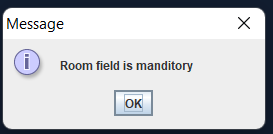


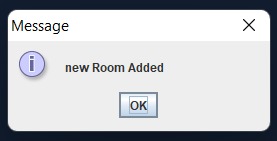
**Add Rooms**



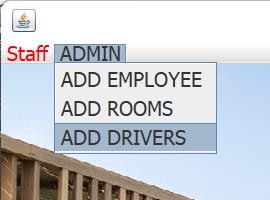
**Validation**

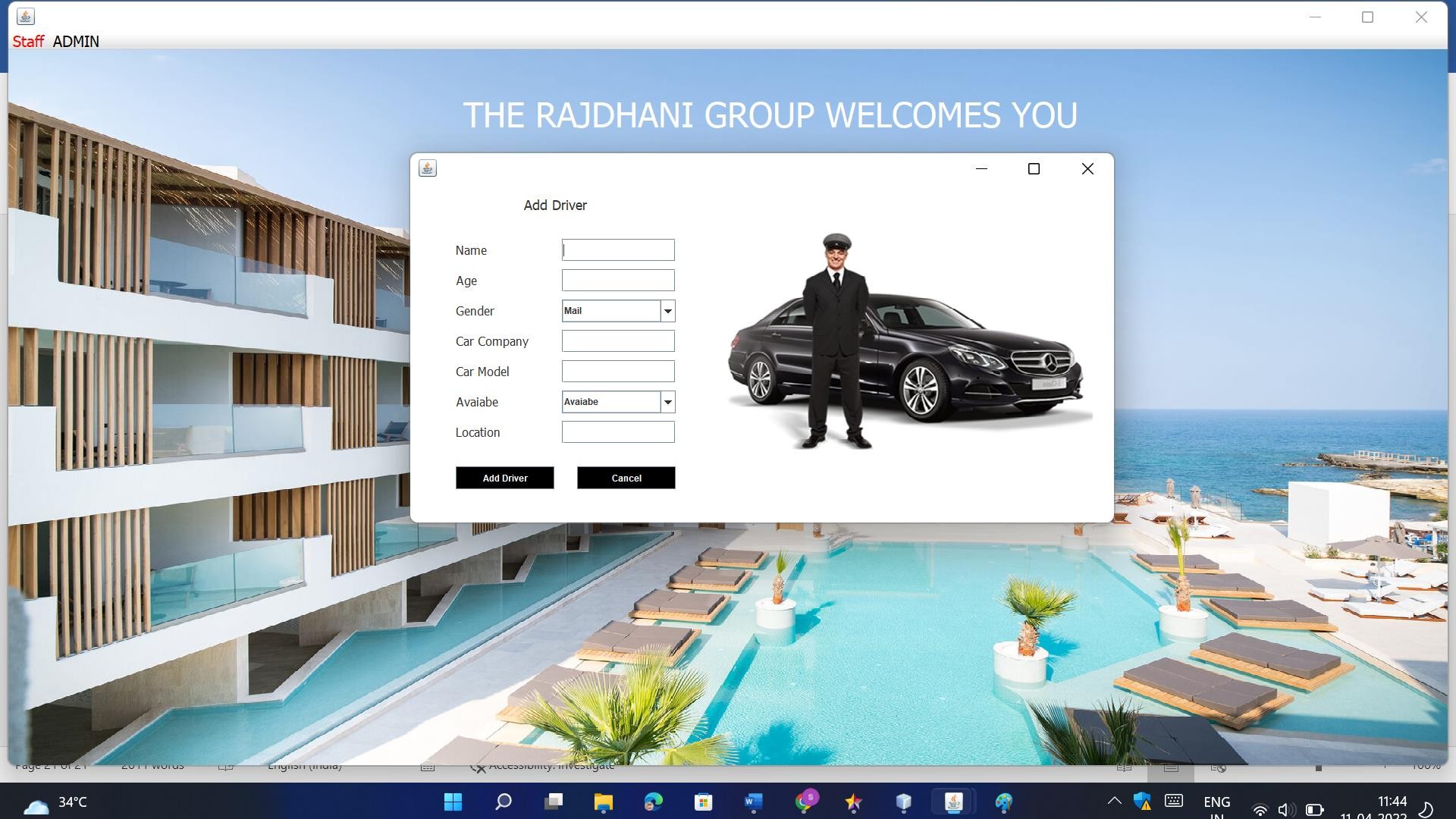






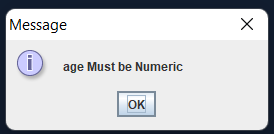
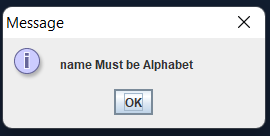
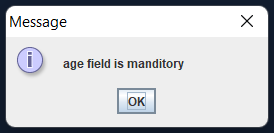
Room Added Successfully

**Add Driver**

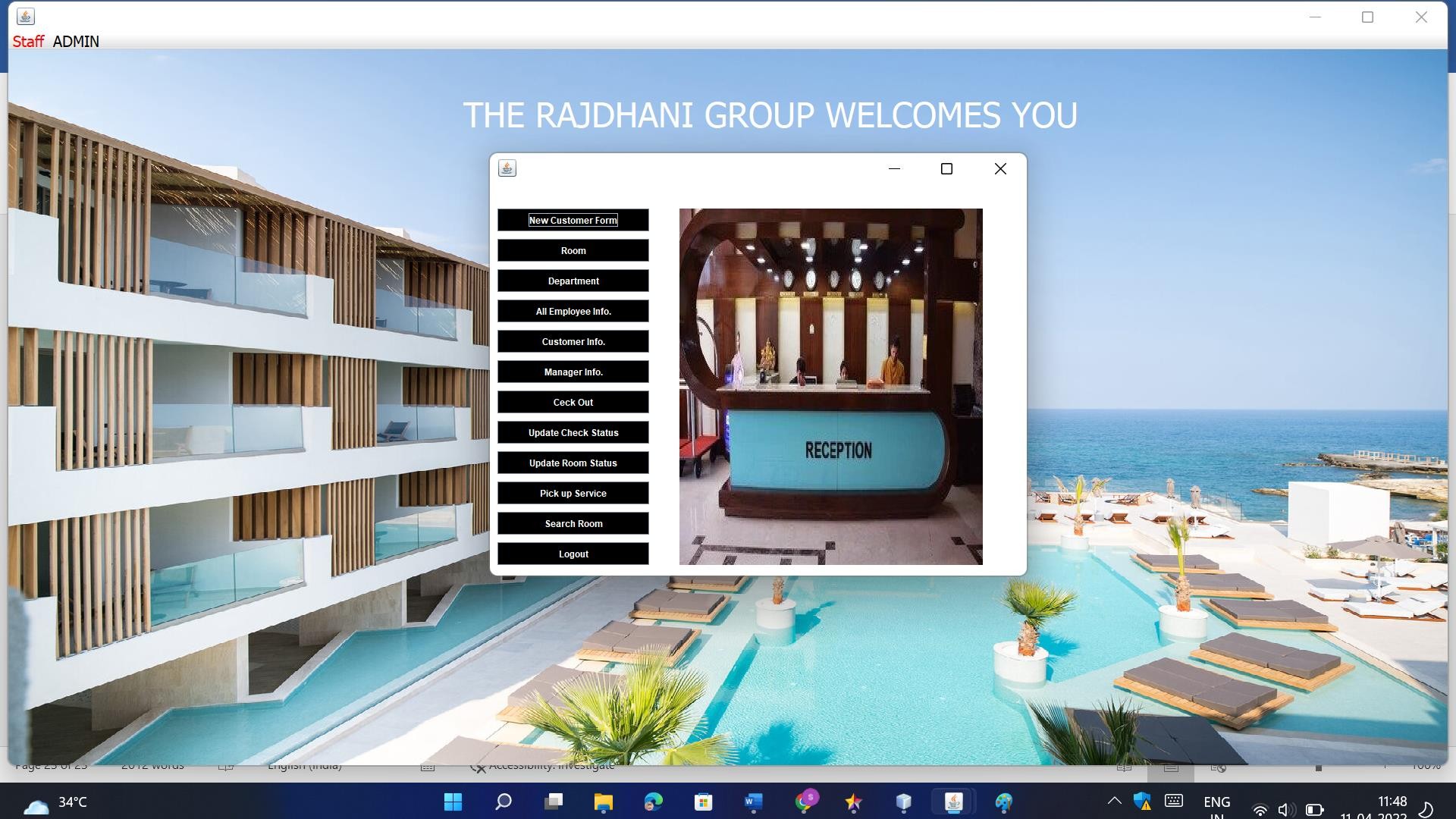


**Validation**



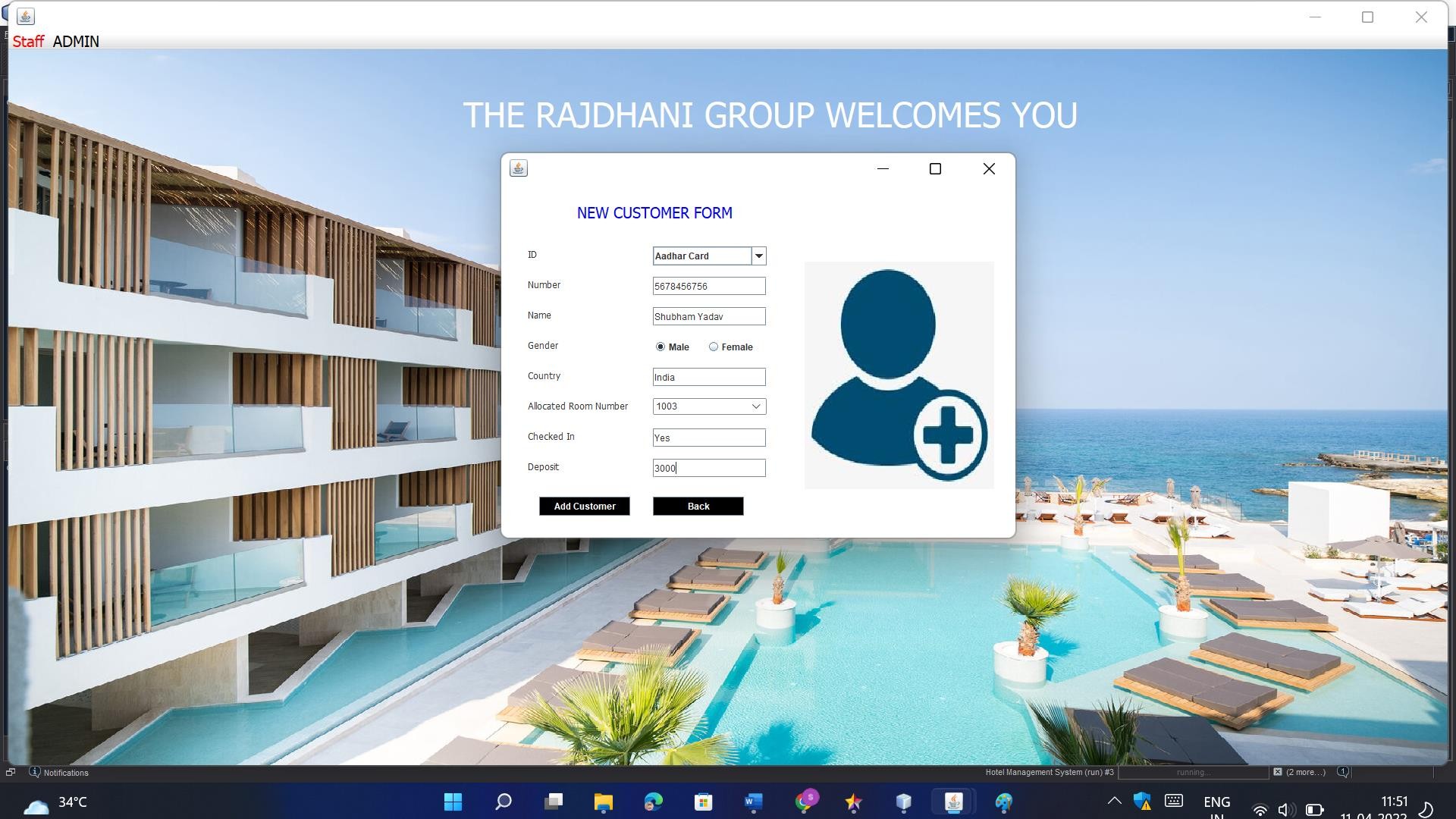


**Reception**

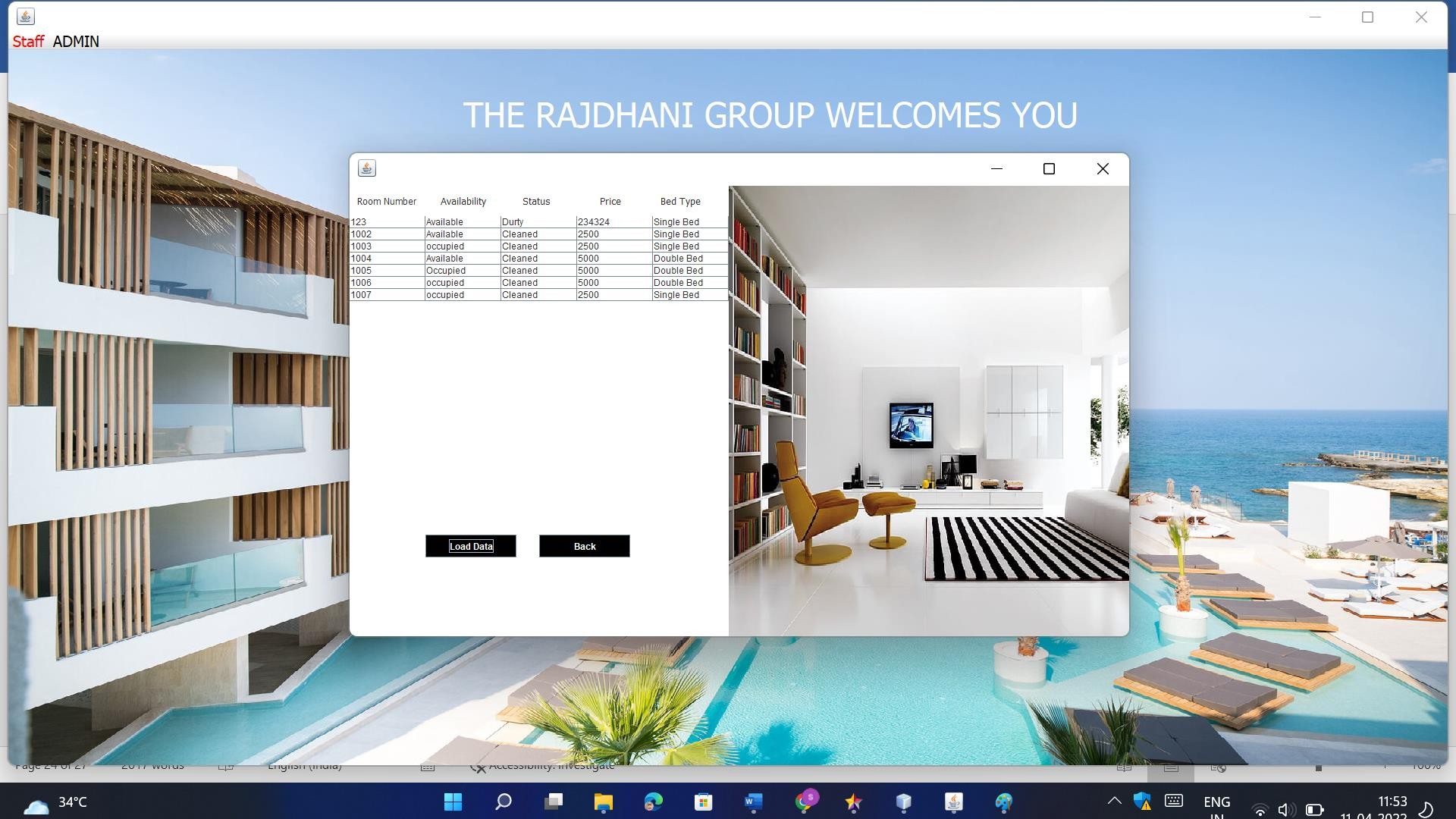


**Add New Customer**



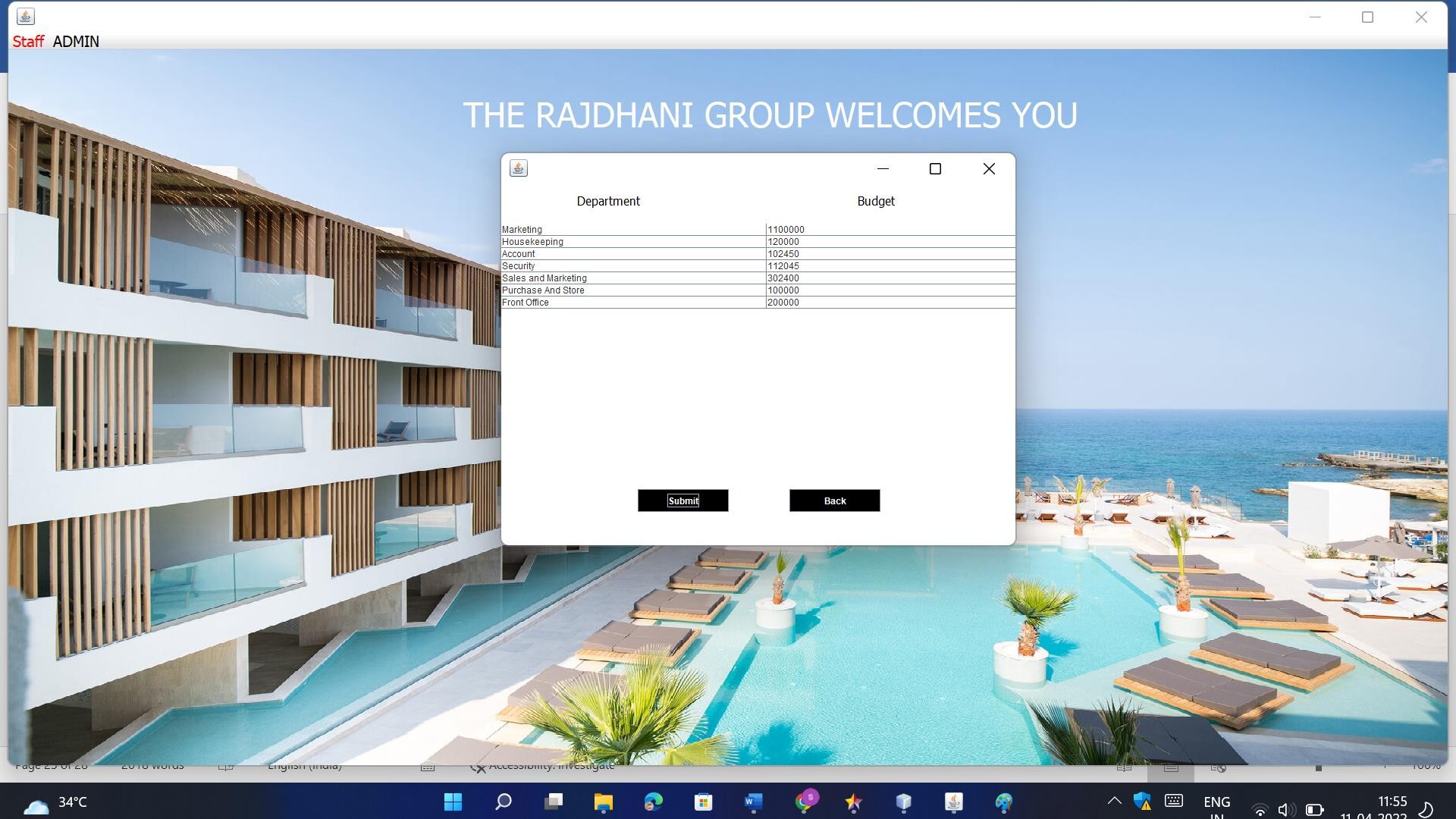


**Room Details**

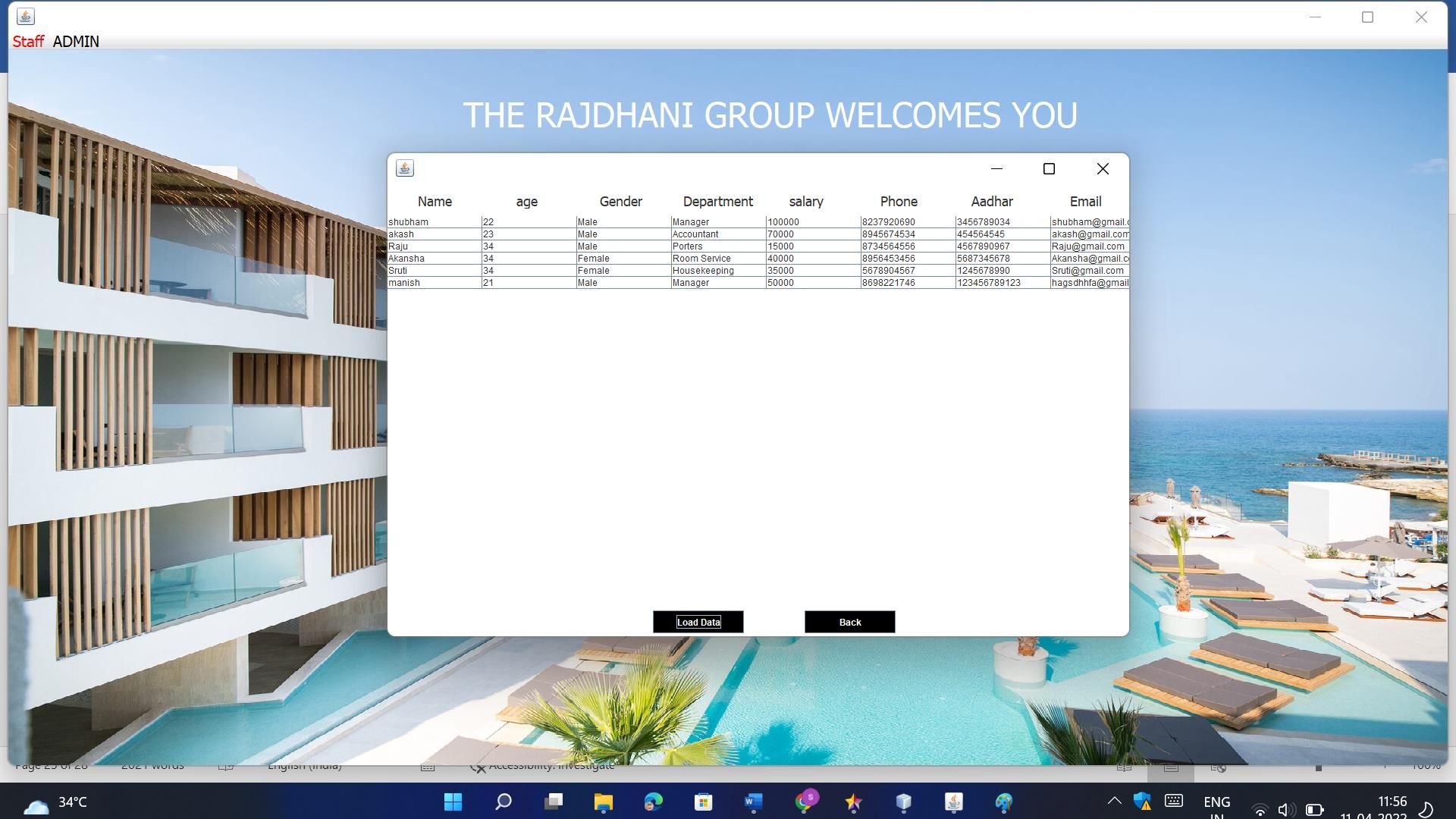


**Department**

****



**All Employee Info.**

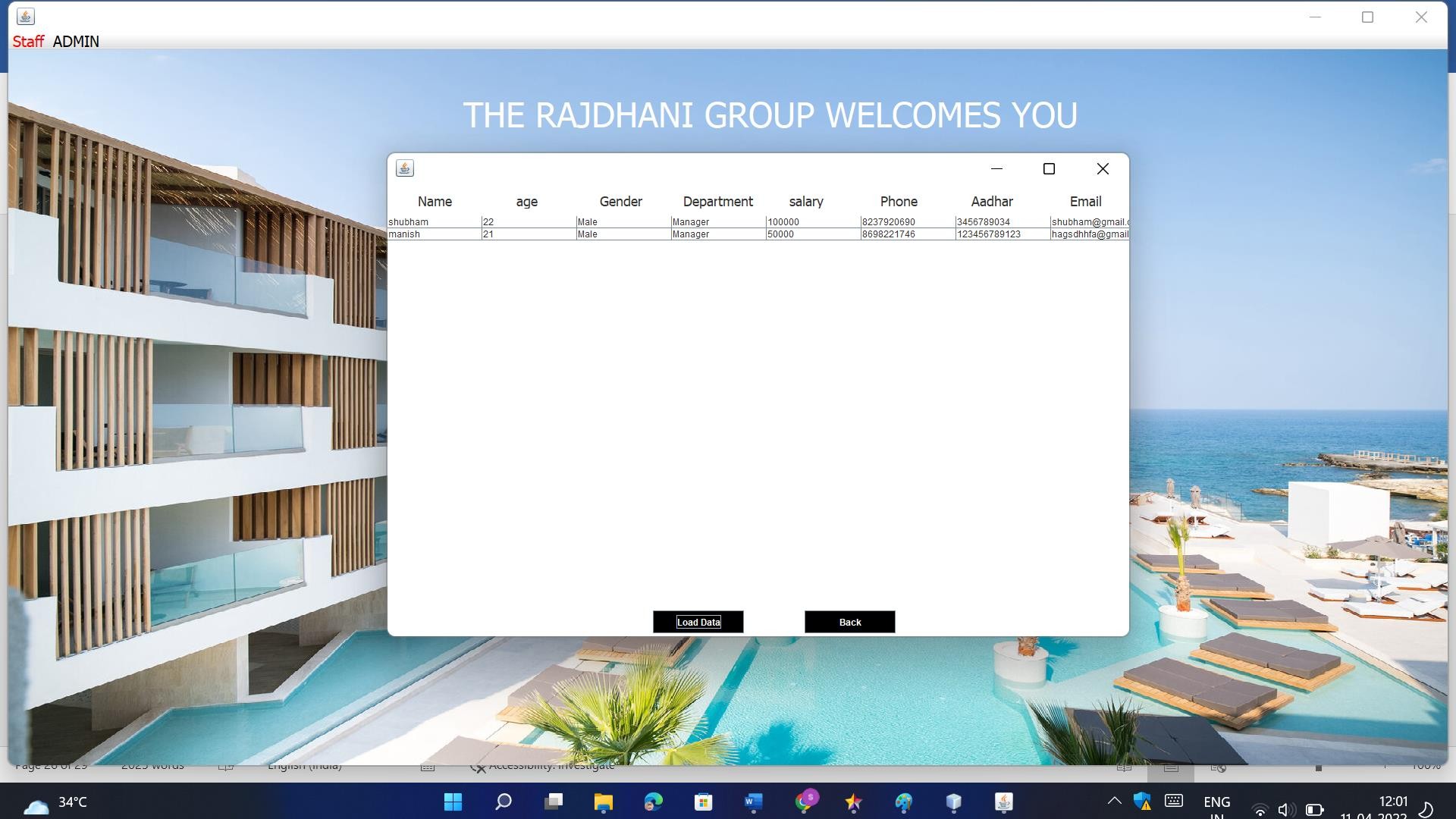


**Customer Info**



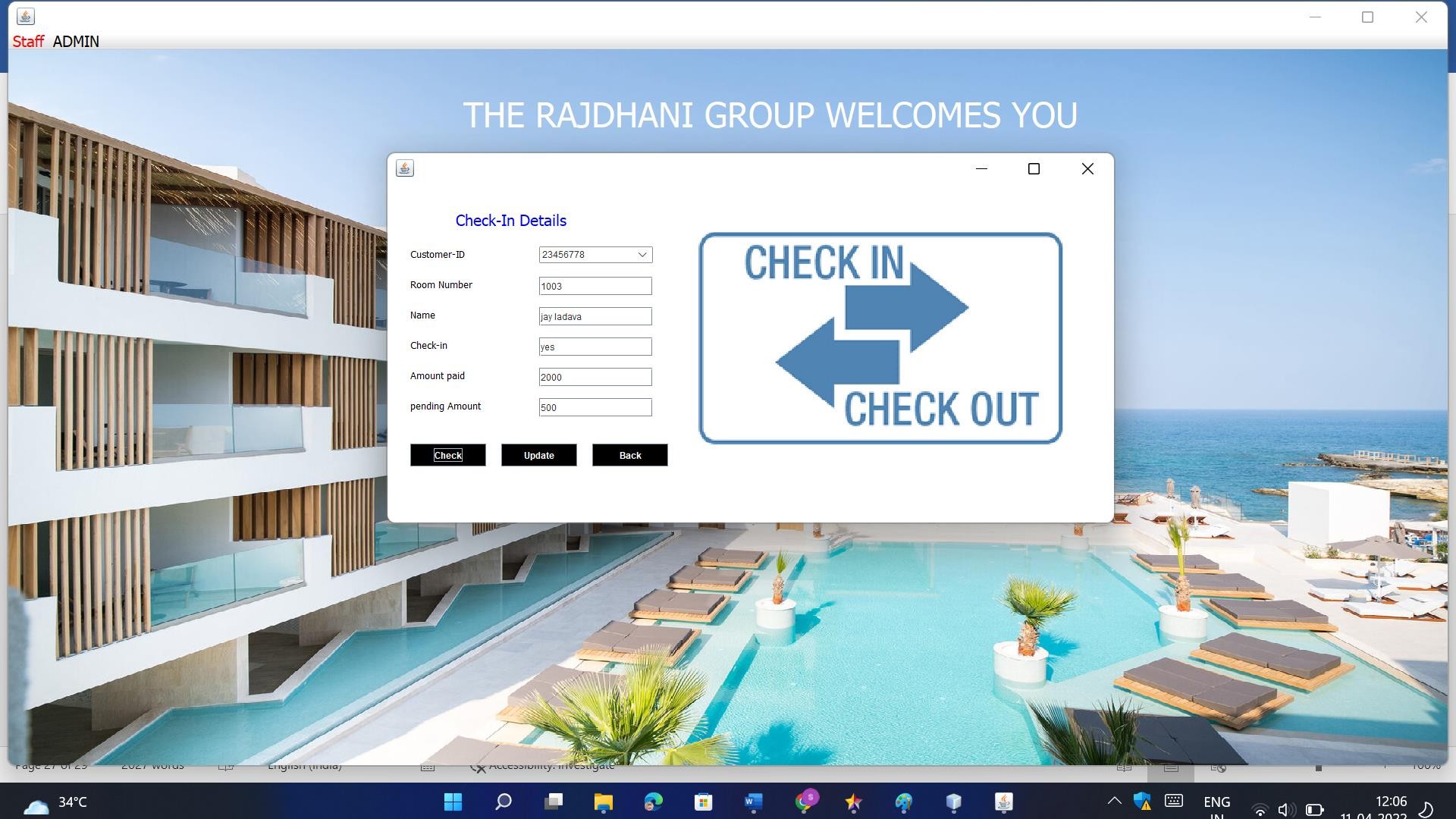


**Manager Info.**

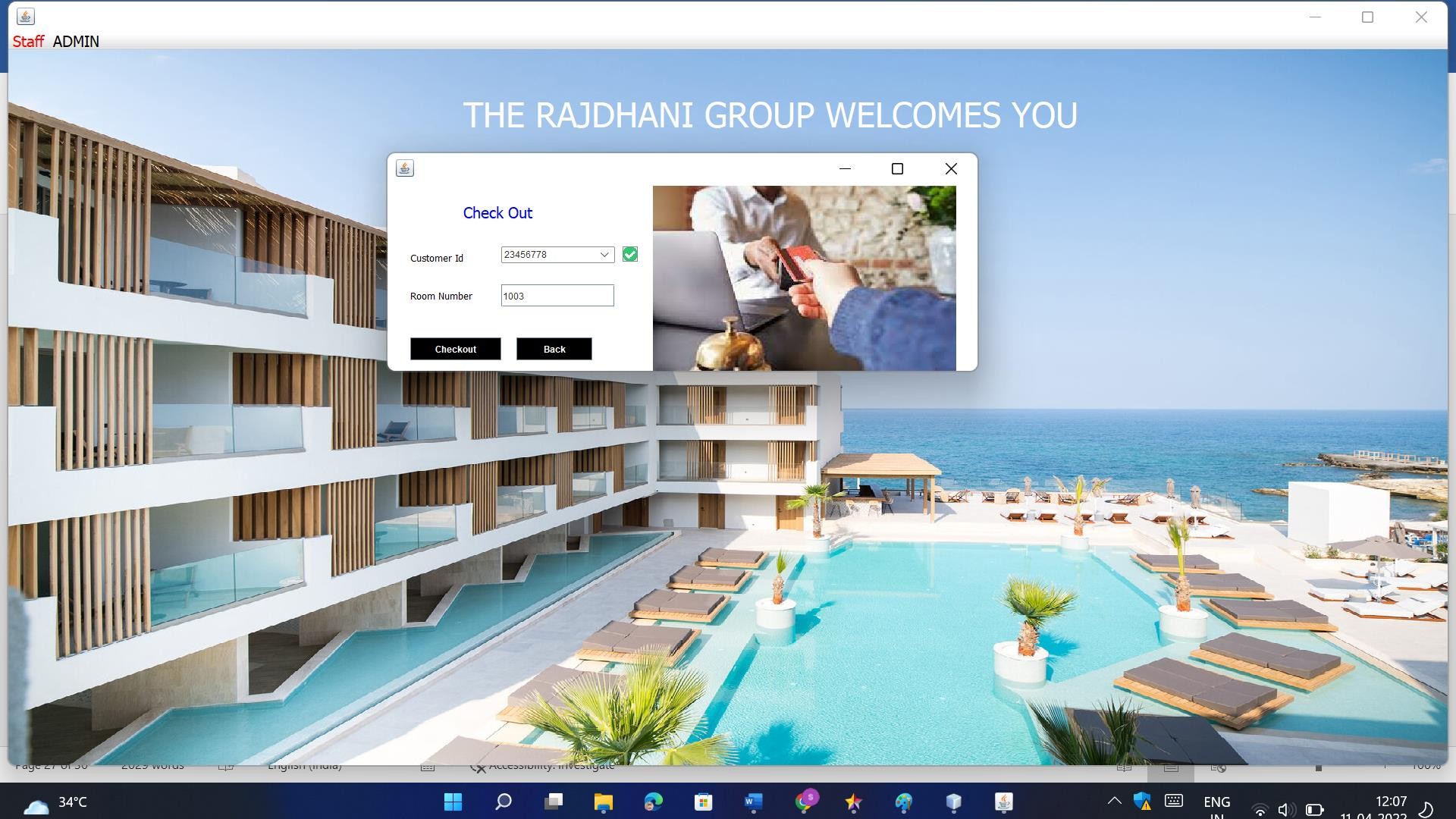


**Check-In Details**



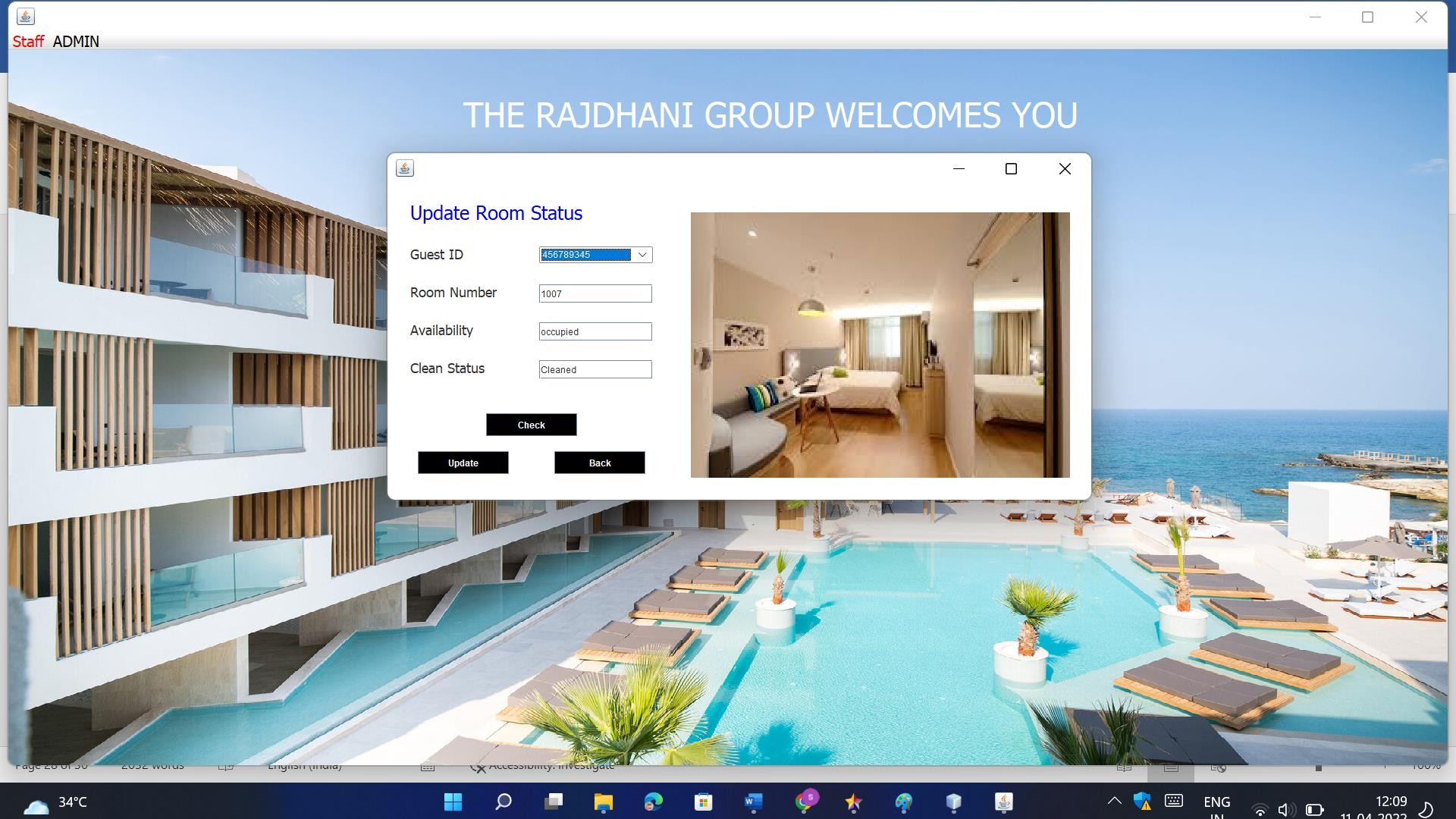


**Check-Out Details**

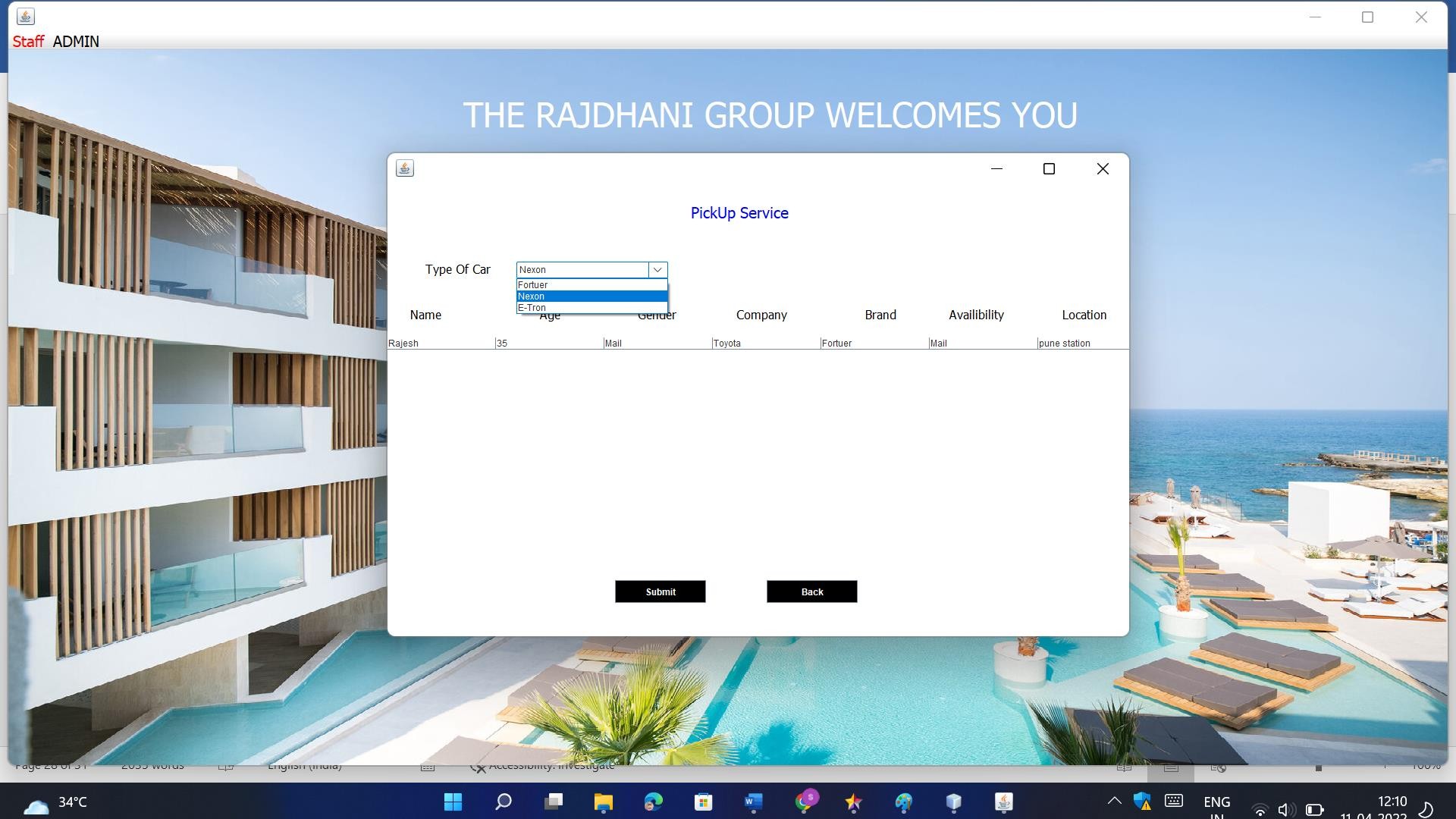


**Update Room Status**



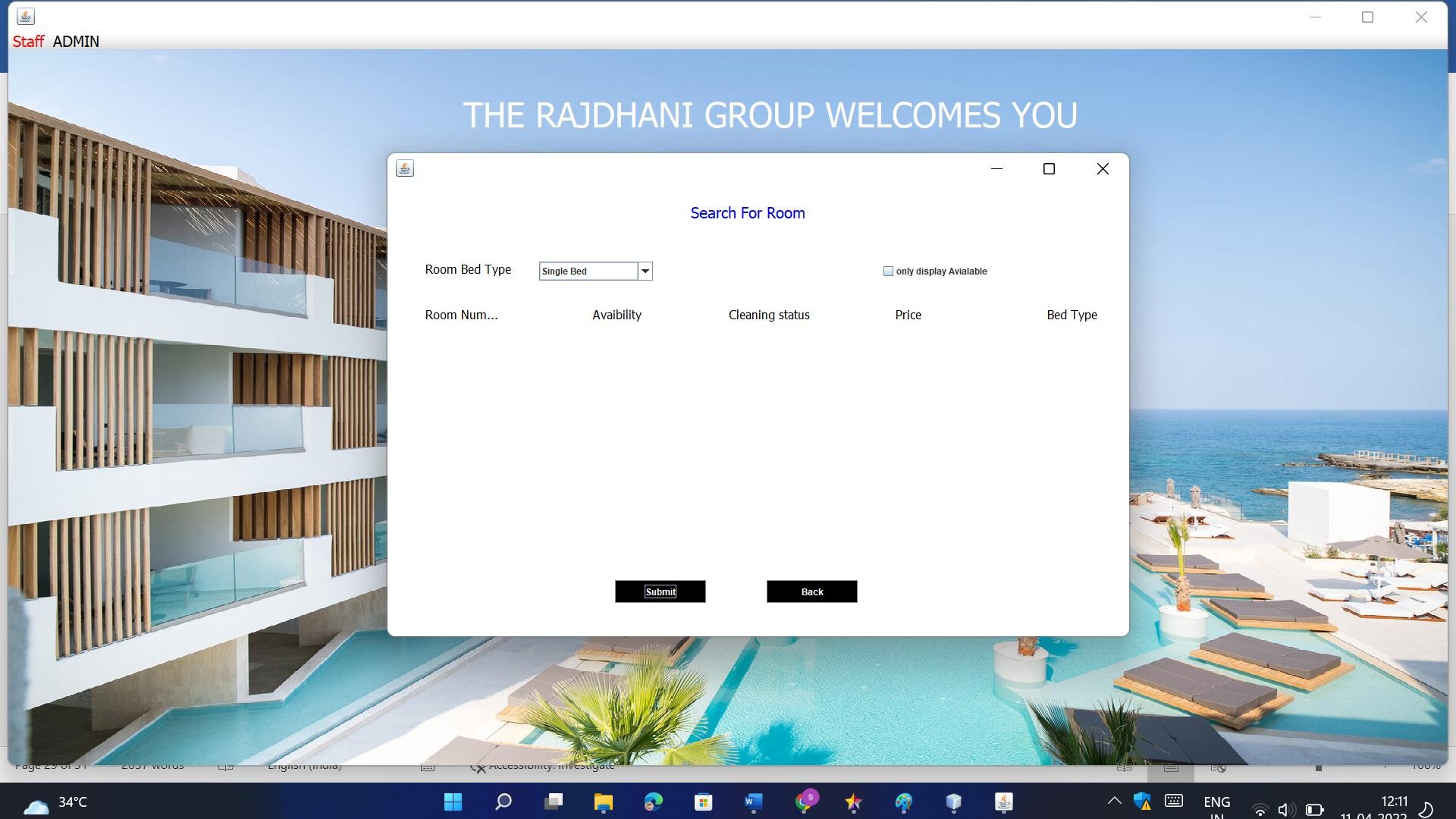


**Pick Up Service**



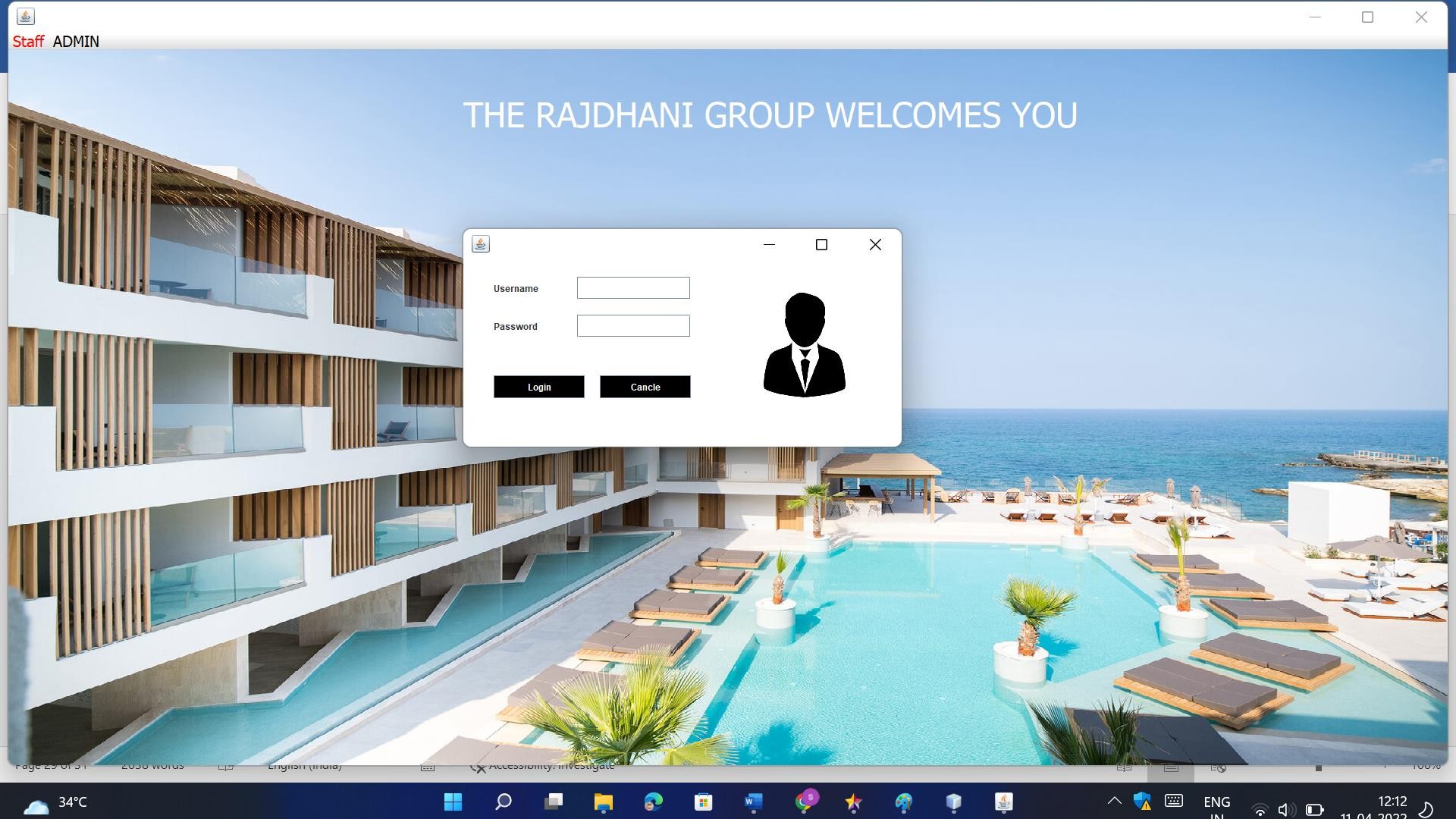
**Search Room**





**Logout**





**4.1 Limitations:**

The booking process usually requires a customer identity, which the system cannot detect.

It requires a reliable internet connection

**4.2 Future Enhancement:**

The world is changing rapidly and so is the meaning of the Rajdhani System. Today hotel is not only confined to hotels but has gone deep into tourism, catering, clubs, etc. making it a very paying and an exciting career option.

With the rapid growth of the hotel industry pushed forward by foreign and domestic tourism and business travels, the demand for well trained and quality personnel too has gone up high. India is one of the preferred tourist and travel destinations. Approx. 4.4 million tourists visit our country every year. The growth of 20% has been recorded in the tourist and hospitality industry over a few years and more growth is expected in coming years. At present, there are about 200 million of jobs available in the industry, out of which 20% of the job opportunities are in India.

The Rajdhani Hotel has a lot of enhancement options. In future more features may be added category-wise. It may try to analyze the user behavior and preferences and accordingly suggest. All concepts can be applied to make the Rajdhani Hotel more efficient.

**BIBLOGRAP**

website: <https://www.w3schools.com/css/>

<https://www.tutorialspoint.com/>

Refer Books:

MySQL - For Database

Java - E-balaguruswami

Core Java – by Dr. Nageshwar Raos

UML – By Jason T Roff

Software Project Management – By Bob Hughes, Mike Cotterell

**Annexure:**

**Sample Program Code**

**Login**

package hotel.management.system;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.sql.\*;

public class Login extends JFrame implements ActionListener{

JLabel l1,l2;

JTextField t1;

JPasswordField t2;

JButton b1,b2,b3;

Login(){

setUndecorated(true);

l1 = new JLabel("Username");

l1.setBounds(40,50,100,30);

l1.setFont(new Font("Tahoma",Font.PLAIN,20));

add(l1);

l2 = new JLabel("Password");

l2.setBounds(40,100,100,30);

l1.setFont(new Font("Tahoma",Font.PLAIN,20));

add(l2);

t1 = new JTextField();

t1.setBounds(150,50,150,30);

t1.setFont(new Font("Tahoma",Font.PLAIN,20));

add(t1);

t2 = new JPasswordField();

t2.setBounds(150,100,150,30);

l2.setFont(new Font("Tahoma",Font.PLAIN,20));

add(t2);

b1 = new JButton("Login");

b1.setBackground(Color.BLACK);

b1.setForeground(Color.WHITE);

b1.setBounds(40,170,120,30);

b1.setFont(new Font("Tahoma",Font.PLAIN,16));

b1.addActionListener(this);

add(b1);

b2 = new JButton("Cancel");

b2.setBackground(Color.BLACK);

b2.setForeground(Color.WHITE);

b2.setBounds(180,170,120,30);

b2.setFont(new Font("Tahoma",Font.PLAIN,16));

b2.addActionListener(this);

add(b2);

b3 = new JButton("regitration");

b3.setBackground(Color.BLACK);

b3.setForeground(Color.WHITE);

b3.setBounds(110,220,120,30);

b3.setFont(new Font("Tahoma",Font.PLAIN,16));

b3.addActionListener(this);

add(b3);

ImageIcon i1 =new ImageIcon(ClassLoader.getSystemResource("hotel/management/system/icons/second.jpg"));

Image i2 = i1.getImage().getScaledInstance(200,200, Image.SCALE\_DEFAULT);

ImageIcon i3 = new ImageIcon(i2);

JLabel l3 = new JLabel(i3);

l3.setBounds(310,10,200,200);

add(l3);

getContentPane().setBackground(Color.WHITE);

setLayout(null);

setBounds(600,300,540,260);

setVisible(true);

}

public void actionPerformed(ActionEvent ae){

if(ae.getSource()==b2){

System.exit(0);

}

if(ae.getSource()==b1){

String username = t1.getText();

String password = t2.getText();

String firstname = t2.getText();

String lastname = t2.getText();

if(username.isEmpty()){

JOptionPane.showMessageDialog(null,"username field is manditory");

}

if(password.isEmpty()){

JOptionPane.showMessageDialog(null,"password field is manditory");

}

conn c = new conn();

String str = "select \* from registration where username = '"+username+"' and password = '"+password+"'";

try{

ResultSet rs = c.s.executeQuery(str);

if(rs.next()){

new Dashboard().setVisible(true);

this.setVisible(false);

}else{

JOptionPane.showMessageDialog(null,"Invalid username and password ");} } }

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

new registration().setVisible(true);

}

public static void main(String[] args){

new Login();

}}