

**Report of In-Plant Training**

**Perfect Software Solution**

**On**

**Development of Hotel Management System Using C# and**

**ASP.NET with reference to Backend development**

**Amar Ganesh Pawar**

**Submitted in partial fulfillment of the requirement for**

**Post Graduation Degree In Masters Of Computer Applications (MCA)**

**of**

**Dr. Babasaheb Ambedkar Marathwada University**

**Chh. Sambhajinagar.**



**Department of Computer Applications**

**Maharashtra Institute of Technology Chh.  
Sambhajinagar  
(An Autonomous Institute)**

**2023-2024**

**G.S. Mandal's  
Maharashtra Institute of Technology,  
Chh. Sambhajinagar**

**( An Autonomous Institute )**

**Department of Computer Applications**



**CERTIFICATE**

This is to certify that the In-Plant Training Report  
Submitted By

**Mr. Amar Ganesh Pawar ( Roll No : 45 )**

Is Completed as per the requirement of the Dr. Babasaheb Ambedkar Marathwada University , Chh.Sambaji Nagar in partial fulfillment of Post Graduation Degree in Master of Computer Application (MCA)

For the academic Year 2023 - 2024

**Dr. Vaishali Bhagile  
Internal Guide**

**Dr. Prashant Chintal  
Head of Department**

**Dr. Nilesh Patil  
Director**

---

**External Examine**



# Perfect Software Solution

• S/W Development

• Web Development

Gulmandi Corner, Aurangpura, Aurangabad-431001. • Ph. No. off. : 0240-2342994 Mob.: 9168012341, 9422291397

• E-mail : uday\_Chatupale@yahoo.in

• Website : [www.perfectsoftwaresolution.com](http://www.perfectsoftwaresolution.com)

14<sup>th</sup> May, 2024

To,

Amar Pawar

Maharastra Institute Of Technology,  
C.Sambhajinagar - 431136

## CERTIFICATE OF INTERNSHIP

Dear Amar,

He is to certify that Mr. Amar Pawar has done his internship in .Net development project with us from 14<sup>th</sup> Feb to 14<sup>th</sup> May 2024.

He has worked on a project titled "Hotel Management System". The project was aimed to managing internal process through portal. As part of the project, he has worked on design and development using programming languages C# .NET, SQL Server and related plugins.

During internship, he has demonstrated skills with self-motivation to learn new technology stack. His performance was as per our expectations and was able to complete his part from the project on time.

We wish all the best for him upcoming career.

Authorized Signature

Perfect Software Solution,

## **DECLARATION**

To,  
The Head,  
Computer Applications Department,  
Maharashtra Institute of Technology, Chh Sambhaji Nagar.

Respected Sir,

I **Mr. Amar Ganesh Pawar** hereby declare that, the project titled "**Development of Hotel Management System Using C# and ASP.NET with reference to Backend development**" developed and submitted as the partial fulfillment of submission for SYMCA, under the guidance of **Dr. Vaishali Bhagile** is my original work and has not duplicated from any other sources.

**Date :**

**Place : Chh. Sambhajinagar**

Your Sincerely,

**AMAR GANESH PAWAR**

## **ACKNOWLEDGMENT**

A successful project is the result of teamwork and co-ordination that includes not only the group of developers who put forth the ideas, logic and efforts but also those who guide them. I would like to express my gratitude and sincere to the following people to whom I am grateful for their support and help without which I would not have been able to do this project entitled **“Development of Hotel Management System Using C# and ASP.NET with reference to Backend development”**.

I would like to express my honor gratitude to **Mr. Uday.C.Chaatupale** , who has given all co-operation and help to complete this project. I wish to place on my record my deep sense of gratitude to my project guide, **Dr. Vaishali Bhagile** for her constant motivation and valuable help through the project work. Express my gratitude to **Dr. Prashant Chintal** Head of MCA Department, Maharashtra Institute of Technology (MIT) , Chh.Sambhaji Nagar for his valuable suggestions and advice throughout the MCA course, I also extend my thanks to other Faculties for their Co-operation during my Course.

**Amar Ganesh Pawar**

# INDEX

<b>Sr. No.</b>	<b>Number</b>	<b>Contents</b>	<b>Page</b>
<b>1.</b>		<b>INTRODUCTION</b>	
	1.1	Company profile	1
	1.2	Existing System and Need for System	3
	1.3	Scope of Work	4
	1.4	Operating Environment at Server Side - Hardware and Software	4
	1.5	Operating Environment at Client Side -Hardware and Software	5
	1.6	Detailed Description of Technology Used	6
<b>2.</b>		<b>PROPOSED SYSTEM</b>	
	2.1	Proposed System	8
	2.2	Objectives of System	9
	2.3	User Requirements	9
<b>3.</b>		<b>ANALYSIS &amp; DESIGN</b>	
	3.1	Data Flow Diagram (DFD)	10
	3.2	Entity Relationship Diagram (ERD)	11
	3.3	UML Diagrams (Use Case, Activity, Sequence, Class Diagrams)	12
	3.4	Table Design	15
	3.5	Menu Tree Diagram (Site Map in Case of Web Application)	18
	3.6	Menu Screen	20
	3.7	Test Procedures and Implementation – With a Test Case	22
<b>4.</b>		<b>USER MANUAL</b>	
	4.1	User Manual	24
	4.2	Operations Manual / Menu Explanation	24
	4.3	Forms and Report Specifications	25
<b>5.</b>		<b>Drawbacks and Limitations</b>	26
<b>6.</b>		<b>Proposed Enhancements</b>	27
<b>7.</b>		<b>Conclusion</b>	28
<b>8.</b>		<b>Bibliography</b>	29
<b>9.</b>		<b>ANNEXURES</b>	
	9.1	Input Forms With Data	30
	9.2	Output Reports With Data(If Applicable)	31
	9.3	Sample Code	34

## **1.INTRODUCTION**

### **1.1 Company Profile:**

Perfect Software Solution is a professional software, website development company based in Aurangabad, Maharashtra that endeavors on highly proficient, timely delivered and cost effective software, website development services, We are highly experienced in offering offshore software development and project management. Perfect Software Solution is a professional software. website development company based in Aurangabad, Maharashtra that endeavors on highly proficient, timely delivered and cost effective software, website development services, We are highly experienced in offering offshore software development and project management.



- **LOCATION**

- ✓ **Head Office :**

New Shreya Nagar, Shreya Nagar , Ramanagar, Chh.Sambhaji Nagar -431001

- ✓ **Company Guide Details :**

**Name :** Uday.C. Chaatupale

**Email :** [uday\\_chaatupale@PerfectSoftwareSolution.com](mailto:uday_chaatupale@PerfectSoftwareSolution.com)

**Mobile :** +91 9422291397

**Website :** [www.perfectsoftwaresolution.com](http://www.perfectsoftwaresolution.com)

## **1.2 Existing System and Need for System**

To overcome the drawbacks of the existing system, this project is proposing a smarter way of communication. As discussed earlier an automated notification system can be developed using a very popular web based platform which will provide a Desktop based application

### **Existing System**

- The existing system of Hotel Management was manual. All the daily routines are carried out manually and the records are maintained in the record books or the registers.
- As computers have merged with man as a single entity so a computerized application can be developed that can handle Hotel Management System(HMS).
- As all the activities that happen in the Hotel such as enquiry, check status booking, food order etc. can be handled on this system simultaneously.

### **Need for System**

- **Streamlined Operations :-**

A hotel management system automates and integrates various operational tasks such as reservations, check-ins and check-outs, room assignments, housekeeping schedules, inventory management, billing, and reporting. This streamlines day-to-day operations, reduces manual errors, and improves overall efficiency.

- **Resource Management :-**

Efficient allocation of resources such as rooms, staff, and amenities is critical for maximizing profitability. A hotel management system provides real-time visibility into resource availability and utilization, enabling better decision-making and resource optimization.

- **Competitive Advantages :-**

Implementing a modern hotel management system not only improves operational efficiency but also gives hotels a competitive edge in the market. By offering superior guest experiences, optimizing revenue, and staying ahead of industry trends, hotels can attract more guests and achieve sustainable growth.

### **1.3 Scope of Work**

A hotel management system helps in enhancing the guest experience by providing them with a hassle-free check-in and check-out process. It allows guests to easily book rooms online and access real-time information about room availability, pricing, and services offered.

### **1.4 Operating Environment at Server Side Hardware and Software**

#### **Hardware requirements**

- Multi-core processors (such as Intel Xeon or AMD Ryzen) are recommended to handle current requests efficiently.
- For larger hotels or systems with heavy database usage, 32 GB or more of RAM may be necessary to ensure optimal performance.
- Solid-state drives (SSDs) are preferred over traditional hard disk drives (HDDs) for faster data access and improved system responsiveness.

#### **Software requirements**

- Versions such as Windows Server 2012, 2016, or 2019 offer robust features and support for server applications.
- SQL Server: Offers scalability, performance, and a range of features for data management.
- PostgreSQL: A powerful open-source database system known for its reliability, extensibility, and compliance with SQL standards.

## **1.5 Operating Environment at Client Side – Hardware and Software**

### **Hardware requirements**

- Processor: Dual-Core 1.6 Ghz or higher
- Processor speed : 250MHz to 833 MHz
- RAM : 2GB
- Hardware Disk : 500 MB free disk space
- Other : Network card is required

### **Software requirements**

- Operating System: Windows 10 or later.
- Development Environment: Visual Studio with .NET Framework or .NET Core SDK.
- Database Management System: SQL Server or alternative like MySQL or PostgreSQL.
- Web Server: Internet Information Services (IIS).

### **Supported operating systems**

It goes without saying that using current versions of web browsers require current/supported operating systems on the computer or mobile device. The general rule is that the two latest operating systems versions are supported. It can be applied for

- Windows, Mac-OS , Linux distributions
- Mobile systems.

## **1.6 Detailed Description Of Technology Used**

### **Front End Technologies**

#### **HTML:**

The HyperText Markup Language or HTML is the standard markup language for documents designed to be displayed in a web browser. It is often assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for its appearance.

#### **JavaScript:**

JavaScript is a high-level, interpreted scripting language that conforms to the ECMAScript specification. JavaScript has curly-bracket syntax, dynamic typing, prototype-based object orientation, and first-class functions. Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it, and major web browsers have a dedicated JavaScript engine to execute it. As a multi-paradigm language, JavaScript supports event driven, functional, and imperative (including object-oriented and prototype-based) programming styles.

#### **CSS:**

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .

### **Bootstrap:**

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation and other interface components. Bootstrap is the third most starred project on GitHub, with more than 135,000 stars, behind only free Code Camp (almost 305,000 stars) and marginally behind Vue.js framework. According to Alexa Rank, Bootstrap getbootstrap.com is in the top-2000 in the US while vuejs.org is in the top-7000 in the US. Bootstrap is a web framework that focuses on simplifying the development of informative web pages (as opposed to web apps).

### **Programming**

#### **languages: C# :**

The C# programming language was designed by Anders Hejlsberg from Microsoft in 2000 and was later approved as an international standard by Ecma (ECMA-334) in 2002 and ISO/IEC (ISO/IEC 23270) in 2003. Microsoft introduced C# along with .NET Framework and Visual Studio, both of which were closed-source. At the time, Microsoft had no open-source products. Four years later, in 2004, a free and open-source project called Mono began, providing a cross-platform compiler and runtime environment for the C# programming language. A decade later, Microsoft released Visual Studio Code (code editor), Roslyn (compiler), and the unified .NET platform (software framework).

#### **SQL :**

Structured Query Language (SQL) is a domain-specific language used in programming and designed for managing data held in a relational database management system (RDBMS), or for stream processing in relational data stream management system (RDSMS). It is particularly useful in handling structured data, i.e., data incorporating relations among entities and variables. SQL offers two main advantages over older read-write APIs such as ISAM or VSAM.

#### **ASP.NET :**

ASP.NET is an open-source, server-side web-application framework designed for web development to produce dynamic web pages. It was developed by Microsoft to allow programmers to build dynamic web sites, applications and services.

## **2.PROPOSED SYSTEM**

### **2.1 Proposed System**

Administrator Interface

Design. Security

Authentication. Reports.

General end-users.

**The following are the modules included in this system :**

- Admin
- Owner
- Employee
- Staff

## **2.2 Objectives Of System**

- Provide superior guest experiences through features such as online booking, mobile check-in/check-out, personalized services, quick response to guest requests, and efficient problem resolution.
- Maximize revenue through dynamic pricing strategies, yield management techniques, upselling opportunities, and effective management of room rates, packages, and promotions.
- Efficiently allocate resources including rooms, staff, amenities, and inventory to minimize waste, reduce costs, and maximize profitability.
- Empower hotel staff with user-friendly tools, training programs, and support resources to enhance their productivity, job satisfaction, and ability to deliver exceptional service to guests.

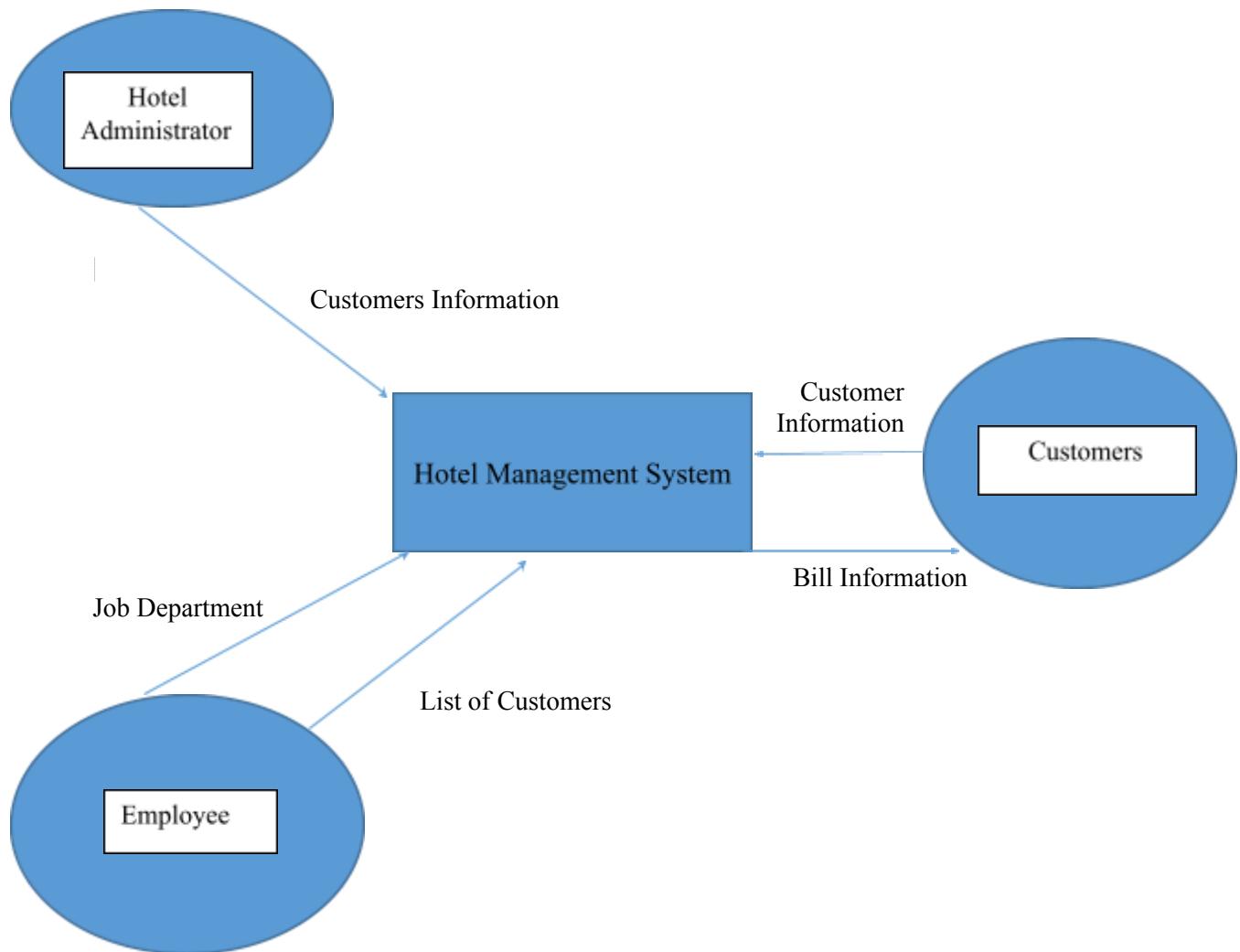
## **2.3 User Requirements**

- A user-friendly interface for quick and easy check-in and check-out processes.
- Ability to assign rooms based on guest preferences, availability, and special requests.
- Access to guest information, preferences, and booking history for personalized service.
- Real-time updates on room status (e.g., clean, dirty, under maintenance) for efficient housekeeping scheduling.

### **3. ANALYSIS AND DESIGN**

#### **3.1 Data flow diagram (DFD)**

- Data Flow Diagram – Hotel Management System
- A data flow diagram is a graphical view of how data is processed in a system in terms of input and output.
- The Data flow diagram (DFD) contains some symbol for drawing the data flow diagram

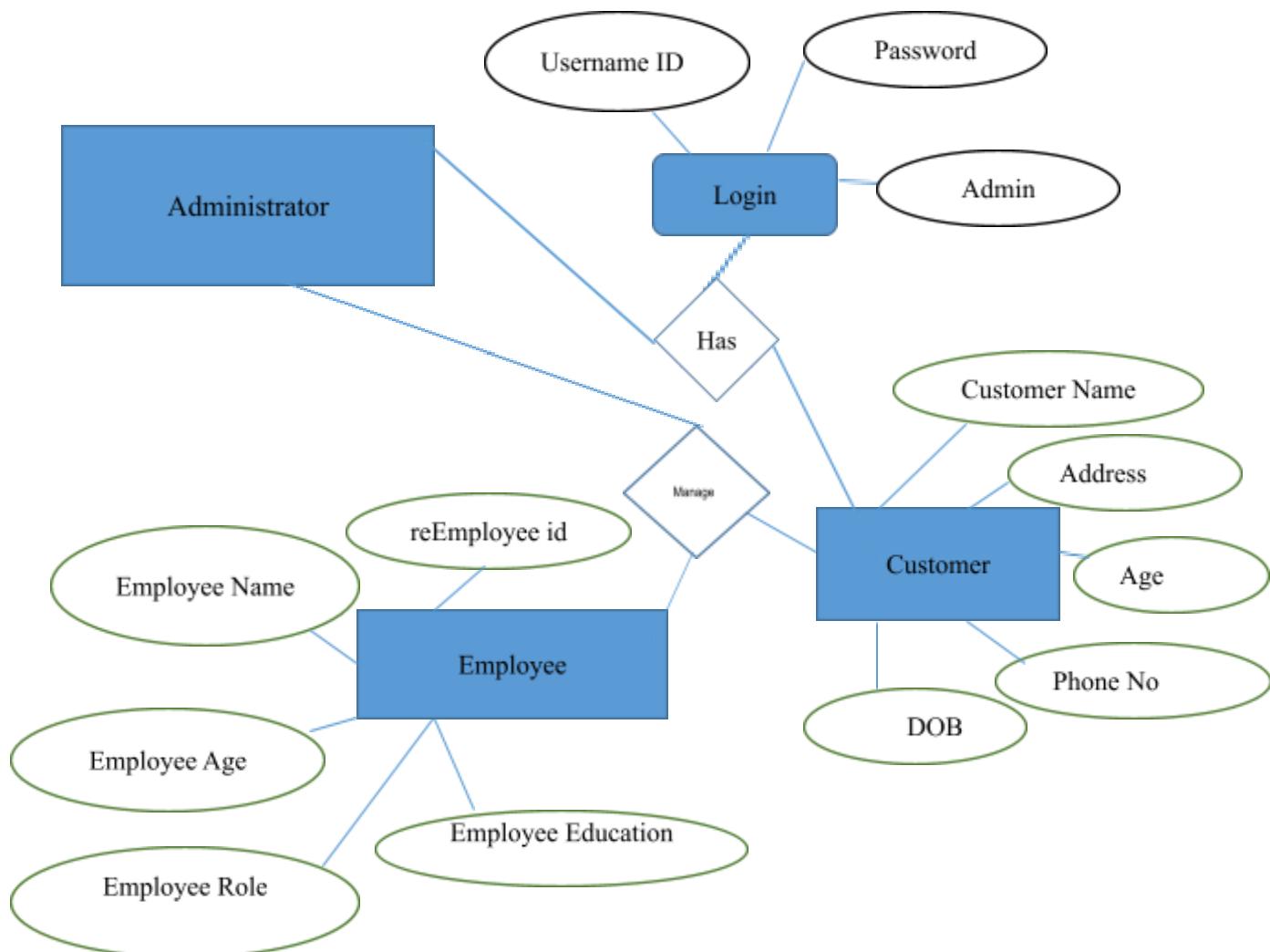


DFD Diagram for Hotel Management System

### 3.2 Entity Relationship Diagram (ERD)

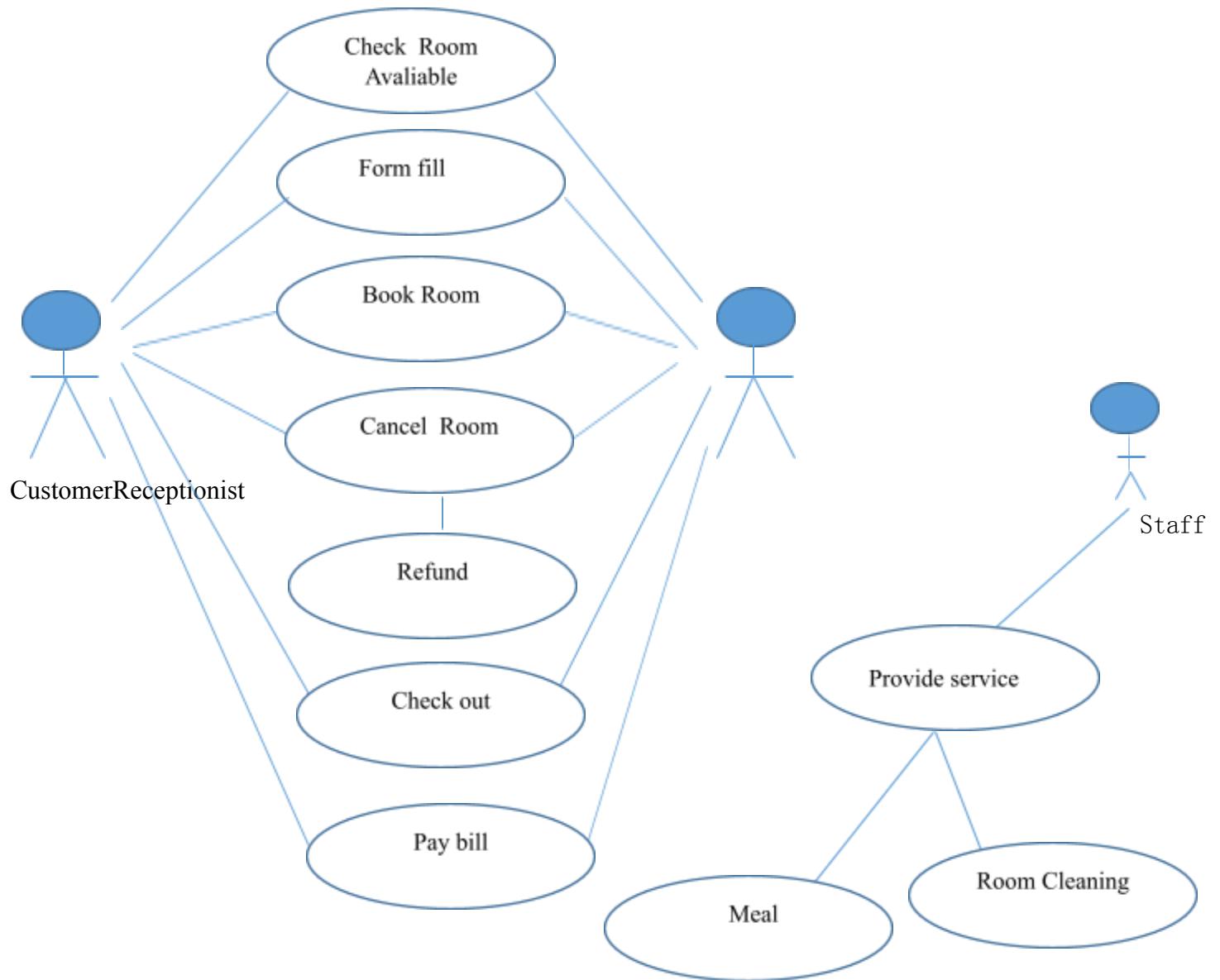
E-R (Entity-Relationship) Diagram is used to represent the relationship between entities in a table. ER diagrams represent the logical structure of databases. ER Diagram represent relationship between two database tables.

E-R diagram means Entity Relationship diagram. Entity is a object of a system, generally we refer entity as database table , the e-r diagram represent the relationship between each table of database. E-R diagram represent entity with attributes, attributes are the properties of an entity. If we assume entity is a database table then all the columns of table are treat as attributes



ER Diagram for Hotel Management System

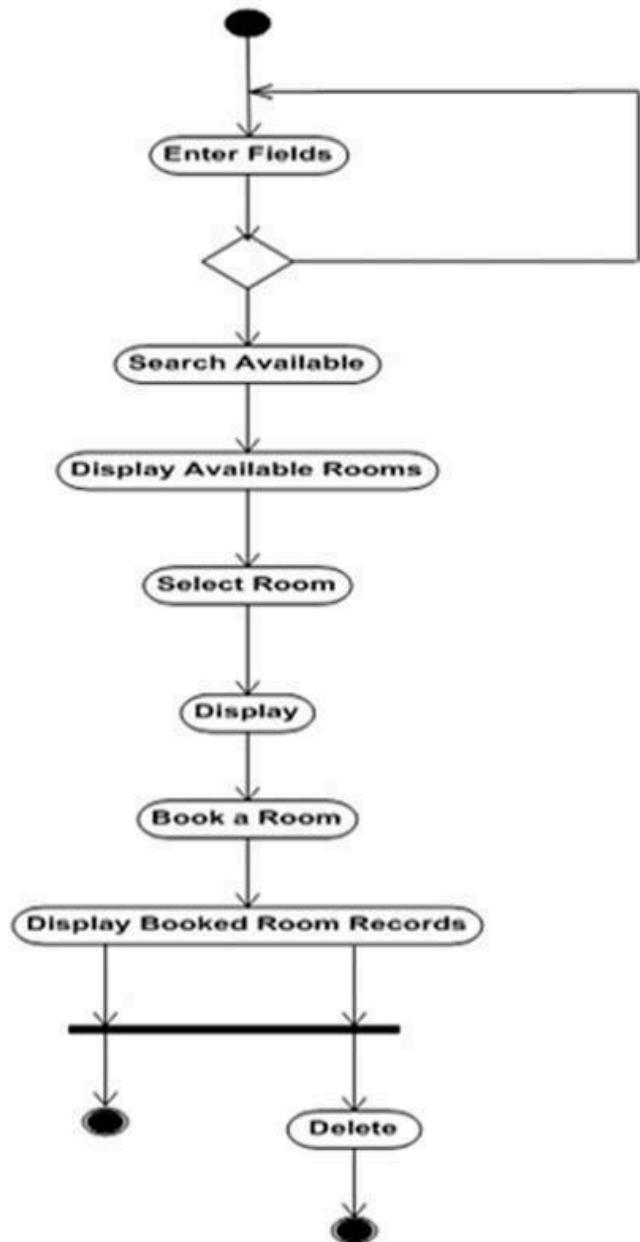
### 3.3 UML Diagram



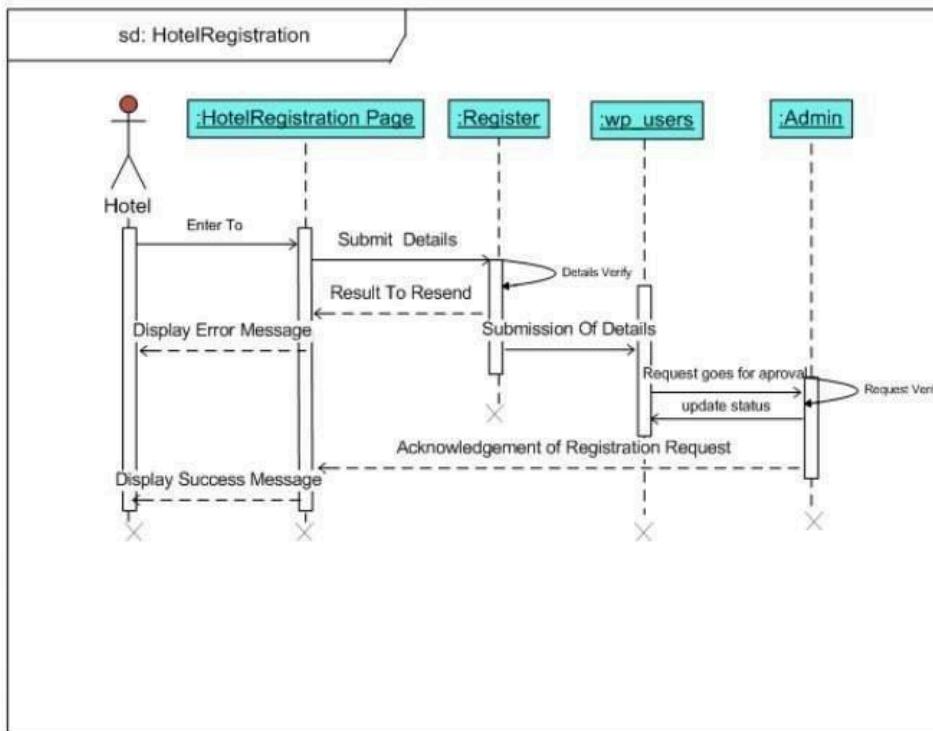
Use Case Diagram for Hotel Management System

## Activity Diagram

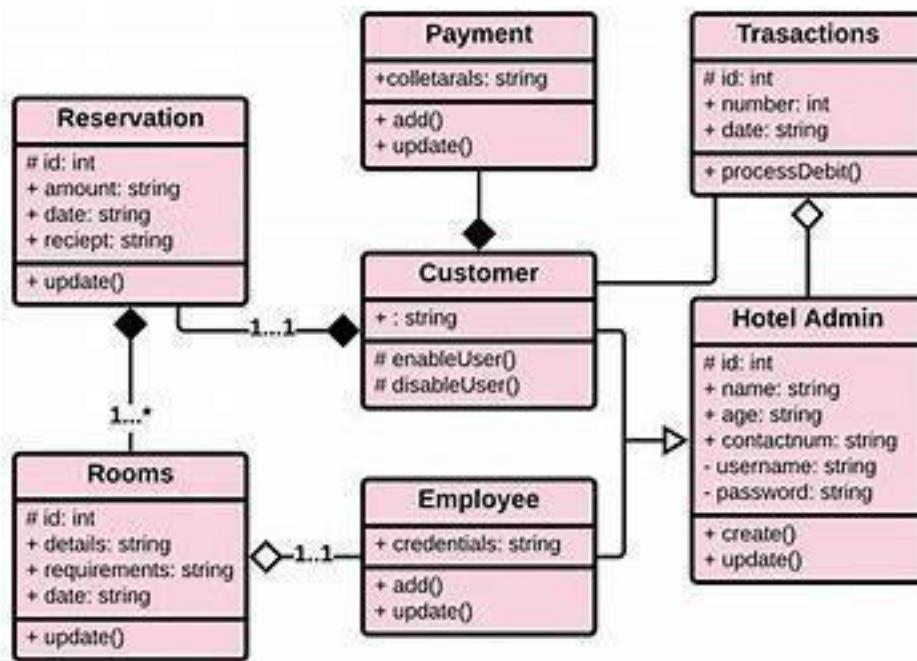
### Hotel Management System - Activity Diagram - Room Booking



## Sequence Diagram



## Class Diagram



### 3.4 Table Design

**Table Name : User Table**

Name	Data Type	Allow Nulls	Default
User ID	int	Not Null	True
User Name	Varchar(20)	Null	False
password	Varchar(30)	Null	False
User role	int	Null	False

**Table Name : Countries**

Name	Data Type	Allow Nulls
CountryID	int	Not Null
CountryName	Nvarchar(100)	Null
IsDeleted	bit	Null

**Table Name: State**

Name	Data Type	Allow Nulls
StateID	int	Not Null
StateName	Nvarchar(100)	Null
IsDeleted	bit	Null

**Table Name: District**

Name	Data Type	Allow Nulls
DistrictID	int	Not Null
DistrictName	Nvarchar(100)	Null
IsDeleted	bit	Null

**Table Name: City**

Name	Data Type	Allow Nulls
CityID	int	Not Null
CityName	Nvarchar(100)	Null
DistrictName	Nvarchar(100)	Null
StateName	Nvarchar(100)	Null
CountryName	Nvarchar(100)	Null
IsDeleted	bit	Null

**Table Name: Designation**

Name	Data Type	Allow Nulls
DesignationID	int	Not Null
DesignationName	Nvarchar(100)	Null
IsDeleted	bit	Null

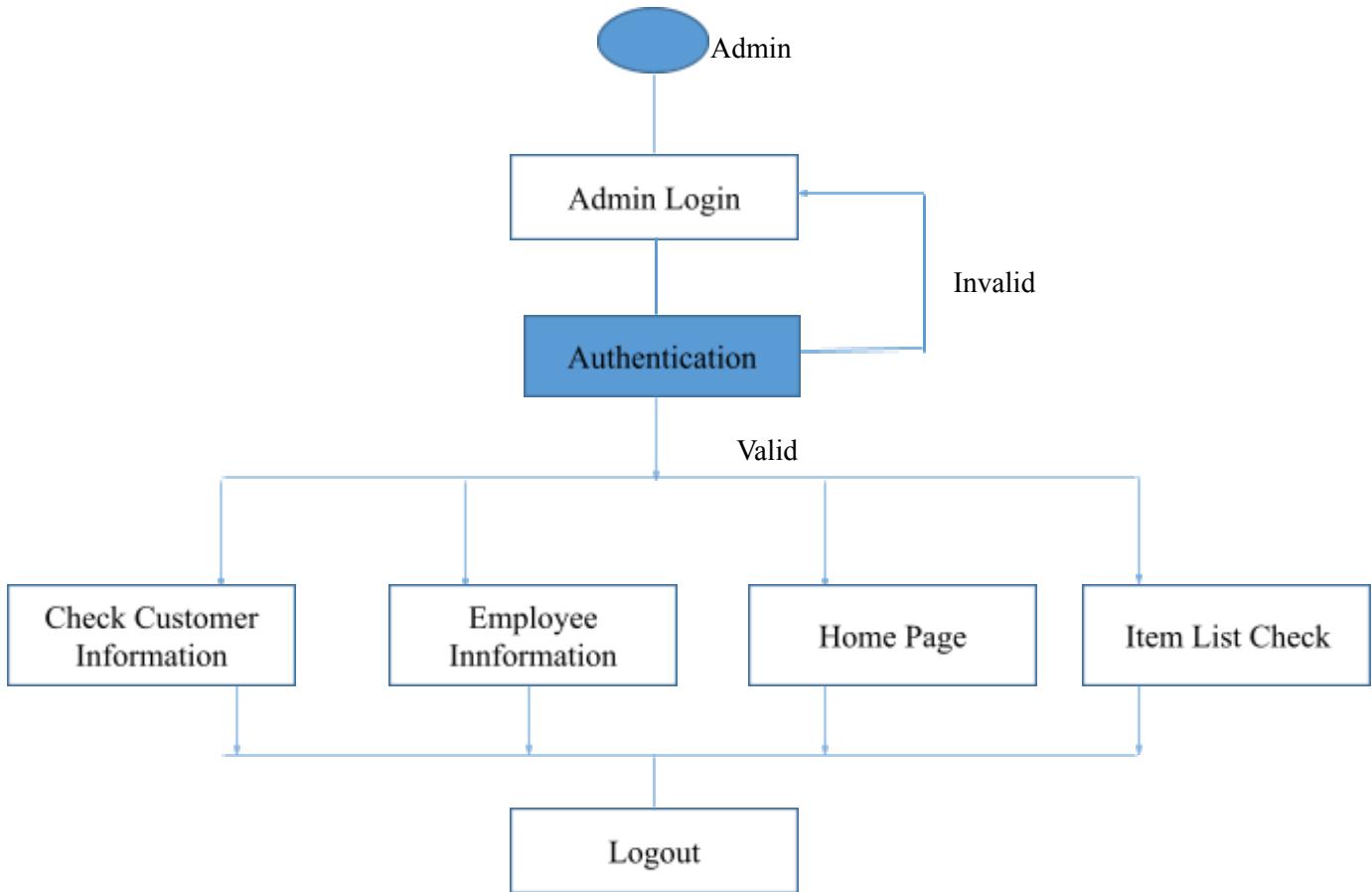
**Table Name: Customer**

Name	Data Type	Allow Nulls
CustomerID	int	Not Null
CustomerName	Nvarchar(100)	Null
Address	Nvarchar(100)	Null
DateofBirth	Nvarchar(100)	Null
WhatsappNo	bigint	Not Null
GstNo	bigint	Not Null
IsDeleted	bit	Null

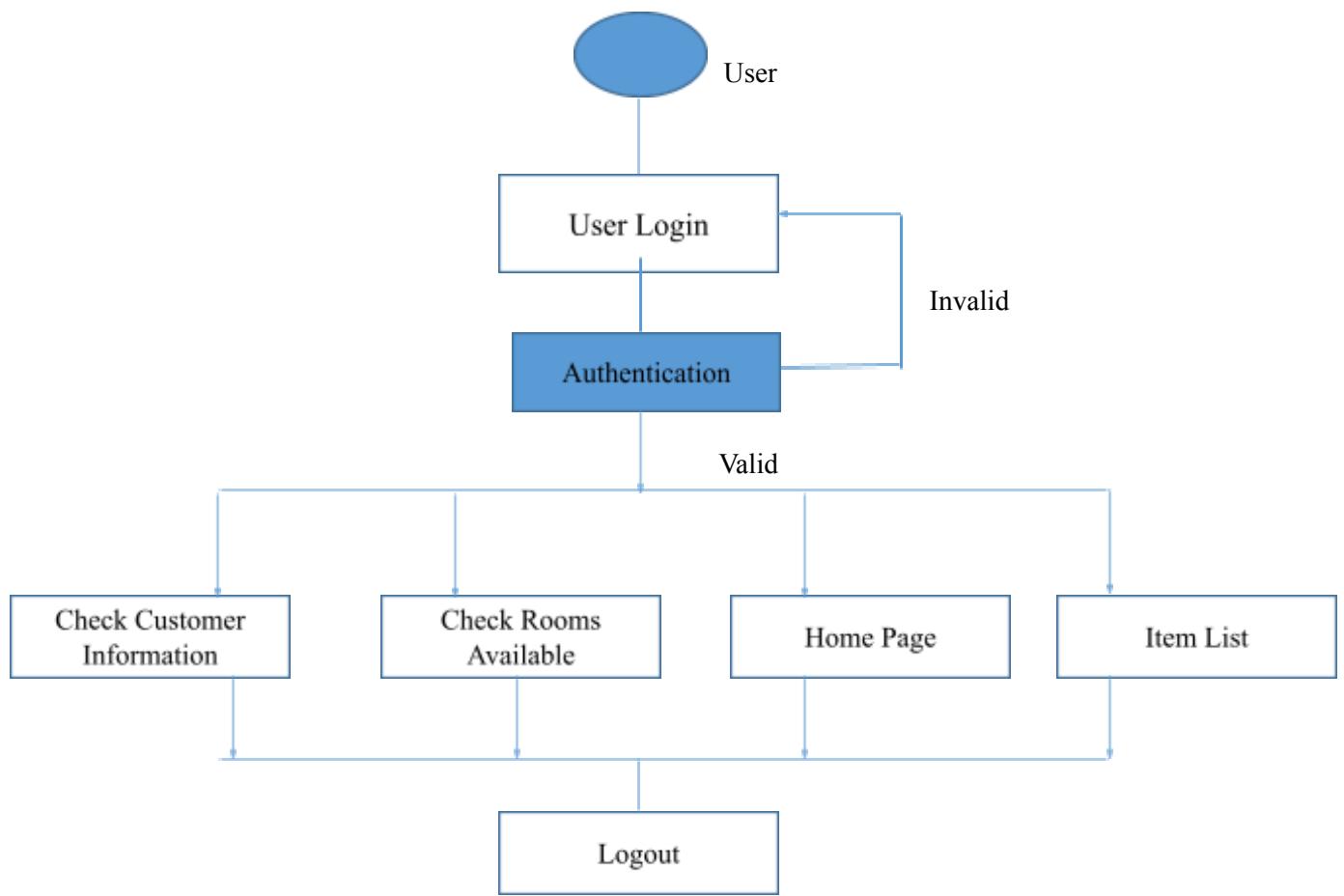
**Table Name: Employee**

Name	Data Type	Allow Nulls
EmployeeID	int	Not Null
EmployeeName	Nvarchar(100)	Null
Address	Nvarchar(100)	Null
Education	Nvarchar(100)	Null
MobileNo	bigint	Not Null
DesignationName	Nvarchar(100)	Null
IsDeleted	bit	Null

### 3.5 Menu Tree Diagram

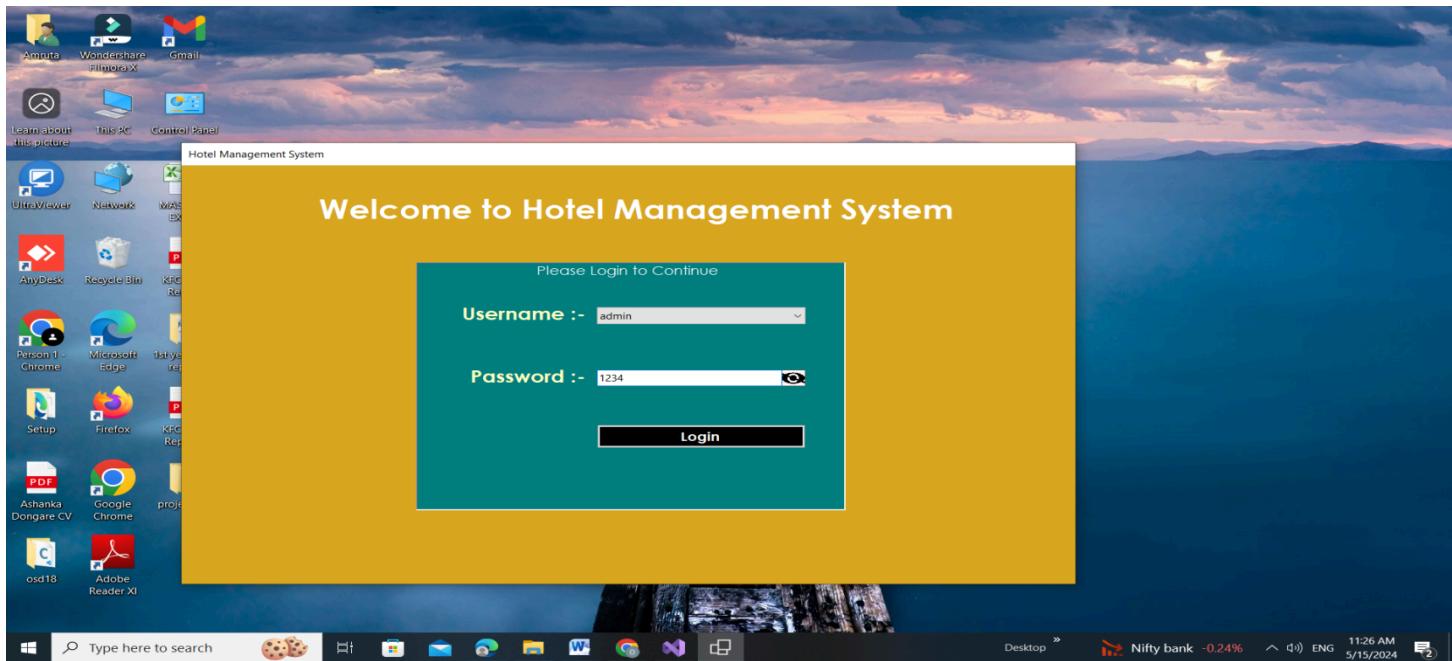


Admin - Activity Diagram



User – Activity Diagram

### 3.6 Menu Screens



## Customer Information

Customer ID

1009

Customer Type \*

Diamond

Customer Name \*

Whatsapp No \*

Address \*

Alternate Mobile No

City Name \*

GST No

Date of Birth \*

Wednesday, May 15, 2024

Aadhar No \*

Aniversary Date

Wednesday, May 15, 2024

Remark

**Save****Edit****Delete****Close**

	CustomerID	CustomerName	Address	CityName	DateofBirth	AniversaryDate	CustomerType	WhatsappNo	AlternateMobileN	GSTNo	AadharNo	Remark
▶	1009	omkar	waluj	Sambhajinagar	Wednesday, Ma...	Wednesday, Ma...	Diamond	8249749339	883985847	9287425970	8240842938	20382972040
■	1008	Shyam	midc	pune	Thursday, July 2...	Wednesday, Fe...	Golden	8498758765	80579689	986540595	5589580995	good
*												



### **3.7 Test Procedure and Implementation – with a Test Case**

#### **Basis – Path Testing**

It is a white-box testing technique which enables the test case designer to derive a logical complexity measure as a guide for defining a basis set of execution paths. Test cases derived to exercise the basis set are guaranteed to execute every statement in the program at least once during testing.

#### **Test Case**

Testing For Login Page

Test Engineer:	Group members
Test Case ID:	TC1
Date:	13-4-2024
Purpose:	For valid outcome, and restriction to fill every form (Login, Signup,)
Pre-Req:	The customer will first of all enter his credentials in order to get himself registered with the system.
Test Data:	Foam consisting of different variables and data are tested in this test case. A member cannot leave a field empty.
Steps:	Steps to carry out the test. See step formatting rules below. <ol style="list-style-type: none"><li>1. Visit the Hotel Management System official website.</li><li>2. Enter Username ( Required )</li><li>3. Enter Password ( Required )</li><li>4. Submit</li></ol>
Status:	Pass

Test Engineer:	Group members
Test Case ID:	TC1
Date:	10-5-2024
Purpose:	To check everything is in order because Admin has all authority.
Pre-Req:	First, an Admin must be logged in to the system. (Required)
Test Data:	<p>All data and variables related to the admin panel are used in this testing. The Following major components</p> <ul style="list-style-type: none"> <li>· Add Items</li> <li>· Check Rooms</li> <li>· Customer Information</li> <li>· Employee Information</li> </ul>
Steps:	<ol style="list-style-type: none"> <li>1. Login, (required).</li> <li>2. Admin will add Items</li> <li>3. Admin will check All Customer Information</li> <li>4. Admin will check all available Rooms</li> <li>5. Admin will maintain Software</li> <li>6. Submit</li> </ol>
Status:	Pass

## **4.USER MANUAL**

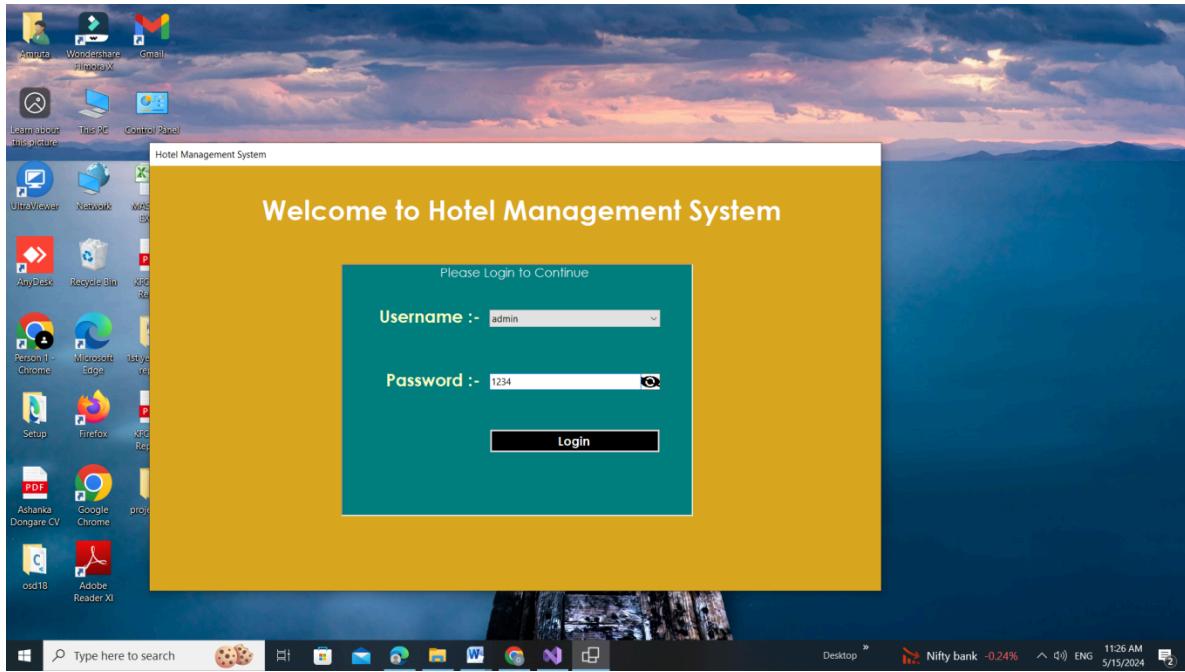
### **4.1 User manual:**

- **The process of logging in and registering can be found here:**
  - You can register or sign up on the Hotel Management System Application.
  - Provide the required information such as name, email, contact number, and other details.
  - Once registered, use your credentials to log in to the Application.

### **4.2 Operations Manual /Menu Explanation**

- **The process of Entering the Customer information and Employee Information:**
  - You can add customer information and employee information the Hotel Management System Application.
  - Provide the hotel services to the customer such as menu items, cleaning
  - Once registered, use your credentials to log in to the Application.
  - After, click on the logout button.

### 4.3 Forms and Specification



## **5. DRAWBACKS AND LIMITATIONS**

Hotel management systems (HMS), while powerful tools, do come with some drawbacks and limitations.

- **Cost:** Implementing and maintaining an HMS can be expensive, especially for smaller hotels. Subscription fees, training costs, and potential hardware upgrades can add up.
- **Dependence on Technology:** HMS relies heavily on a stable internet connection. Any outages can disrupt operations, leading to frustrated guests and potential revenue loss.
- **Security Risks:** Storing guest data in a digital system makes it a target for cyberattacks. Robust security measures are essential to protect sensitive information.
- **Limited Customization:** Not all HMS are created equal. Some may not offer the level of customization needed for a hotel's specific needs, potentially hindering workflow and guest experience.
- **Learning Curve:** Staff needs to be trained on the new system, which can take time and resources. A user-friendly interface can help, but a learning curve exists nonetheless.
- **Reduced Human Touch:** While HMS streamlines tasks, it can also impersonalize guest interactions. Finding the right balance between efficiency and fostering genuine hospitality is crucial.

## **6. PROPOSED ENHANCEMENTS**

Here are some proposed enhancements for Hotel Management Systems (HMS) to address current drawbacks and limitations:

- **Scalable Solutions:** Develop tiered pricing structures and cloud-based options to make HMS more affordable for smaller hotels.
- **Offline Functionality:** Implement features that allow basic operations to continue during internet outages, minimizing disruptions.
- **Enhanced Security:** Integrate advanced cybersecurity measures including encryption, multi-factor authentication, and regular security audits.
- **Open API Architecture:** Allow for easier customization and integration with third-party applications to cater to specific hotel needs.
- **Intuitive Interface Design:** Focus on user-friendly interfaces with clear instructions and training modules to minimize the learning curve for staff.
- **Guest Communication Tools:** Integrate features for personalized communication channels like SMS and in-app messaging, fostering guest interaction and reducing the feeling of impersonal service.
- **AI-powered Features:** Utilize Artificial Intelligence (AI) for tasks like automated guest support through chatbots, personalized recommendations based on guest preferences, and predictive maintenance for hotel equipment.
- **Integration with IoT Devices:** Enable seamless integration with Internet of Things (IoT) devices in guest rooms, allowing guests to control lighting, temperature, and entertainment systems through their smartphones, further enhancing the guest experience.

By implementing these enhancements, HMS can become even more robust and user-friendly, addressing current limitations and propelling hotels towards a future of efficient operations, exceptional guest experiences, and a competitive edge in the hospitality industry.

## **7. CONCLUSION**

The hotel management project has been a comprehensive endeavor aimed at revolutionizing the hospitality experience through strategic innovation and operational optimization. Extensive staff training sessions were pivotal in ensuring that the new systems were utilized to their full potential, promoting a culture of excellence and continuous professional development. Feedback mechanisms were established to monitor performance and gather insights, allowing for ongoing adjustments and improvements. The project's success is evident in the tangible benefits achieved: a marked increase in occupancy rates, more efficient resource allocation, and a noticeable uplift in guest reviews and ratings.

In addition to operational improvements, the project emphasized sustainability and eco-friendly practices, integrating energy-saving technologies and waste reduction strategies. This commitment not only enhances the hotel's market appeal but also aligns with growing consumer expectations for sustainable travel options. The project's success serves as a testament to the power of strategic planning, technological integration, and a dedicated team, setting a new standard for excellence in the hospitality industry.

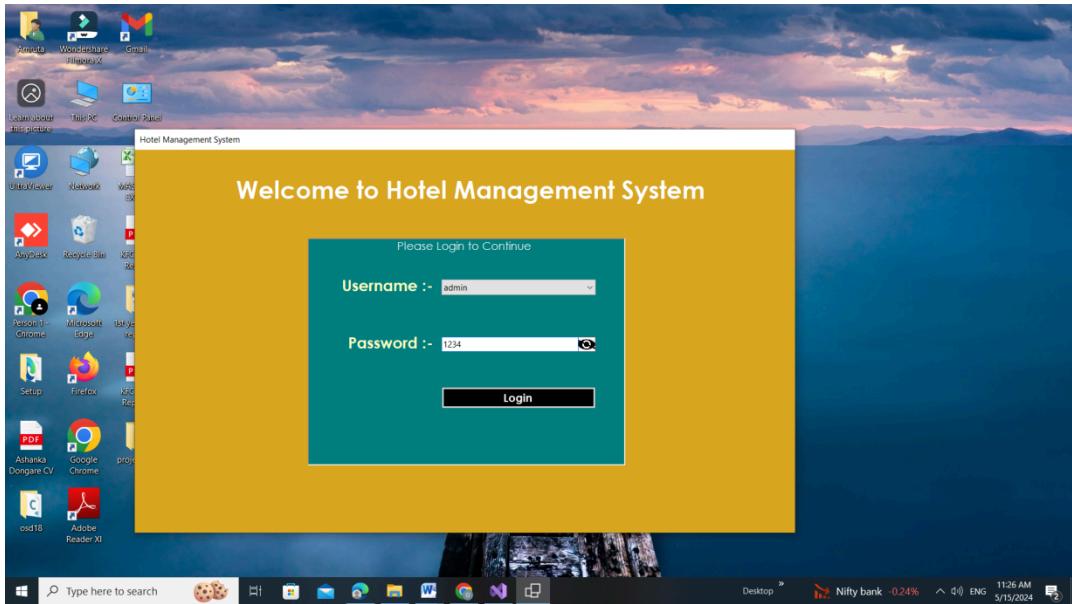
Ultimately, a well-optimized hotel management system not only optimizes operational efficiency and revenue generation but also fosters heightened guest satisfaction and loyalty. Positioned strategically, it fortifies the hotel's competitive edge and sets the stage for sustained success in an ever-evolving industry landscape.

## **8. BIBLIOGRAPHY**

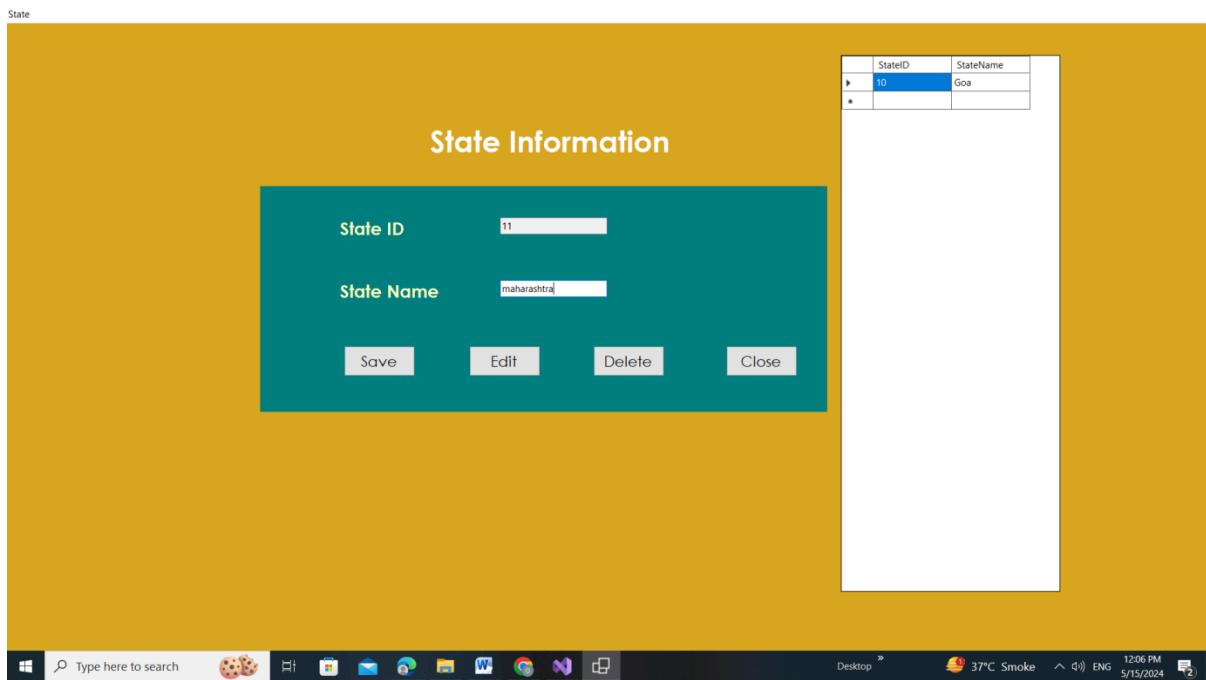
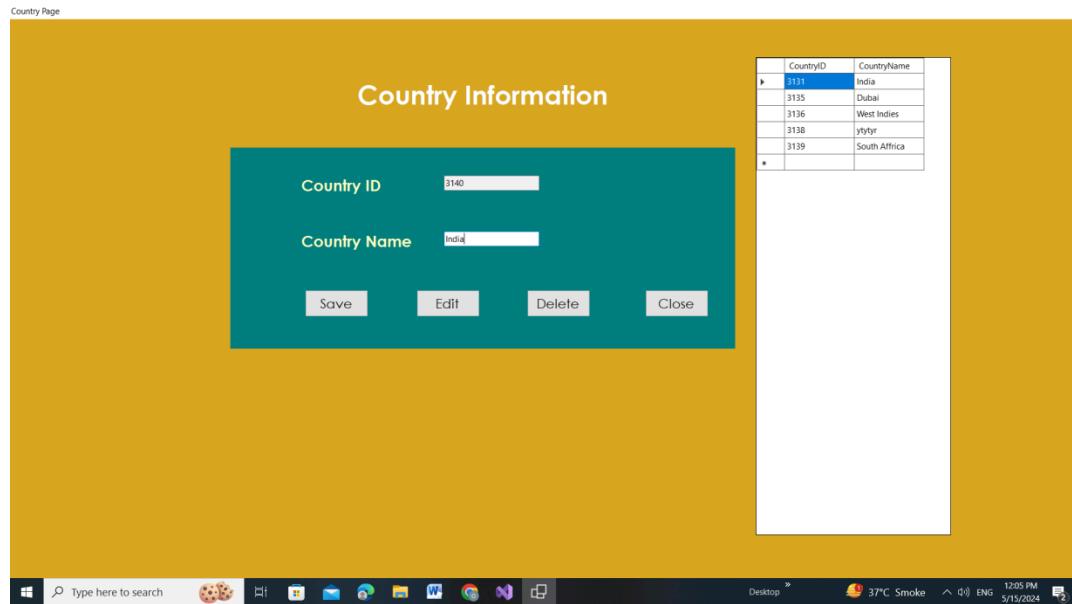
- Steven Feuerstein, Bill Pribyl ., PL/SQL Programming, O'Reilly Media, 2014
- Jon Duckett ., Java Script Programming, Wiley, 2014
- Elliotte Rusty Harold ., JAVA Networking, O'Reilly Media, 2010
- Rene Enriquez ., JAVA Security, Packt Publishing, 2014
- [www.irjmets.com](http://www.irjmets.com)
- [www.emerald.com](http://www.emerald.com)
- [www.researchgate.com](http://www.researchgate.com)
- [www.studocu.com](http://www.studocu.com)

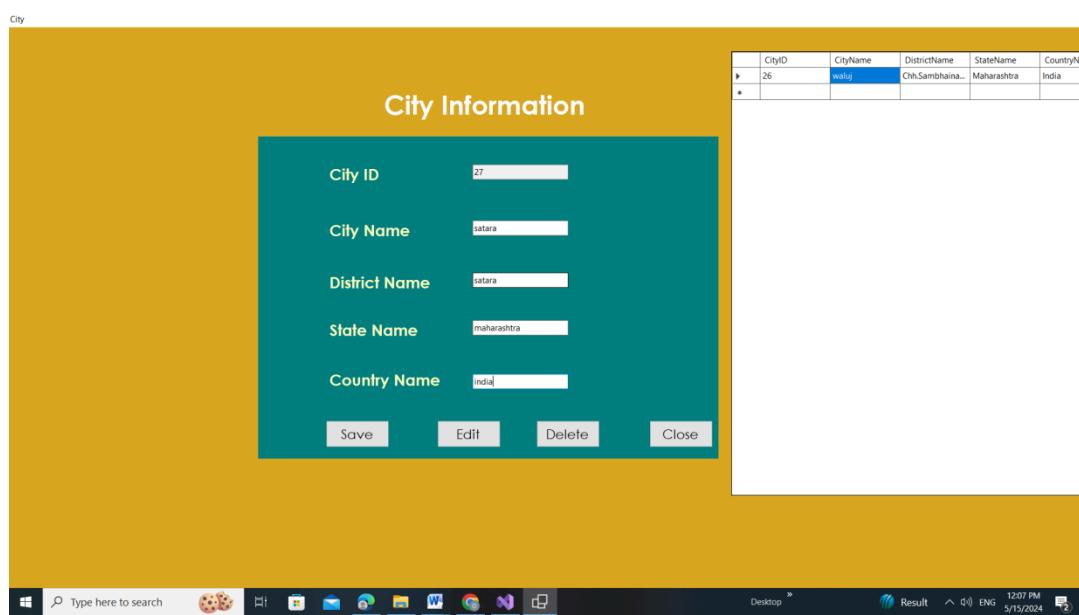
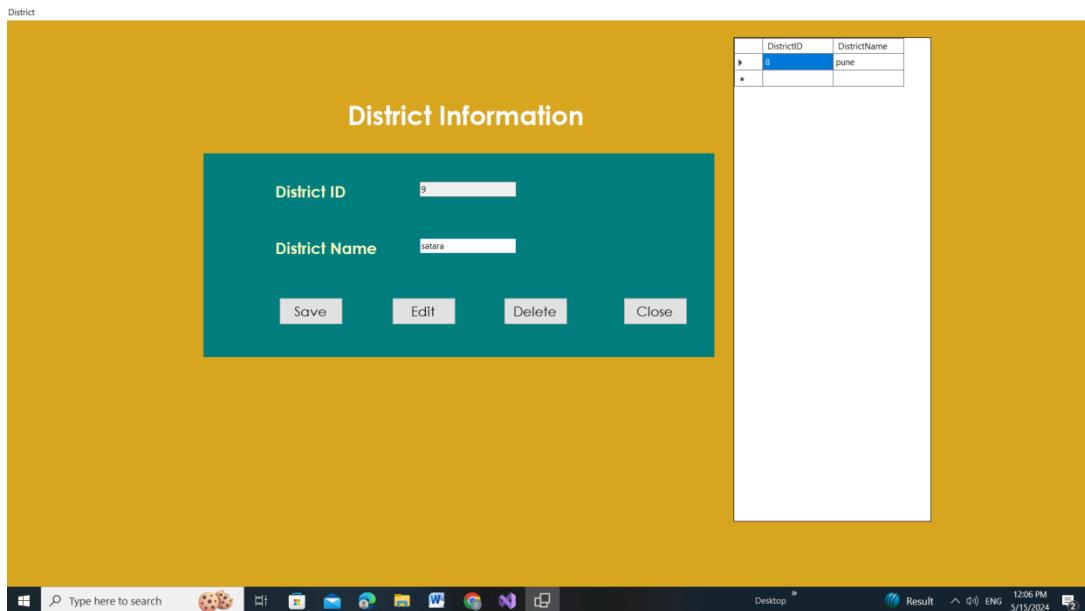
## 9. ANNEXURES

### 9.1 Input Forms with Data



## 9.2 Output Reports with Data





Customer

### Customer Information

Customer ID	1009	Customer Type *	Diamond
Customer Name *		Whatsapp No *	
Address *		Alternate Mobile No	
City Name *		GST No	
Date of Birth *	Wednesday, May 15, 2024	Aadhar No *	
Aniversary Date	Wednesday, May 15, 2024	Remark	

Save    Edit    Delete    Close

CustomerID	CustomerName	Address	CityName	DateofBirth	AniversaryDate	CustomerType	WhatsappNo	AlternateMobileNo	GSTNo	AadharNo	Remark
▶ 1009	omkar	waluj	Sambhajinagar	Wednesday, Ma...	Wednesday, Ma...	Diamond	6249749339	883985847	9287425970	8240842938	20382972040
1008	Shyam	midc	pune	Thursday, July 2...	Wednesday, Fe...	Golden	8498758765	80579689	986540595	5588580995	good
▪											

Windows Type here to search Cookies Desktop > Result 12:08 PM ENG 5/15/2024

Designation

### Designation Information

District ID	4
District Name	

Save    Edit    Delete    Close

DesignationID	DesignationName
▶ 4	manager
2	Cook
3	Waiter
▪	

Windows Type here to search Cookies Desktop > 37°C Smoke 12:09 PM ENG 5/15/2024

### 9.3 Sample Code

#### Login Page

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using static System.Windows.Forms.VisualStyles.VisualStyleElement;

namespace Hotel_Management_System
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
            Username_Click();
        }

        private void Username_Click()
        {
            try
            {

                using (SqlConnection connection = new SqlConnection(connectionString))
                {
                    connection.Open();
                    SqlCommand command = new SqlCommand("SELECT Username FROM user_table", connection);
                    SqlDataReader reader = command.ExecuteReader();
                    while (reader.Read())
                    {
                        comboBoxUsername.Items.Add("admin");
                        comboBoxUsername.Items.Add("manager");
                        comboBoxUsername.Items.Add("owner");
                        comboBoxUsername.Items.Add("staff");
                    }
                }
            }
        }
    }
}
```

```

        }

    }

}

catch (Exception ex)
{
    MessageBox.Show("Error loading usernames: " + ex.Message);
}
}

private void buttonLogin_Click(object sender, EventArgs e)
{

    string username = comboBoxUsername.Text;
    string password = textBoxPassword.Text;

    // Implement your authentication logic here
    if (AuthenticateUser(username, password))
    {

        // MessageBox.Show("Login Successful!");
        FormHome formHome = new FormHome();
        formHome.Show();
        this.Hide();

        // Proceed to next steps after successful login
    }
    else
    {
        if (string.IsNullOrEmpty(username) || string.IsNullOrEmpty(password))
        {
            DialogResult dialogResult = MessageBox.Show("No empty fields allowed", "You cannot continue ", MessageBoxButtons.OK);
        }
        else
        {
            MessageBox.Show("Username Or Password is incorrect!", "You cannot continue ");
        }
    }
}

private bool AuthenticateUser(string username, string password)
{
    try
    {
        using (SqlConnection connection = new SqlConnection(connectionString))

```

```

{
connection.Open();
SqlCommand command = new SqlCommand("SELECT COUNT(*) FROM user_table WHERE Username = @Username AND Password = @Password", connection);
command.Parameters.AddWithValue("@Username", username);
command.Parameters.AddWithValue("@Password", password);
int count = (int)command.ExecuteScalar();
return count > 0;
}
}
catch (Exception ex)
{
MessageBox.Show("Error authenticating user: " + ex.Message);
return false;
}
}

private void Password_Click(object sender, EventArgs e)
{
}

private void comboBoxUsername_SelectedIndexChanged(object sender, EventArgs e)
{
}

private void Form1_Load(object sender, EventArgs e)
{
//textBoxPassword.Focus();

comboBoxUsername.Items.Clear();

// Add unique usernames to the ComboBox
AddUniqueUsername("admin");
AddUniqueUsername("manager");
AddUniqueUsername("owner");
AddUniqueUsername("staff");

textBoxPassword.Focus();
}
private void AddUniqueUsername(string username)
{
// Check if the username is already in the ComboBox
if (!comboBoxUsername.Items.Contains(username))
{
// Add the username to the ComboBox
}
}

```

```

comboBoxUsername.Items.Add(username);

}

}

private void comboBoxUsername_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
    {
        e.SuppressKeyPress = true; // Suppress the default Enter key behavior (newline in TextBox)
        textBoxPassword.Focus(); // Move the focus to the password TextBox
    }
}

private void textBoxPassword_KeyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Enter)
    {
        e.SuppressKeyPress = true; // Suppress the default Enter key behavior (newline in TextBox)
        buttonLogin.Focus(); // Perform login or any other action here
    }
}

private void pictureBoxShow_MouseHover(object sender, EventArgs e)
{
    toolTip1.SetToolTip(pictureBoxShow, "Show Password");
}

private void pictureBoxHide_MouseHover(object sender, EventArgs e)
{
    toolTip1.SetToolTip(pictureBoxHide, "Hide Password");
}

private void pictureBoxShow_Click(object sender, EventArgs e)
{
    pictureBoxShow.Hide();
    textBoxPassword.UseSystemPasswordChar = false;
    pictureBoxHide.Show();
}

private void pictureBoxHide_Click(object sender, EventArgs e)
{
    pictureBoxHide.Hide();
    textBoxPassword.UseSystemPasswordChar = true;
    pictureBoxShow.Show();
}
}

```

## **RoomManagementForm**

```
using System;
using System.Collections.Generic;
using System.Windows.Forms;

public partial class RoomManagementForm : Form
{
    private List<Room> rooms; // Replace with your data structure to store rooms (e.g., database connection)

    public RoomManagementForm()
    {
        InitializeComponent();

        // Load room data from your data source (e.g., database)
        rooms = LoadRooms(); // Implement LoadRooms() to retrieve room data
        UpdateRoomList(); // Update displayed room information
    }

    private List<Room> LoadRooms()
    {
        // Implement logic to retrieve room data from your data source (e.g., database)
        // Replace with your actual implementation
        List<Room> roomList = new List<Room>();
        roomList.Add(new Room { RoomNumber = 101, RoomType = "Single", Availability = true });
        roomList.Add(new Room { RoomNumber = 102, RoomType = "Double", Availability = false });
        return roomList;
    }

    private void UpdateRoomList()
    {
        // Clear existing list items (if any)
        listViewRooms.Items.Clear();

        // Add each room information to the list view
        foreach (Room room in rooms)
        {
            ListViewItem item = new ListViewItem(room.RoomNumber.ToString());
            item.SubItems.Add(room.RoomType);
            item.SubItems.Add(room.Availability ? "Available" : "Occupied");
            listViewRooms.Items.Add(item);
        }
    }

    private void btnAddRoom_Click(object sender, EventArgs e)
    {
    }
```

```

// Open a form for adding a new room
AddRoomForm addRoomForm = new AddRoomForm();
addRoomForm.ShowDialog();

// If a new room was added successfully (implement logic in AddRoomForm.cs)
if (addRoomForm.DialogResult == DialogResult.OK)
{
    rooms.Add(addRoomForm.NewRoom); // Add the new room to the data structure
    UpdateRoomList(); // Refresh the displayed room list
}

private void btnEditRoom_Click(object sender, EventArgs e)
{
    // Get the selected room from the list view
    if (listViewRooms.SelectedItems.Count == 0)
    {
        MessageBox.Show("Please select a room to edit.");
        return;
    }
    int selectedRoomNumber = int.Parse(listViewRooms.SelectedItems[0].Text);

    // Find the corresponding room in the data structure
    Room selectedRoom = rooms.Find(room => room.RoomNumber == selectedRoomNumber);

    // Open a form for editing the selected room
    EditRoomForm editRoomForm = new EditRoomForm(selectedRoom);
    editRoomForm.ShowDialog();

    // If the room was edited successfully (implement logic in EditRoomForm.cs)
    if (editRoomForm.DialogResult == DialogResult.OK)
    {
        // Update the room information in the data structure
        selectedRoom.RoomType = editRoomForm.EditedRoom.RoomType;
        selectedRoom.Availability = editRoomForm.EditedRoom.Availability;
        UpdateRoomList(); // Refresh the displayed room list
    }
}

// Implement similar logic for btnDeleteRoom (handle confirmation and data structure update)
}

public class Room // Example class to represent a room
{
    public int RoomNumber { get; set; }
    public string RoomType { get; set; }
    public bool Availability { get; set; }
}

```

```
}
```

## MenuForm

```
using System;
using System.Windows.Forms;

public partial class MenuForm : Form
{
    public MenuForm()
    {
        InitializeComponent();
    }

    private void btnRoomManagement_Click(object sender, EventArgs e)
    {
        RoomManagementForm roomForm = new RoomManagementForm();
        roomForm.MdiParent = this; // Set this form as the MDI parent
        roomForm.Show();
    }

    private void btnGuestManagement_Click(object sender, EventArgs e)
    {
        // Open the guest management form (implement logic)
        GuestManagementForm guestForm = new GuestManagementForm();
        guestForm.MdiParent = this;
        guestForm.Show();
    }

    private void btnBookings_Click(object sender, EventArgs e)
    {
        // Open the bookings form (implement logic)
        BookingsForm bookingsForm = new BookingsForm();
        bookingsForm.MdiParent = this;
        bookingsForm.Show();
    }

    private void btnExit_Click(object sender, EventArgs e)
    {
        Application.Exit();
    }
}
```

## ItemListForm

```
using System;
using System.Collections.Generic;
using System.Windows.Forms;

public partial class ItemListForm : Form
{
    private List<Item> items; // List to store items

    public ItemListForm()
    {
        InitializeComponent();
        items = new List<Item>(); // Initialize items list
        UpdateItemList(); // Initially display an empty list
    }

    private void btnAddItem_Click(object sender, EventArgs e)
    {
        // Open a form to add a new item (implement logic)
        AddItemForm addItemForm = new AddItemForm();
        addItemForm.ShowDialog();

        if (addItemForm.DialogResult == DialogResult.OK)
        {
            items.Add(addItemForm.NewItem); // Add the new item
            UpdateItemList(); // Update displayed items
        }
    }

    private void btnEditItem_Click(object sender, EventArgs e)
    {
        // Get the selected item from the list view
        if (listViewItems.SelectedItems.Count == 0)
        {
            MessageBox.Show("Please select an item to edit.");
            return;
        }

        int selectedIndex = listViewItems.SelectedIndices[0];
        Item selectedItem = items[selectedIndex];

        // Open a form to edit the selected item
        EditItemForm editItemForm = new EditItemForm(selectedItem);
        editItemForm.ShowDialog();

        if (editItemForm.DialogResult == DialogResult.OK)
```

```

    {
        // Update the item information in the list
        items[selectedIndex] = editItemForm.EditedItem;
        UpdateItemList(); // Refresh the displayed list
    }
}

private void btnDeleteItem_Click(object sender, EventArgs e)
{
    // Get the selected item from the list view
    if (listViewItems.SelectedItems.Count == 0)
    {
        MessageBox.Show("Please select an item to delete.");
        return;
    }

    if (MessageBox.Show("Are you sure you want to delete the selected item?", "Confirmation",
        MessageBoxButtons.YesNo) == DialogResult.Yes)
    {
        int selectedIndex = listViewItems.SelectedIndices[0];
        items.RemoveAt(selectedIndex);
        UpdateItemList(); // Refresh the displayed list
    }
}

private void UpdateItemList()
{
    listViewItems.Items.Clear(); // Clear existing list items

    foreach (Item item in items)
    {
        ListViewItem listViewItem = new ListViewItem(item.Name);
        listViewItem.SubItems.Add(item.Description); // Add additional properties as needed
        listViewItems.Items.Add(listViewItem);
    }
}

public class Item // Example class to represent an item
{
    public string Name { get; set; }
    public string Description { get; set; } // Add additional properties as needed
}

```

## BillingForm

```
using System;
using System.Collections.Generic;
using System.Windows.Forms;

public partial class BillingForm : Form
{
    private List<BillingItem> billingItems; // List to store billing items
    private double totalAmount = 0.0; // Total amount before taxes

    public BillingForm()
    {
        InitializeComponent();
        billingItems = new List<BillingItem>(); // Initialize billing items list
        UpdateBillingList(); // Initially display an empty list
    }

    private void btnAddItem_Click(object sender, EventArgs e)
    {
        // Open a form to add a new billing item (implement logic)
        AddItemForm addItemForm = new AddItemForm();
        addItemForm.ShowDialog();

        if (addItemForm.DialogResult == DialogResult.OK)
        {
            billingItems.Add(addItemForm.NewBillingItem); // Add the new item
            UpdateBillingList(); // Update displayed billing items
            CalculateTotal(); // Recalculate total amount
        }
    }

    private void UpdateBillingList()
    {
        listViewBillingItems.Items.Clear(); // Clear existing list items

        foreach (BillingItem item in billingItems)
        {
            ListViewItem listItem = new ListViewItem(item.Description);
            listItem.SubItems.Add(item.Quantity.ToString());
            listItem.SubItems.Add(item.UnitPrice.ToString("C")); // Format as currency
            listItem.SubItems.Add(item.CalculateTotal().ToString("C")); // Format as currency
            listViewBillingItems.Items.Add(listItem);
        }
    }

    private void CalculateTotal()
```

```

{
    totalAmount = 0.0; // Reset total before recalculating

    foreach (BillingItem item in billingItems)
    {
        totalAmount += item.CalculateTotal();
    }

    lblSubtotal.Text = totalAmount.ToString("C"); // Format as currency
    // Implement logic to calculate and display taxes based on your requirements
    // (e.g., apply tax rate to total amount and display the result)
    lblGrandTotal.Text = totalAmount.ToString("C"); // Placeholder for grand total (including tax)
}

private void btnGenerateBill_Click(object sender, EventArgs e)
{
    // Implement logic to generate a bill (e.g., print or save as PDF)
    // You can use libraries like iTextSharp for PDF generation
    MessageBox.Show("Bill generated successfully!"); // Placeholder for success message
}
}

public class BillingItem // Example class to represent a billing item
{
    public string Description { get; set; }
    public int Quantity { get; set; }
    public double UnitPrice { get; set; }

    public double CalculateTotal()
    {
        return Quantity * UnitPrice;
    }
}

```

