HOTEL BOOKING SYSTEM USING JSP

Department of Computer Science and Engineering

MINI PROJECT REPORT

19CS30E - ADVANCED JAVA PROGRAMMING

NATIONAL ENGINEERING COLLEGE

K.R. NAGAR

அறிவே ஆக்கம்

Estd : 1984

SUBMITTED BY

Shivram U - 2112088

Dharun Ram K - 2112118

TABLE OF CONTENTS

INTRODUCTION

A hotel booking system, or hotel reservation software, is a technology platform that enables hoteliers to accept direct bookings through the hotel's website and through various distribution channels.

Also known as a booking system or online booking engine, it is a tool that allows hotel guests to schedule the dates of their stay, choose rooms at the time of booking, and take payment from them.

This system brings efficiency and accessibility, allowing potential guests to book rooms at any time of the day or night without the need for staff oversight.Reservation systems enable direct bookings from your website, cutting out the middleman and ensuring that guests get the best rates—and you keep the most revenue.Real-time updates in the hotel reservation system prevent overbooking and keep everything in sync, while the system itself enhances the guest experience by allowing them to tailor their stay according to their preferences.

ABSTRACT

The Hotel Booking System is a project implemented for Hail Hotel, which is an imaginary hotel. It provides people all over the world with an easy and fast way to book hotel rooms online. The interface of the Hotel Booking System is Web pages that can be accessed with a Web site browser. The system is implemented using JSP(Java Server Page) with database connectivity using MySQL. Users can perform room booking activities at Hail Hotel anytime and anywhere by accessing it via Internet.

The Hotel Booking System is an easy-to-use application. Everyone who knows how to use a Web browser can easily carry out booking, change the booking details, cancel the booking, change the personal profile, view the booking history, or view the hotel information by following its simple and clear GUI (Graphical user interface) design.

Traditional methods of hotel booking often involve time-consuming processes and paperwork, leading to inefficiencies and potential errors. The Hotel Booking System addresses these challenges by providing an online platform that allows guests to browse available hotels, view room details, and make reservations effortlessly. For hotel owners and managers, it offers a user-friendly interface to manage room inventory, reservations, and guest information.

TECHNOLOGY USED

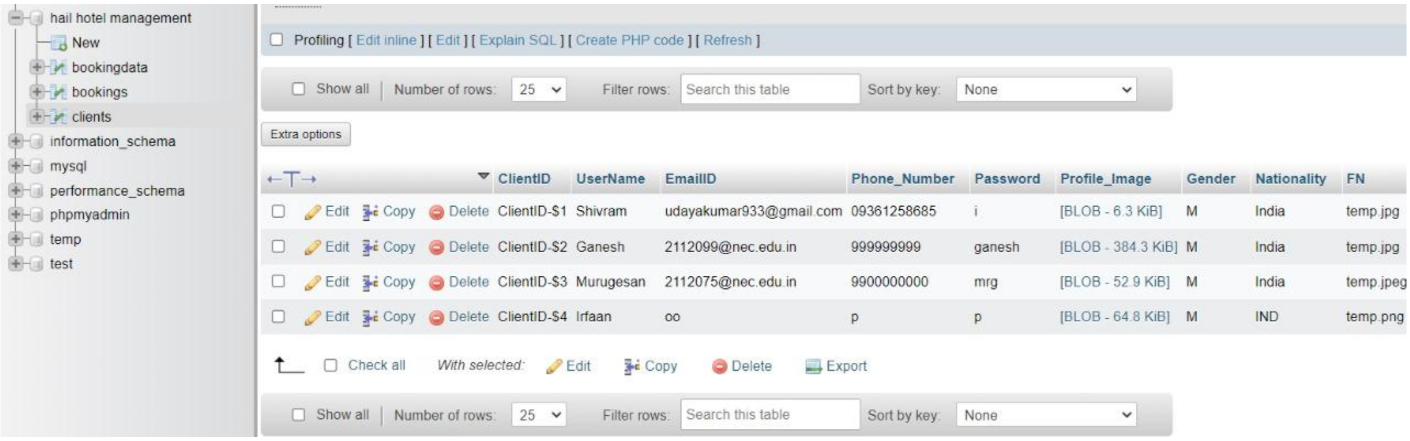
- HTML(Hyper Text Markup Language)
- CSS(Cascading Style Sheets)
- JS(Javascript)
- JSP(Java Server Page)
- MySQL(My Structured Query Language)

SOFTWARE USED

- Visual Studio Code (For Frontend)
- phpMyAdmin (For Backend)
- XAMPP(Server)

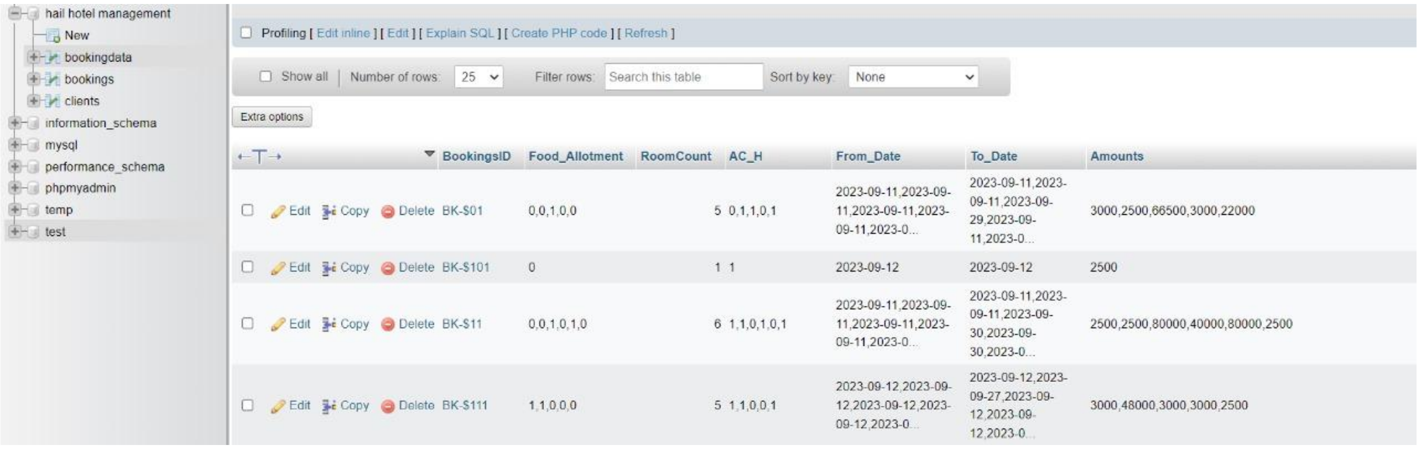
LIST OF TABLES

Clients


hail hotel management
New
bookingdata
bookings
clients
information_schema
mysql
performance_schema
phpmyadmin
temp
test
□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None
Extra options
↑ □ Check all With selected: Edit Copy Delete Export
□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

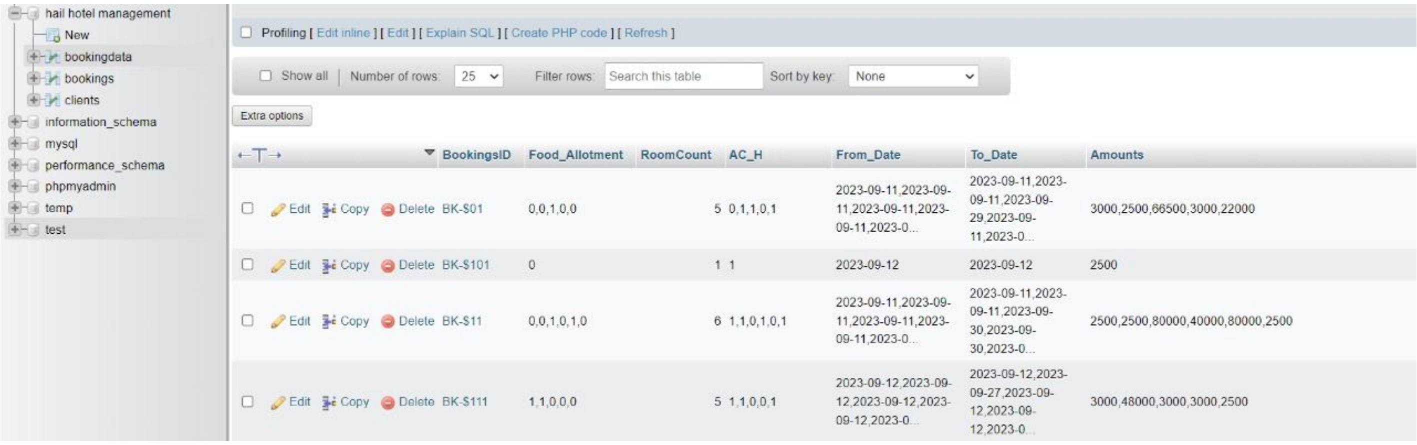
	ClientID	UserName	EmailID	Phone_Number	Password	Profile_Image	Gender	Nationality	FN
□ Edit Copy Delete	ClientID-$1	Shivram	udayakumar933@gmail.com	09361258685	i	[BLOB - 6.3 KiB]	M	India	temp.jpg
□ Edit Copy Delete	ClientID-$2	Ganesh	2112099@nec.edu.in	999999999	ganesh	[BLOB - 384.3 KiB]	M	India	temp.jpg
□ Edit Copy Delete	ClientID-$3	Murugesan	2112075@nec.edu.in	9900000000	mrg	[BLOB - 52.9 KiB]	M	India	temp.jpeg
□ Edit Copy Delete	ClientID-$4	Irfaan	oo	p	p	[BLOB - 64.8 KiB]	M	IND	temp.png

BookingData


hail hotel management
New
bookingdata
bookings
clients
information_schema
mysql
performance_schema
phpmyadmin
temp
test
□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None
Extra options

	BookingsID	Food_Allotment	RoomCount	AC_H	From_Date	To_Date	Amounts
□ Edit Copy Delete	BK-$01	0,0,1,0,0	5	0,1,1,0,1	2023-09-11,2023-09-11,2023-09-11,2023-09-11,2023-0...	2023-09-11,2023-09-11,2023-09-29,2023-09-11,2023-0...	3000,2500,66500,3000,22000
□ Edit Copy Delete	BK-$101	0	1	1	2023-09-12	2023-09-12	2500
□ Edit Copy Delete	BK-$11	0,0,1,0,1,0	6	1,1,0,1,0,1	2023-09-11,2023-09-11,2023-09-11,2023-09-11,2023-0...	2023-09-11,2023-09-11,2023-09-30,2023-09-30,2023-0...	2500,2500,80000,40000,80000,2500
□ Edit Copy Delete	BK-$111	1,1,0,0,0	5	1,1,0,0,1	2023-09-12,2023-09-12,2023-09-12,2023-09-12,2023-0...	2023-09-12,2023-09-27,2023-09-12,2023-09-12,2023-0...	3000,48000,3000,3000,2500

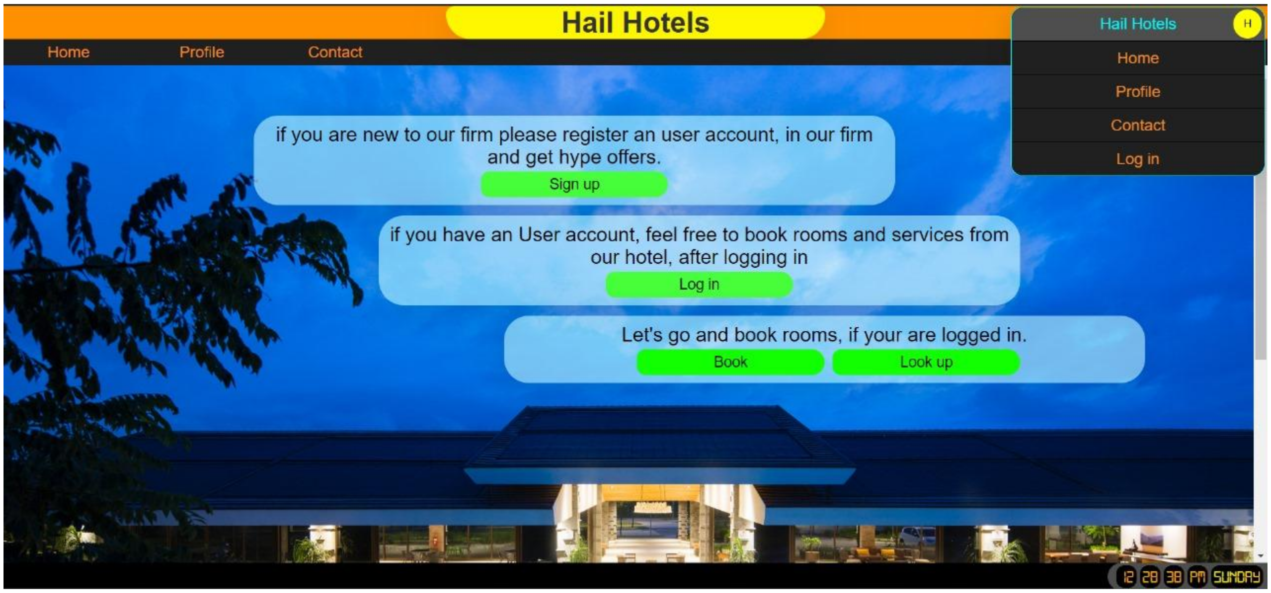
Bookings


hail hotel management
New
bookingdata
bookings
clients
information_schema
mysql
performance_schema
phpmyadmin
temp
test
□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None
Extra options

	BookingsID	Food_Allotment	RoomCount	AC_H	From_Date	To_Date	Amounts
□ Edit Copy Delete	BK-$01	0,0,1,0,0	5	0,1,1,0,1	2023-09-11,2023-09-11,2023-09-11,2023-09-11,2023-0...	2023-09-11,2023-09-11,2023-09-29,2023-09-11,2023-0...	3000,2500,66500,3000,22000
□ Edit Copy Delete	BK-$101	0	1	1	2023-09-12	2023-09-12	2500
□ Edit Copy Delete	BK-$11	0,0,1,0,1,0	6	1,1,0,1,0,1	2023-09-11,2023-09-11,2023-09-11,2023-09-11,2023-0...	2023-09-11,2023-09-11,2023-09-30,2023-09-30,2023-0...	2500,2500,80000,40000,80000,2500
□ Edit Copy Delete	BK-$111	1,1,0,0,0	5	1,1,0,0,1	2023-09-12,2023-09-12,2023-09-12,2023-09-12,2023-0...	2023-09-12,2023-09-27,2023-09-12,2023-09-12,2023-0...	3000,48000,3000,3000,2500

IMPLEMENTATION

Welcome Page

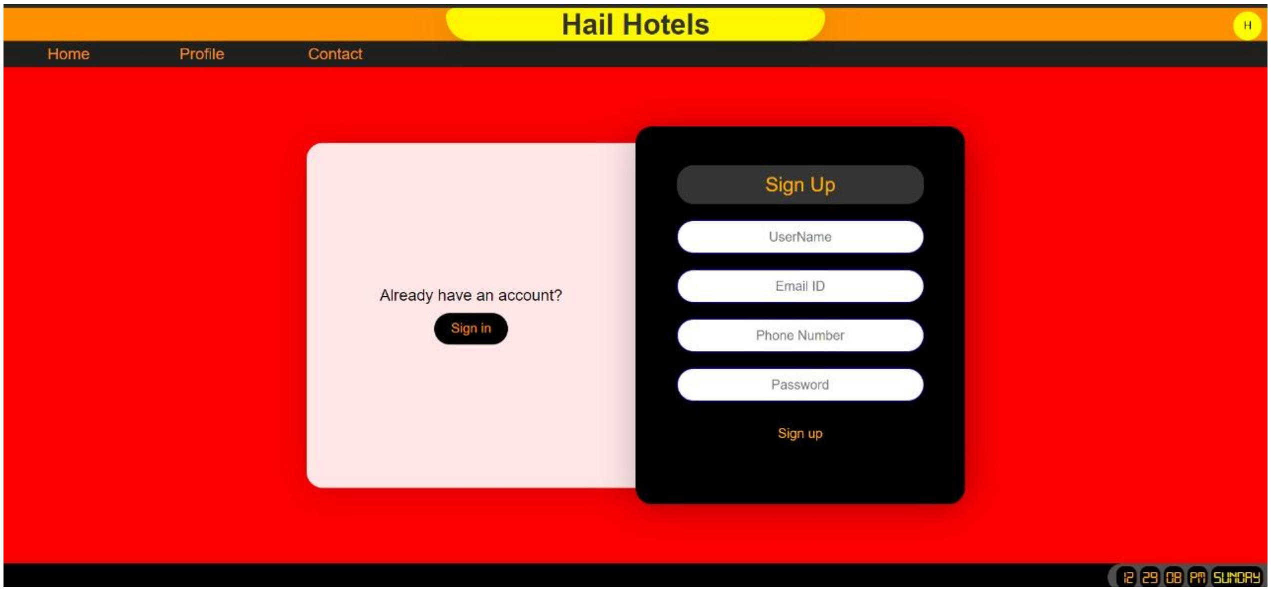


if you are new to our firm please register an user account, in our firm and get hype offers.

if you have an User account, feel free to book rooms and services from our hotel, after logging in

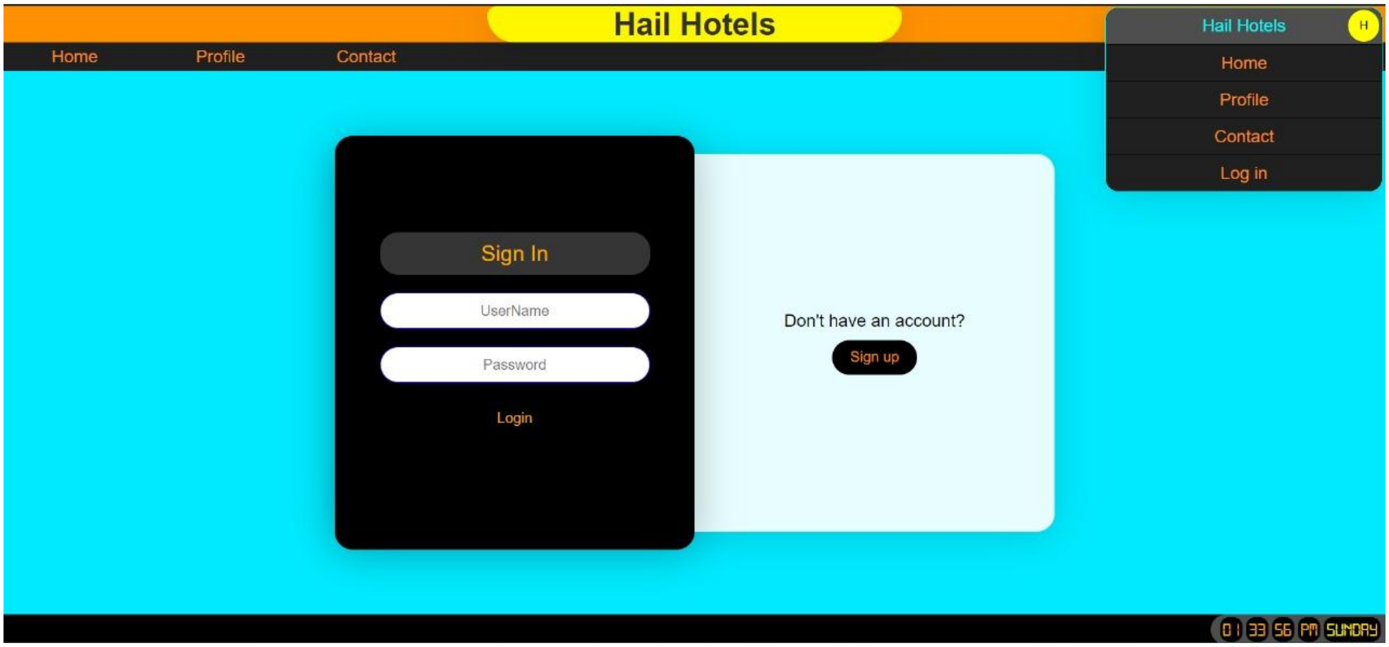
Let's go and book rooms, if your are logged in.

Sign up Page

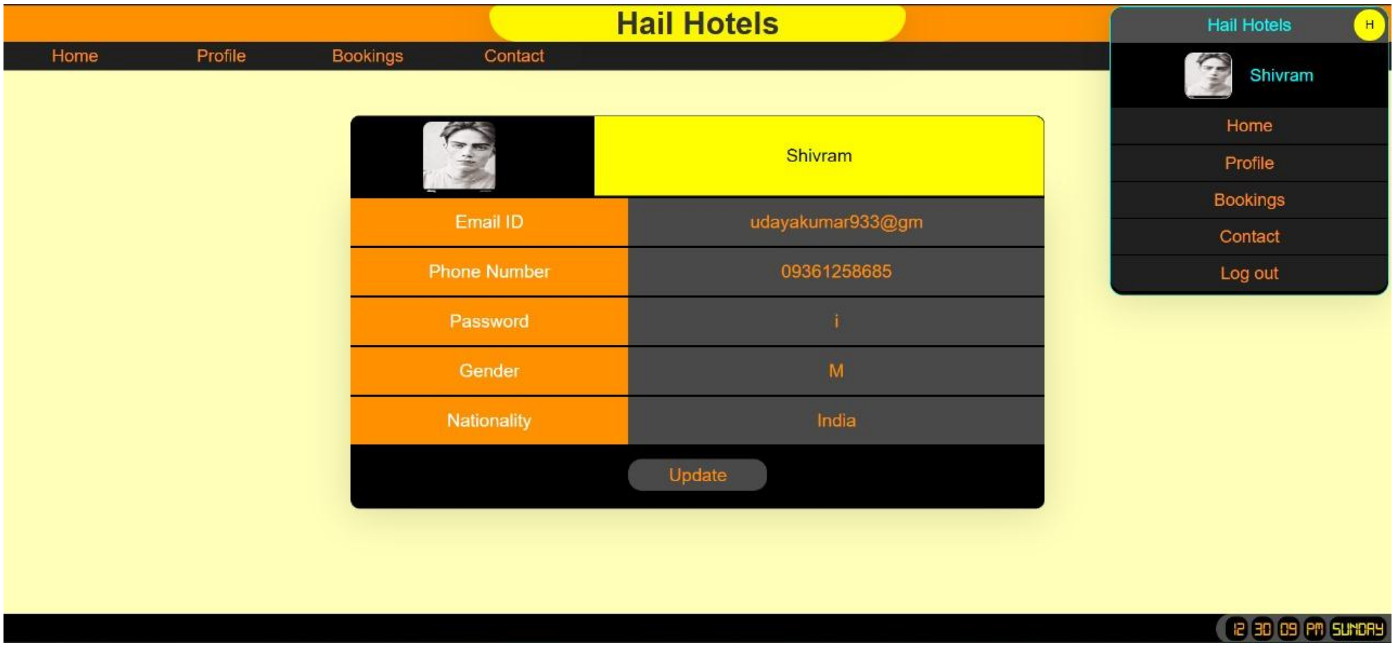


Already have an account?

Sign in Page

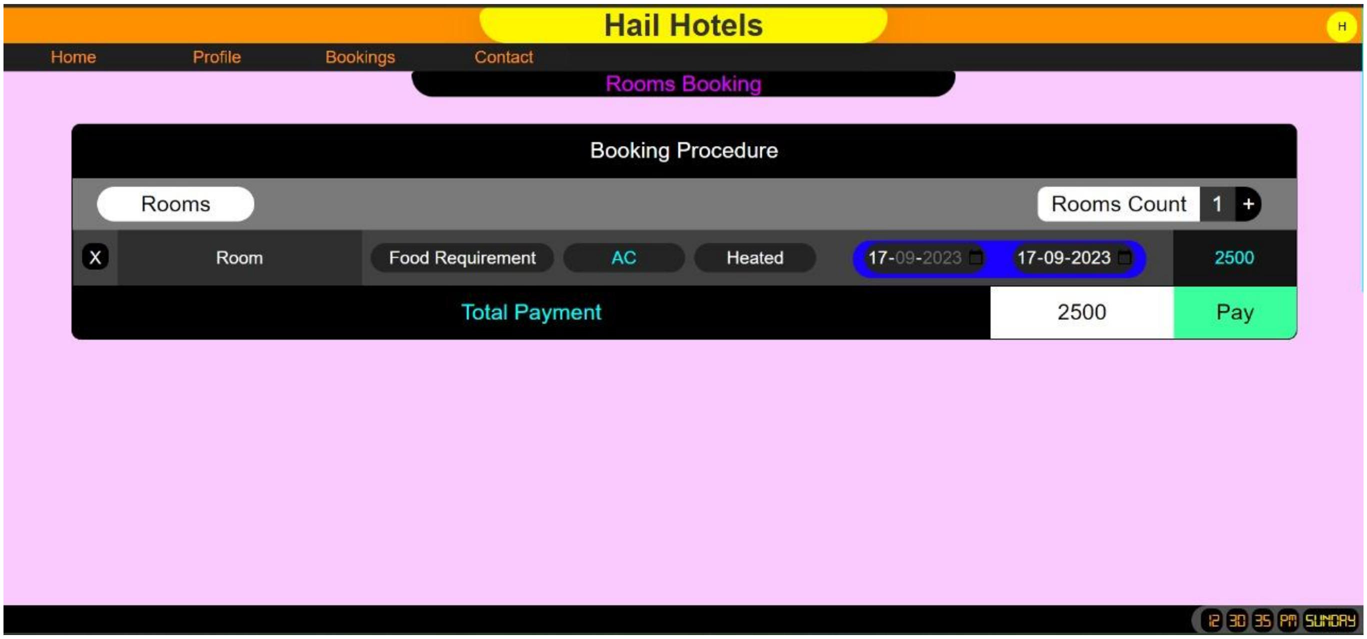


Profile Page



	Shivram
Email ID	udayakumar933@gm
Phone Number	09361258685
Password	i
Gender	M
Nationality	India

Booking Page



Hail Hotels

Rooms Booking

Booking Procedure

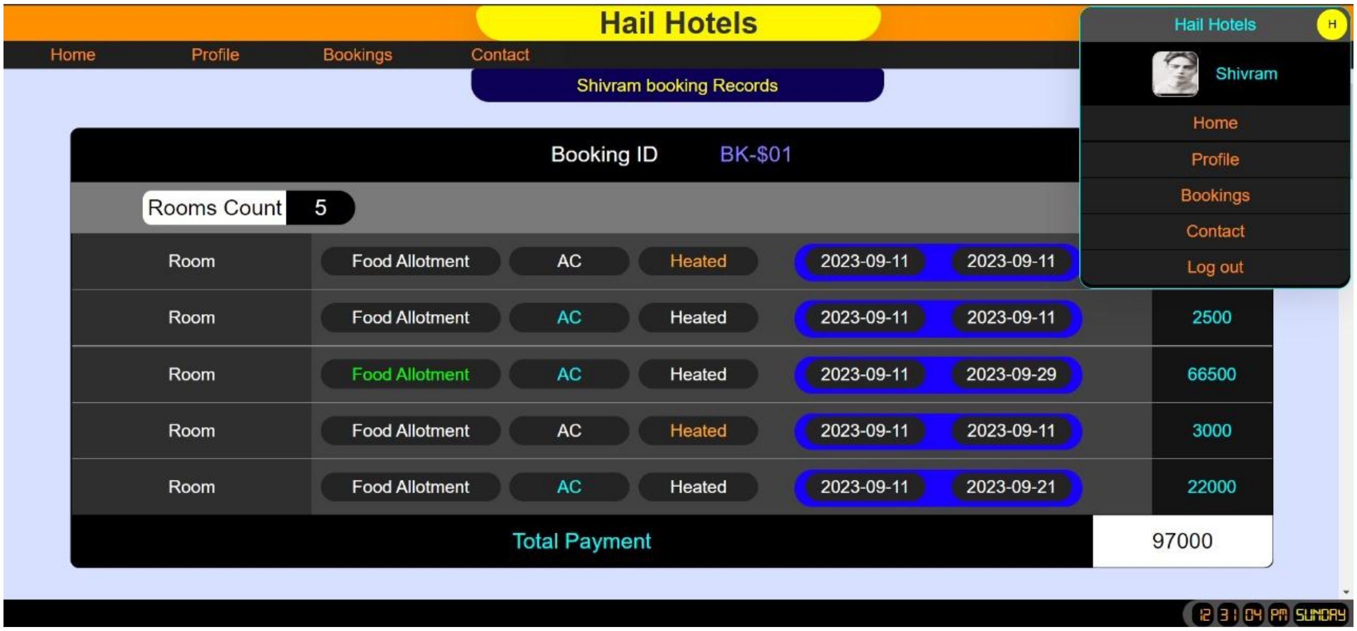
Rooms

Rooms Count 1 +

X	Room	Food Requirement	AC	Heated	17-09-2023	17-09-2023	2500

Total Payment 2500 Pay

View Bookings Page



Hail Hotels

Shivram booking Records

Booking ID BK-$01

Rooms Count 5

Room	Food Allotment	AC	Heated	2023-09-11	2023-09-11	
Room	Food Allotment	AC	Heated	2023-09-11	2023-09-11	2500
Room	Food Allotment	AC	Heated	2023-09-11	2023-09-29	66500
Room	Food Allotment	AC	Heated	2023-09-11	2023-09-11	3000
Room	Food Allotment	AC	Heated	2023-09-11	2023-09-21	22000

Total Payment 97000

Source Code

Home.jsp

```
<html>
   <script>
      var pg = "Login.jsp";
   </script>
   <%
      if(session.getAttribute("UserName")!=null)
      {
   %>
      <script>
         pg = "User.jsp";
      </script>
   <%
      }
   %>
   <link rel="stylesheet" href="CSS\Home.css">
   <body>
      <img src="CSS\RES\Media\HF.jpg" alt="Description of the image" id="bg">
      <div class="cont" id="sg">
            if you are new to our firm please register an user account, in our firm and
get hype offers.
         <button id="b1" class="bt">Sign up</button>
      </div>
      <div class="cont" id="lgin">
            if you have an User account, feel free to book rooms and services from our
hotel, after logging in
         <button id="b2" class="bt">Log in</button>
      </div>
      <div class="cont" id="bk">
         Let's go and book rooms, if your are logged in.
         <div>
```

```
                                                        <button      id="b3"
style="display:inline;"class="bt1">Book</button><button                id="b4"
style="display:inline;" class="bt1">Look up</button>
      </div>
    </div>
  </body>
  <script>
    var b1,b2,b3,b4;
    b1 = document.getElementById("b1");
    b2 = document.getElementById("b2");
    b3 = document.getElementById("b3");
    b4 = document.getElementById("b4");
    function lgin()
    {
                                              parent.window.location.href     =
"index.jsp?Tab="+encodeURIComponent(pg);
    }
    b1.onclick = () => {lgin()};
    b2.onclick = () => {lgin()};
    b3.onclick = () => { window.location.href = "forward.jsp?to=Book.jsp"; };
     b4.onclick = () => { window.location.href = "forward.jsp?to=Bookings.jsp";
};
    function Check_Parameters()
    {
      var urlparams = new URLSearchParams(window.location.search);
      var cmd = urlparams.get("cmd");
      var msg = urlparams.get("message");
      if(cmd!=null)
      {
        if(cmd=="Logout")
        {
          window.location.href = "reset.jsp";
        }
      }
      if(msg!=null)
```

```
            {
                var mesg = decodeURIComponent(msg);
                msg = document.createElement("div");
                msg.className="msg";
                msg.textContent=mesg;
                document.body.appendChild(msg);
                setTimeout(removemsg,3000);
            }
        }
        function removemsg()
        {
            var msgd = document.querySelector(".msg");
            if(msgd)
            {
                msgd.parentNode.removeChild(msgd);
            }
        }
        window.onload = Check_Parameters
    </script>
</html>
```

Book.jsp

```
<html>
    <link rel="stylesheet" href="CSS\Book.css">
    <h1> Rooms Booking </h1>
    <body>
        <div id="Bookp">Booking Procedure</div>
        <div id="RC">
                        <div class="cont" id="rc1">Rooms Count</div><div class="cont"
id="val">1</div><button class="bt" id="max">+</button>
        </div>
        <div id="container">
        </div>
        <div id="FN">
                <div id="TP">Total Payment</div><div id="payval">0</div><button id="pay">Pay</button>
        </div>
    </body>
    <script>
        var count =0,rms=[];
        var rc = document.getElementById("val");
        var cont = document.getElementById("container");
        var b1 = document.getElementById("max");
        var paym = document.getElementById("payval");
        var paybt = document.getElementById("pay");
        b1.onclick = () =>
        {
            var t = Number(rc.innerText);
            rc.innerText = t+1;
        };
        function Create_RM()
        {
            var rm = document.createElement("div");
            rm.innerText = "Room";
            rm.classList.add("rmn");
            return rm;
        }
        function Create_FA()
        {
            var fa = document.createElement("button");
            fa.innerText = "Food Allotment";
            fa.classList.add("rmsfa");
            return fa;
        }
        function Create_PM()
        {
```

```
    var pm = document.createElement("div");
    pm.innerText = "200";
    pm.classList.add("rmspm");
    pm.id = "rmspmid";
    return pm;
}
function Rooms()
{
    var s = Number(rc.innerText);
    for(i=count;i<s;i++)
    {
        var t = document.createElement("div");
        t.classList.add("rms");
        var bt = document.createElement("button");
        bt.classList.add("rmsbt");
        bt.innerText = "X";
        t.appendChild(bt);
        t.appendChild(Create_RM());
        t.appendChild(Create_FA());
        t.appendChild(Create_PM());
        t.addEventListener('click', function (event) {
            if (event.target.classList.contains('rmsfa')) {
                console.log(event.target.style.background);
                if(event.target.style.background == "rgb(0, 255, 0)")
                {
                    event.target.style.background = "#ff0000";
                    const pm = event.target.parentNode.querySelector("#rmspmid");
                    //console.log(pm);
                    if(pm)
                    {
                        console.log(pm.innerText);
                        pm.innerText = Number(pm.innerText)-50;
                    }
                }
                else
```

```
                        {
                            event.target.style.background = "#00ff00";
                            const pm = event.target.parentNode.querySelector("#rmspmid");
                            if(pm)
                            {
                                pm.innerText = Number(pm.innerText)+50;
                            }
                        }
                    }
                });
                cont.appendChild(t);
                rms.push(t);
            }
            count = s;
            let pmt =0;
            function add(i)
            {
                //console.log(i);
                pmt+=Number((i.querySelector("#rmspmid")).innerText);
            }
            rms.forEach(add);
            paym.innerText = pmt;
        }
        // Event Delegation Concept
            // -> Where Events can be added to a parent element instead of adding to
every single element. It refers to the process of using event propagation (bubbling)
to handle events at a higher level in the DOM than the element on which the event
originated.
        cont.addEventListener('click', function (event) {
            if (event.target.classList.contains('rmsbt')) {
                rc.innerText = Number(rc.innerText) - 1;

event.target.parentNode.parentNode.removeChild(event.target.parentNode);
                rms = rms.filter(item => item!=event.target.parentNode);
            }
```

```
    });

    paybt.onclick = function()
    {
      var rmfad = [];
      rmfad.push(Number(rc.innerText));
      rmfad.push(Number(paym.innerText));
      function anal(i)
      {

rmfad.push([Number((i.querySelector("#rmspmid")).innerText),(i.querySelector(".
rmsfa")).style.background == "rgb(0, 255, 0)" ? "1" : "0"]);
      }
      rms.forEach(anal);
      console.log(rmfad);
      console.log(rmfad[2]);
      var rmfadString = rmfad.join(',');
       var url = "http://localhost:8080/Hotel_Booking_System/Bookings_DB.jsp"
+
        "?Rooms=" + encodeURIComponent(rmfadString);
      // Redirect to the new URL
      window.location.href = url;
    }
    setInterval(Rooms,100);
  </script>
</html>
<!--
Try
  try to pass the javascript array in JSON format ans intrepret it from JSP.
-->
```

Login.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8" %>
<!DOCTYPE html>
<html>
  <link rel="stylesheet" type="text/css" href="CSS/Login.css">
  <body>
    <div class="container">
      <div class="bluebg">
        <div class="Box Signin">
          <h2>Already have an account?</h2>
          <button class="signin">Sign in</button>
        </div>
        <div class="Box Signup">
          <h2>Don't have an account?</h2>
          <button class="signup">Sign up</button>
        </div>
      </div>
      <div class="form">
        <div class="frm signinform">
          <form id="signinfrm" action="Signin.jsp">
            <h3>Sign In</h3>
                              <input id="f1" type="text" name="UserName"
placeholder="UserName">
                         <input id="f2" type="password" name="Password"
placeholder="Password">
            <input id="sn" type="submit" value="Login">
          </form>
        </div>
        <div class="frm signupform">
          <form id="signupfrm" action="Signup.jsp">
            <h3>Sign Up</h3>
                           <input id="f3" type="text" name = "UserName"
placeholder="UserName">
```

```
                        <input id="f4" type="text" name = "EmailID" placeholder="Email
ID">
                        <input id="f5" type="text" name = "Phone_Number"
placeholder="Phone Number">
                        <input id="f6" type="password" name = "Password"
placeholder="Password">
                        <input id="sup" type="submit" value="Sign up">
                    </form>
                </div>
            </div>
        </div>
        <script>
            const signin = document.querySelector(".signin");
            const signup = document.querySelector(".signup");
            const signinf = document.getElementById("signinfrm");
            const signupf = document.getElementById("signupfrm");
            const form = document.querySelector(".form");
            const body = document.querySelector('body');
            const f = []
            f.push(document.getElementById("f1"));
            f.push(document.getElementById("f2"));
            f.push(document.getElementById("f3"));
            f.push(document.getElementById("f4"));
            f.push(document.getElementById("f5"));
            f.push(document.getElementById("f6"));
            signup.onclick = function()
            {
                form.classList.add('active');
                body.classList.add('active');
            }
            signin.onclick = function()
            {
                form.classList.remove("active");
                body.classList.remove("active");
            }
        var sn = document.getElementById("sn"),sup = document.getElementById("sup");
        sn.onclick = function()
        {
          event.preventDefault();
          if(f[0].value!="" && f[1].value!="")
            signinf.submit();
          else
            alert("Please enter the values and then submit")
        }
        sup.onclick = function()
        {
          event.preventDefault();
          for(i=3;i<6;i++)
          {
            if(f[i].value == "")
            {
              alert("Please enter the values and then submit");
              return;
            }
          }
          signupf.submit();
        }
        function displayMessage()
        {
          var urlparams = new URLSearchParams(window.location.search);
          var message = urlparams.get("message");
          console.log("meow"+message);
          if(message!=null)
          {
            var mesg = decodeURIComponent(message);
            var msg = document.createElement("div");
            msg.className="msg";
            msg.textContent=mesg;
            document.body.appendChild(msg);
            setTimeout(removemsg,2000);
            console.log("meowwwqq")
                if(mesg == "Authentication Successfull" || mesg == "Registration Successfull")
            {
              setTimeout(function(){parent.reld("User.jsp");},2000);
              console.log("reload")
            }
          }
        }
        function removemsg()
        {
          var msgd = document.querySelector(".msg");
          if(msgd)
          {
            msgd.parentNode.removeChild(msgd);
          }
        }
        window.onload = displayMessage;
      </script>
    </body>
  </html>
```

<u>Bookings_mgmt.java</u>

```
package DB;
import java.io.IOException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.Map;
// Java Beans Concept
public class Bookings_Mgmt
{
    private String jdbcUrl,username,password,dbname,sql,name,em,ph,pswrd;
    private Connection connection;
    private PreparedStatement pst;
    private ResultSet rs;
    public boolean con = false,tsk=false,auth=false;
     public Bookings_Mgmt(String Username,String password,String jdbcurl,String
databasename)
    {
        this.jdbcUrl = jdbcurl; // Update with your MySQL database URL
        this.username = Username;
        this.password = password;
        this.dbname = databasename;
        this.jdbcUrl+=databasename;
        // Drivers Import
        try
        {
            Class.forName("com.mysql.cj.jdbc.Driver");
            this.connection = null;
            this.pst = null;
        }
        catch(Exception e)
        {
            System.out.println(e.getMessage());
        }
    }
    public Bookings_Mgmt()
    {
        this.jdbcUrl = "jdbc:mysql://localhost:3306/hail hotel management"; // Update with your MySQL database URL
        this.username = "root";
        this.password = "";
        this.DataBaseConnectionEstablishment();
    }
    public void DataBaseConnectionEstablishment()
    {
        try
        {
            this.connection = null;
            this.pst = null;
            Class.forName("com.mysql.cj.jdbc.Driver");
            connection = DriverManager.getConnection(jdbcUrl, username, password);
            this.con = true;
            // Drivers Import
        }
        catch(Exception e)
        {
            System.out.println(e.getMessage());
        }
    }
    public String Book(String name,int rmt,ArrayList<Integer> rmfa,int amt)
    {
        this.tsk = false;
        int cnt=0;
        if(this.con)
        {
```

```
try
{
   int rn=0;
   String clid="";
   sql = "SELECT COUNT(*) from bookings";
   pst = connection.prepareStatement(sql);
   rs = pst.executeQuery();
   if(rs.next())
      rn = rs.getInt(1);
   sql = "SELECT ClientID from clients where UserName = '"+name+"';";
   pst = connection.prepareStatement(sql);
   rs = pst.executeQuery();
   if(rs.next())
      clid = rs.getString(1);
   String bk = "BK-$"+rn+1;
   sql = "Insert INTO bookings VALUES(?,?,?)";
   pst = connection.prepareStatement(sql);
   int k=1;
   pst.setString(k++,clid);
   pst.setString(k++,bk);
   pst.setString(k++,String.valueOf(amt));
   pst.executeUpdate();
   String fas = "",r="";
   for(int i=0;i<rmt-1;i++)
   {
      fas+=String.valueOf(rmfa.get(i))+",";
      r+="R"+",";
   }
   fas+=String.valueOf(rmfa.get(rmt-1));
   r+="R";
   sql = "Insert into bookingdata VALUES(?,?,?)";
   pst = connection.prepareStatement(sql);
   k=1;
   pst.setString(k++,bk);
   pst.setString(k++,r);
```

```
                pst.setString(k++,fas);
                pst.executeUpdate();
                this.tsk = true;
                return "Success";
            }
            catch (Exception e)
            {
                return e.getMessage();
            }
        }
        return "Server Error, please try again later";
    }
    public ArrayList<String> Get_Bookings(String name)
    {
        ArrayList<String> al = new ArrayList<String>();
        if(this.con)
        {
            try
            {
                String clid="",bk="",t="";
                int amt=0;
                sql = "SELECT ClientID from clients where UserName = '"+name+"';";
                pst = connection.prepareStatement(sql);
                rs = pst.executeQuery();
                if(rs.next())
                    clid = rs.getString(1);
                   sql = "SELECT BookingsID,Amount from bookings where ClientID =
'"+clid+"';";
                pst = connection.prepareStatement(sql);
                rs = pst.executeQuery();
                if(rs.next())
                {
                    bk = rs.getString("BookingsID");
                    amt = rs.getInt("Amount");
                }
                if(bk!="")
                {
                    al.add(bk);
                    al.add(String.valueOf(amt));
                      sql = "SELECT food_allotment from bookingdata where BookingsID
= '"+bk+"';";
                    pst = connection.prepareStatement(sql);
                    rs = pst.executeQuery();
                    if(rs.next())
                    {
                        t = rs.getString("Food_Allotment");
                    }
                    al.add(t);
                }
                return al;
            }
            catch (Exception e)
            {
                al.add(e.getMessage());
                return al;
                // TODO: handle exception
            }
        }
        return null;
    }
    public static void main(String[] args)
    {
        DataBase_Mgmt db = new DataBase_Mgmt();
        System.out.println(db.Check_UP("i","i"));
    }
}
```

Database_Mgmt.java

```
package DB;
import java.io.*;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.Map;
// Java Beans Concept
public class DataBase_Mgmt
{
   private String jdbcUrl,username,password,dbname,sql,name,em,ph,pswrd;
   private Connection connection;
   private PreparedStatement pst;
   private ResultSet rs;
   public boolean con = false,tsk=false,auth=false;
                                          public String[] nms =
{"UserName","EmailID","Phone_Number","Password","Gender","Nationality"};
    public DataBase_Mgmt(String Username,String password,String jdbcurl,String
databasename)
   {
      this.jdbcUrl = jdbcurl; // Update with your MySQL database URL
      this.username = Username;
      this.password = password;
      this.dbname = databasename;
      this.jdbcUrl+=databasename;
      // Drivers Import
      try
      {
         Class.forName("com.mysql.cj.jdbc.Driver");
         this.connection = null;
```

```
            this.pst = null;
        }
        catch(Exception e)
        {
            System.out.println(e.getMessage());
        }
    }
    public DataBase_Mgmt()
    {
        this.jdbcUrl = "jdbc:mysql://localhost:3306/hail hotel management"; // Update with your MySQL database URL
        this.username = "root";
        this.password = "";
        this.DataBaseConnectionEstablishment();
    }
    public void DataBaseConnectionEstablishment()
    {
        try
        {
            this.connection = null;
            this.pst = null;
            Class.forName("com.mysql.cj.jdbc.Driver");
            connection = DriverManager.getConnection(jdbcUrl, username, password);
            this.con = true;
            // Drivers Import
        }
        catch(Exception e)
        {
            System.out.println(e.getMessage());
        }
    }
    public String Check_UP(String un,String ps)
    {
        this.auth = false;
        if(this.con)
```

```
{
   boolean[] vals = new boolean[2];
   vals[0]=false;
   vals[1]=false;
   HashMap<String,String> mp = null;
   mp = new HashMap<String,String>();
   // SQL query to insert a record
   sql = "SELECT UserName,Password from clients;";
   try
   {
      pst = connection.prepareStatement(sql);
      rs = pst.executeQuery();
      while(rs.next())
      {
         mp.put(rs.getString("UserName"),rs.getString("Password"));
      }

      for(Map.Entry<String,String> ent : mp.entrySet())
      {
         //s+=ent.getKey()+","+ent.getValue()+",";
         if(ent.getKey().equals(un))
         {
            vals[0] = true;
            if(ent.getValue().equals(ps))
            {
               vals[1]= true;
            }
         }
      }
      if(vals[0])
      {
         if(vals[1])
         {
            this.auth=true;
            return "Authentication Successfull";
```

```
                    }
                    else
                        return "Password incorrect, please check and try again";
                }
                else
                {
                              return "Authentication failed, please check the User name and
password ";
                }
            }
            catch(Exception e)
            {
                System.out.println(e.getMessage());
                return "Server Error, please try again later";
            }
        }
        else
            return "Server Error, please try again later";
    }
    public String Add_User(String name,String em,String phn,String pswrd)
    {
        this.tsk = false;
        int cnt=0;
        if(this.con)
        {
            // SQL query to insert a record
            try
            {
                String sql = "SELECT COUNT(*) FROM clients";
                pst = connection.prepareStatement(sql);
                rs = pst.executeQuery();
                if(rs.next())
                {
                    cnt = rs.getInt(1);
                }
                sql = "INSERT INTO clients
(ClientID,UserName,EmailID,Phone_Number,Password,Gender,Nationality)
VALUES (?,?,?,?,?,?,?)";
                pst = connection.prepareStatement(sql);

                int k=1;
                pst.setString(k++, "ClientID-$"+(cnt+1));
                pst.setString(k++, name);
                pst.setString(k++, em);
                pst.setString(k++,phn);
                pst.setString(k++, pswrd);
                pst.setString(k++, "N");
                pst.setString(k++, "NIL");
                // Execute the query
                pst.executeUpdate();
                this.tsk = true;
                return "Registration Successfull";
            }
            catch(Exception e)
            {
                String message = e.getMessage();
                //out.println(e.getMessage());
                if(e instanceof java.sql.SQLIntegrityConstraintViolationException)
                {
                    if(message.contains("UserName"))
                    {
                        return "User name already exists";
                    }
                    if(message.contains("Phone_Number"))
                    {
                        return "Phone Number already exists";
                    }
                    else if(message.contains("EmailID"))
                    {
                        return "EmailID already exists";
```

```
                }
                else
                {
                    return "Password already exists";
                }
            }
        }
    }
    return "Server Error, please try again later";
}
public ArrayList<String> Get_User(String un)
{
    ArrayList<String> al = new ArrayList<String>();
    try
    {
        sql = "Select * from clients where UserName=?";
        pst = connection.prepareStatement(sql);
        pst.setString(1,un);
        ResultSet rs = pst.executeQuery();
        if(rs.next())
        {
            for(int i=0;i<6;i++)
                al.add(rs.getString(this.nms[i]));
            return al;
        }
    }
    catch (Exception e)
    {
        System.out.println(e.getMessage());
    }
    return null;
}

public String Update_User(String name,String pms,String[] vals)
{
    this.tsk = false;
    int cnt=0,i=0,k=0;
    String Error="",clid=null;
            String sql = "Select ClientID from clients where UserName =
'"+name+"';",tsql;
    try
    {
      pst = connection.prepareStatement(sql);
      rs = pst.executeQuery();
      if(rs.next())
        clid = rs.getString("ClientID");
    }
    catch(Exception e){}
    if(this.con)
    {
      sql = "UPDATE clients set ";
      // SQL query to insert a record
      for(i=0;i<6;i++)
      {
        try
        {
          if(pms.charAt(0) == '1')
          {
            tsql = sql;
            tsql = tsql+" "+this.nms[i]+" = "+"'"+vals[k++]+"' where ClientID =
'"+clid+"';";
            pst = connection.prepareStatement(tsql);
            pst.executeUpdate();
          }
        }
        catch(Exception e)
        {
          String message = e.getMessage();
          //out.println(e.getMessage());
          if(e instanceof java.sql.SQLIntegrityConstraintViolationException)
```

```
                {
                    if(message.contains("UserName"))
                    {
                        Error+="User name already exists";
                    }
                    if(message.contains("Phone_Number"))
                    {
                        Error+="Phone Number already exists";
                    }
                    else if(message.contains("EmailID"))
                    {
                        Error+="EmailID already exists";
                    }
                    else if(message.contains("Password"))
                    {
                        Error+="Password already exists";
                    }
                }
            }
        }
        this.tsk = true;
    }
    else
        Error+="Server Error, please try again later";
    if(Error!="")
        return Error;
    else
        return "Success";
}
public String Update_User_profile(String name,File f,String fn)
{
    try
    {
        byte[] imageData = readFileToByteArray(f);
        // SQL statement to insert the image into the database
```

```
                      String sql = "UPDATE clients set FN = ?,Profile_Image =? where
UserName = ?";

            pst = connection.prepareStatement(sql);
                // Bind the image data to the parameter
            pst.setString(1,fn);
            pst.setBytes(2, imageData);
            pst.setString(3, name);

                // Execute the SQL statement
            int rowsAffected = pst.executeUpdate();
            if (rowsAffected > 0)
            {
                return"Image inserted successfully.";
            }
            else
            {
                return "Failed to insert image.";
            }
        }
        catch(Exception e)
        {
            return e.getMessage();
        }
    }
    public File Get_Profile(String name)
    {
        int imageId = 1; // Replace with the actual image_id you want to retrieve
        try
        {
                      String sql = "SELECT Profile_Image,FN FROM clients WHERE
UserName = ?";
            pst = connection.prepareStatement(sql);
            pst.setString(1, name);
            rs = pst.executeQuery();
```

```
      if (rs.next())
      {
        // Retrieve the image data and file name
        byte[] imageData = rs.getBytes("Profile_Image");
        String fileName = rs.getString("FN");
        // Determine the file extension based on the file name
        String fileExtension = getFileExtension(fileName);
        // Create a File object to save the image
        File outputFile = new File("temp." + fileExtension);
        // Write the binary data to the file
        try (FileOutputStream fos = new FileOutputStream(outputFile)) {
          fos.write(imageData);
          System.out.println("Image retrieved and saved as: " + outputFile.getAbsolutePath());
        } catch (IOException e) {
          e.printStackTrace();
        }
        return outputFile;
      } else {
        System.err.println("Image not found.");
        return null;
      }
    }
    catch(Exception e)
    {
      return null;
    }
  }
  private static String getFileExtension(String fileName) {
    int lastDotIndex = fileName.lastIndexOf(".");
    if (lastDotIndex != -1) {
      return fileName.substring(lastDotIndex + 1);
    }
    return "";
  }
```

```
    private static byte[] readFileToByteArray(File file) throws IOException {
        FileInputStream fis = new FileInputStream(file);
        byte[] data = new byte[(int) file.length()];
        fis.read(data);
        fis.close();
        return data;
    }
    public static void main(String[] args)
    {
        DataBase_Mgmt db = new DataBase_Mgmt();
        System.out.println(db.Check_UP("i","i"));
    }
}
```

PrfImg_Upl.java

```
package WebApp;
import java.io.*;
import DB.*;
import javax.servlet.ServletException;
import javax.servlet.http.*;
import com.oreilly.servlet.MultipartRequest;

// Class
// Extending HttpServlet class
public class PrfImg_Upl extends HttpServlet {

    // Method
    private String getFileExtension(String fileName) {
        int lastDotIndex = fileName.lastIndexOf(".");
        if (lastDotIndex != -1) {
            return fileName.substring(lastDotIndex);
        }
        return "";
    }
    // To handle request response mechanism
    public void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {

        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
                                                                        String        directoryPath        =
"C:\\Applications\\Application_Folders\\Servers\\Xampp\\tomcat\\webapps\\Hotel
_Booking_System\\CSS\\RES\\Media\\temp";

        String Name = request.getParameter("on");
        // Create a File object for the directory
        File directory = new File(directoryPath);
        // Check if the directory exists
        if (directory.exists() && directory.isDirectory()) {
            // List all files in the directory
            File[] files = directory.listFiles();
            // Delete each file
            for (File file : files) {
                if (file.isFile()) {
                    file.delete();
                }
            }
        }
        MultipartRequest m = new MultipartRequest(request,directoryPath );
        // Get the original file name
        String originalFileName = m.getOriginalFileName("file");
        // Generate a new file name (you can implement your own logic)
        String newFileName = "temp";
        String fileExtension = getFileExtension(originalFileName);
        newFileName = newFileName+fileExtension;
        // Get the file object
        File uploadedFile = m.getFile("file");
        // Rename the file
        File renamedFile = new File(uploadedFile.getParentFile(), newFileName);
        uploadedFile.renameTo(renamedFile);
        response.getWriter().append(newFileName);

        DataBase_Mgmt db = new DataBase_Mgmt();
        String msg = db.Update_User_profile(Name, renamedFile,newFileName);
        // response.getWriter().append(msg);
        //response.redirect("Hotel_Booking_System\\index.jsp?Tab=User.jsp");
    }
}
```

CONCLUSION

The Hotel Booking System provides an environment for users to book hotel rooms, perform booking activities, and manage personal account at Hail Hotel with a Web browser. The system uses XAMPP Server running on Windows platform. The database server is MySQL. To implement the system, we have used JavaScript, HTML, CSS and JSP. All dynamic contents are handled by JSP.

Persistent data are saved in the database. Hotel Booking System is a user-friendly and easy-to-use system of a Web-based application. Everyone who knows how to use a Web browser can register and then login to book a room, change booking details, cancel booking, and view or modify personal profile online. It is easy and fast to make a reservation.

There can still be improvements for our system and it can be enhanced according to the user needs.

REFERENCES

https://www.javatpoint.com/jsp-tutorial

https://www.w3schools.com/js/

https://www.w3schools.com/html/

https://www.javatpoint.com/css-tutorial

https://www.roseindia.net/jsp/connect-jsp-mysql.shtml

https://people.utm.my/shaharil/mysql-and-phpmyadmin-xampp/