**ONLINE HOTEL MANAGEMENT SYSTEM**

Course of

**EXL- Certified Software Test Engineer**

Submitted By

**ROSHNI SUNIL POKHARKAR. S**

**BATCH : 2022 – 7669 (Pune – Chinchwad)**

**ENROLLMENT NUMBER : EBE0722630305**

Under the Guidance of

Ms. A. MOHANA PRIYA Centre Trainer



**JANUARY -2023**

**CONTENT**

|  |  |
| --- | --- |
| **S.NO** | **INDEX** |
| **1.** | INTRODUCTION |
| **2.** | SCOPE AND OBJECTIVES |
| **3.** | SYSTEM SPECIFICATION |
| **4.** | SYSTEM DESIGN |
| **5.** | IMPLEMENTATION |
| **6.** | SYSTEM TESTING |
| **7.** | CONCLUSION |

# 1.INTRODUCTION

## 1.1 Project Outline ( Purpose And Goal )

The aim of this Online Hotel Management System project is to build a system that will able to automate many operations in a hotel. Modern day hotels aim to create a user friendly atmosphere with the availability of concierges who remember frequent visitors and making it possible to call and make reservations. While such hotels are extremely expensive, such a service can also be provided in a cost – effective manner with the use of computers.

The process begins when a customer books a room, the booking can be placed online or through an Interactive Voice Response System (IVRS). The system will remember client preferences and can provide options accordingly. Once the room is booked, the visitor only needs to turn up and present identification. The system will also be able to send messages to the visitor’s phone to remind them of restaurants they may have visited in the past and other options. The system can accept bookings for restaurants present in the hotel. Things – to – do around the locality can be advertised as well through messages.

Every hotel has an intricate system of cleaning and replacement of room sheets, etc. The scheduling of the same can be completely managed by this system and it can also take into account guest preferences. The guest can use the IVRS at the hotel to inform the system about the best time to clean the room.

When the guest vacates the room, the system will create the bill, schedule clean up and allot the next guest. The customer can keep their credit card linked with the system to avoid the hassle of continually making payments. All payments in the hotel can be automatically credited from this account.

# 

# 2.SCOPE AND OBJECTIVES

The Online Hotel Management System depends on many highly trained individuals to take care of guests. Such a system is prone to error and delays. Visitors may not have a consistent experience and have to be bothered about remembering mundane things such as payments and restaurants. A lot of paperwork is generated and it is difficult for a supervisor to go through all these documents. Allocation of rooms based on expected vacancies is also difficult as it requires extra work on the part of the employees. The costs of running such a hotel is also great.

The Online Hotel Management System is much more efficient in terms of human and monetary resources. The number of staff to ensure successful implementation is reduced. Lead time of service is reduced and it is easy to predict room availabilities. Electronic documentation of hotel operations help management in keeping the hotel in check. A visitor can be sure of consistent service even across other hotels of the same chain.

At check out the user can enter a review which will be uploaded to the website, the system will collect user reviews from other sites and from past guests, making them available to future visitors. The Online Hotel Management System will make it easy for higher level management to easily review the operation of the hotel.

# 3. SYSTEM SPECIFICATION

## 3.1 Software Requirements

Processors will continue to get faster, smaller and cheaper, whereas memory will continue to get faster, larger and cheaper. The trend except to have a reasonable memory to a powerful processor.

* Windows XP
* PHP 5
* MS SQL Server 2005 Express
* CSS
* Apache Server

## 3.2 Hardware Requirements

When an application project is considered the three basic software requirements are the platform in which the project is developed, the front-end tool that provides the interaction with the users and the back-end tool that stores the data.

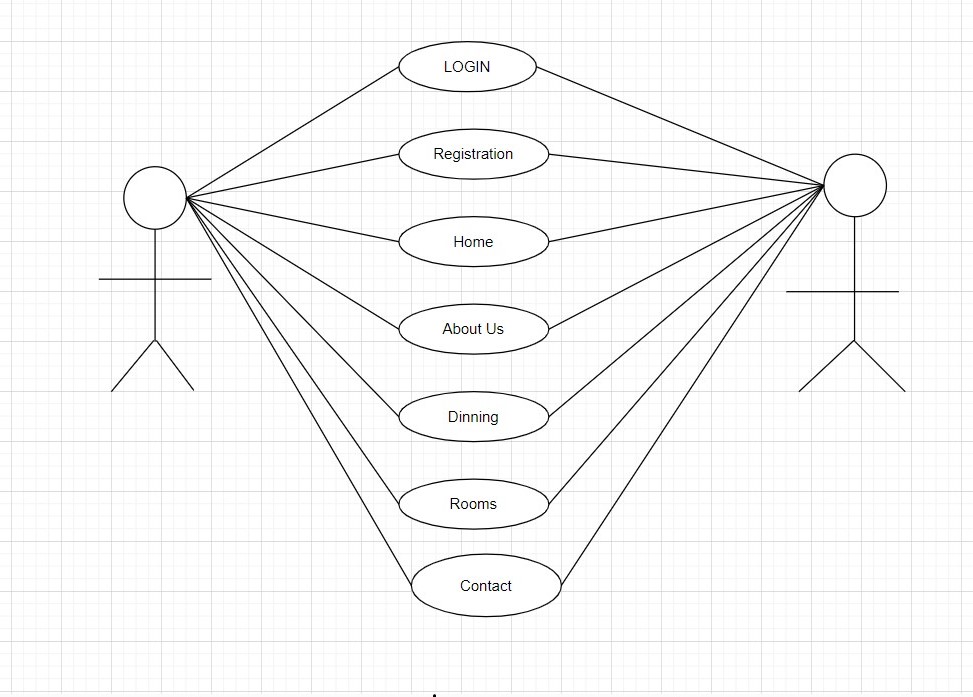
* Hard Disk – 2 GB
* RAM – 1 GB

**3.3 Feature of Learning in Online Hotel Management system.**

1. Automating the online system.
2. Reducing time taken between student and staff to communicate
3. Making the student easily accessible the portal
4. Speeding up learning.
5. Reduce time in the automation system by using in your place.
6. Reduce the cost of money occupied the whole year being used.

# 4.SYSTEM DESIGN

## 4.1 Use case Diagram



**4.1.1 Use case Descriptions**

A use case is a methodology used in system analysis to identify, clarify and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal.

The method creates a document that describes all the steps taken by a user to complete an activity.

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **USECASE** | **ACTOR** | **DESCRIPTION** |
| 1 | Login | Administrator, Customer (user) and Visitor. | Enter user name, and password |
| 2 | Registration | Administrator, Customer (user) and Visitor. | First name, last name, email id, password. |
| 3 | Contact us | Administrator, Customer (user) and Visitor. | First name, last name, email id, password, subject and comment box. |

**4.1.2 Use case Explanations**

In this use case diagram they mention the step process

### 4.1.3 Use case 001: login

INTORDUCTION: this use case outlines the step that need to be followed in order to login into the system.

**Start**

ACTOR: Administrator, Customer (user)

PRE-CONDITION: user has to have a valid credential

POST-CONDITION: the system display the relevant page

BASIC FLOW: the user enter the user name and password

**Scenario:**

|  |  |
| --- | --- |
| **ACTOR** | **SOFTWARE REACTION** |
| User name password | If it is valid user can enter into system If it is invalid user it indicates it is wrong user name or password |

ALTERNATE FLOW: the user can enter the reg no

SPECIAL REQUIREMENTS: None

ASSOCIATED USE CASE(S): None

### 4.1.4 Use case 002: Registration

INTORDUCTION: this use case outlines the step that need to be followed in order to register form

ACTOR: Customer (user) and admin.

PRE-CONDITION: user has to have a valid credential

POST-CONDITION: the system display the relevant page

BASIC FLOW: the system confirms that the entered details is correct or not.

**Scenario:**

|  |  |
| --- | --- |
| **ACTOR** | **SOFTWARE REACTION** |
| First name  Last name  Email id  Password | The system verifies that the above items has been filled out  If any data is missing, the system warns the user and the steps continues with software reaction  If all data has been entered the system ask the user to view the result.  If the user indicate they want to end the scenario here  And If the user indicate that their marks are not correct the scenario continues with software reaction. |

ALTERNATE FLOW: None

SPECIAL REQUIREMENTS: None

ASSOCIATED USE CASE(S): None

### 4.1.5 Use case 003: Contact us

INTORDUCTION: this use case outlines the step that need to be followed in order to fill out the data

ACTOR: Customer (user) and admin

PRE-CONDITION: user has to have a valid credential

POST-CONDITION: the system display the relevant page

BASIC FOLW: the user enter the details **Scenario:**

|  |  |
| --- | --- |
| **ACTOR** | **SOFTWARE REACTION** |
| First name  Last name  Email id  Password  Subject  Comment | The system verifies that the above items has been filled out  If any data is missing, the system warns the user and the steps continues with software reaction  If all data has been entered the system ask the user to view the result.  If the user indicate they want to end the scenario here  And If the user indicate that their marks are not correct the scenario continues with software reaction. |

ALTERNATE FLOW: click the help option

SPECIAL REQUIREMENTS: None

ASSOCIATED USE CASE(S): None

**2. Activity Diagram**

An activity diagram **shows Online and software processes as a progression of actions .**

User

ADMIN

LOGIN

Update Booking Details

View

Check For

Authentication

View Rooms

Details

Select

Rooms

Luxuary Roo Room

Room available

or not

If available

Fill Registration

Form

Sumnit Form

Allow

Check For

Authentication

Confirm

Registration

**3.Class Diagram**

The purpose of class diagram is to model the static view of an application. Class diagrams are the only diagrams which can be directly mapped with object-oriented languages and thus widely used at the time of construction.

**User**

Name: string

Location: string

Login ()

Logout ()

**Administrator**

Date: date

Number: string

**Admit ()**

**Close ()**

**4 Entity Relational Diagram (ER)**

LOGIN

REGISTER

TO

EMAIL ID

LOGIN

PASSWORD

FIRST

NAME

CONFORM

PASSWOR

LAST

NAME

EMAIL.ID

HAS

CONTACT US

FIRST

NAME

LAST NAME

EMAIL.ID

SELECTED

HOTEL ROOMS

COMMENT

**5 Data Flow Diagram**

**A** data flow diagram (DFD) is a graphical model the show all of the main requirement

For an information system in a diagram : input and output, process, and data storage. A DFD

Describe what data flows rather than how it is processed. Everyone working on a development project can see all aspects of the system working

Together at once with DFD. That is one of the reason for its popularity, the DFD is also easy To read because it is graphical model. The DFD is mainly used during problem analysis. End

DFD with minimal training.

### 4.5.1 DFD Symbol

1. Process

1. Data Flow

1. External Entity

|  |  |
| --- | --- |
|  |  |
|  |  |

1. Data Store

**4.5.2 Data Flow Diagram**

**Level-0:**

LOGIN

USER/CUSTOMERS

CUSTOMERS

DATABASE

**Level-1:**

SYSTEM

USER/CUSTOMERS

CUSTOMERS

DATABASE

LOGIN

REGISTER

CONTACT

LOGIN DETAILS

TABLE

REGISTER DETAILS

TABLE

CONTACT DETAILS

TABLE

#### 4.6 Table Structure

##### **4.6.1 Login**

Table name: login

Primary key: email id

|  |  |  |
| --- | --- | --- |
| **COLUMN NAME** | **DATA TYPE** | **DESCRIPTION** |
| EMAIL ID | TEXT | EMAIL ID |
| PASSWORD | NUMBER | PASSWORD |

##### **4.6.2 Register**

Table Name: Register

Primary key: email id

|  |  |  |
| --- | --- | --- |
| **COLUMN NAME** | **DATA TYPE** | **DESCRIPTION** |
| FIRST NAME | TEXT | FIRST NAME |
| LAST NAME | TEXT | LAST NAME |
| EMAIL ID | TEXT | EMAIL ID |
| CONFORM PASSWORD | NUMBER | CONFORM PASSWORD |

##### **4.6.3Contact us**

Table Name: Contact us

Primary key: email id

Foreign Key: Comment box

|  |  |  |
| --- | --- | --- |
| **COLUMN NAME** | **DATA TYPE** | **DESCRIPTION** |
| FIRST NAME | TEXT | FIRST NAME |
| LAST NAME | TEXT | LAST NAME |
| EMAIL ID | TEXT | EMAIL ID |
| COMMENT BOX | TEXT | COMMENT BOX |

**5. IMPLEMENTATION**

## 5.1 Modules

An admin can create accounts for the user and present reports to management. Details about employees and cleaning schedules are also updated by the admin.

**5.1.1 Home**

“Online Hotel Management System” you can learn for your future start learning.... ”

This is my quotation mentioned in my home page this will attract the learners quickly and I have mentioned the website link to check out the strategies to attract the student is necessary for the survival and growth of your business.

**5.1.5 Contact us**

Hotel management system allows students to share their ideas on various subjects with each other **”**.customer online discussions typically motivate deeper understanding as well as yield interesting personal applications of course concepts and theories.

## 5.2 (frontend) Webpage

I have used HTML , CSS , JAVASCRIPT to develop the frontend design

### 5.2.1 HTML (Hypertext Markup Language)

**HTML** stands for Hyper Text Markup Language. It is used to design web pages using a markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between web pages. A markup language is used to define the text document within the tag which defines the structure of web pages.

language is used to annotate (make notes for the computer) text so that a machine can understand it and manipulate text accordingly. Most markup languages (e.g. HTML) are human-readable. The language uses tags to define what manipulation has to be done on the text.

1.HTML Modules(.html) 2.PHP Modules (.php)

**5.2.2 HTML page structure:**

The basic structure of an HTML page is laid out below. It contains the essential building-block elements (i.e. doctype declaration, HTML, head, title, and body elements) upon which all web pages are created.

* It is easy to learn and easy to use.
* It is platform-independent.
* Images, videos, and audio can be added to a web page.
* Hypertext can be added to the text.
* It is a markup language.

### 5.2.3 CSS ( Cascading Style Sheet )

**C**ascading **S**tyle **S**heets, fondly referred to as **CSS**, is a simply designed language intended to simplify the process of making web pages presentable. CSS allows you to apply styles to web pages. More importantly, CSS enables you to do this independent of the HTML that makes up each web page.

CSS lets developers and designers define how it behaves, including how elements are positioned in the browser. While html uses tags, css uses rulesets. CSS is easy to learn and understand, but it provides powerful control over the presentation of an HTML document.

### 5.2.4 types

* **CSS saves time:** You can write CSS once and reuse the same sheet in multiple HTML pages.
* **Easy Maintenance:** To make a global change simply change the style, and all elements in all the webpages will be updated automatically.
* **Search Engines:** CSS is considered a clean codingtechnique, which means search engines won’t have to struggle to “read” its content.
* **Superior styles to HTML:** CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.
* **Offline Browsing:** CSS can store web applications locally with the help of an offline cache. Using this we can view offline websites.

#### 5.2.5 JavaScript

**JavaScript** is a lightweight, cross-platform, and interpreted compiled programming language which is also known as the scripting language for webpages. It is well-known for the development of web pages, many non-browser environments also use it.

JavaScript can be used for CLIENT-SIDE developments as well as SERVER\_SIDE developments. Javascript is both imperative and declarative type of language. JavaScript contains a standard library of objects, like array, date and math, and a core set of language elements like **operators**, **control structures**, and **statements**.

**5.2.6 Client-side:**

It supplies objects to control a browser and its Document Object Model (DOM).

Like if client-side extensions allow an application to place elements on an HTML form and respond to user events such as **mouse clicks**, **form input**, and **page navigation**.

**5.2.7 Server-side:**

It supplies objects relevant to running JavaScript on a server. Like if the server-side extensions allow an application to communicate with a database, it provide continuity of information from one invocation to another of the application, or perform file manipulations on a server.

### 5.2.8 Imperative language –

In this type of language we are mostly concern about how it is to be done . It simply control the flow of computation . The procedural programming approach , object, oriented approach comes under this like async await we are thinking what it is to be done further after async call.

**5.2.9 Declarative programming-** In this type of language we are concern about how it is to be done basically here logical computation require .Here main goal is to describe the desired result without direct dictation on how to get it like arrow function do .

### 5.3 Coding

<html>

<head>

<title> Roshni </title>

<link rel="stylesheet" type="text/css" href="css/style.css">

<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/@fortawesome/fontawesomefree@6.1.2/css/fontawesome.min.css">

<center>

<br>

<br>

<br>

<br>

<div class="title">

<h1>WELLCOME TO HOTEL MANAGEMENT SYSTEM </h1>

</center>

</div>

<br>

<br>

<br>

<br>

<br>

<style>

\* { padding: 0; margin: 0; box-sizing: border-box;

} body

{ background-image: url(https://c4.wallpaperflare.com/wallpaper/624/380/1000/life-resorthotel-resort-hotel-wallpaper-preview.jpg); background-size: cover; background-position: center; font-family: sans-serif;

} section

{

padding:2rem 7%;

}

.heading { text-align: center; color: #ff1ac6; text-transform: uppercase; padding-bottom: 3rem; font-size: 1rem; }

.heading span { color:var(--main-color); text-transform: uppercase;

}

.menu-bar { background: rgb(0,0,30); text-align: center; }

.menu-bar ul { display: inline-flex; list-style: none; color: #fff; }

.menu-bar ul li { width: 120px; margin: 15px; padding: 15px; }

.menu-bar ul li a { text-decoration: none; color: #fff;

}

.active, .menu-bar ul li:hover

{ background: #ff1ac6; border-radius: 100px; }

#login-form { display: none;

/\* this block is for hiding the form \*/

}

.form-box

{ width:650px; height:480px; position:relative; margin:2% auto; background:rgba(0,0,0,0.3); padding:10px; overflow:hidden;

/\*creating te form with width and height\*/

}

.button-box { width:220px; margin:35px auto; position:relative; box-shadow: 0 0 20px 9px #ff1ac6; border-radius: 30px; }

.toggle-btn { padding:10px 30px; cursor:pointer; background:transparent; border:0;

outline: none; position: relative; }

#btn { top: 0; left: 0; position: absolute; width: 110px; height: 100%; background:#ff1ac6; border-radius: 30px; transition: .5s; }

.input-group-login { top: 150px; position:absolute; width:280px; transition:.3s; }

.input-group-register { top: 120px; position:absolute; width:280px; transition:.3s; }

.input-field { width: 100%; padding:10px 0; margin:5px 0; border-left:0; border-top:0; border-right:0; border-bottom: 1px solid #999; ontline:none; background: transparent;

}

.submit-btn { width: 85%; padding: 10px 35px; cursor: pointer; display: block; margin: auto; background: dodgerblue; border: 0; outline:none; border-radius: 30px; }

.check-box { margin: 30px 10px 34px 0; } span { color:#ff1ac6; font-size:12px; bottom:68px; position:absolute; }

#login { left:40px; }

#login input { color:white; font-size:15; }

#register { left:340px; }

#register input { color:white; font-size: 15; }

.about .row { display:flex; align-items: center; background:var(--blue); flex-wrap: wrap; }

.about .row .image { flex: 1 1 45rem; }

.about .row .image img { width: 400%;

}

.about .row .content

{ flex:1 1 45rem; padding:2rem;

}

.about .row .content h2

{ font-size:2rem; color: #ff1ac6;

}

.about .row .content p

{ font-size:1rem; color: #ff1ac6; padding:0.5rem 0; line-height: 1; } menu.box-container

{ display: grid; grid-template-columns: repeat(auto-fit, minmax(30rem, 1fr)); gap: 1.5rem; } menu.box-container .box

{ padding:5rem; text-align: center; border:var(--border); } menu.box-container .box img

{ height: 10rem;

}

menu.box-container .box h2

{ color: #fff; font-size: 2rem; padding:1rem 0; } menu.box-container .box .price

{ color: #fff font-size: 2.5rem; padding: 5rem 0; } menu.box-container .box .price span

{

color: #fff; font-size: 1.5rem; text-decoration: line-through;

} menu.box-container .box:hover

{ background:#fff; } menu.box-container .box:hover>\*

{ color:var(--blue); }

</style>

</head>

<body>

<div class="Menu-bar">

<ul>

<li class="active">

<a href="#">home</a>

</li>

<li>

<a href="#">About Us</a>

</li>

<li>

<a href="#">Dining</a>

</li>

<li>

<a href="#">Rooms</a>

</li>

<li>

<a href="#">Contact</a>

</li>

<li>

<button class='Loginbtn'onclick="document.getElementById('Loginform').style.display='block'"style="width:auto;">Login</button>

</li>

</ul>

</div>

<div id='Login-form'class='Login-page'>

<div class="form-box">

<div class='button-box'>

<div id='btn'>

</div>

<button type='button'onclick='Login()'class='toggle-btn'>Login</button>

<button type='button'onclick='register()'class='toggle-btn'>register</button>

</div>

<!--creating login form-->

<form id='Login' class='input-group-login'>

<input type='text'class='input-field'placeholder='email id'required>

<input type='password'class='input-field'placeholder='enter [password'required>

<input type='checkbox'class='check-box'>

<span>remember password</span>

<button type='submit'class='submit-btn'>login</button>

</form>

<form id='register'class='input-group-register'>

<input type='text'class='input-field'placeholder='first name'required>

<input type='text'class='input-field'placeholder='last name'required>

<input type='email'class='input-field'placeholder='email id'required>

<input type='password'class='input-field'placeholder='confirm password'required>

<input type='checkbox'class='check-box'>

<span>i agree to the terms and conditions</span>

<button type='submit'class='submit-btn'>register</button>

</form>

</div>

</div>

<!---Rooms section is start-->

<br><br><br><br><br><br><br><br><br><br><br><br><br><br><br>

<div class="">

<p align="center">

<font face="sans-serif" color="#fff" size="4">ROOMS</font>

</p>

<div class="image">

<center>

<table align="center" cellspacing="6" cellpadding="4">

<tr>

<td>

<img src="https://encryptedtbn0.gstatic.com/images?q=tbn:ANd9GcQyuGuiaPYpolPmg6bYtPETfOKJUS09\_BVQ3w& usqp=CAU" height="200" width="350"/>

</td>

<br><br><br>

<td>

<img src="https://encryptedtbn0.gstatic.com/images?q=tbn:ANd9GcShpjYhowE70zjPLTllphHG18IvbKp3YzI4H5M6DSfkKojWO1emg6XVuMZnw0NMxFFUhQ&usqp=CAU" height="200" width="350"/>

</td>

<td>

<img src="https://encrypted-

tbn0.gstatic.com/images?q=tbn:ANd9GcTHY1n8NHL\_07K7bg9kPvmLrEhJiZBx6\_7Kvg&u sqp=CAU" height="200" width="350"/>

</td>

</tr>

</table>

</center>

<br>

<div class="content">

<h2>What Makes Us Feel Spcial?</h2>

<p>1. A friendly, welcoming smile when your guests check-in. Linen quality that provides the feeling of luxury. </p>

<p>2. Lighting to create a calm ambiance.</p>

<p>3. An efficient service can make the difference between a relaxing stay and a stay that makes your guests feel anxious.</p>

<p>4. Where staff feel like they have too much to do while working long shifts, this frustration may be taken out on guests.</p>

<a href="#" class="btn">learn more</a>

</div>

</div>

</section>

<br><br><br><br><br><br><br><br><br><br><br><br><br><br><br><br><br><br><br> <br>

<div class="contact ">

<p align="center">

<font face="sans-serif" color="#000000

" size="5">CONTACT </font>

</p>

<!-- Contact from --->

<br>

<br>

<br>

<center>

<form name="F1" id="contact form">

<table align="center" cellspacing="15" cellpadding="10" border="1">

<tr>

<td>

<input type="text" name="First Name" placeholder="First Name"required="required" />

</td>

<td>

<input type="text" name="Last Name" placeholder="Last Name"required="required" />

</td>

</tr>

<tr>

<td>

<input type="email" name"Email" placeholder="Email" required="required" />

</td>

<td>

<input type="text" name"subject" placeholder="Subject" required="required" />

</td>

</tr>

<tr>

<th colspan="2">

<textarea name="message" placeholder="Enter Your Message" required="required" minlength="20" maxlength="100"> </textarea>

</th>

</tr>

<tr>

<th colspan="2">

<input type="Submit" name="contact-submit" form="contact-form" value="send" />

</th>

</tr>

</table>

</center>

</form>

</center>

</div>

<!-- footer --->

<br>

<br>

<br>

<center>

<footer>

<p align="center">

<a href="index.html">

<font face="sans-serif" size="7" color="#0a140c">ENJOY YOUR DAY</font>

</a>

<br>

<br>

<font face="sans-serif" size="5" color="#0a140c">"Every luxury must be paid for, and everything is a luxury, starting with being in this world."</font>

</p>

</footer>

</center> <script> var x=document.getElementById('login'); var y=document.getElementById('Register'); var z=document.getElementById('btn'); function register()

{

x.style.left='-400px';

y.style.left='50px';

z.style.left='110px'; } function login()

{

x.style.left='50px';

y.style.left='450px';

z.style.left='0px';

}

</script>

<!--this code is for the when you click out the login or registration page the form-box disappears--> <script> var modal = document.getElementById('login-form'); window.onclick = function(event)

{ if (event.target == modal) { modal.style.display= "none";

}

}

</script>

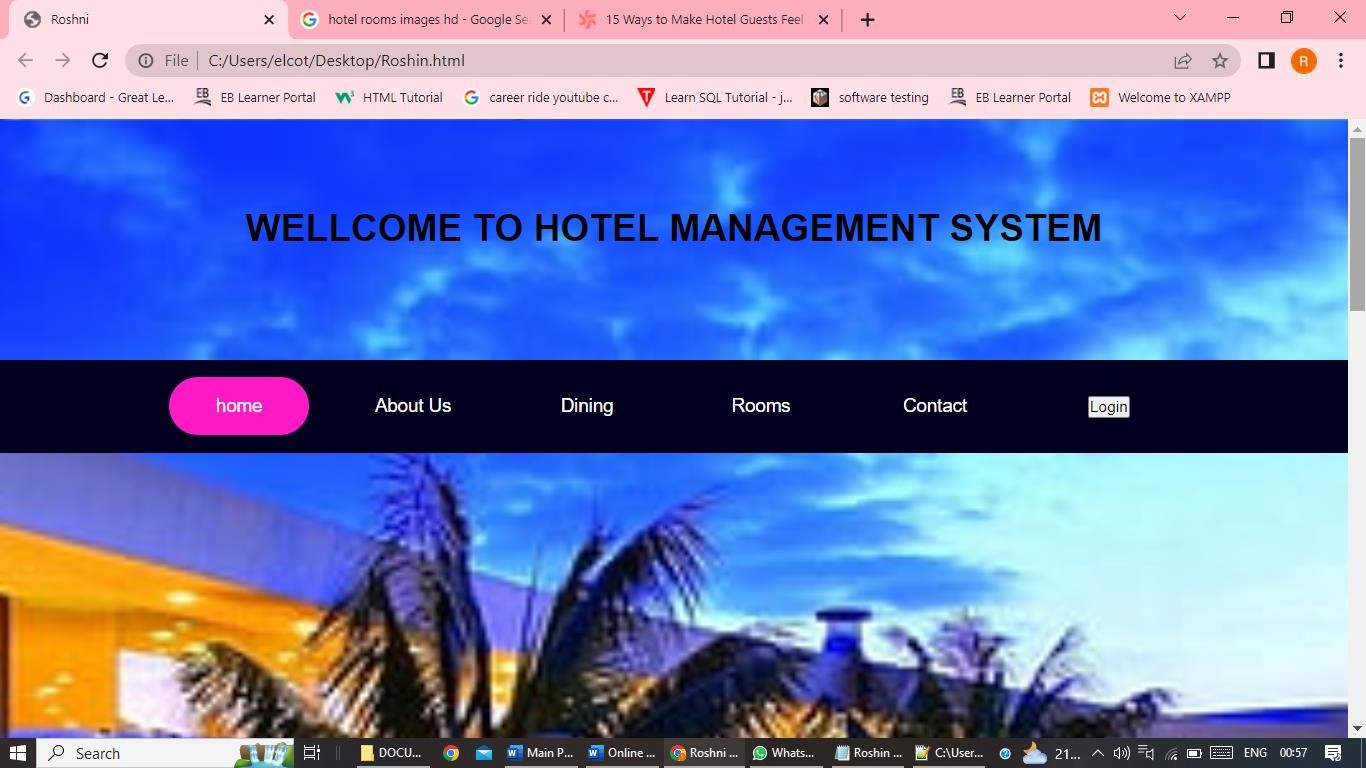
<!--let us run and see-->

</body>

</html>

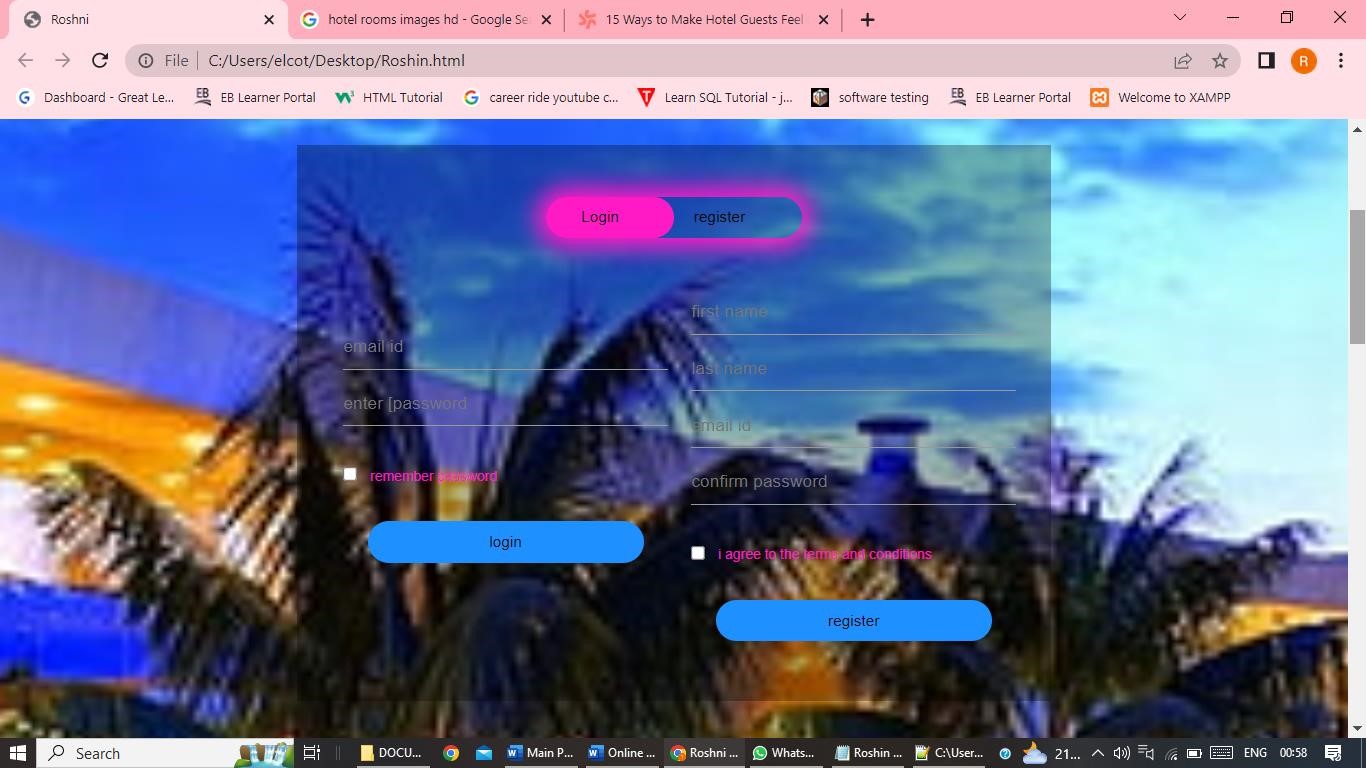
**5.4 Output**

#### 5.4.1 Home page



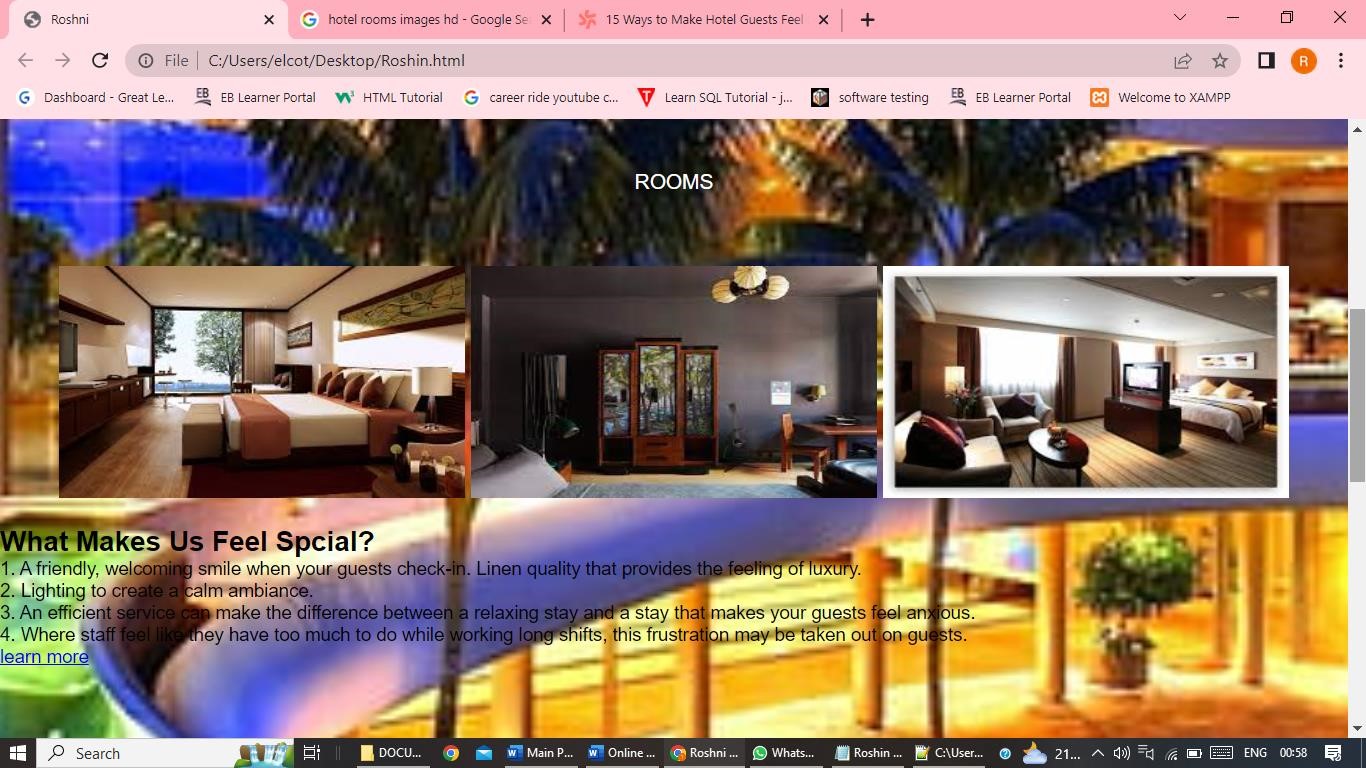
In above figure we can able to view home page

#### 5.4.2 Login and Registration Form page



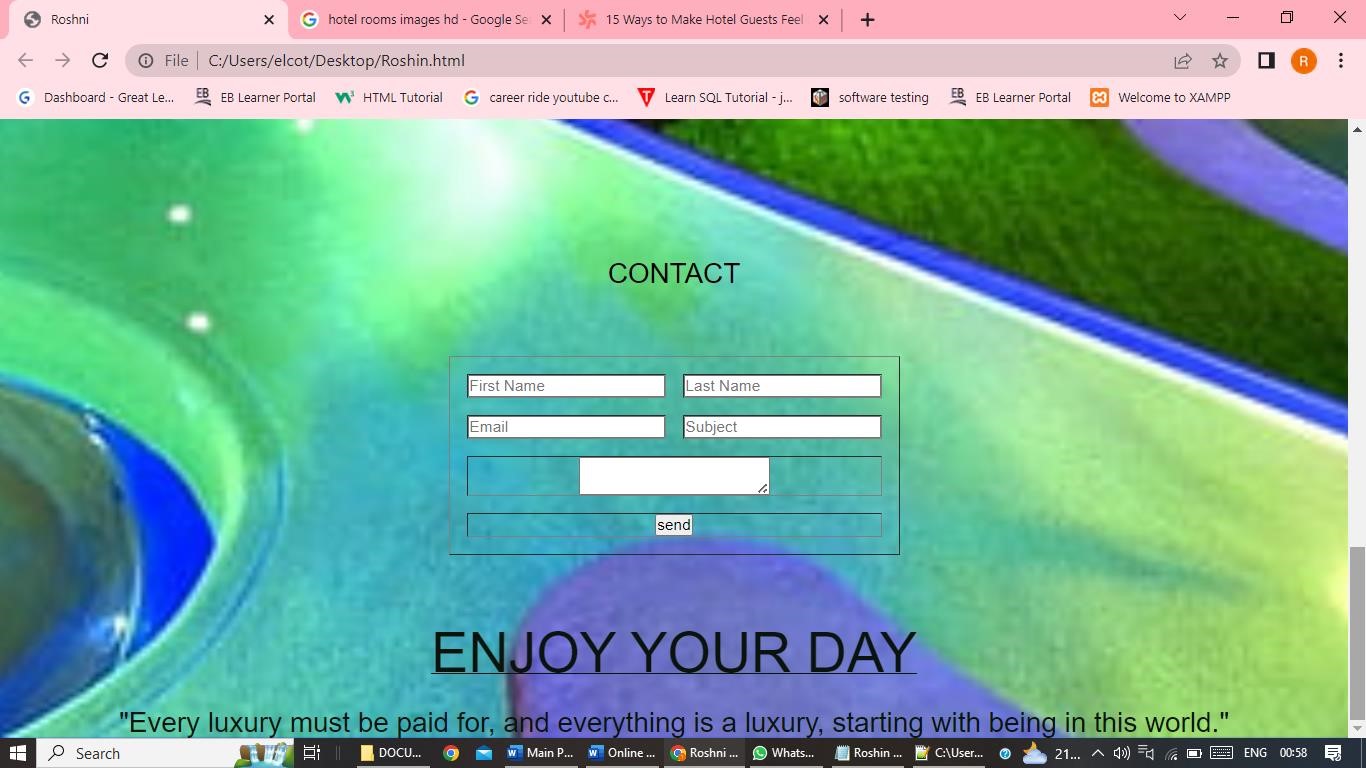
In above figure we can able to view login page

#### 5.4.3 Rooms page



In above figure we can able to view room’s

**5.4.3 Contact us page**



In above figure we can able to view contact us page.

# 6.SYSTEM TESTING

## 6.1 Selenium

Selenium is one of the most widely used open source Web UI (User Interface) automation testing suite. It was originally developed by Jason Huggins in 2004 as an internal tool at Thought Works. Selenium supports automation across different browsers, platforms and programming languages. Selenium can be easily deployed on platforms such as Windows, Linux, Solaris and Macintosh. Moreover, it supports OS (Operating System) for mobile applications like iOS, windows mobile and android. Selenium supports a variety of programming languages through the use of drivers specific to each language.

Languages supported by Selenium include **C#, Java, Perl, PHP, Python** and **Ruby.** Currently, Selenium Web driver is most popular with Java and C#. Selenium test scripts can be coded in any of the supported programming languages and can be run directly in most modern web browsers. Browsers supported by Selenium include **Internet Explorer, Mozilla Firefox, Google Chrome** and **Safari.**

|  |  |  |
| --- | --- | --- |
|  |  |  |

Selenium Suite

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Selenium IDE |  | Selenium RC |  | Selenium WebDriver |  | Selenium Gird |

Selenium can be used to automate functional tests and can be integrated with automation test tools such as **Maven**, **Jenkins**, **& Docker** to achieve continuous testing.

It can also be integrated with tools such as **TestNG**, & **JUnit** for managing test cases and generating reports.

## 6.2 Testing

**System Testing** is a level of testing that validates the complete and fully integrated software product. The purpose of a system test is to evaluate the end-to-end system specifications.

Usually, the software is only one element of a larger computer-based system. Ultimately, the software is interfaced with other software/hardware systems.

System Testing is defined as a series of different tests whose sole purpose is to exercise the full computer-based system. Two Category of Software Testing

* Black Box Testing
* White Box Testing

System test falls under the **black box testing** category of software testing.

**White box testing** is the testing of the internal workings or code of a software application. In contrast, black box or System Testing is the opposite. System test involves the external workings of the software from the user’s perspective.

## 6.2 1 Testing Methodologies

Software Testing Methodology is defined as strategies and testing types used to certify that the Application under Test meets client expectations.

Test Methodologies include functional and non-functional testing to validate the AUT. Examples of Testing Methodologies are [**Unit Testing,**](https://www.guru99.com/unit-testing-guide.html) [**Integration Testing,**](https://www.guru99.com/integration-testing.html) [**System Testing,**](https://www.guru99.com/system-testing.html) [**Performance Testing**](https://www.guru99.com/performance-testing.html) etc.

Each testing methodology has a defined test objective, test strategy, and deliverables. There are tons of methodologies available for software development and its corresponding testing. Each testing technique and methodology is designed for a specific purpose and has its relative merits and demerits.

Selection of a particular methodology depends on many factors such as the **nature of a project, client requirement, project schedule,** etc.

From a testing perspective, some methodologies push for testing input early in the development life cycle, while others wait until a working model of the system is ready.

## 6.2. 2 Equivalence Partitioning

Equivalence classes are evaluated for given input conditions. Whenever any input is given, then type of input condition is checked, then for this input conditions, Equivalence class represents or describes set of valid or invalid states.

Let us consider an example of any college admission process. There is a college that gives admissions to students based upon their percentage. Consider percentage field that will accept percentage only between 50 to 90 %, more and even less than not be accepted, and application will redirect user to an error page.

This technique tries to define test cases that uncover classes of errors, thereby reducing the total number of test cases that must be developed. An advantage of this approach is reduction in the time required for testing software due to lesser number of test cases.

Equivalence partitioning is typically applied to the inputs of a tested component, but may be applied to the outputs in rare cases. The equivalence partitions are usually derived from the requirements specification for input attributes that influence the processing of the test object.

**6.2.3 Software level testing can be majorly classified into 4 levels:**

1. **Unit Testing:** A level of the software testing process where individual units/components of a software/system are tested. The purpose is to validate that each unit of the software performs as designed.
2. **Integration Testing:** A level of the software testing process where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units.
3. **System Testing:** A level of the software testing process where a complete, integrated system/software is tested. The purpose of this test is to evaluate the system’s compliance with the specified requirements.
4. **Acceptance Testing:** A level of the software testing process where a system is tested for acceptability. The purpose of this test is to evaluate the system’s compliance with the business requirements and assess whether it is acceptable for delivery.

## 6.3Testcase

**Testcase for Online Hotel Management System Website.** Project Name: Online Hotel Management System

Reference: Project Functionality Requirement Specification

Created By: <http://www.onlinehotelmanagementsystem.com>team

Date of created: 01-October -2022

Date of review: 19-January -2023.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario 001-Login** | | | | | |  |  |
| **Test case** | **Test**  **Objective** | **Precon dition** | **Step:** | **Test data** | **Expected result** | **Post Condition** | **Result** |
|  |  | A valid account to login | Email id: password: click login button. | [roshni2610@gmail.com](mailto:roshni2610@gmail.com)  1464 | Logged successfully | Login  Inbox  Is shown | pass |
| **Test Scenario 002-Register** | | | | | |  |  |
|  |  | A valid account to login | First name:  Last name: Email id:  Password:  Click register Button. | Roshni Pokharkar  roshni2610@ gmail.com  1464 | Registered successfully | registered inbox is shown | pass |
| **Test Scenario 003-Home** | | | | | |  |  |
|  |  | All details | Click the About us | Details of online booking for hotel. | It navigate to about us page | Information’s are shown | pass |
| Click rooms | Details of rooms. | It navigate to about us page | Information’s are shown | pass |
| Click dining | Details of dining in hotel. | It navigate to dining page | Information’s are shown | pass |
| Click contact us  First name  Last name  Email  Password  Comment  Click sendbutton | Roshni Pokharkar  Roshni2610@gmail.com  1464  Enjoy your special day for  Your happieness in your life. | Contact us form is send successfully | sent inbox is shown | pass |

**6.4Website Automation Testing Using Selenium**

**Automation Testing** is a software testing technique that performs using special automated testing software tools to execute a test case suite.

On the contrary, Manual Testing is performed by a human sitting in front of a computer carefully executing the test steps.

The automation testing software can also enter test data into the System Under Test, compare expected and actual results and generate detailed test reports.

Software Test Automation demands considerable investments of money and resources.

**6.4.1 Automation testing process:**

Following steps are followed in an Automation Process

**Step 1.** Test Tool Selection

**Step 2.** Define scope of Automation

**Step 3.** Planning, Design and Development

**Step 4.** Test Execution

**Step 5.** Maintenance

### 6.4.2 Selenium Automation

Selenium is a free, open-source automation testing suite for web applications across different browsers and platforms.

It is somewhat similar to HP Quick Test Pro (QTP, currently UFT). However, Selenium focuses on automating web-based applications.

Testing done using Selenium is usually referred to as Selenium testing. Remember, only testing web applications is possible with Selenium. You cannot use it to test desktop applications or mobile applications. I have created an website for online courses named as Online Hotel Management System And I have done test Automation on my project by using selenium and the syntax for testing my website is

**6.5.Selenium code for Online Hotel Management System Website.:**

**import** org.openqa.selenium.By;

**import** org.openqa.selenium.JavascriptExecutor;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.chrome.ChromeDriver; **public** **class** OnlineHotelManagementSystem

{

**public** **static** **void** main(String[] args) **throws** InterruptedException

{

System.*setProperty*("webdriver.chrome.driver",

"C:\\Users\\elcot\\Documents\\chromedriver\_win32\\chromedriver.exe");

WebDriver driver=**new** ChromeDriver(); driver.get("file:///C:/Users/elcot/Desktop/ OnlineHotelManagementSystem.HTML"); driver.manage().window().maximize(); Thread.*sleep*(2000); driver.findElement(By.*xpath*("//button[@class='Loginbtn']")).click();

JavascriptExecutor js = (JavascriptExecutor) driver;

//Scroll down till the bottom of the page js.executeScript("window.scrollBy(0,300)"); driver.findElement(By.*className*("input-field")).sendKeys("ruby@gmail.com"); driver.findElement(By.*className*("input-field")).sendKeys("2720"); driver.findElement(By.*xpath*("//button[@class='check-box']")).click(); driver.findElement(By.*className*("submit-btn")).click(); driver.findElement(By.*xpath*("//button[@class='Loginbtn']")).click();

JavascriptExecutor js1 = (JavascriptExecutor) driver;

//Scroll down till the bottom of the page js1.executeScript("window.scrollBy(0,300)"); driver.findElement(By.*className*("input-field")).sendKeys("Roshni"); driver.findElement(By.*className*("input-field")).sendKeys("Pokharkar"); driver.findElement(By.*className*("input-field")).sendKeys("roshni2610@gmail.com"); driver.findElement(By.*className*("input-field")).sendKeys("1464"); driver.findElement(By.*xpath*("//button[@class='check-box']")).click(); driver.findElement(By.*xpath*("//button[@class='submit-btn']")).click();

JavascriptExecutor js2 = (JavascriptExecutor) driver;

//Scroll down till the bottom of the page js2.executeScript("window.scrollBy(0,document.body.scrollHeight)"); driver.findElement(By.*name*("First Name")).sendKeys("Roshni");

driver.findElement(By.*name*("Last Name")).sendKeys("S"); driver.findElement(By.*name*("email")).sendKeys("roshni2610@mail.com");

driver.findElement(By.*name*("message")).sendKeys(" i have selected online hotel management system for my happieness"); driver.findElement(By.*xpath*("//[@type='Submit']")).click();

}

}

**6.6Output Automation testing for Amazon website.**

package SimplePro;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

public class ElementIdAmazon {

public static void main(String[] args)

{

System.setProperty("WebDriver.chrome.driver","C:\\Users\\ROSHNI\\Downloads\\chromedriver\_win32\\chromedriver.exe");

WebDriver driver=new ChromeDriver();

driver.get("https://www.amazon.com/");

driver.findElement(By.id("nav-link-accountList-nav-line-1")).click();

driver.findElement(By.id("ap\_email")).sendKeys("8530501464");

driver.findElement(By.id("continue")).click();

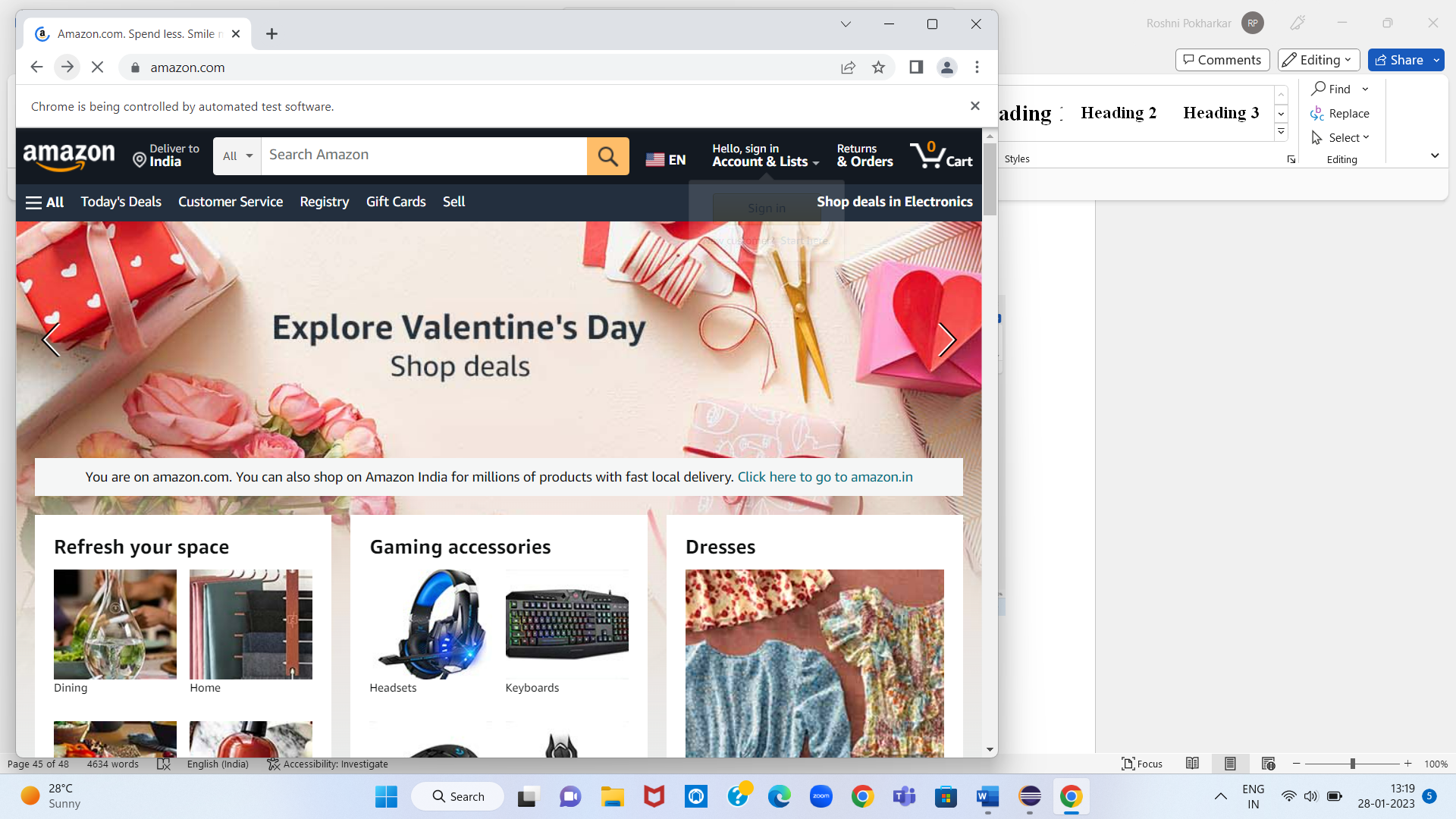
driver.findElement(By.name("password")).sendKeys("Roshni@1999");

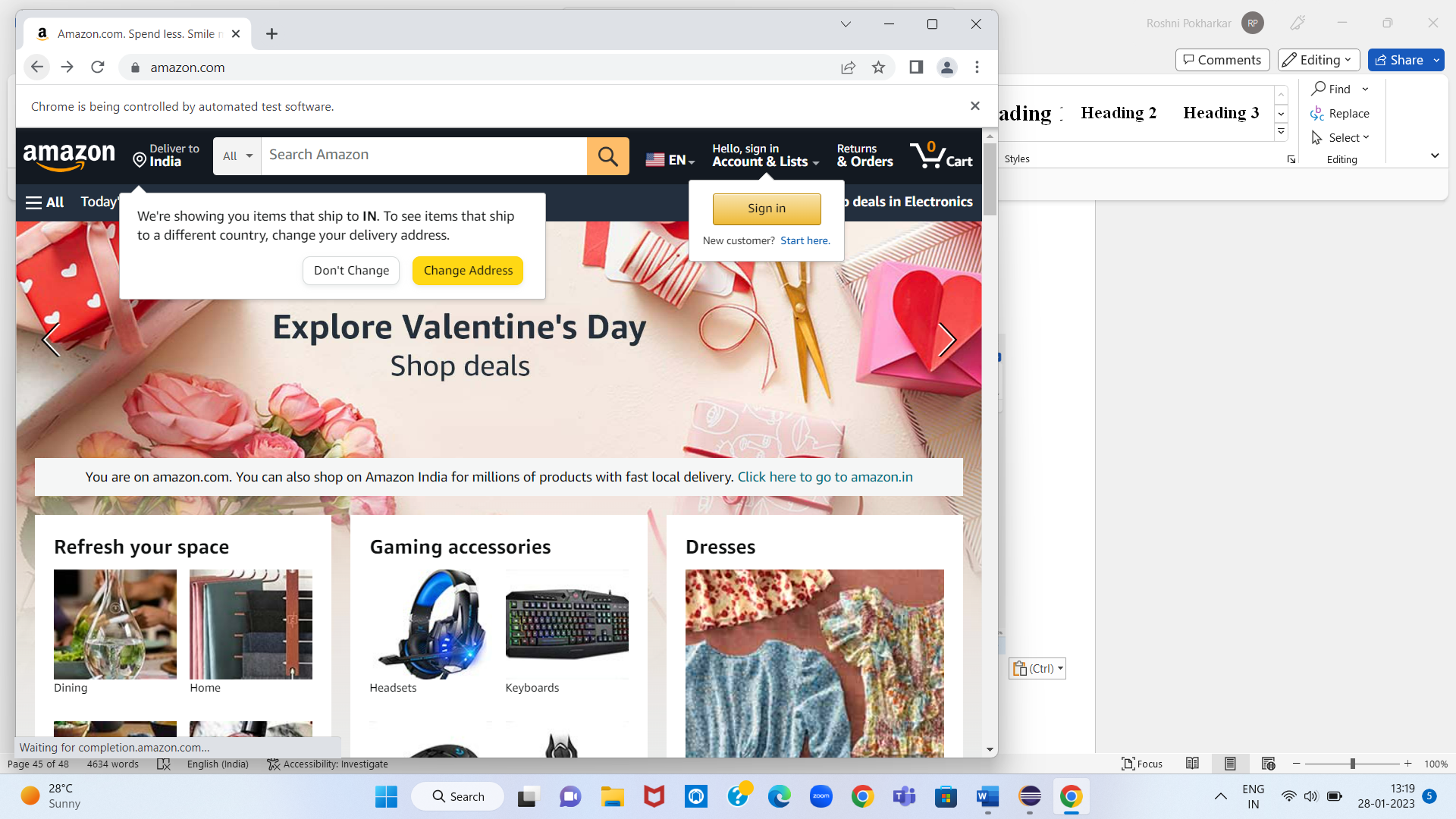
driver.findElement(By.className("a-button-input")).click();

driver.findElement(By.id("twotabsearchtextbox")).sendKeys("watches for womens ");

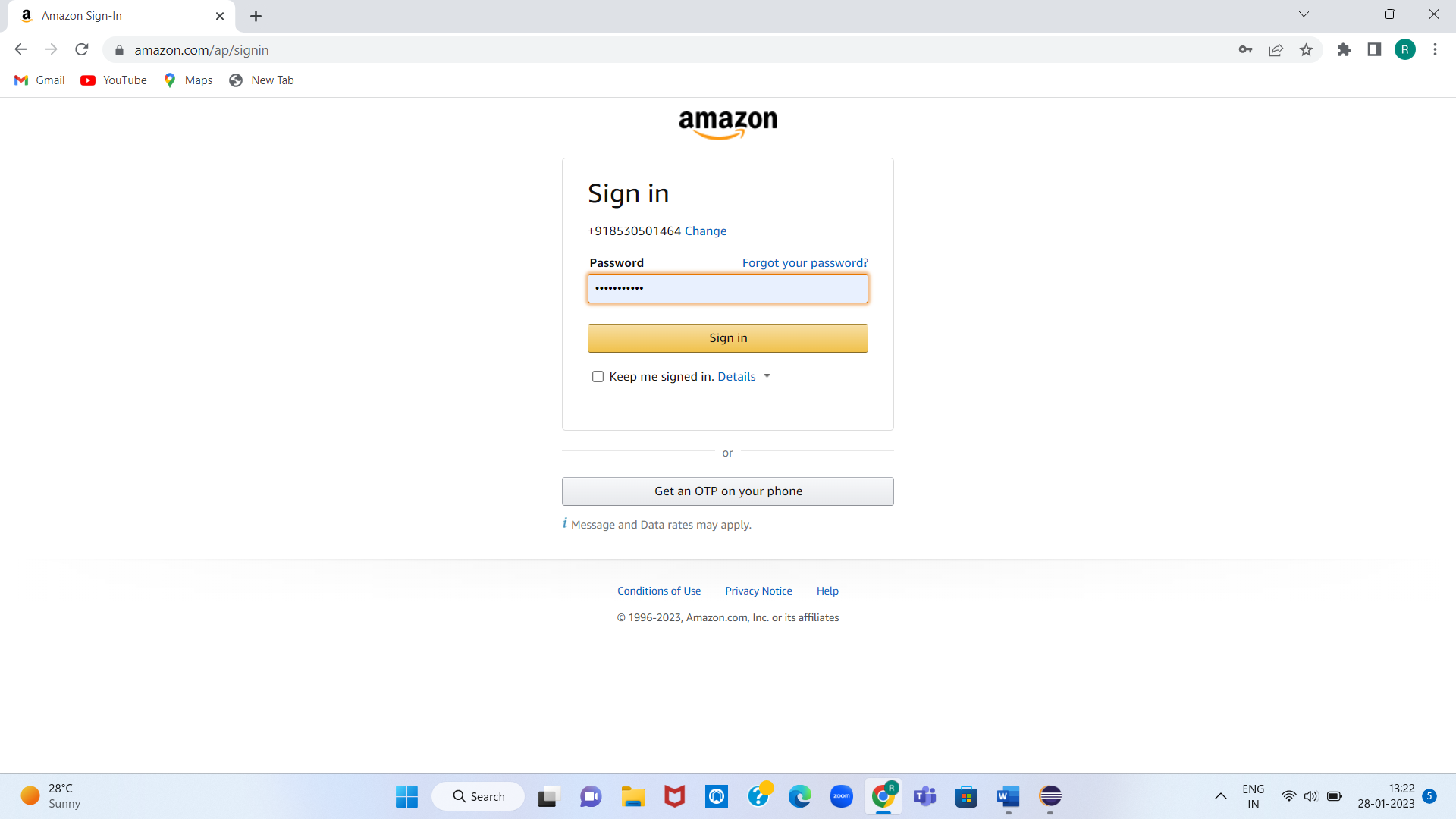
driver.findElement(By.id("nav-search-submit-button")).click();

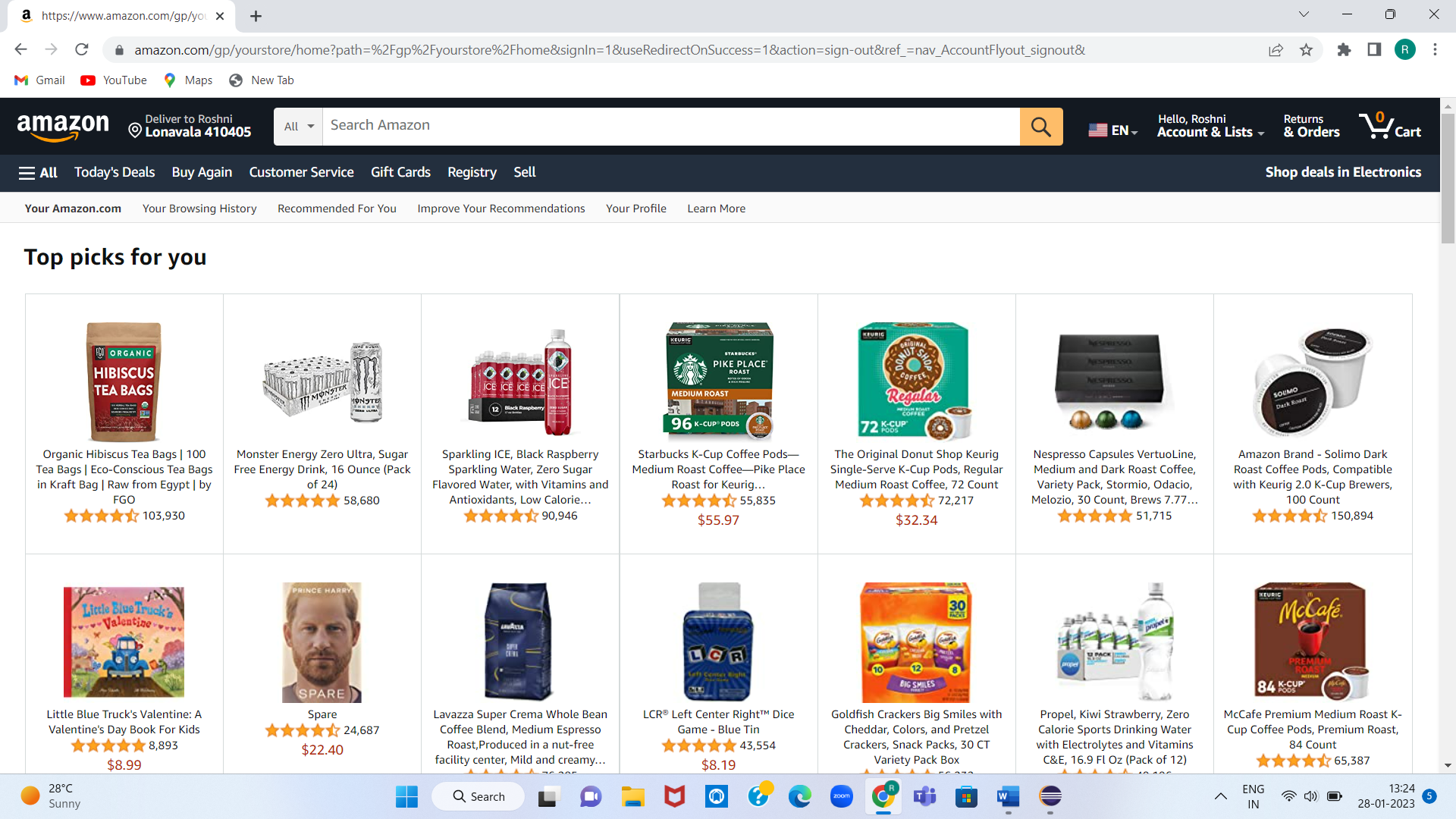
**Output:**

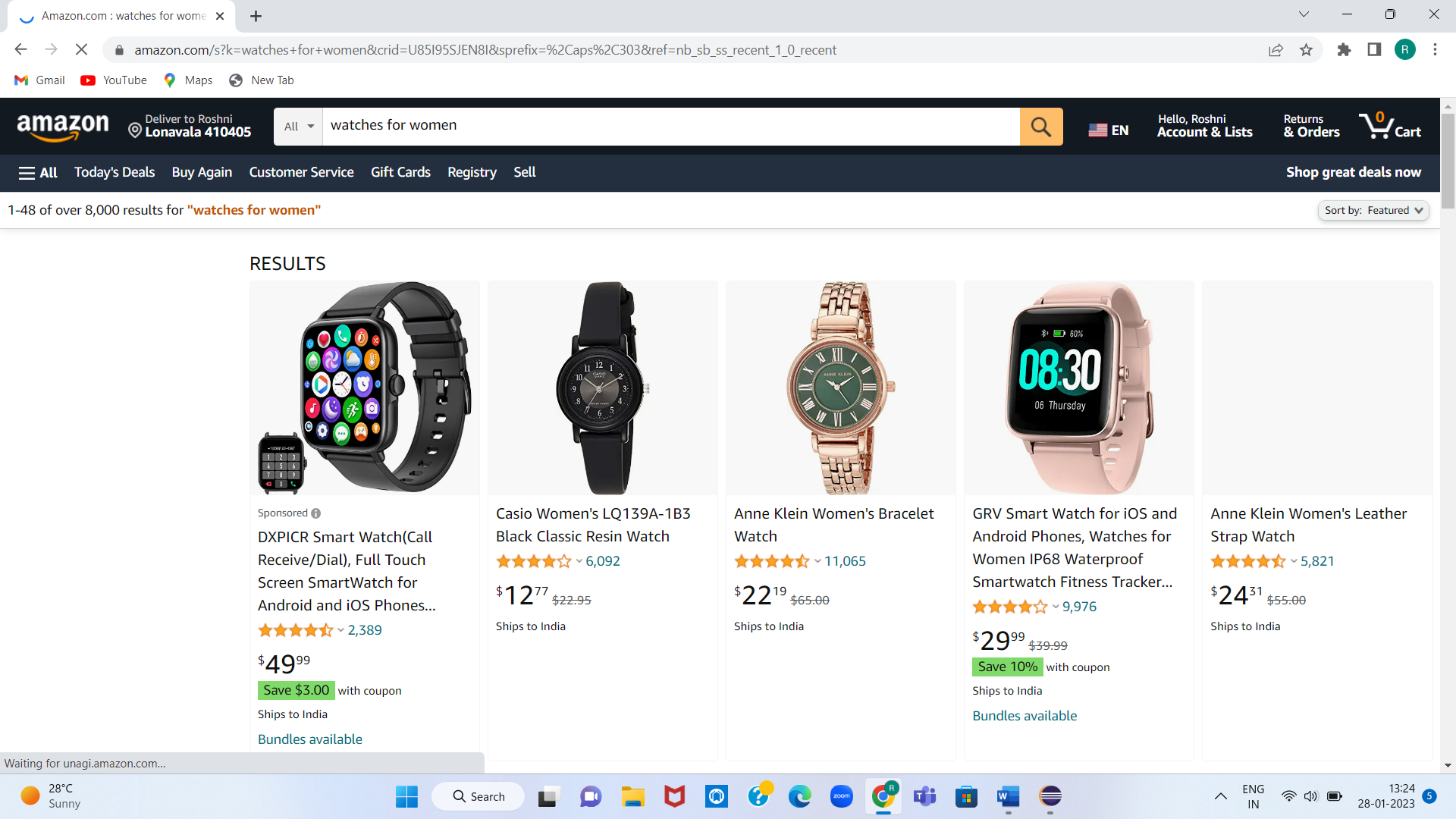












**7.Conclusion**

Online Hotel Management System has all the basics modules and also it makes online system fully computerized which is very fast and efficient. Normal in online system we use maximum manual application .it consumes lots of time and paper. there is always a distance between the system and end user. Now-a-days everything is becoming computerized in order to reduce the distance between the system and the user and bring it to a close relationship as using paper and pens, this Career bridge has been developed.

The owner can enter into the system very easily. The system is totally a user-friendly and timesaving system .it is also cost effective. All the modules are designed in a way that a zone can understand the system very easily.

7.Conclusion

Online Hotel Management System has all the basics modules and also it makes online system fully computerized which is very fast and efficient. Normal in online system we use maximum manual application .it consumes lots of time and paper. there is always a distance between the system and end user. Now-a-days everything is becoming computerized in order to reduce the distance between the system and the user and bring it to a close relationship as using paper and pens, this Career bridge has been developed.

The owner can enter into the system very easily. The system is totally a user-friendly and timesaving system .it is also cost effective. All the modules are designed in a way that a zone can understand the system very easily.