

SWEET FACE AESTHETICS PROJECT REPORT

Course of

EXL- Certified Software Test Engineer

Submitted By

JAYABHARATHI.M

BATCH: 2022 - 7607

ENROLLMENT NUMBER: EBEON0722634518

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

Nothing can match the feeling you get when you glide your fingers over your face and feel that soft and milky skin. Most of you follow a daily skin care routine at home. It might not be elaborate, but it at least helps you maintain your skin. But once in a while, it's good to let the professionals take care of your skin. Yes, I am talking about the benefits of facials.

I am doing my project on Beauty Parlour Management System. There is the need for best management of a network based system for handling customers orders. It is a era of digital technology & development of technology. The growth of internet users is increasing day by day. Beauty parlour Management System is a web-based parlour management application with appointment scheduling functionality In this system user can review parlour & stylist online & can make appointment online. At present many people run their business (online) without any physical setup of their business. Like digital marketing agency, digital marketing & many mores.

There are various types of online websites available, ranging from online E-commerce website (AMAZON, FLIPKART) online food delivery system (ZOMATO, DOMINOZ). In this system beauty parlour owner can make their account online & give details of the services. They offer and describe the stylist information. Using this platform user can view all the parlour which are partner with our parlour management system & also view their services. User can review the Salons & their services using this system & can make appointment with, favour stylist online user can also cancel the appointments online customer search the near beauty parlours then the system will show the next blank slots their nearest location.

It will be more useful for customers because now they have their booking with the available slot. It will also remove all extra paperwork as the owner will have access to all customers details & their records to save them better. In this project MYSQL & PHP are used to back the interface with strong database functionality & the frontend is HTML, CSS & JavaScript are used. This project will target the major web browser like google chrome, fire fox & safari as the initial platform for our version. As Beauty & Fashion, style is become a great trend in all generation peoples is no matter what age & gender.

2.SCOPE AND OBJECTIVES

2.1 Purpose

The beauty parlour shop is the need of every age of men women and children famous poet defined the word beauty as : " A thing of beauty is a joy for ever: Beauty parlour is a very important shop to make the people good looking by application of cosmetics treatment of hair and nourishment of skin by various.

2.2 Scope

Beautician tasks include removing hair, recommending skin care therapies and managing client appointments beautician should be well-versed in beauty treatments, including hairstyling, makeup, facials, hair removal, manicures and pedicures

A Beauty Parlour Management System is a website that managed the appointment. This website system is .connected to users & beauty parlours in an online platform users can write & read the review of the Beauty .Parlour Management. Beauty Parlour can also book the appointments for the customers. The 1st is Admin Panel & 2nd is User Panel. The user can make meeting with the admin of parlour & therefore admin approves it. All activities are creating appointment like choose service, date, time are going to be record in the database

3.SYSTEM SPECIFICATION

3.1 Hardware 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.

Processor Intel Dual Core Processor

Ram:1 GB RAM

Hard Drive:160 GB

Monitor:17 INCHES

Keyboard:104 keys

Mouse: Logitech Optical Mouse

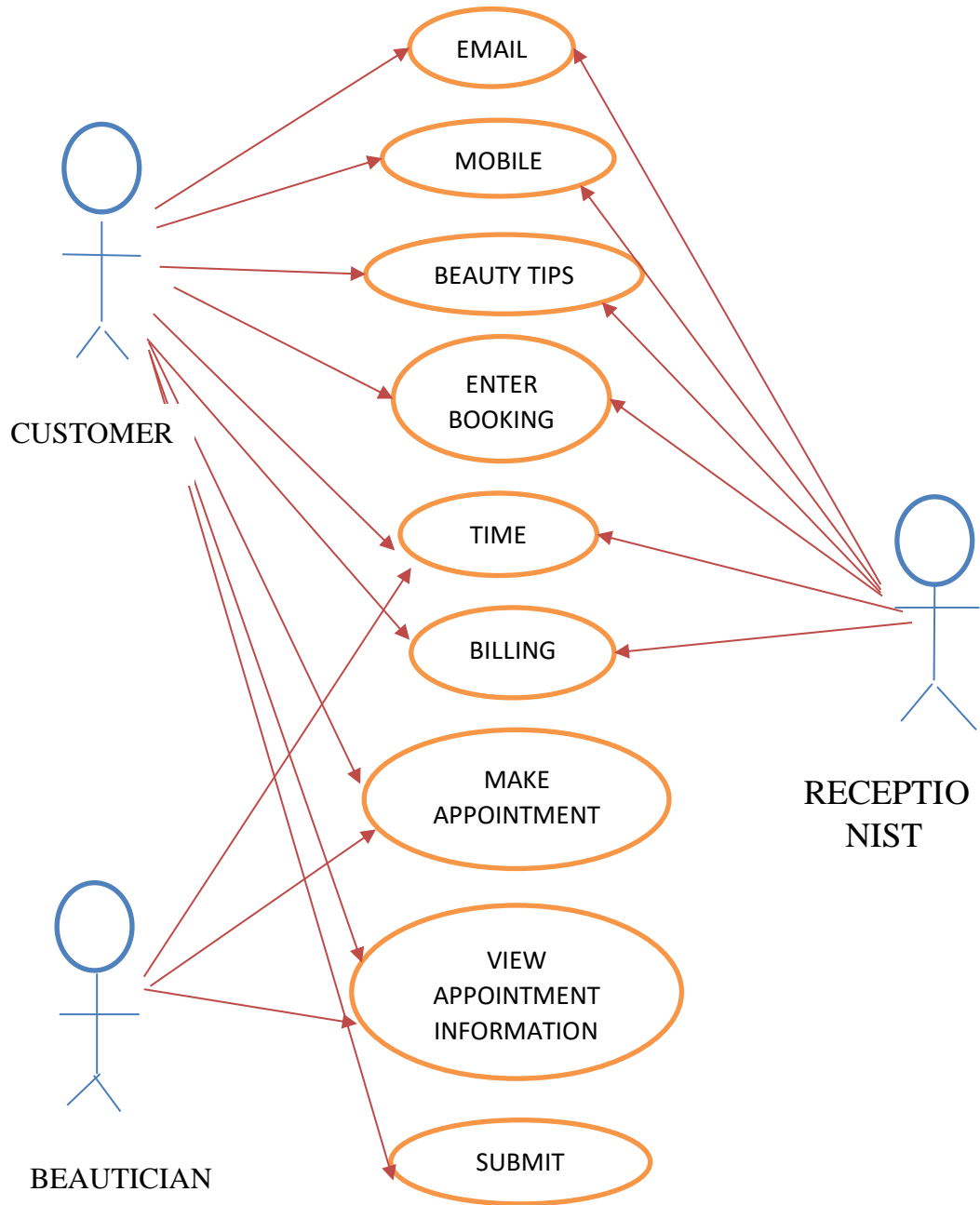
3.2 software Requirements.

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 backend tool that stores the data.

Operating System: Windows 10

Front-end: Notepad, browser.

4. SYSTEM DESIGN



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.	Email id	Receptionist, customer	Enter your valid mail id
2.	Mobile	Receptionist, customer	Enter your mobile number
3.	Booking date	Receptionist, customer, beautician	Enter your booking date and timing

4.1.2 Use case Explanations

In this use case diagram, they mention the step process

4.1.3 Use case 001: login

Use case 001: login

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

Start

ACTOR: Administrator, Student (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's name and password

Scenario:

ACTOR	SOFTWARE REACTION
Mail id	If it is valid user can enter into the system If it is invalid user it indicates is shows wrong mail Id

ALTERNATE FLOW: the user can enter the email id

SPECIAL REQUIREMENTS: None

ASSOCIATED USE CASE(S): None

4.1.4 Use case 002: Registration

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

ACTOR: student (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	SPECIFICATION
Mobile number	The system verifies that the above phone number is correct or not Suppose if any one number is missing or number is wrong the system doesn't Consider the mobile number You have to enter the correct mobile number

ALTERNATE FLOW: None

SPECIAL REQUIREMENTS: None

ASSOCIATED USE CASE(S): None

4.1.4 Use case 003: submit

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

ACTOR: student (user) and admin

PRE-CONDITION: user has to have a valid credential

POST-CONDITION: the system display the relevant page

BASIC FLOW: the user enter the details

Scenario:

ACTOR	SOFTWARE SPECIFICATION
Enter and Sumit the email id and mobile number	The system verifies that the above mail id and mobile number is valid or not Then if any data is missing the system warns the user and the steps continue With software reaction If all the data has been valid the system is considered In case invalid the system doesn't Considered .

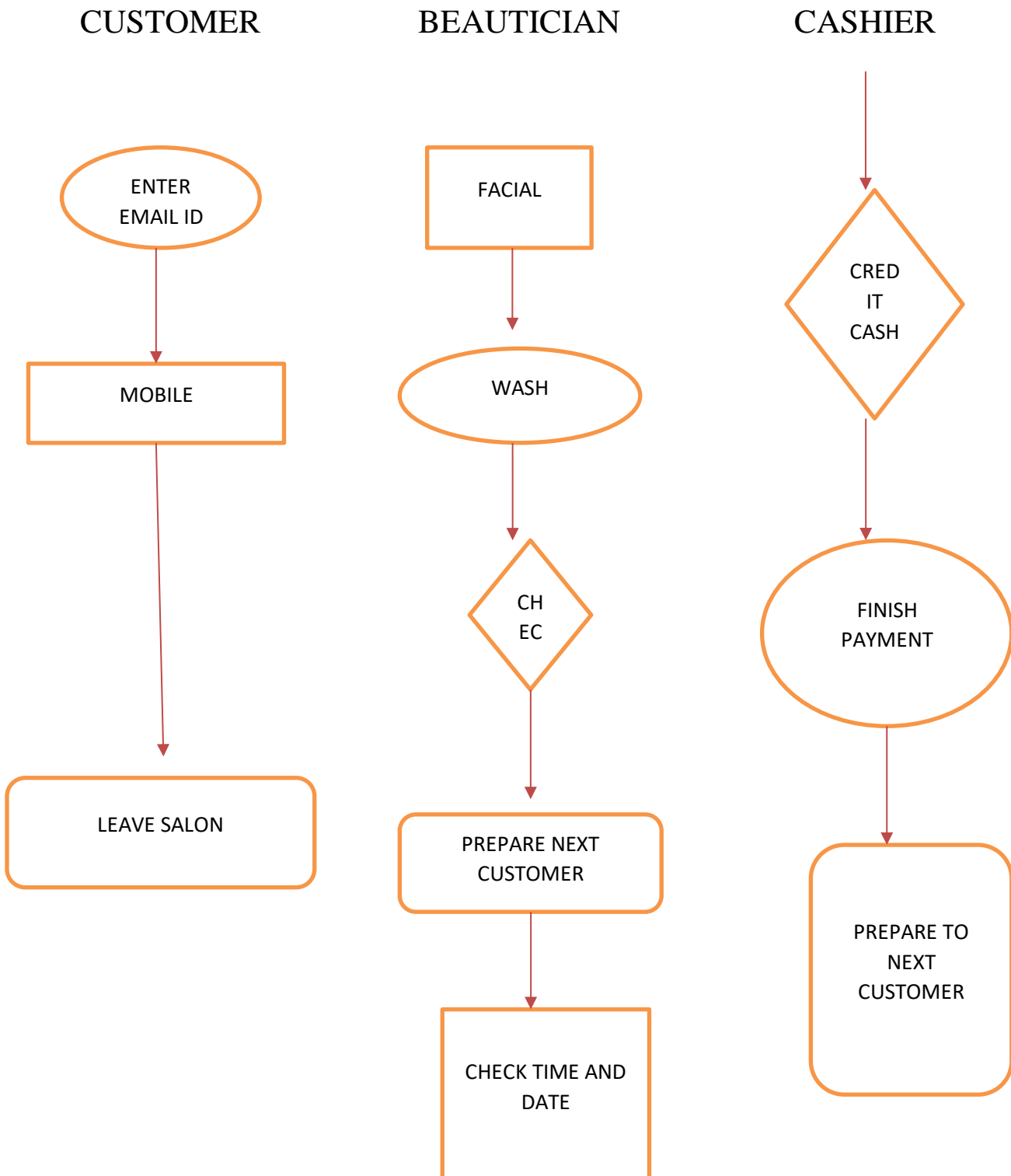
ALTERNATE FLOW: click the help option

SPECIAL REQUIREMENTS: None

ASSOCIATED USE CASE(S): None

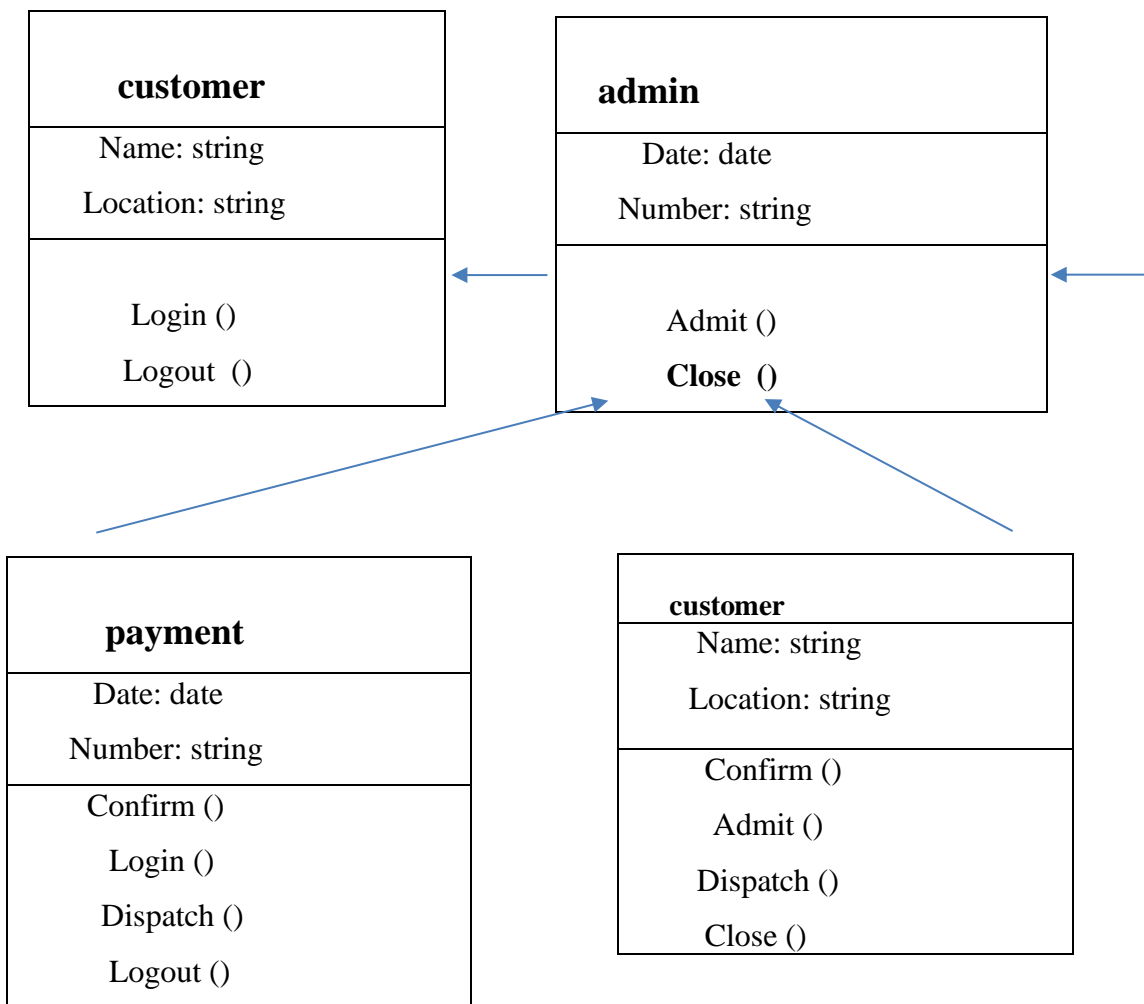
4.2 Activity Diagram

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

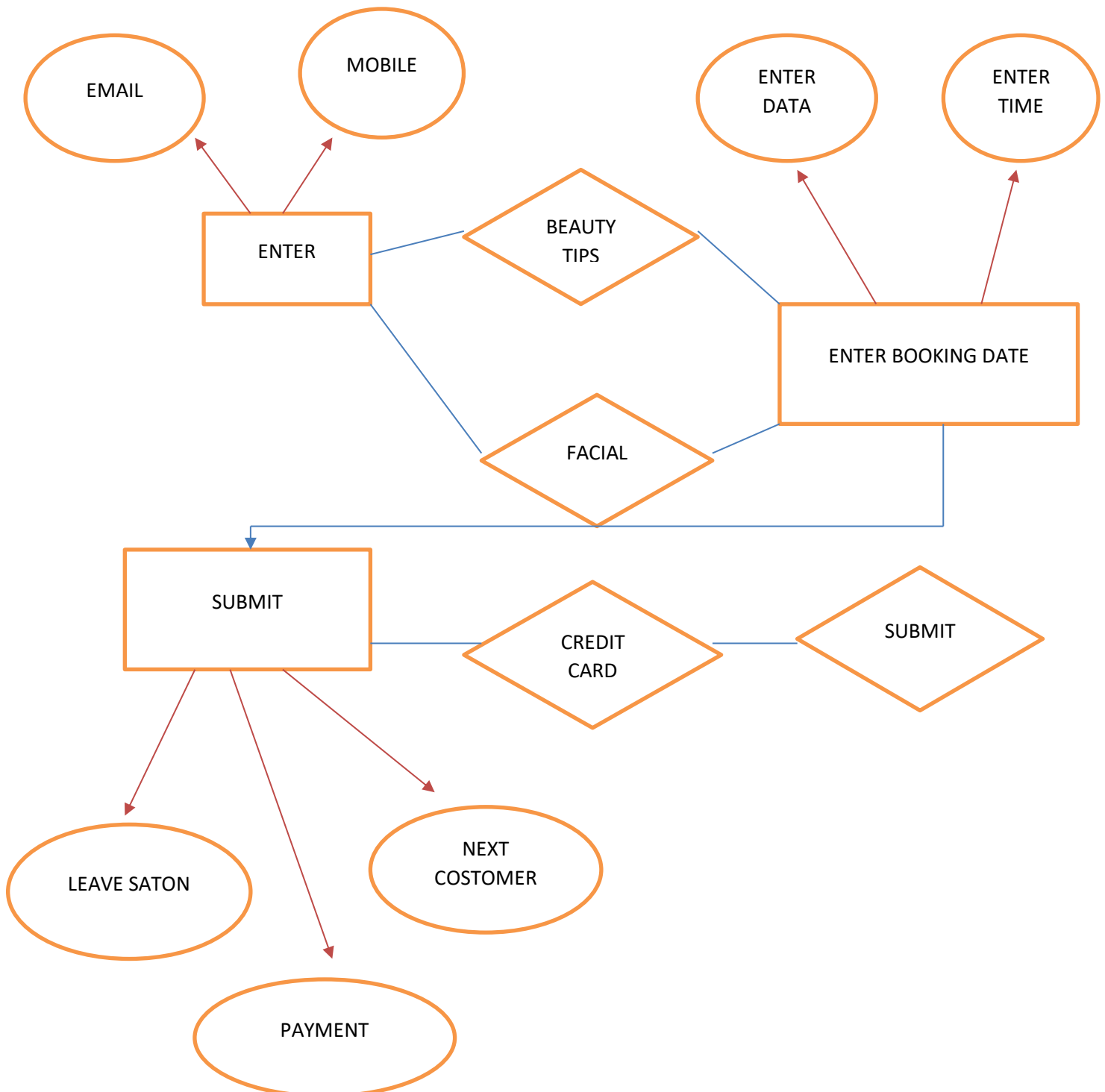


4.3 Class Diagram

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.



4.4 Entity Relational Diagram (ER)



4.5 Data Flow Diagram

A data flow diagram (DFD) is a graphical model that shows all of the main requirements for an information system in a diagram: input and output, process, and data storage. A DFD

describes 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 reasons for its popularity, the DFD is also easy to read because it is a graphical model. The DFD is mainly used during problem analysis. End

DFD with minimal training.

4.5.1 DFD Symbol

1. Process



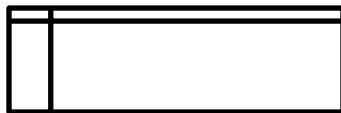
2. Data Flow



3. External Entity

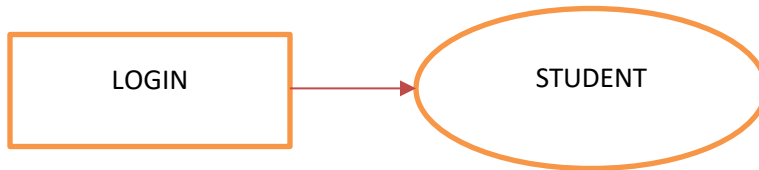


4. Data Store

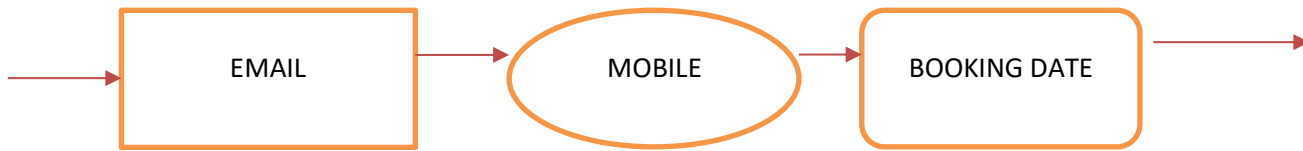


4.5.2 Data Flow Diagram

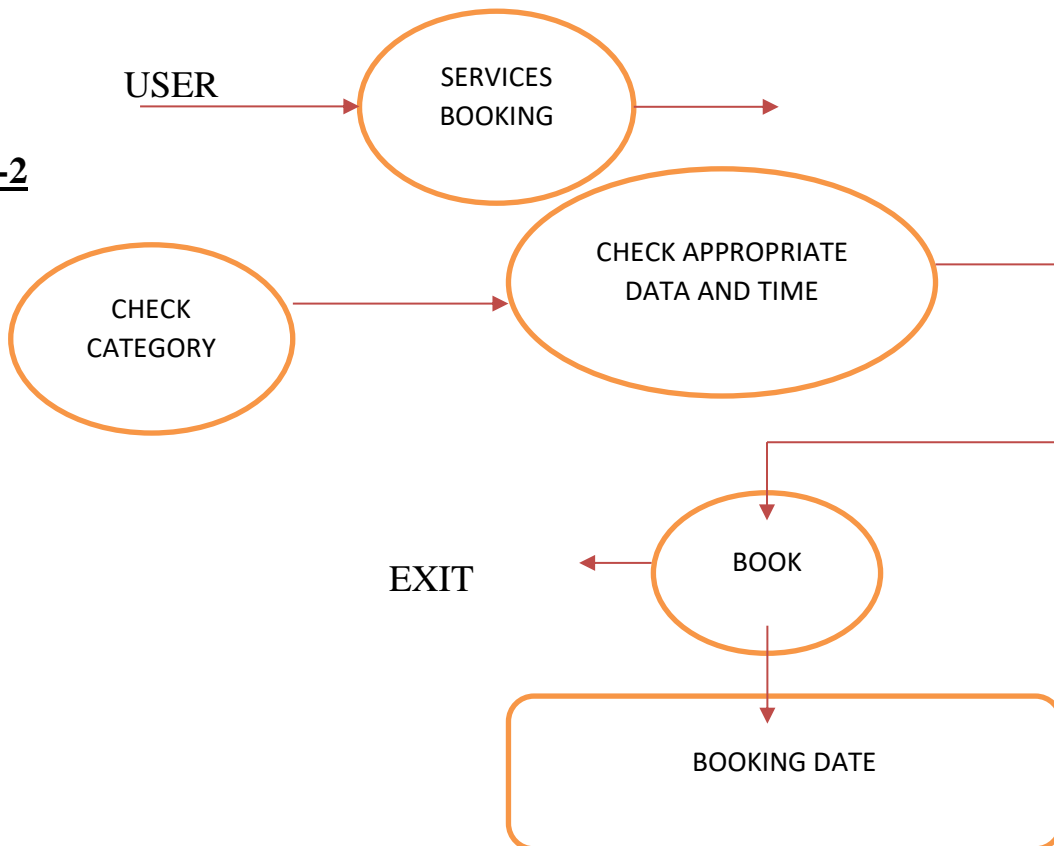
Level-0



Level-1



Level-2



4.6 Table Structure

4.6.1 Login

Table Name: Email

Primary Key: password

COLUMN NAME	DATA TYPE	DESCRIPTION
EMAIL ID	TEXT	EMAIL ID
PASSWORD	NUMBER	PASSWORD

4.6.1 Booking

Table Name: Booking The Date

Primary key: Email id

COLUMN NAME	DATA TYPE	DESCRIPTION
EMAIL ID	TEXT	EMAIL ID
PASSWORD	NUMBER	PASSWORD
BOOKING DATE	NUMBER	BOOKING DATE

4.6.3 payment

Table name: Submit

Primary key : email id

Foreign Key: Payment

COLUMN NAME	DATA TYPE	DESCRIPTION
EMAIL ID	TEXT	EMAI ID
PASSWORD	NUMBER	PASSWORD
BOOKING DATE	NUMBER	BOOKING DAYE
SUMBIT	TEXT	SUMBIT
PAYMENT	NUMBER	PAYMENT

5.IMPLEMENTATION

5.1.Modules

In this project I have created the beauty salon for maintain glow and good look and using online booking I have named as sweet face aesthetics.

5.1.1Email id

Enter your valid mail id

5.1.2.Mobile Number

Enter your valid Mobile Number

5.1.3.Booking date

In which date and time do you want you can select and booking the date .Once you visit our beauty salon we will be given a discount on your next time

5.1.4.Beauty tips

Beauty parlour is a service oriented establishment in which women receive treatment to increase their beauty. Skin care, facial make up, hair nourishment and model hair cutting are the most important nourishing activities of a beauty parlour. The use of machinery, ayurvedic formulations and approved synthetic chemicals and medicines vary from person to person as per need.

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)

Cascading Style Sheets, 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 coding technique, 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.

5.3 Coding

```
<!DOCTYPE html>

<head>

<h1>BHARATHI BEAUTY PARLOUR</h1>

</head>

<br>

<body bgcolor="violet">

<hr/> <br>

<style>

body{

background-color: violet;

color: black;

padding:40px;

text-align: center;}

. BEAUTY FACIAL{

background-color: black;

color: white;

padding:10px;

}

.content{

margin-left:190px;

border-left:1px solid #d4d4d4;

padding:1em;
```

overflow: hidden;

}

</style>

<body>

<h3>LETS GO BEAUTY FACIAL WELCOME TO ALL</h3>

<p>Taking Care Oh Your Skin Is More Important Than Covering It Up </P>

<h2 class="facial">golden facial</h2>

<p> Improve your skin elasticity making your skin look fresh and rejuvenated</p>

<h2 class="facial">silver facial</h2>

<p>Detoxifies and purifies your skin making it glowing</p>

<h2 class="facial">whiting facial</h2>

<p>It lightness,freckles,pigmentation,dark spots production</p>

</body>

<button on click="document. Location='default.asp'"> beauty tips </button></br>

<label>Email:<input type="text" name="bharathimano2001@gmail.com"></label>

<label>mobile:<input type="text" name="04435678390"></label>

<h3>once you visit our beauty salon we will be given a discount on your next time </h3>

<text area name="comments" rows="3">Enter Your Booking Date</text area>

<label for="app">choose a time for your wish:</label>

<input type="time" id="apt" name="app" min="09.00" max="18.00" required>

<small>parlor timing are 9am to 6am</small>

<input type="submit "value="submit"/>

home

career

contact

about

</div>

</div>

</div>

</nav>

<body>

<iframe width="420" height="345"

src="https://youtu.be/tKkJnauWFgA">

</iframe>

<h3 style="color:green;">once you click this you can see read more details & <p>lots of
information to

read</p>

</h3>

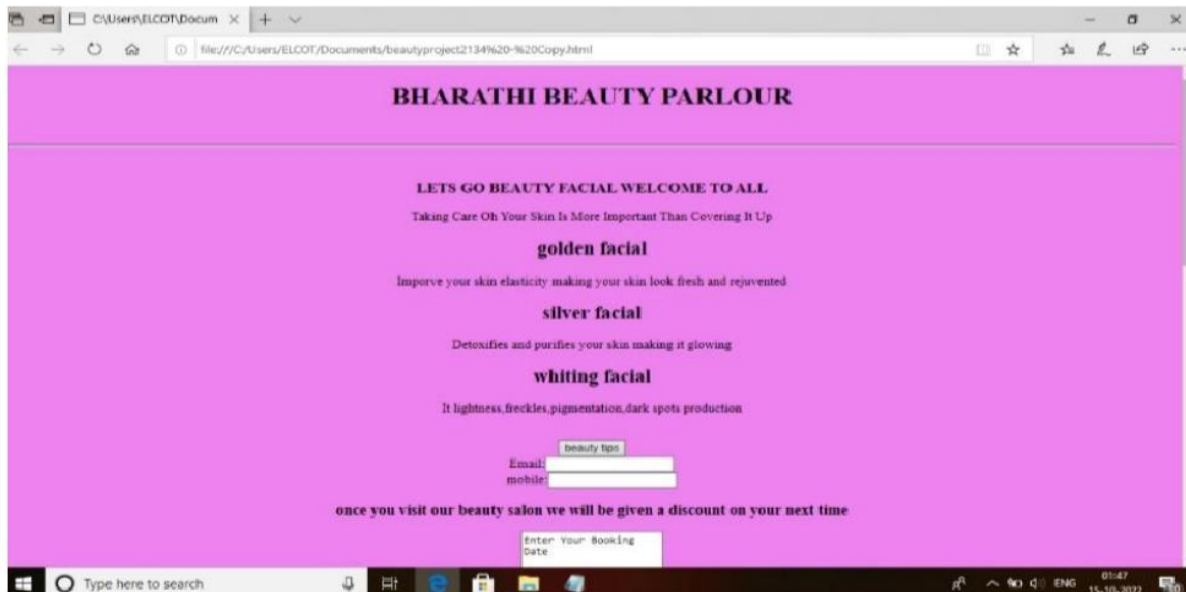
<a href="https://m.economictimes.com/news/how-to/7-aesthetic-treatments-to-get-your-face-
glowing-thiswedding-season/articleshow/87680677.cms">about makeup

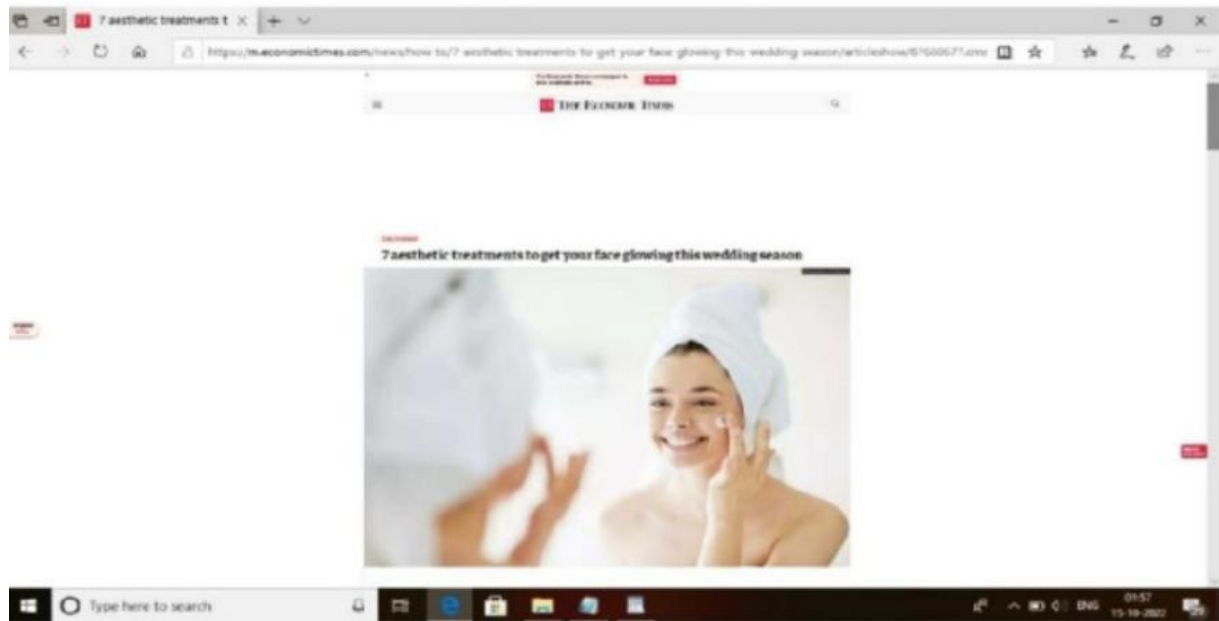
<p> welcome again </p>

</body>

</html>

5.4 Output





6. SYSTEM TESTING

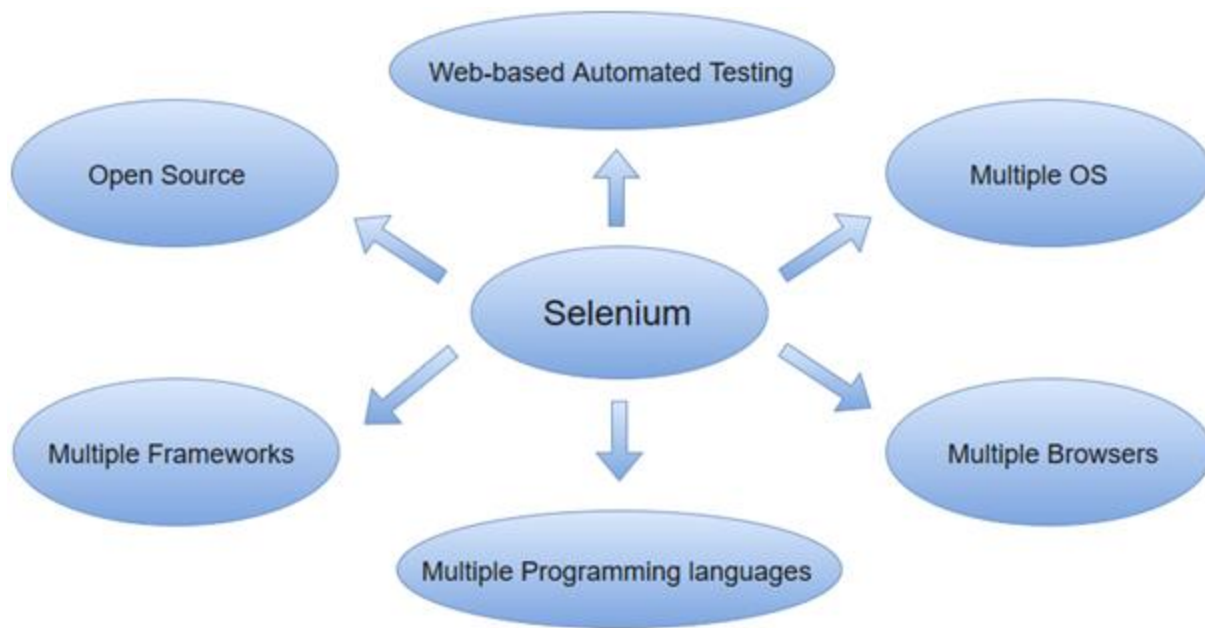
6.1 Selenium

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 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



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 **Chrome** and **Safari**.

- 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, Integration Testing, System Testing, Performance Testing 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.

6.3 Testcase

Testcase for sweet face aesthetics salon

Project Name: sweet face aesthetics

Reference : project Functionality Requirements Specification

Created By: <http://WWW.sweetfaceasesthetics.com>

Date of created: 01-October -2022

Date of Review : 19-January -2023

S. NO	TESTCASE-ID	TESTCASE	DESCRIPTION	INPUT	PRECONDITION	EXPECTED OUTPUT
Test scenario 001-open	TC-CP-01	Open the website	Open the website or path of webpage	Path	The path page should be open	Open related page open link
Test scenario -002 click	TC-CP-02	Click	Click on the beauty tips button	Click the button	User should click the beauty tips	Open the link
Test scenario-003-mail id	TC-CP-03	Enter	Enter your correct mail id	Click the button	User should enter into the mail id	Enter into valid mail id
Test scenario -004-Mobile number	TC-CP-04	Enter	Enter your mobile number	Enter Mobile number	User should enter into the mobile number	Enter into valid Mobile number
Test scenario -005-booking date and time	TC-CP-05	Choose	Choose the booking date and timing	Enter date and time	User should enter into the booking date and timing	Enter into appropriate date and time

6.4 Website 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 Sweet face aesthetic 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 Sweet face aesthetic:

```
package newselenium;

import org.openqa.selenium.By;

import org.openqa.selenium.JavascriptExecutor;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

public class beauty

{

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

    {

        // TODO Auto-generated method stub

        System.setProperty("webdriver.chrome.driver","C:\\\\Users\\DEEPA\\Downloads\\chromedriver_

win32\\chromedriver.exe");

        WebDriver driver=new ChromeDriver();

        driver.get("file:///C:/Users/DEEPA/Documents/beautysalon.html");

        driver.manage().window().maximize();
```

```
Thread.sleep(3000);
```

```
driver.findElement(By.xpath("/html/body/lable/input")).sendKeys("bharathimano2001@gmail.com");
```

```
driver.findElement(By.xpath("/html/body/lable/lable/input[1]")).sendKeys("6978542475");
```

```
driver.findElement(By.xpath("/html/body/lable/lable/textarea")).sendKeys("20.01.2023");
```

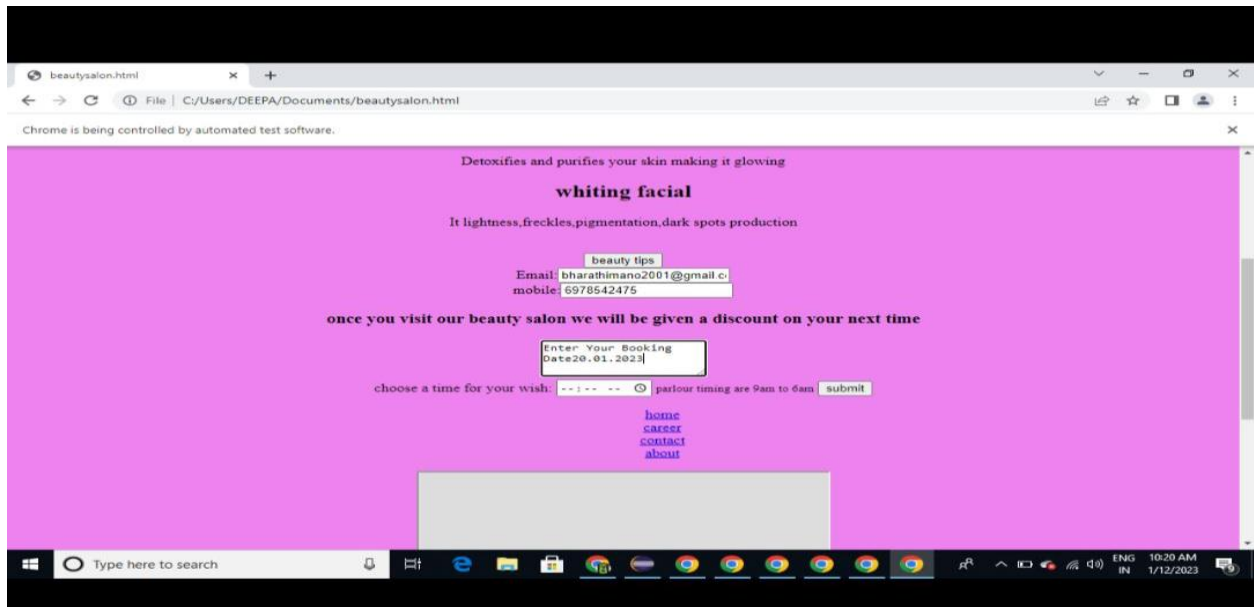
```
driver.findElement(By.xpath("/html/body/lable/lable/input[2]")).sendKeys("11.00 pm");
```

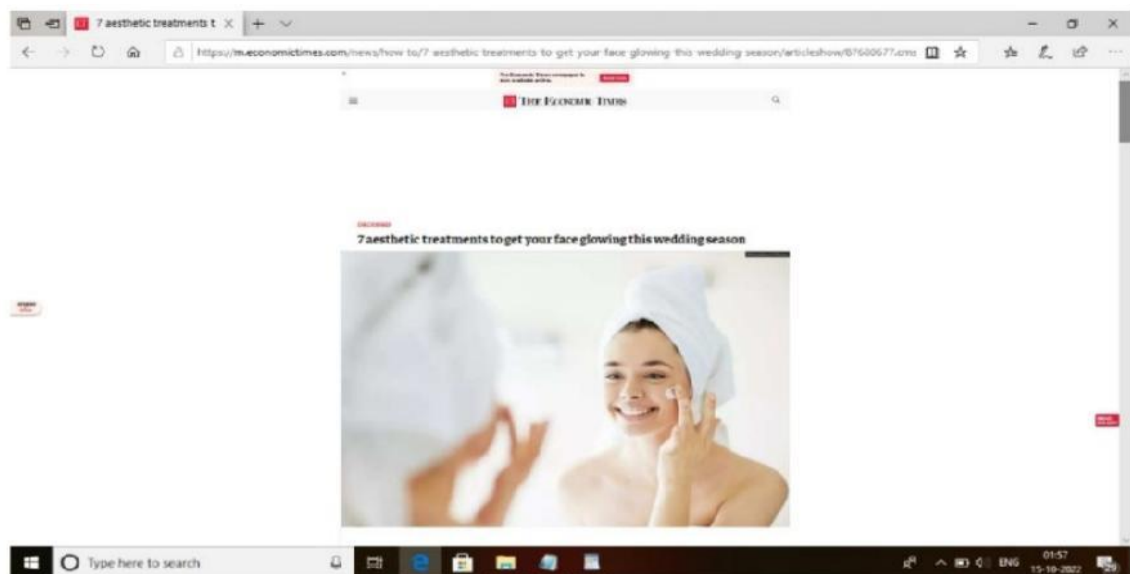
```
driver.findElement(By.xpath("//[@type='submit']")).click();
```

```
}
```

```
}
```

6.6 Output





6.7 Automation testing for Amazon website

```
package seleproject;

import java.util.Iterator;

import java.util.Set;

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

public class mainproject {

    public static void main(String[] args) {

        // TODO Auto-generated method stub

        System.setProperty("webdriver.chrome.driver","C:\\Users\\DEEPA\\Downloads\\chromedriver_
win32\\chromedriver.exe");

        WebDriver driver=new ChromeDriver();

        driver.manage().timeouts().implicitlyWait(30,TimeUnit.SECONDS);

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

        driver.manage().window().maximize();

        driver.findElement(By.id("twotabsearchtextbox")).sendKeys("realme");

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

        driver.get("https://www.amazon.com/OnePlus-Android-Display-Unlocked-
Charging/dp/B0B14LBNL6/ref=sr_1_2_sspa?keywords=realme&qid=1673181221&sr=8-2-
spons&psc=1&spLa=ZW5jcnlwdGVkUXVhbGlmaWVyPUEyN01LRkdaVVQwNzZUJmVuY3
J5cHRlZElkPUEwNTc3NzgXMkJCVCk1OU1ZUIVlKMCZlbmNyeXB0ZWZlZElkPUEwODU5
```

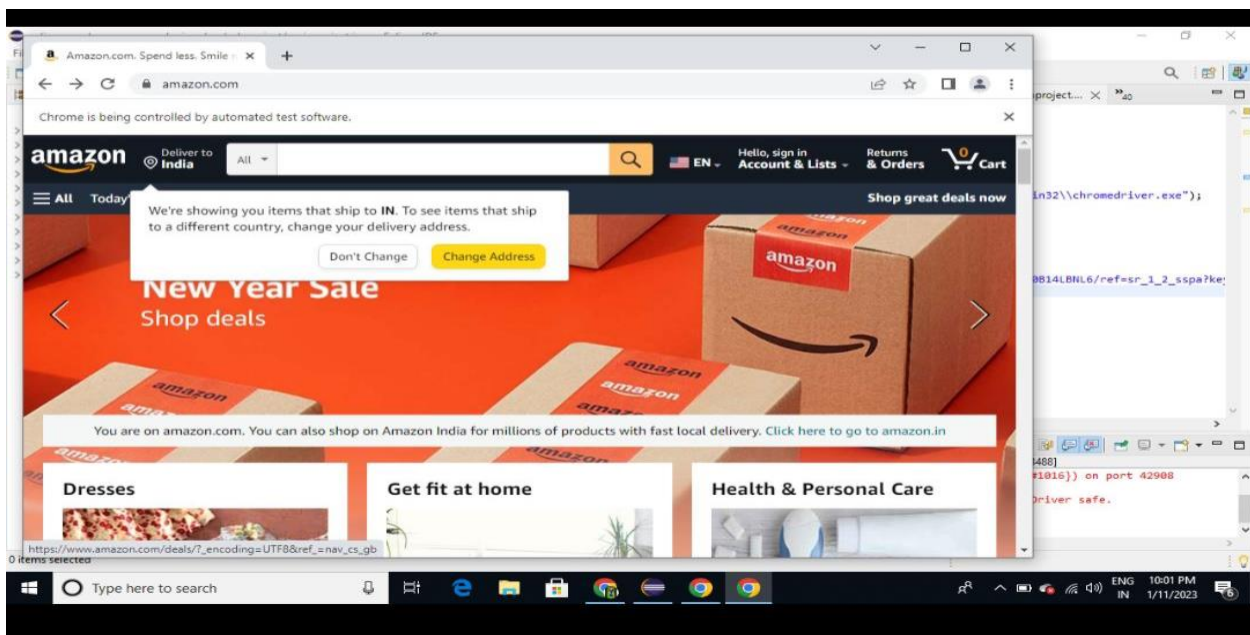
```
NDcwV0tPWFBFRFIDVENQJndpZGdldE5hbWU9c3BfYXRmJmFjdGlvbj1jbGlja1JlZGlyZW  
N0JmRvTm90TG9nQ2xpY2s9dHJlZQ==");
```

```
driver.findElement(By.id("add-to-cart-button")).click();
```

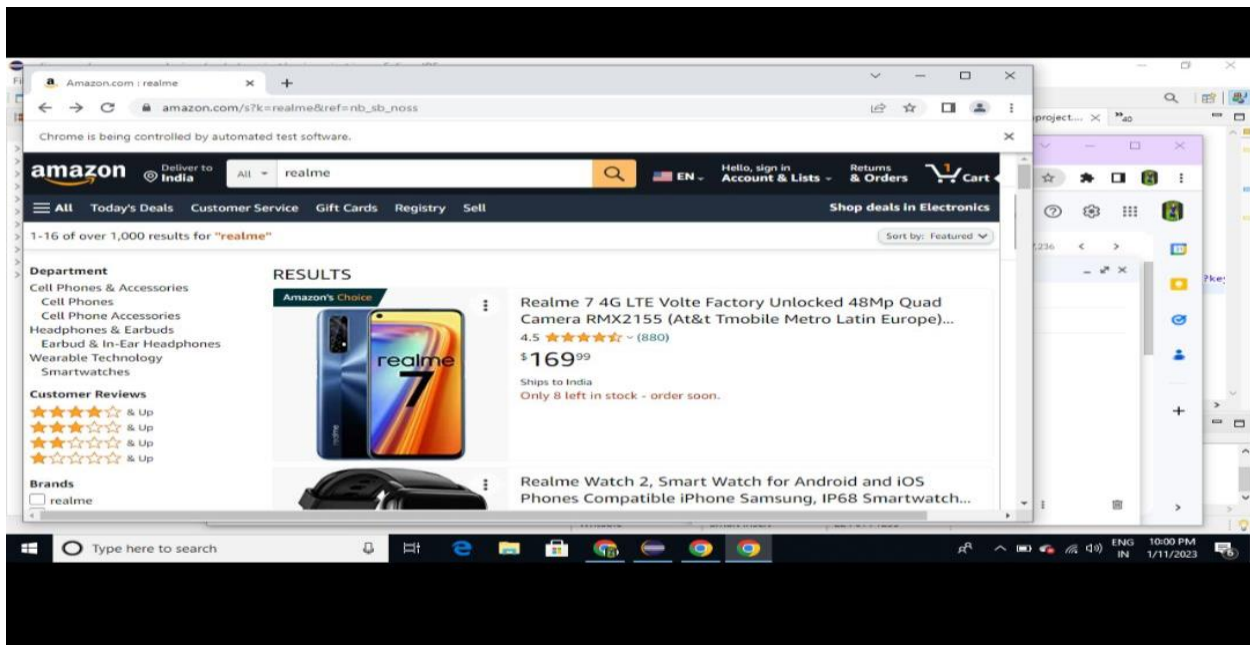
```
}
```

```
}
```

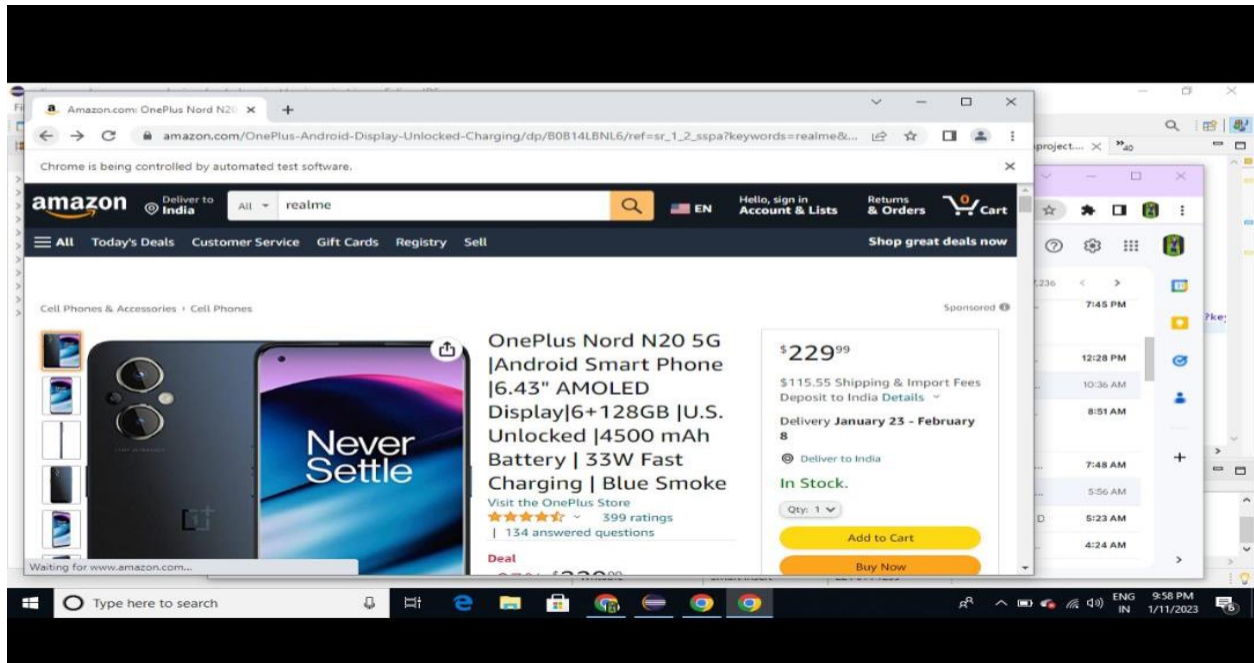
6.8 Output



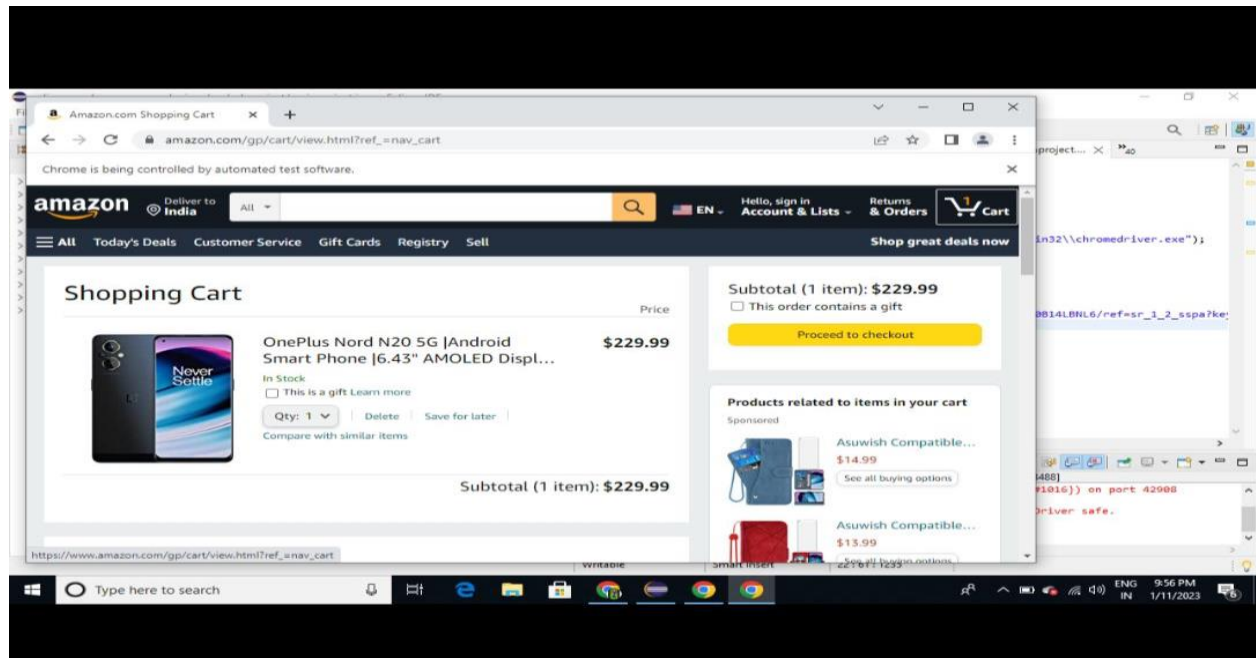
Display the amazon page



Display the realme phone



Display the add to card option



Finally display the add to card for the realme phone

DB CONNECTIVITY

7. Database connectivity:

A database connection is a facility in computer science that allows client software to talk to database server software, whether on the same machine or not. A connection is required to send commands and receive answers, usually in the form of a result set.

Connections are a key concept in data-centric programming. Since some DBMS engines require considerable time to connect, connection pooling was invented to improve performance. No command can be performed against a database without an “open and available” connection to it.

Connections are built by supplying an underlying driver or provider with a connection string, which is a way of addressing a specific database or server and instance as well as user authentication credentials (for example, Server=sql_box;Database=Common;UserID=uid;Pwd=password;). Once a connection has been built it can be opened and closed at will, and properties (such as the command time-out length, or transaction, if one exists) can be set. The Connection String is composed of a set of key/value pairs as dictated by the data access interface and data provider being used.

CODE1

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>
```

```
form site
```

```
</title>
```

```
<style type="text/css">
```

```
form{
```

```
padding-top:70px;
```

```
text-align:center;
```

```
font-size:30px;
```

```
}
```

```
input{
```

```
width:250px;
height:40px;
font-size:30px;
}
</style>
</center>
</head>
<body>
<form method="POST" method="connect.php">
username:<input type="text" name="username"><br></br>
password:<input type="password" name="password"><br></br>
<input type="sumbit" value="sumbit">
</form>
</body>
</html>
```

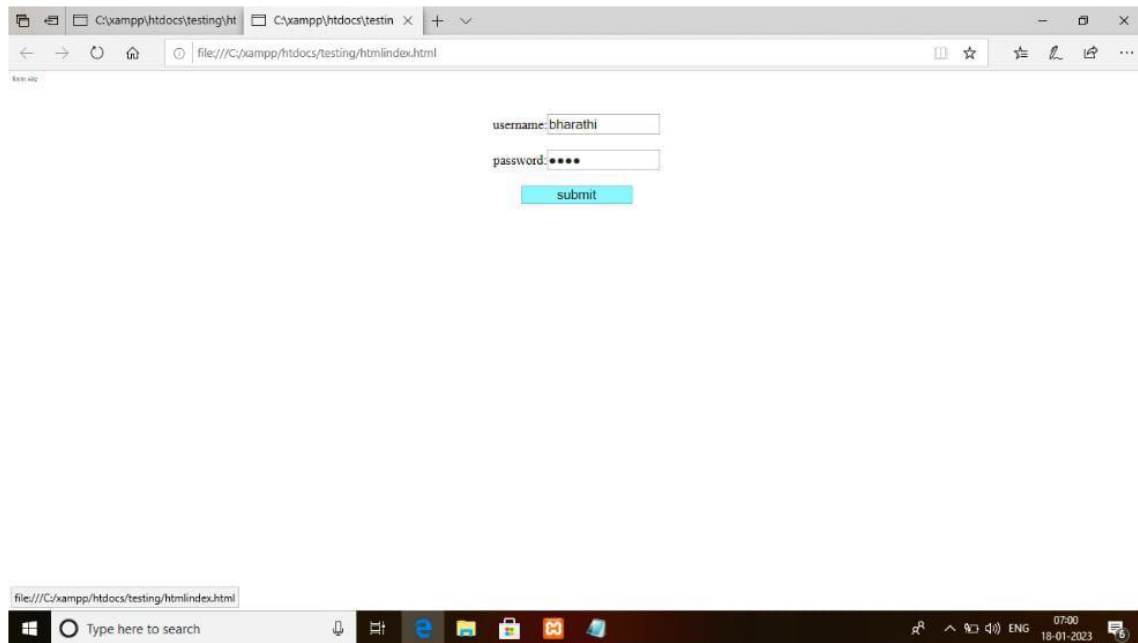
CODE2

```
<?php
$username =filter_input(INPUT_POST, 'username');
$password =filter_input(INPUT_POST, 'password');
if(!empty($username))
{
if(!empty($password))
{
$host="localhost";
$dbusername="root";
$dbpassword="";
$dbname="projectnew";
$conn=new mysqli($host,$dbusername,$dbpassword,$dbname);
if(mysqli_connect_error())
{
die('Connect Error(' . mysqli_connect_error() . ') ' . mysqli_connect_error());
}
else
{
$sql="INSERT INTO form (username,password) values ('$username','$password')";
if($conn->query($sql))
{
echo "New record is inserted sucessfully";
}
else
{
echo"Error: " . $sql . "<br>" . $conn->error;
```



```
}  
$conn->close();  
}  
}  
else  
{  
    echo"Password should not be empty";  
    die();  
}  
}  
else  
{  
    echo"Username should not be empty";  
    die();  
}  
?>
```

7.1 OUTPUT



CUCUMBER

8 Cucumber: Cucumber in Selenium

Cucumber Framework in Selenium is an open-source testing framework that supports Behavior Driven Development for [automation testing of web applications](#). The tests are first written in a simple scenario form that describes the expected behavior of the system from the user's perspective.

The Cucumber Framework: BDD Framework for BDD framework mainly consists of three major parts – **Feature File**, **Step Definitions**, and the **Test Runner File**.

8.1 FEATURE FILE

Cucumber tests are written in these Feature Files that are stored with the extension – “.feature”. A Feature File can be given a description to make the documentation more legible.

Example:

The Login function on a website

Feature File Name: userLogin.feature

Description: The user shall be able to login upon entering the correct username and password in the correct fields. The user should be directed to the homepage if the username and password entered are correct.

Keywords such as GIVEN, WHEN, and THEN used to write the test in Cucumber are called Annotations.

8.2STEP DEFINITION

Now that the features are written in the feature files, the code for the related scenario has to be run. To know which batch of code needs to be run for a given scenario, Steps Definitions come into the picture. A Steps Definitions file stores the mapping data between each step of a scenario defined in the feature file and the code to be executed.

Test runner

To run the test, one needs a Test Runner File, which is a JUnit Test Runner Class containing the Step Definition location and the other primary metadata required to run the test.

Example:

Test Runner Class in cucucucumberskage, with the feature files in “src/test/Feature” location and Step Definition files in the “src/main/step definition” folder.

#Author

#Date

#Description

Feature: feature to test login functionality

Scenario: check login is successful with valid credentials

Given user is on login page

When user enters username and password

And clicks on login button

Then user is navigated to the home page

```
package StepDefinations;
```

```
import io.cucumber.java.en.And;
```

```
import io.cucumber.java.en.Given;
```

```
import io.cucumber.java.en.Then;
```

```
import io.cucumber.java.en.When;
```

```
public class LoginSteps {
```

```
@Given("user is on login page")
```

```
public void user_is_on_login_page() {
```

```
    System.out.println("Inside Step-user is on login page");
```

```
}
```

```
@When("user enters username and password")
```

```
public void user_enters_username_and_password() {
```

```
    System.out.println("Inside Step-user enters username and password");
```

```
}
```

```
@And("clicks on login button")
```

```
public void clicks_on_login_button() {
```

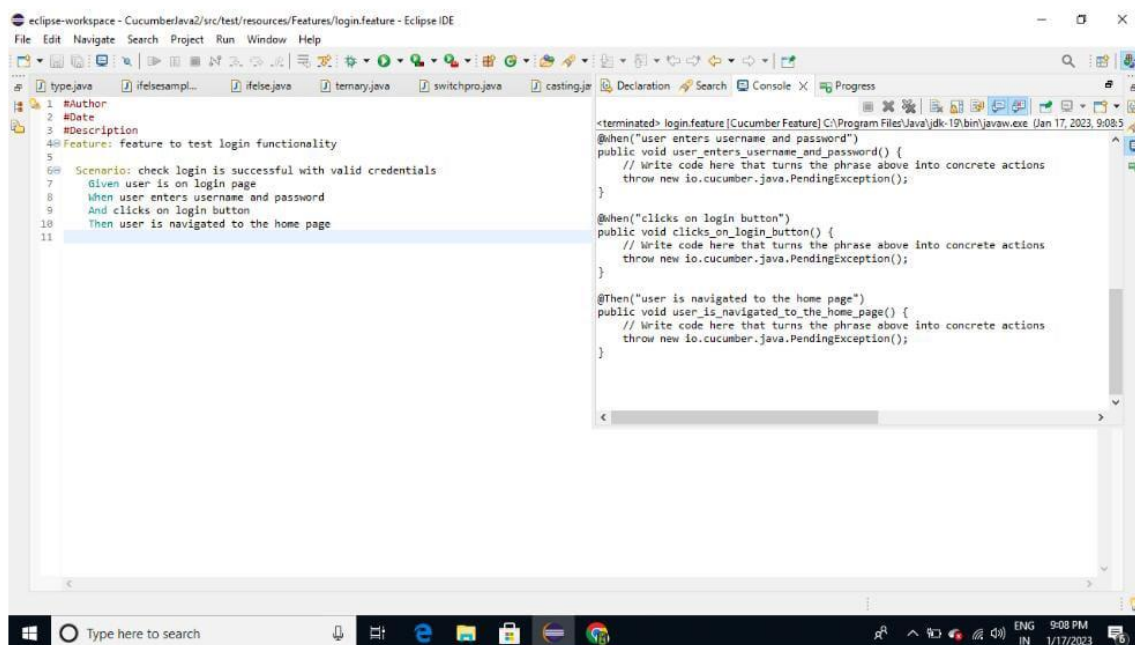
```
    System.out.println("Inside Step-clicks on login button");
```

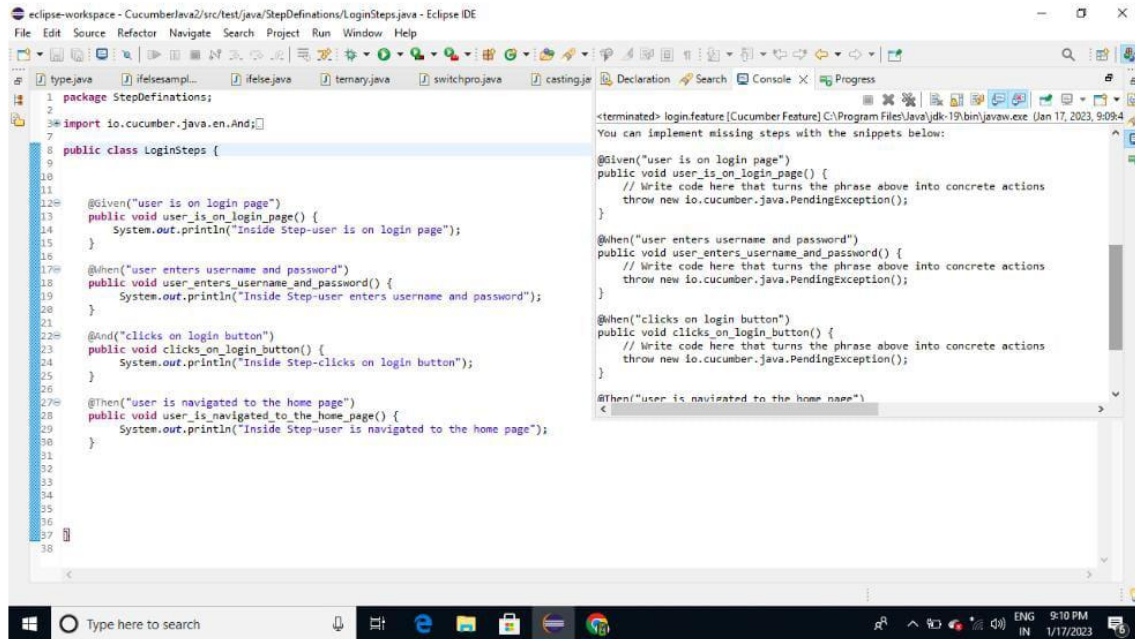
```
}
```

```
@Then("user is navigated to the home page")
```

```
public void user_is_navigated_to_the_home_page() {  
    System.out.println("Inside Step-user is navigated to the home page");  
}
```

8.3 OUTPUT





9. CONCLUSION

The conclusion of this discussion will be bootable for you in this sense if keep it in your mind the er that the beauty salon business plan is a bone for the staff and organization makes it makes available a clear outlook of the business plan for the upcoming era. The beauty salon business plan allows you to determine the budget required for the daily service costs. Inan market individual can judge the position of his company's position through this