# ONLINE CAKE SHOP

A dissertation submitted in partial fulfilment of the requirements for the Award of

the

Course of

**Certified Software Test Engineer**

Submitted By

**Sonal Bhujbal**

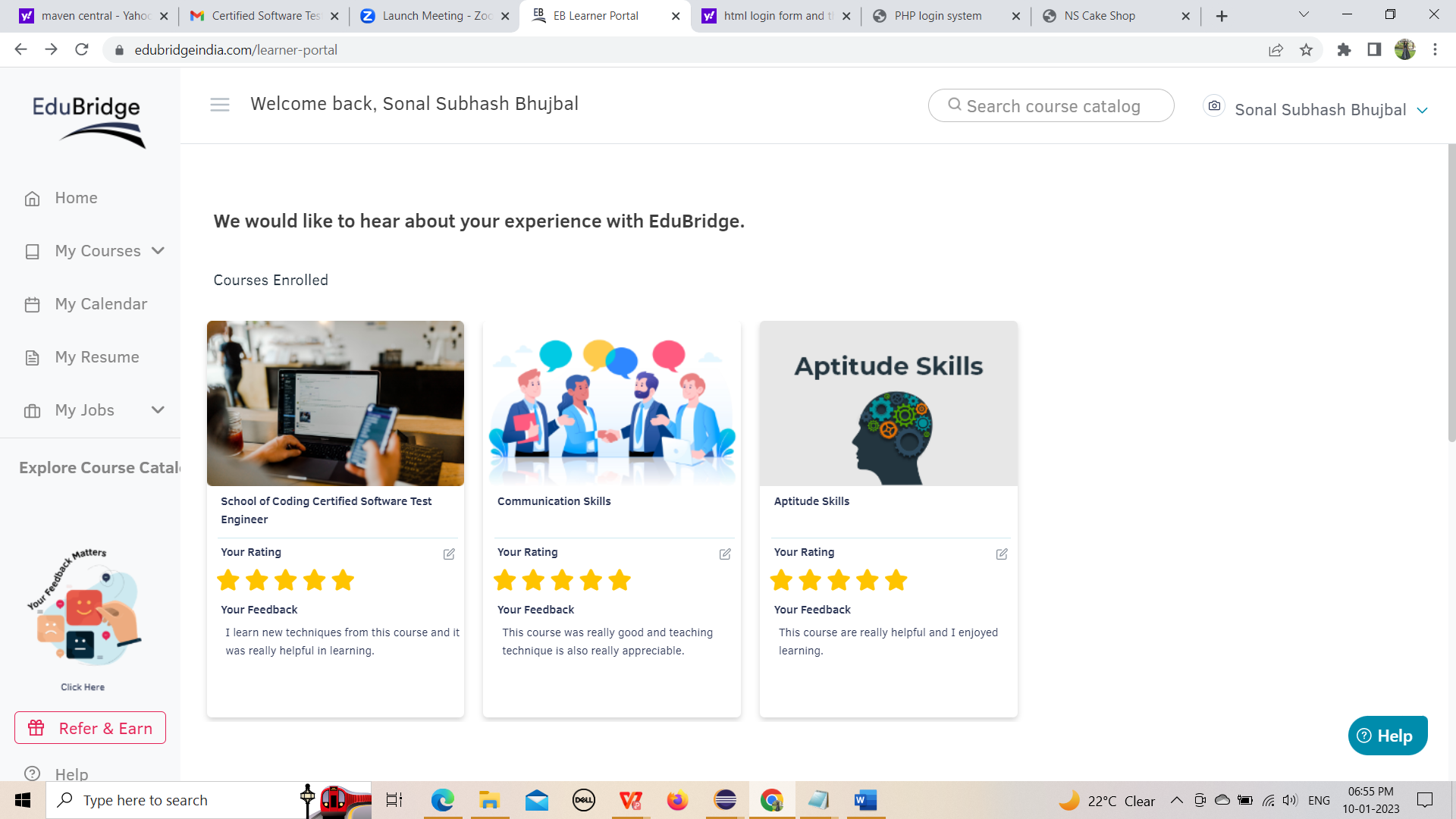
Batch: 2022-7669

Enrollment Number: EBEON0722628265

Guided By

**Ms. Mohana Priya**

Trainer



**CERTIFICATE**

This is to certify that Mrs. Sonal Bhujbal have

Completed the project work on

**“ONLINE CAKE SHOP”**

Under my teacher/trainer guidance & supervision in a satisfactory manner for the Practical/project fulfillment of Course in year 2022-2023.

**Project Guide**

**Ms. Mohana Priya**

Date:

Place:

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Sr.no** | **Description** | **Page no** |
| **1.** | **Introduction** | **5** |
| **2.** | **Scope of the System** | **6** |
| **3.** | **Existing System**  3.1. Existing System  3.2. Advantages  3.3. Disadvantage  3.4. Fact Finding Technique | **7-8** |
| **4.** | **System Analysis:**  4.1 Feasibility Study  4.2 Objective of the System  4.3 Hardware & Software Requirements | **9-10** |
| **5.** | **System Design:**  5.1 Use Case Diagram  5.2 Activity Diagram  5.3 Class Diagram  5.4 ER Diagram  5.5 Data Flow Diagram | **11-16** |
| **6.** | **System Coding:**  6.1 Coding  6.2 Screenshots | **17-47** |
| **7.** | **System Testing:**  7.1 Types of Testing  7.2 Testing Methodologies  7.3 Equivalence partitioning  7.4 Test Reports | **47-52** |

**INTRODUCTION**

It is a small software project which is helpful in the areas of shop. The activities like customer entry, Login, Customer managing, the data summary, Help, and About Cake Shop etc are included.

All the data are saved automatically in the defined place and are retrieve from there directly by using this software. In terms of security this software is totally secure as it doesn't give access to unauthorized user as the username and password facility is provided.

The project consists of list of Cakes and bakery products displayed in various categories. The user may browse through these items as per categories. If the user likes a Product, he may add it to his shopping cart. the online Cake shopping project brings an entire cake shop online and makes it easy for both buyer and seller.

**2.SCOPE OF THE SYSTEM**

The scope of the specification includes the following scenarios:

1. We can use this system in Online Cake shops.
2. This system provides the facility to add the details of Customer, and dealer.
3. This system also provides facility to delete the details of Customer and dealer.
4. System will be ideal information centre for the dealer and customer.

1. System saves lot of time that waste during searching the records.
2. The system reduces the human efforts & error.

vii)The system is user friendly.

**3.1EXISTING SYSTEM**

Consumes more time and effort for updating.

• Reports are not in attractive manner.

• Efficiency and accuracy cannot be expected.

• Quick and timely services are not provided.

• Slow processing speed.

• Ready-made data retrieval is difficult.

**3.2 Advantages**

**1.User-Freindly**

The system has got much simplified screen which makes the system fast enhances quick and accurate data.

**2. Validations Check:**

The system has various validation checks and it also gives appropriate error message.

**3. Report Generation:**

The system can generate various reports giving dealer details, stock details, bill details, Purchase details etc.

**4. Easy to learn:**

This system is easy to understand, this mainly due to simple functionality.

1. Helps cake shops to automate bakery selling online.
2. Helps cake shops to take cc payments.
3. Provides email confirmation on payment success.

**3.3Disadvantages**

This system is design to be standalone & does not work on network site

1. The system is not platform independence.
2. Does not keep track of stock.

**4.SYSTEM ANALYSIS:**

**4.1Feasibility study**

## It is necessary to determine whether the development of new program will be beneficial to the end user/Customer or not. This study examines the aspect like hardware and software requirements, costs and benefits and the time that are needed to develop a program.

1. After doing the project ‘Online Cake Shop’, study and analysing all the existing or required functionalities of the system, the next task is to do the feasibility study for the project. All project is feasible- given unlimited resource and infinite time.
2. Feasibility study includes of all the possible ways to provide a solution to the given problem. the solution should satisfy all the user requirement and should be flexible enough so that future changes can be easily done based on the future upcoming requirement.

**1. Economical Feasibility:**

* 1. This is a very important aspect to be considered while developing a project. we decided the technology based on minimum possible cost factor.
  2. All hardware and software cost has to be borne by the organization.
  3. Overall, we have estimated that the benefits the organization is going to receive from the proposed system will surely overcome the initial costs and the later on running cost for system.

**2. Technical Feasibility:**

* 1. This included the study of function, performance and constraints that may affect the ability to achieve an acceptable system.
  2. For this feasibility study, we studied complete functionality to be provided in the system, as described in the system Requirement Specification (SRS), and checked if everything was possible using different type of front-end and back-end platform.

**3.Operational Feasibility:**

No doubt the proposed system is fully GUI based that is very friendly and all inputs to be taken all self-explanatory.

**4.2 OBJECTIVE OF THE SYSTEM**

1. Increased Speed of work.
2. Decreased cost and increased Profit of Business.
3. Saves time.
4. Quick searching of any product
5. Give all data that we require.
6. Keep security of data.

**4.3 HARDWARE AND SOFTWARE REQUIREMENTS**

### **Hardware Requirement**

Processor : Intel core Duo 1.3 GHz or more

RAM : 256 or more

Hard-disk : 4GB or more

Monitor : LCD Monitor

Keyboard : Normal or Multimedia

Mouse : Compatible mouse

### **Software Requirement**

### Front End: Notepad

Operating System: Windows 10

**5.SYSTEM DESIGN:**

**5.1 Use Case Diagram**

**Administrator**

**Customer**

Use Case Diagram: Online Cake Shop

**5.2 ACTIVITY DIAGRAM**

**Login**

**Authentication**

**Invalid**

**Check**

**View Order Details**

**Edit Profile**

**Choose Flavour**

**Change Password**

**My Account**

**Order Report**

**Add to cart**

**Cancel Order**

**Payment Report**

**Make Payment**

**Confirm Order**

**Sign Out**

**5.3 CLASS DIAGRAM**

|  |
| --- |
| **Customer** |
| +ID: Integer  **#**Password**:** encrypted |
| +Login  +Select Flavour  +Place Order  +Payment |

|  |
| --- |
| **System Order** |
| +Cake List  +Customer id & Password  +Payment Option |

|  |
| --- |
| **Payment** |
| +Amount  +Payment Option |
|  |

|  |
| --- |
| **Cake Shop** |
| +Check the Order  +Delivered  the Order |

|  |
| --- |
| **Bank** |
| -Account No  -Account Type  -Online Payment ID & Password |

|  |
| --- |
| **Cash on Delivery** |
| -Customer Name  -Customer Address  -Amount |

**5.4 ER DIAGRAM**

ITEM

Order

User

Generate Bill

Bill\_No

Bill

**5.5DATA FLOW DIAGRAM**

Category Management

Cake Management

Order Management

Online Cake Shop System

System User Management

Customer Management

Login Management

1. **SYSTEM CODING**

**HTML:**

HTML is an acronym which stands for **Hyper Text Markup Language** which is used for creating web pages and web applications. Let's see what is meant by Hypertext Markup Language, and Web page.

**Hyper Text:** Hyper Text simply means "Text within Text." A text has a link within it, is a hypertext. Whenever you click on a link which brings you to a new webpage, you have clicked on a hypertext. Hyper Text is a way to link two or more web pages (HTML documents) with each other.

**Markup language:** A markup language is a computer language that is used to apply layout and formatting conventions to a text document. Markup language makes text more interactive and dynamic. It can turn text into images, tables, links, etc.

**Web Page:** A web page is a document which is commonly written in HTML and translated by a web browser. A web page can be identified by entering an URL. A Web page can be of the static or dynamic type. **With the help of HTML only, we can create static web pages**.

**Tags in HTML:**

All HTML documents must start with a document type declaration: <!DOCTYPE html>.

The HTML document itself begins with <html> and ends with </html>.

The visible part of the HTML document is between <body> and </body>.

HTML headings are defined with the <h1> to <h6> tags.

<h1> defines the most important heading. <h6> defines the least important heading:

**Index.html**

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8" />

<title>NS Cake Shop</title>

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

<!--[if IE 8]>

<link rel="stylesheet" type="text/css" href="css/ie8.css" />

<![endif]-->

<!--[if IE 7]>

<link rel="stylesheet" type="text/css" href="css/ie7.css" />

<![endif]-->

<!--[if IE 6]>

<link rel="stylesheet" type="text/css" href="css/ie6.css" />

<![endif]-->

</head>

<body>

<div id="header">

<div>

<div>

<div id="logo">

<a href="index.html"><img src="C:\Users\Sonal\Desktop\cake shop\lgo.jpg" alt="Logo"width="300"height="100" allign="top"/></a>

</div>

<div>

<div>

<a href="signup.html">My Account</a>

<a href="index.html">Help</a>

</div>

</div>

</div>

<ul>

<li class="current"><a href="index.html">Home</a></li>

<li><a href="product.html">Online Cake shop</a></li>

<li><a href="about.html">About us</a></li>

<li><a href="contact.html">Contact Us</a></li>

</ul>

<div id="section">

<ul>

<li><a href="index.html">Chocolate</a></li>

<li><a href="product.html">Vanilla</a></li>

<li><a href="about.html">Red-velvet</a></li>

<li class="selected"><a href="index.html">White-Forest</a></li>

<li><a href="index.html">Butterscotch</a></li>

<li><a href="index.html">Rasmalai</a></li>

<li><a href="index.html">Strawberry</a></li>

<li><a href="index.html">Blueberry</a></li>

<li><a href="index.html">Dryfruit</a></li>

</ul>

<a href="index.html"><img src="C:\Users\Sonal\Desktop\cake shop\choc.jpg" alt="Image" width="700" height="400"/></a>

</div>

</div>

</div>

<div id="content">

<div class="home">

<div class="aside">

<center><h1>Welcome to our site</h1></center><br><br>

<p><h2><span>We provide You Online Cake Delivery Services for Saving Your Time and Built Great Experience for Dealing with Us!</span></h2></p>

</div>

<li class="first">

<a href="index.html"><img src="C:\Users\Sonal\Desktop\cake shop\red.jpg" alt="Image" width="300" height="200" /></a>

<a href="index.html"><img src="C:\Users\Sonal\Desktop\cake shop\vf.jpg" alt="Image" width="300" height="200" /></a>

</li>

</ul>

</div>

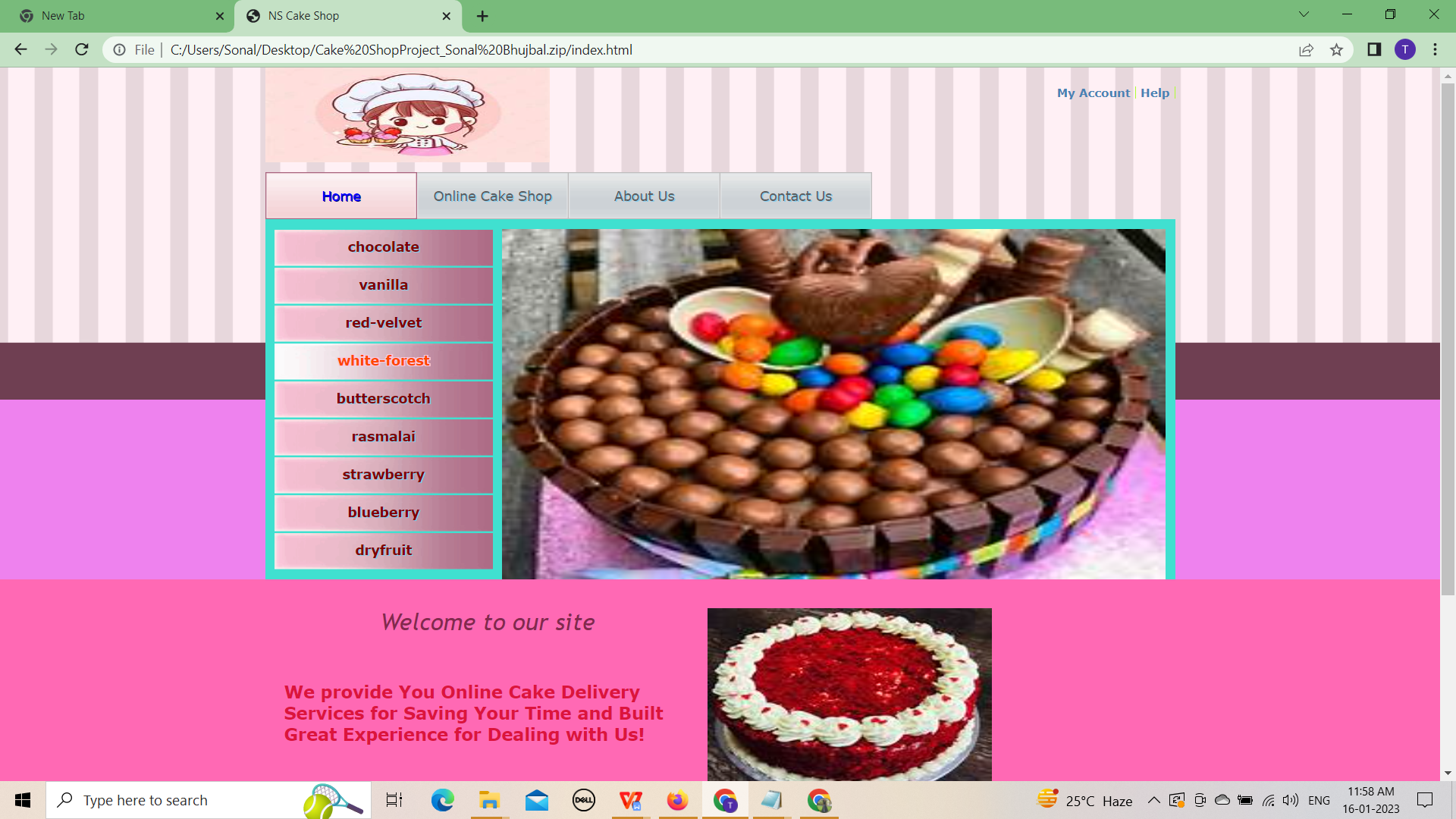
</div>

</div>

</body>

</html>

**Screenshot**



**Product.html**

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8" />

<title>Online Cake Shop</title>

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

<!--[if IE 8]>

<link rel="stylesheet" type="text/css" href="css/ie8.css" />

<![endif]-->

<!--[if IE 7]>

<link rel="stylesheet" type="text/css" href="css/ie7.css" />

<![endif]-->

<!--[if IE 6]>

<link rel="stylesheet" type="text/css" href="css/ie6.css" />

<![endif]-->

</head>

<body>

<div id="header">

<div>

<div>

<div id="logo">

<a href="index.html"><img src="C:\Users\Sonal\Desktop\cake shop\lgo.jpg" alt="Logo"width="300"height="100" align="top"/></a>

</div>

<div>

<div>

<a href="signup.html">My Account</a>

<a href="index.html">Help</a>

</div>

</div>

</div>

<ul>

<li><a href="index.html">Home</a></li>

<li class="current"><a href="product.html">Online Cake Shop</a></li>

<li><a href="about.html">About us</a></li>

<li><a href="contact.html">Contact Us</a></li>

</ul>

<div class="section">

<center><a href="index.html"><img src="C:\Users\Sonal\Desktop\cake shop\r.jpg" alt="Image"width="200"/></a></center>

</div>

</div>

</div>

<div id="content">

<div>

<h1>Online Cake Shop</h1>

<ul>

<li>

<div>

<div>

<h1><a href="index.html">Chocolate cake</a></h1><br>

<br>

</div>

<a href="index.html"><img src="C:\Users\Sonal\Desktop\cake shop\a1.jpg" alt="Image" width="400" height="300" allign="left"/></a>

</div>

</li>

<li>

<div>

<div>

<h1><a href="index.html">Vanilla</a></h1>

</div>

<a href="index.html"><img src="C:\Users\Sonal\Desktop\cake shop\a7.jpg" alt="Image"width="400" height="300" allign="left"/></a>

</div>

</li>

<li>

<div>

<div>

<h1><a href="index.html">Red Velvet</a></h1>

</div>

<a href="index.html"><img src="C:\Users\Sonal\Desktop\cake shop\red.jpg" alt="Image"width="400" height="300" allign="left"/></a>

</div>

</li>

<li>

<div>

<div>

<h1><a href="index.html">White Forest</a></h1><br><br>

</div>

<a href="index.html"><img src="C:\Users\Sonal\Desktop\cake shop\white.jpg" alt="Image"width="400" height="300" allign="right"/></a>

</div>

</li>

<li>

<div>

<div>

<h2><a href="index.html">Butterscotch</a></h2>

</div>

<a href="index.html"><img src="C:\Users\Sonal\Desktop\cake shop\butter.jpg" alt="Image"width="400" height="300" allign="right"/></a>

</div>

</li>

<li>

<div>

<div>

<h2><a href="index.html">Rasmalai</a></h2>

</div>

<a href="index.html"><img src="C:\Users\Sonal\Desktop\cake shop\ras.jpg" alt="Image"width="400" height="300" allign="right"/></a>

</div>

</li>

</ul>

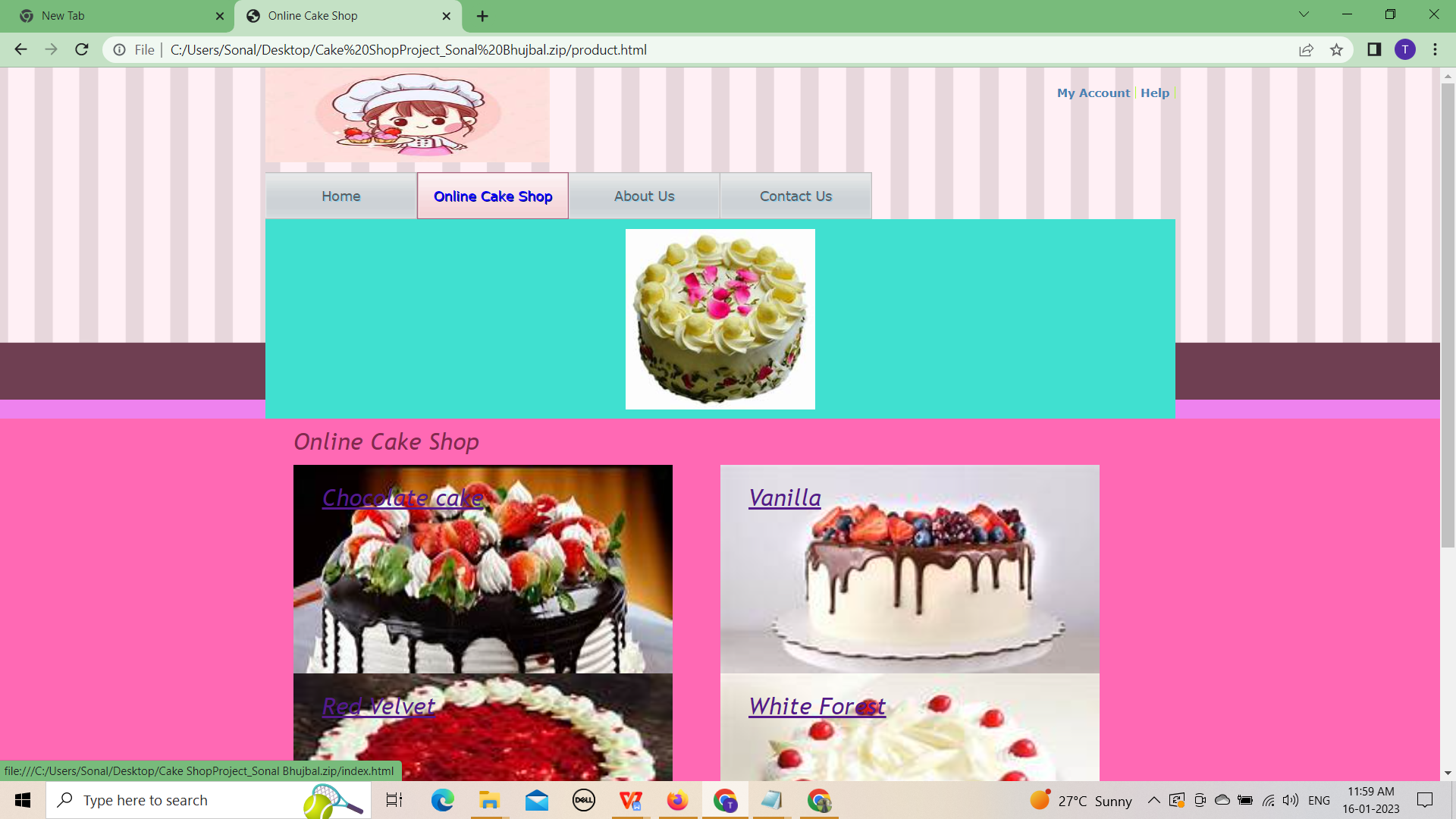
</div>

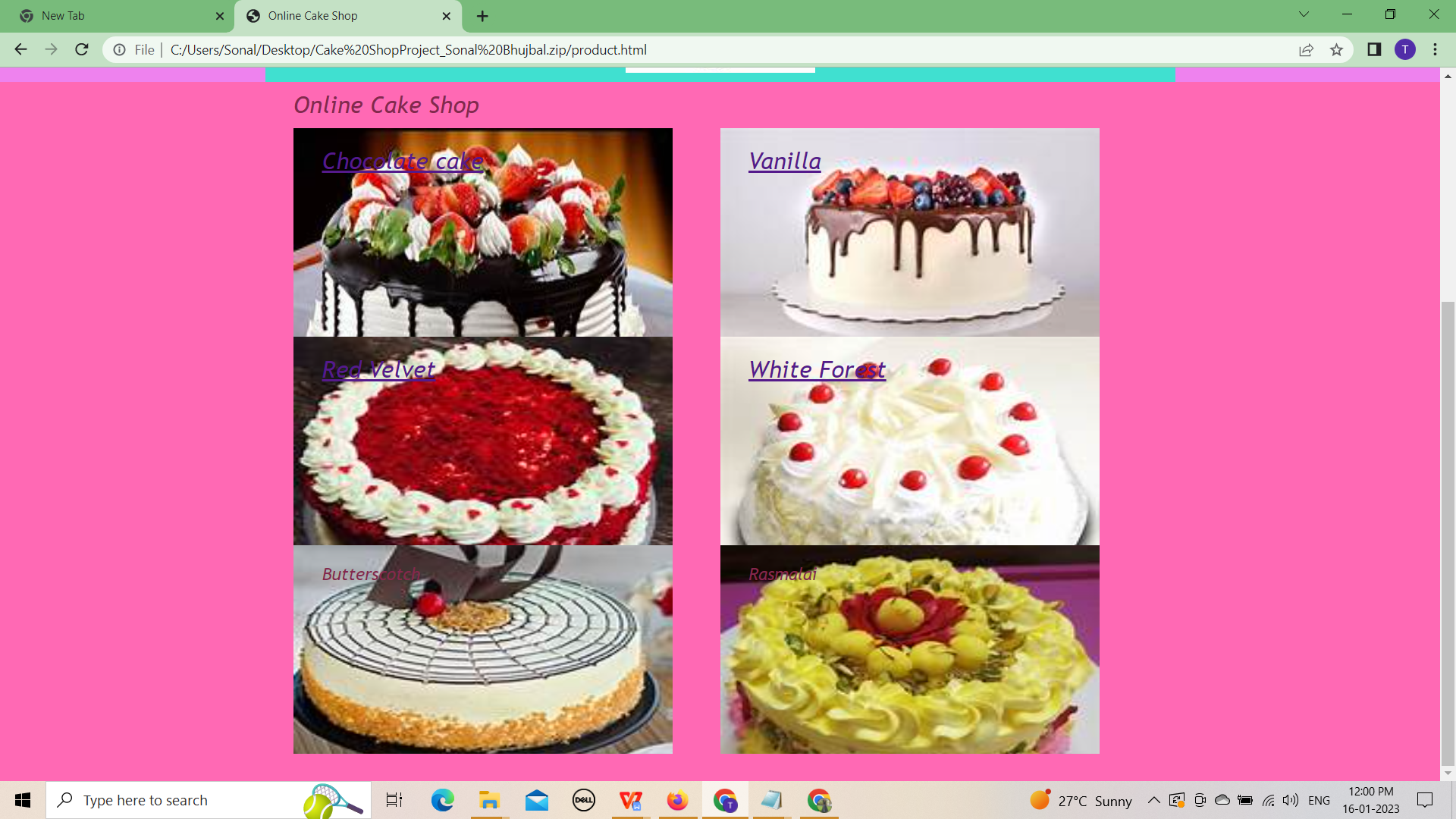
</div></div>

</body>

</html>

**Screenshots:**





CSS: CSS stands for Cascading Style Sheets.

CSS saves a lot of work. It can control the layout of multiple web pages all at once.

Cascading Style Sheets (CSS) is used to format the layout of a webpage.

With CSS, you can control the color, font, the size of text, the spacing between elements, how elements are positioned and laid out, what background images or background colors are to be used, different displays for different devices and screen sizes, and much more!

CSS can be added to HTML documents in 3 ways:

* Inline - by using the style attribute inside HTML elements
* Internal - by using a <style> element in the <head> section
* External - by using a <link> element to link to an external CSS file

**ie6.css:**

#header

div div div div {

height: 1%;

float: right;

padding-bottom: 15px;

width: auto;

}

#header div div#section ul li a {

padding: 0;

}

#content div.home div.section ul li {

margin-top: 10px;

}

#content div div#account div form table tr td label input#rememberme {

vertical-align: middle;

}

#footer div.home div div.section {

height: 271px;

}

#footer div#navigation {

margin-top: 30px;

}

**ie7.css:**

#header div div#section ul li a {

padding: 0;

}

#content div div#account div form table tr td label input#rememberme {

vertical-align: middle;

}

#content div.home div.section ul li {

margin-top: 11px;

}

#footer div.home div div.section {

height: 285px;

}

**ie8.css:**

#content div.home div.section ul li {

margin-top: 11px;

}

**about.html**

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8" />

<title>About -Online Cake Shop</title>

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

<!--[if IE 8]>

<link rel="stylesheet" type="text/css" href="css/ie8.css" />

<![endif]-->

<!--[if IE 7]>

<link rel="stylesheet" type="text/css" href="css/ie7.css" />

<![endif]-->

<!--[if IE 6]>

<link rel="stylesheet" type="text/css" href="css/ie6.css" />

<![endif]-->

</head>

<body>

<div id="header">

<div>

<div>

<div id="logo">

<a href="index.html"><img src="C:\Users\Sonal\Desktop\cake shop\lgo.jpg" alt="logo"width="300"height="100" align="top"/></a>

</div>

<div>

<div>

<a href="signup.html">My Account</a>

<a href="index.html">Help</a>

</div>

</div>

</div>

<ul>

<li><a href="index.html">Home</a></li>

<li><a href="product.html">Online Cake Shop</a></li>

<li class="current"><a href="about.html">About us</a></li>

<li><a href="contact.html">Contact Us</a></li>

</ul>

<div class="section">

<center><a href="index.html"><img src="C:\Users\Sonal\Desktop\cake shop\logo.jpg" alt="Image"width="400"/></a></center>

</div>

</div>

</div>

<div id="content">

<div id="about">

<div class="aside">

<span><h1>About Online cake Shop</h1></span><br><br>

<h1>High quality cakes made to order for weddings,birthdays and special occasions...<h1>

<h1>A speciality cakes are custom made based on your request</h1><br>

<h1>Customize your cake with any flavour and any theme for your special ones.</h1><br>

<a href="index.html"><img src="C:\Users\Sonal\Desktop\cake shop\l.jpg" alt="Image"width="300" height="300" allign="right"/></a>

</div>

<div class="section">

<div id="visitshop">

<div>

<p><span>Sweets from the heart</span> Treat your loveones</p>

<a href="" class="visit">visit the shop</a>

</div>

</div>

<div>

<h3><span>"YOU CAN'T BUY HAPPINESS BUT YOU CAN BUY CAKE AND THAT'S KIND OF THE SAME THING...!"</span></h3>

</div>

<div>

<h3> Branch Address</h3>

<p>Branch-1:Bakori fhata, wagholi <br />Branch-2: Green park, Kharadi <br />Branch-3:Joshi farm, pune <br />Branch-4:Raja rani park, katraj <br />Branch-5:tulsi baag, pune</p>

</div>

<div>

<a href="index.html" class="callus">CALL US <h4>9860066307/7798862592</h4></a>

</div>

</div>

</div>

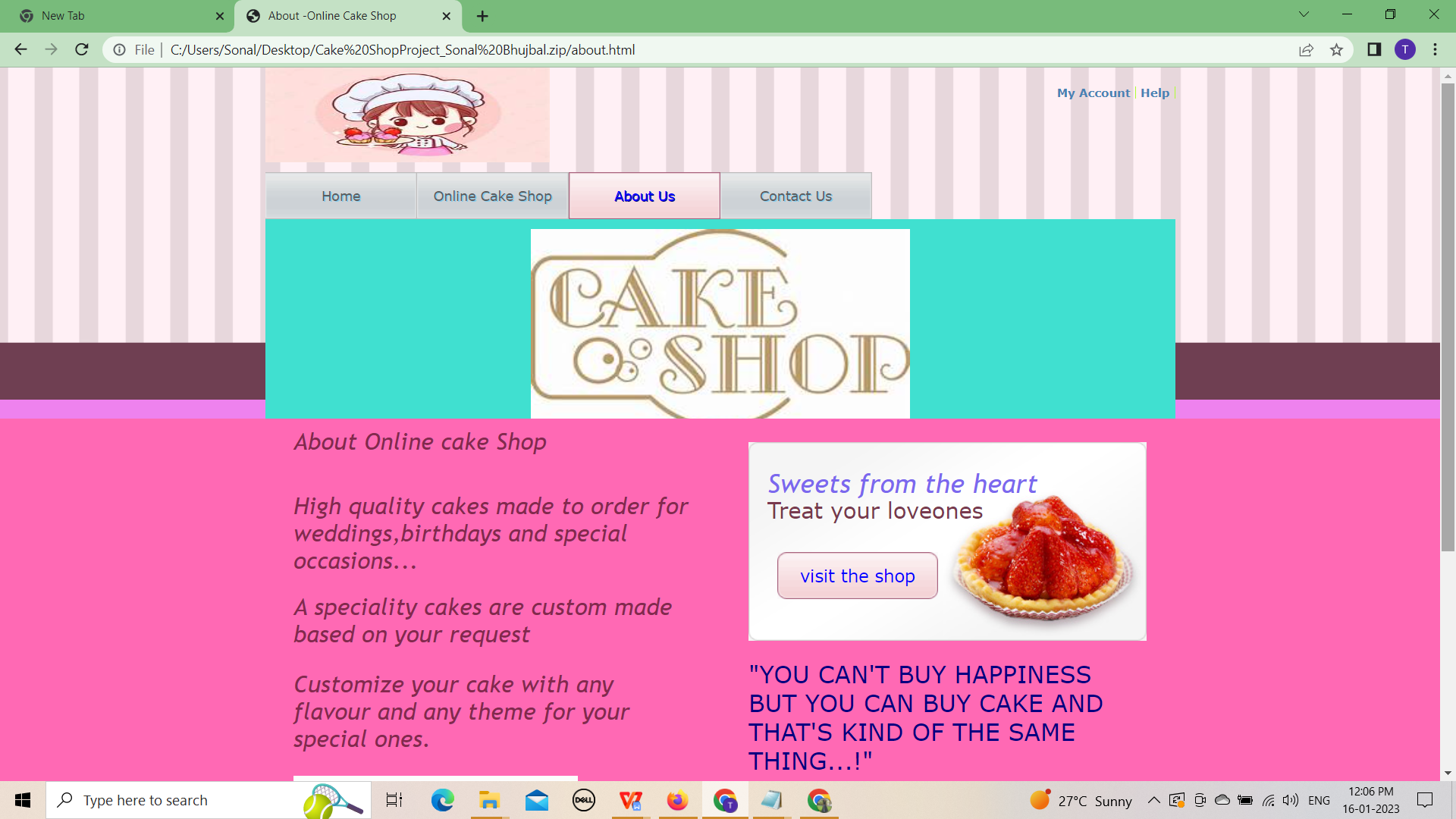
</div>

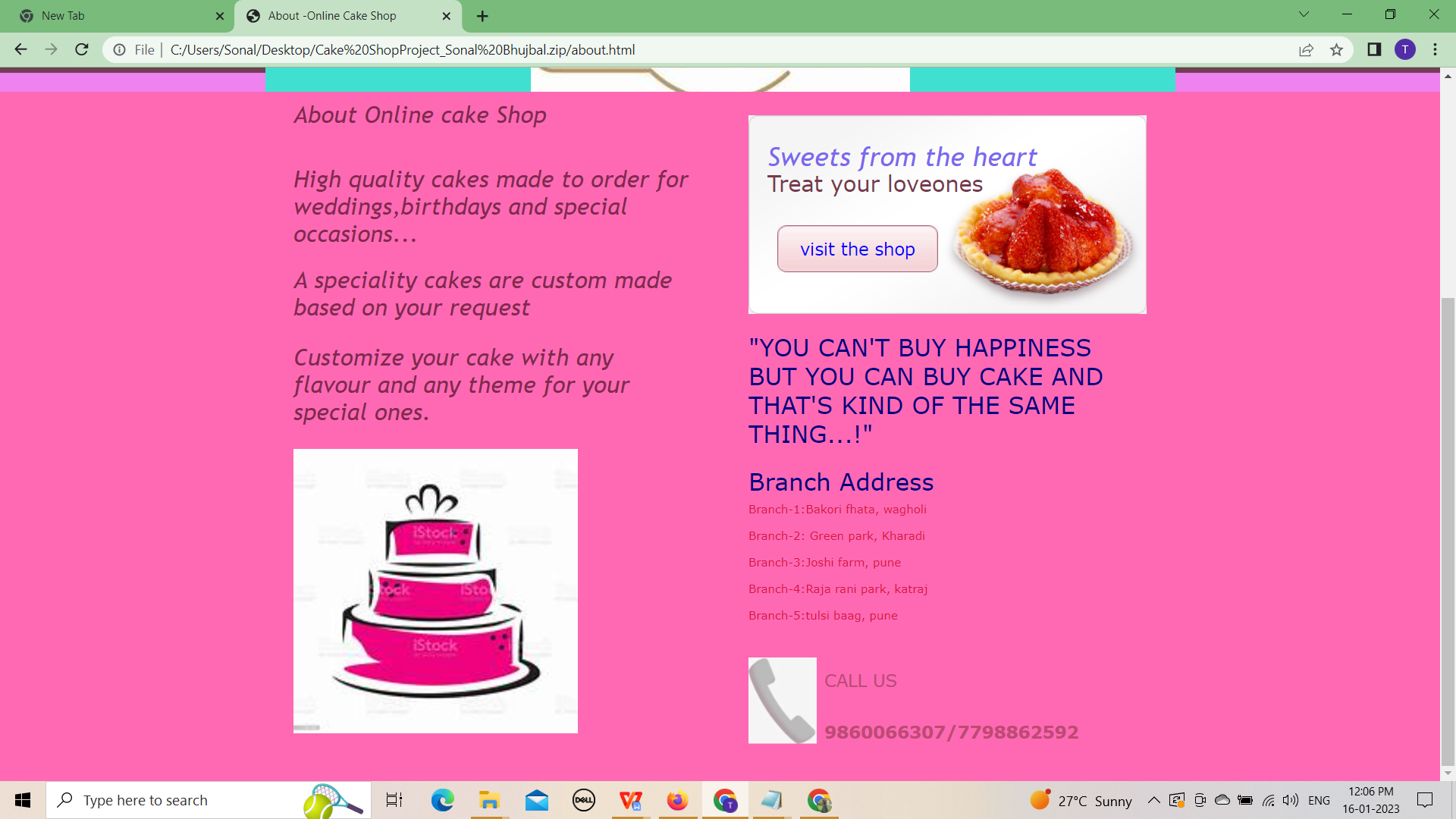
</div>

</body>

</html>

**Screenshots:**





**Contact.html**

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8" />

<title>Contact Us -Online Cake Shop</title>

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

<!--[if IE 8]>

<link rel="stylesheet" type="text/css" href="css/ie8.css" />

<![endif]-->

<!--[if IE 7]>

<link rel="stylesheet" type="text/css" href="css/ie7.css" />

<![endif]-->

<!--[if IE 6]>

<link rel="stylesheet" type="text/css" href="css/ie6.css" />

<![endif]-->

</head>

<body>

<div id="header">

<div>

<div>

<div id="logo">

<a href="index.html"><img src="C:\Users\Sonal\Desktop\cake shop\lgo.jpg" alt="logo"width="300"height="100" align="top"/></a>

</div>

<div>

<div>

<a href="signup.html">My Account</a>

<a href="index.html">Help</a>

</div>

</div>

</div>

<ul>

<li><a href="index.html">Home</a></li>

<li><a href="product.html">Online Cake shop</a></li>

<li><a href="about.html">About us</a></li>

<li class="current"><a href="contact.html">Contact Us</a></li>

</ul>

<div class="section">

<center><a href="index.html"><img src="C:\Users\Sonal\Desktop\cake shop\dry.jpg" alt="Image"width="300"/></a></center>

</div>

</div>

</div>

<div id="content">

<div> <div>

<marquee><h1><p><span>Sweets from the heart</span> Treat your loved ones.</p></h1></marquee>

</div>

<form action="#">

<div>

<center><h2>==========Price list==========</h2>

<table width="40%" id="alignment">

<th>Cakes</th>

<th></th>

<th>Price(Rs.)</th>

</tr>

<tr>

<td>Red Velvet Cake</td>

<td>-</td>

<td>500.00</td>

</tr>

<tr>

<td>White Forest Cake</td>

<td>-</td>

<td>400.00</td>

</tr>

<tr>

<td>Rasmalai Cake</td>

<td>-</td>

<td>350.00</td>

</tr>

<tr>

<td>Vanilla Cake</td>

<td>-</td>

<td>400.00</td>

</tr>

<tr>

<td>Chocolate Cake</td>

<td>-</td>

<td>400.00</td>

</tr>

<tr>

<td>Butterscotch Cake</td>

<td>-</td>

<td>250.00</td>

</tr>

</div>

<div>

</center>

<center><h>Cakes-Order Form</h1>

<table style="background-color:aquamarine;margin-left:50px;"><tr>

<td>Cakes:</td>

<td>

<select>

<option>--Select Flavour--</option>

<option>Chocolate Cake</option>

<option>White Forest Cake</option>

<option>Rasmalai Cake</option>

<option>Red Velvet Cake</option>

<option>Vanilla Cake</option>

<option>Butterscotch Cake</option>

</select>

</td>

</tr>

<tr>

<td>User Name:</td>

<td><input type="text">

</td>

</tr>

<tr><td>Mobile No: </td>

<td><input type="text" id="mblenmb"

onfocusout="checkmob()"></td>

</tr>

</table>

<br>

</center>

<center>

<form name="tblform" >

<table width="80%" id="alignment" >

<tr>

<th>Food Item</th>

<th> </th>

<th>Quantity</th>

<th>Price</th>

</tr>

<tr>

<td>Item 1:</td>

<td>

<select id="Cake1">

<option value="1">--Select Item--</option>

<option value="Chocolate Cake">Chocolate Cake</option>

<option value="Rasmalai Cake">Rasmalai Cake</option>

<option value="Vanilla Cake">Vanilla Cake</option>

<option value="White Forest Cake">White Forest Cake</option>

<option value="Butterscotch Cake">ButterScotch Cake </option>

<option value="Red Velevet Cake">Red Velvet Cake </option>

</select>

</td>

<td><input type="number" value="0" id="q1"></td>

<td><input type="text" id="p1"></td>

</tr>

<tr>

<td>Item 2:</td>

<td>

<select id="Cake2">

<option value="1">--Select Item--</option>

<option value="Chocolate Cake">Chocolate Cake</option>

<option value="Rasmalai Cake">Rasmalai Cake</option>

<option value="Vanilla Cake">Vanilla Cake</option>

<option value="White Forest Cake">White Forest Cake</option>

<option value="Butterscotch Cake">ButterScotch Cake </option>

<option value="Red Velevet Cake">Red Velvet Cake </option>

</select>

</td>

<td><input type="number" value="0" id="q2"></td>

<td><input type="text" id="p2"></td>

</tr>

<tr>

<td></td>

<td></td>

<h2><td><input type="submit" class="button" value="Create Bill" onclick=createBill()></td>

<td><input type="reset" class="button" value="Clear"></td></h2>

</tr>

</table>

</form>

</center>

<script src="Cakes.js"></Script>

</div>

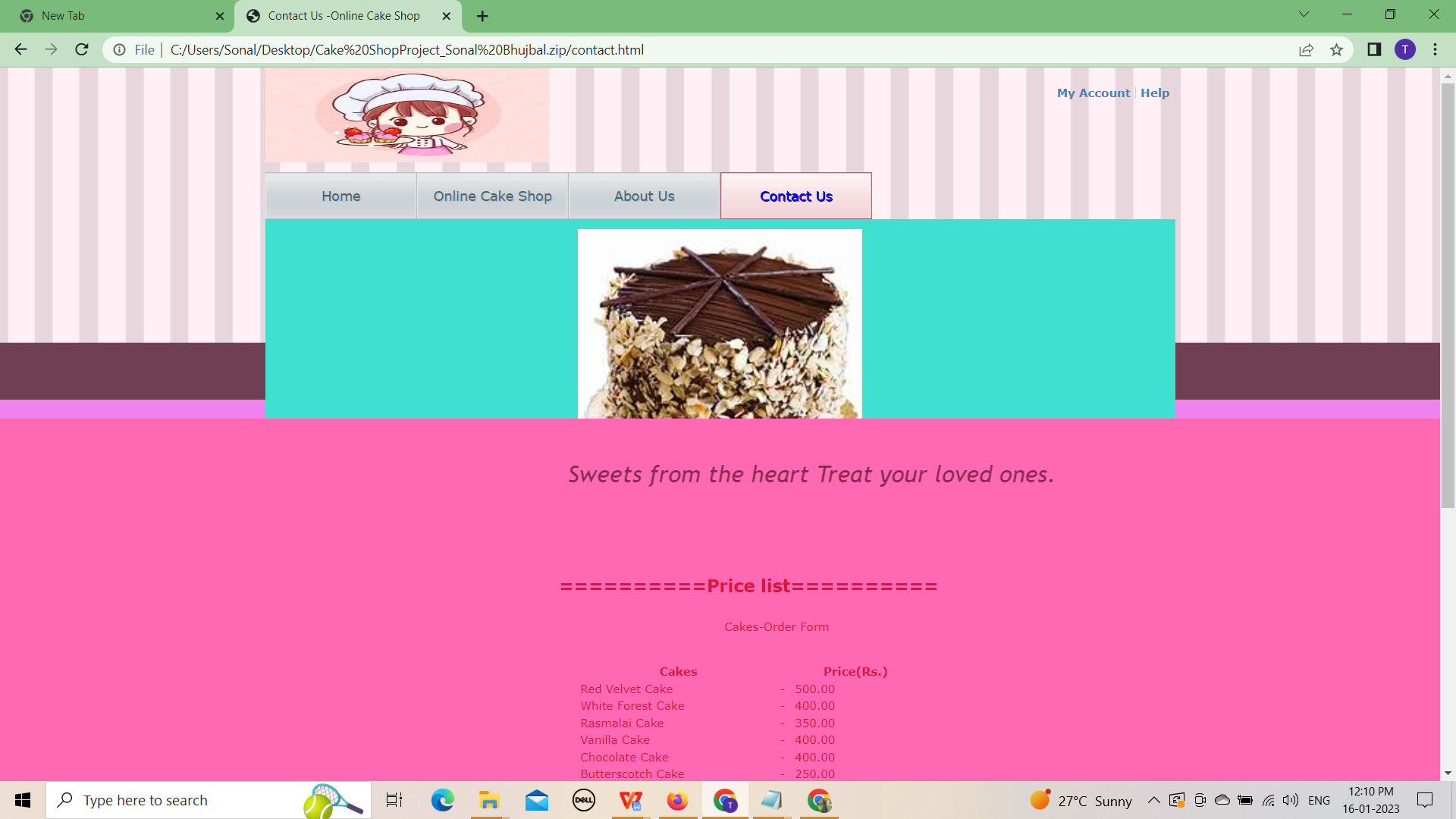
</div></div>

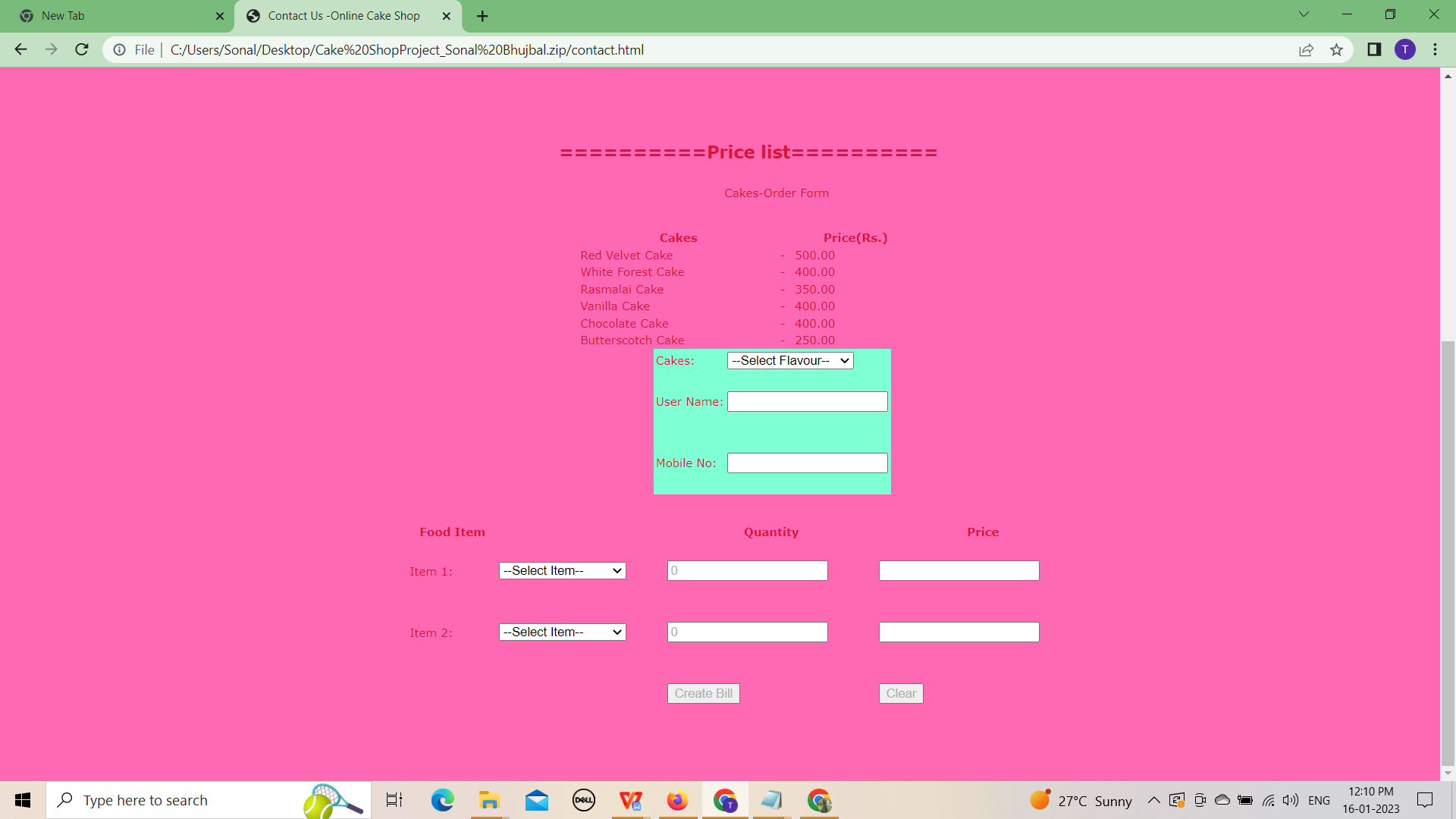
</div>

</body>

</html>

**Screenshots:**





**JavaScript:**

JavaScript is the Programming Language for the Web.

JavaScript can update and change both HTML and CSS.

JavaScript can calculate, manipulate and validate data.

**abc.js:**

<html>

<head>

<title>Login</title>

<script type=”text/javascript” language=”JavaScript” > {

var usname = document.getElementById(‘tbuname’).value;

var pwd = document.getElementById(‘tbpwd’).value;

var cn = new ActiveXObject(“ADODB.Connection”);

var strConn = “Provider=Microsoft.Jet.OLEDB.4.0;

Data Source = C:/my\_db.mdb“;

var rs = new ActiveXObject(“ADODB.Recordset”);

var SQL = “select password from user\_master where uname = ‘” + usname + “‘”;

cn.Open(strConn);

rs.Open(SQL, cn);

if(!rs.eof)

{

if(rs.fields(0).value == pwd)

location.href=’main.htm?uname=’+usname;

else

alert(‘wrong username or password!’);

}

else

alert(‘wrong username or password!’);

rs.Close();

cn.Close();

}

</script>

</head>

<body style=”margin:0;”>

<table width=”1000px” cellpadding=”0″ cellspacing=”0″> 35. <tr>

<td style=”height:200px;background

image:url(images/top.jpg);”>

&nbsp;

</td>

</tr>

<tr>

<td>

&nbsp;

</td>

</tr>

<tr>

<td>

Username : <input type=”text” id=”tbuname” name=”tbuname” /> 48. <br />

Password : <input type=”text” id=”tbpwd” name=”tbuname” /> 50. </td>

</tr>

<tr>

<td>

<input type=”button” id=”btnsbt” name=”btnsbt” value=”Login” onclick=”checkLogin()” /><br />

<br />

<a href=”registration.htm”>Register user</a>

&nbsp;</td>

</tr>

</table>

</body>

</html>

**signup.html**

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8" />

<title>Online Cake Shop</title>

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

<!--[if IE 8]>

<link rel="stylesheet" type="text/css" href="css/ie8.css" />

<![endif]-->

<!--[if IE 7]>

<link rel="stylesheet" type="text/css" href="css/ie7.css" />

<![endif]-->

<!--[if IE 6]>

<link rel="stylesheet" type="text/css" href="css/ie6.css" />

<![endif]-->

</head>

<body>

<div id="header">

<div>

<div>

<div id="logo">

<a href="index.html"><img src="C:\Users\Sonal\Desktop\cake shop\lgo.jpg" alt="Logo"width="300"height="100" align="top"/></a>

</div>

<div>

<div>

<a href="signup.html">My Account</a>

<a href="index.html">Help</a>

</div>

</div>

</div>

<ul>

<li><a href="index.html">Home</a></li>

<li class="current"><a href="product.html">Online Cake Shop</a></li>

<li><a href="about.html">About us</a></li>

<li><a href="contact.html">Contact Us</a></li>

</ul>

<div class="section">

<center><a href="index.html"><img src="C:\Users\Sonal\Desktop\cake shop\logo.jpg" alt="Image"width="300"/></a></center>

</div>

</div>

</div>

<div id="content">

<div>

<div id="account">

<div>

<form action="#">

<span>SIGN-In</span>

<table>

<tr>

<td>

</td>

<td align="left" valign="top"></td>

</tr>

<tr>

<td><label for="email">E-mail</label></td>

<td><input type="text" id="email" /></td>

</tr>

<tr>

<td><label for="login">Login</label></td>

<td><input type="text" id="login" /></td>

</tr>

<tr>

<td><label for="password">Password</label></td>

<td><input type="text" id="password" /></td>

</tr>

<tr>

<td><label for="confirmpass">Confirm<br />Password</label></td>

<td><input type="text" id="confirmpass" /></td>

</tr>

<tr>

<td></td>

<td class="rememberme"><label for="rememberme"><input type="checkbox" id="rememberme" /> Remember me on this computer</label></td>

</tr>

</table>

<center><button type="button" id="myBtn">Sign-In</button></center>

<script src="Cakes.js"></Script>

</form>

</div>

</div>

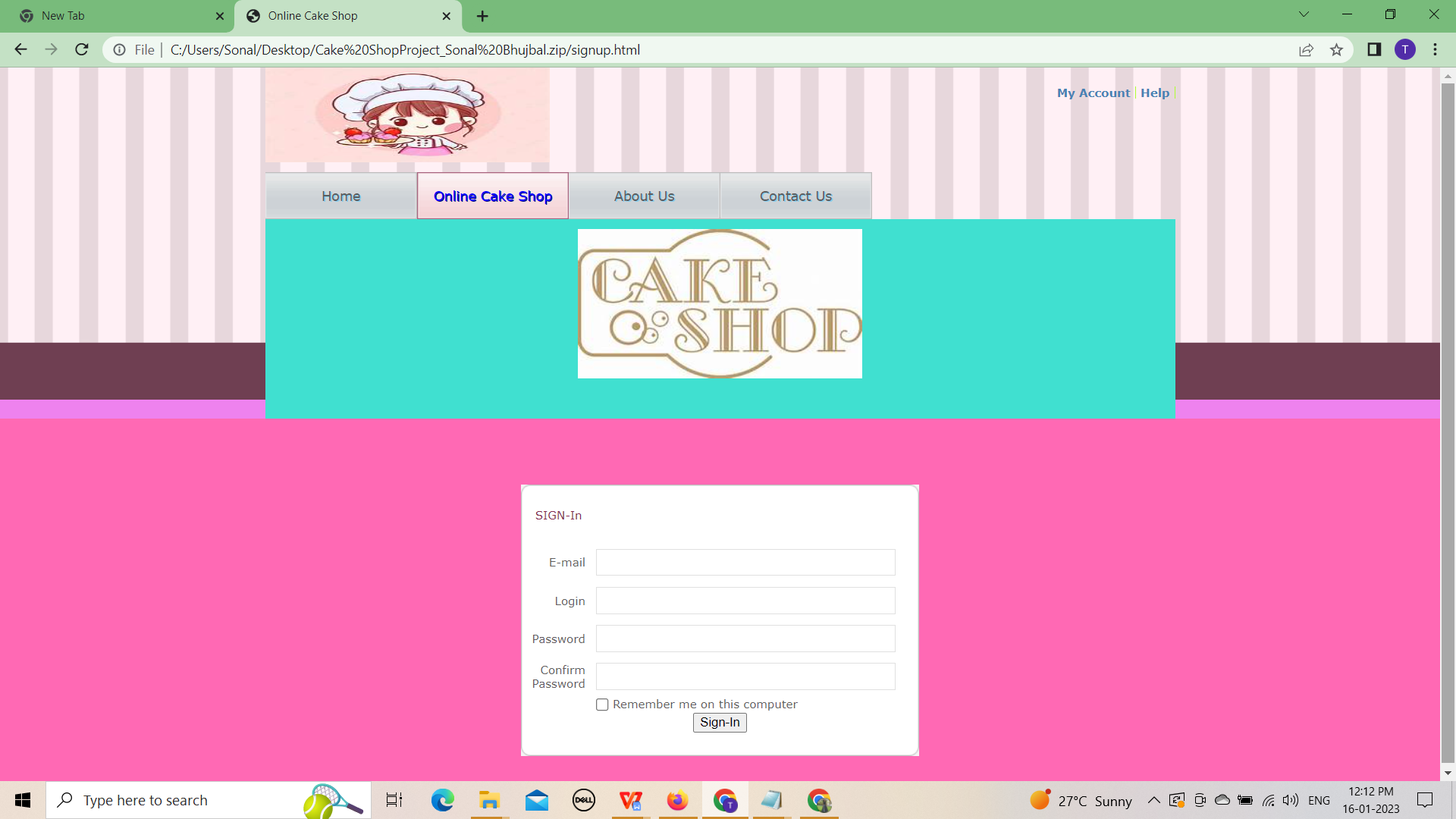
</div>

</div>

</body>

</html>

**Screenshots:**



**6.SYSTEM TESTING**

**6.1 Types of Testing:**

1. **Manual Testing**
2. **Automation Testing**

**1.Manual Testing:** Testing any software or an application according to the client's needs without using any automation tool is known as **manual testing**.

1. **White Box Testing:** In white-box testing, the developer will inspect every line of code before handing it over to the testing team or the concerned test engineers.
2. **Black Box Testing:** **black-box testing**. In this testing, the test engineer will analyze the software against requirements, identify the defects or bug, and sends it back to the development team.
3. **Grey Box Testing:** It is a **collaboration of black box and white box testing**.

**2.Automation Testing:** Automation testing is the best way to enhance the efficiency, productivity, and coverage of Software testing.

It is used to re-run the test scenarios, which were executed manually, quickly, and repeatedly.

1. **Smoke Testing:** In **smoke testing**, we will test an application's basic and critical features before doing one round of deep and rigorous testing.
2. **Sanity Testing:** It is used to ensure that all the bugs have been fixed and no added issues come into existence due to these changes. Sanity testing is unscripted, which means we cannot document it. It checks the correctness of the newly added features and components.
3. **Regression Testing:** Regression testing is the most commonly used type of software testing. Here, the term **regression**implies that we have to re-test those parts of an unaffected application.
4. **User Acceptance Testing:** In user acceptance testing, we analyse the business scenarios, and real-time scenarios on the distinct environment called the **UAT environment.** In this testing, we will test the application before UAI for customer approval.
5. **Security Testing:** It is an essential part of software testing, used to determine the weakness, risks, or threats in the software application.

**6.2 Testing Methodologies:**

Software Testing methodologies are the approaches and strategies used for testing a precise product to make sure it is fit for purpose. It generally entails testing that the product functions along with its specification, has no unwanted side effects when used in modes outside of its design parameters, and the most horrible case, will fail safely.

**The different types of testing methodologies below.**

#### **1.Waterfall Model:** This model for testing works perfectly for small, less complicated projects and is built on a team’s step-by-step growth during the test procedure. As it has fewer players and procedures to tend with, this can result in speedy project completion. But bugs are found at later phases, making them extremely costly.

#### **2.Agile Model:** The [Agile model](https://www.browserstack.com/guide/continuous-integration-with-agile) is dissimilar from the waterfall methodology and is well-suited for big development projects.

1. An agile test is an incremental model where tests are executed at the end of each iteration or increment.
2. It can cover the sphere of tests and software development, and marketing.
3. The outcomes are better with agile methodology when an experienced and strong product Manager can make rapid decisions.

#### **3.Iterative Model:** In the iterative methodology, software developers form basic versions of the software and review & enhance the app in iterations smaller components or steps. It is [data-driven](https://www.browserstack.com/guide/data-driven-framework-in-selenium), and all iteration is based on the outcomes of the previous test cycle.

#### **4.Verification and Validation Methodology (V-Model):** The Verification and Validation method is considered an extension of the waterfall model.

1. It is a step-by-step software test model for small projects with defined software necessities.
2. Follows a ‘V-shaped’ methodology categorized into coding, authentication, and validation.
3. Every development stage goes hand-in-hand with testing, resulting in the early uncovering of bugs at every step.

#### **5.Spiral Model:** In this Quality Assurance technique, the spiral model, incorporates the waterfall and iterative development approaches. It is parallel to the incremental methodology with more concentration on threat analysis. The varied stages of the Spiral model comprise the planning stage, risk analysis, assessment, engineering stage, etc.

#### **Extreme Programming Model:** In this Extreme Programming (XP) methodology, the programmer codes an easy code to obtain feedback on the user’s experience. This approach is based on an agile method that breaks the jobs into smaller sections. After every section is finished, the next section is operated on. This is used where the requirements of users are continuously evolving.

#### **6.3.Equivalence Partitioning:**

#### 

**Equivalence Partitioning Method** is also known as Equivalence class partitioning (ECP). It is a [software testing](https://www.geeksforgeeks.org/software-testing-basics/) technique or [black-box testing](https://www.geeksforgeeks.org/software-engineering-black-box-testing/) that divides input domain into classes of data, and with the help of these classes of data, test cases can be derived. An ideal test case identifies class of error that might require many arbitrary test cases to be executed before general error is observed.

In 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.

**Selenium Code for Checking Login Page:**

package selenium1p;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

public class online1{

public static void main(String[] args) {

System.setProperty("webdriver.gecko.driver",

"C:\\Users\\sonal\\Downloads\\chromedriver\_win32\\chromedriver.exe");

WebDriver driver = new ChromeDriver();

//URL launch

driver.get("file:///C:/Users/Sonal/Desktop/Sonal\_Bhujbal/Cake%20ShopProject\_Sonal%20Bhujbal.zip/signup.html");

//identify username

WebElement l = driver.findElement(By.name("username"));

l.sendKeys("Pass@gmail.com");

//identify password

WebElement p = driver.findElement(By.name("password"));

p.sendKeys("Pass@123");

WebElement b = driver.findElement(By.className("Igw0E"));

b.click();

//obtain value entered for username

String s = l.getAttribute("value");

System.out.println("Value entered for username: " + s);

//quit browser

driver.quit();

}

}

**6.4 TEST REPORTS**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sr.no** | **Test Case ID** | **Feature** | **Description** | **Steps to Execute** | **Test Data/Input** | **Expected Result** |
| 1. | TC\_01 | First Name | 1.Verify that the name field design should be have as per the customer’s Specification. | 1.Enter the name in start with Capital Letter | 1.Sonal | It Should not show any validation Message |
| 2. | TC\_2 | Last Name | 1.Verify that the name field design should be have as per the customer’s Specification. | 1.Enter the name in start with capital letter | 1.Bhujbal | It Should not show any validation Message. |
| 3. | TC\_3 | Email Validation | Check the Email Text Fields with an Email Address without @ Symbol  1. Check the Email text Field with a random string instead of a real email. 2. Check the Email | Enter Invalid Emails.  1. Click on the Register Button. | testAtgmail.com  1. test@gmail 2. @gmail | It should show the validation message for valid emails. |
| 4. | TC\_4 | Email validation | Check all the valid emails | 1. Enter valid Emails 2. Click on the Register Button. | 1. [test.22@gmail.com](mailto:test.22@gmail.com) 2. [test@gmail.com](mailto:test@gmail.com) | It should not show any validation message |
| 5. | TC\_5 | Phone Number validation | Check the phone number when passing alphanumeric data | 1. Enter alphanumeric data in the Phone field 2. Click on Register button | 1. dada5$7567#7 | It should show the validation message for Phone Number |
| 6. | TC\_6 | Phone Number validation | Check the phone number when not pass country code | 1. Enter a valid phone number without country code 2. Click on the Register button | 1.9850930359 | It should show the validation message for country code is required |
| 7. | TC\_7 | Phone Number validation | Check the phone number when passing country code | 1. Enter a valid phone number with country code 2. Click on Register button | 1. +91985093035 | It should not show any validation error message |
| 8. | TC\_8 | Phone Number validation | Verify if the phone number length is incorrect, i.e., more than ten characters | 1. Enter a phone number with less than ten digits. 2. Enter all the required fields. 3. Click on Register Button | 9850930359 | It should show the validation error message for phone number length. |