

# **Reference Thesis**

## **Hotel Parvez Food Menu website**

By  
**Dhattiwala Mohammadparvez Mehboob Ilahi**  
**Enrolment No.**  
**202218100031**

Under the Supervision of

Jai Daga

A Report Submitted to  
Gujarat University  
In Partial Fulfilment of the Requirements for  
the Degree of

**Integrated IT IMS & Cyber Security**

**May-2025**



GUJARAT UNIVERSITY

Estd. 1949

Center for Professional Courses,  
Gujarat University, Ahmedabad

## CERTIFICATE

This is to certify that research work embodied in this report entitled "**Hotel Parvez Food Menu Website**" was carried out by **Dhattiwala Mohammadparvez Mehboob ilahi (Enrolment No:202218100031)** at Centre for Professional Course for partial fulfilment of B.Sc. IT degree to be awarded by Gujarat University. This research work has been carried out under my supervision and is to the satisfaction of department.

Date: 01-05-2025

Place: Ahmedabad

**Soniya Suthar**

Assistant Professor

**(Jai Daga)**

**CPC, Gujarat University**

**Rajendra Rathod**

Program In-Charge

**CPC, Gujarat University**

**Dr. Paavan Pandit**  
**Director**  
**CPC, Gujarat University**  
**Seal of Institute**

## **DECLARATION OF ORIGINALITY**

I hereby certify that I am the sole author of this Project report and that neither any part of this Project report nor the whole of the Project report has been submitted for a degree to any other University or Institution.

I certify that, to the best of my knowledge, my Project report does not infringe upon any one's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my Project report, published or otherwise, are fully acknowledged in accordance with the standard referencing practices.

I declare that this is a true copy of my Project report, including any final revisions, as approved by my Project report review committee.

Date: 01-05-2025

Place: Ahmedabad

Student Name: Dhattiwala Mohammadparvez Mehboob ilahi  
Enrolment No: 202218100031

# PROJECT REPORT APPROVAL

This is to certify that research work embodied in this Project report entitled "**Hotel Parvez Food Menu Website**" was carried out by **Dhattiwala Mohammadparvez Mehboob ilahi (Enrolment No:202218100031)** at Centre for Professional Course for partial fulfilment of B.Sc. IT degree in [IMS & Cyber Security] to be awarded by Gujarat University.

**Date:**

**Place:** Ahmedabad

**Examiner(s):**

( \_\_\_\_\_ ) ( \_\_\_\_\_ ) ( \_\_\_\_\_ )

## **ACKNOWLEDGEMENT**

---

We are sincerely thankful to our guide, **Assi. Prof. Soniya Suthar** for their constant support, stimulating suggestions, and encouragement, which greatly assisted us in successfully completing our project work. Their close supervision over the past few months and helpful insights have been invaluable. Despite their busy schedule, their valuable advice and unwavering support have been an inspiration and a driving force for us. Their experience and knowledge have continuously helped shape our initial ideas into a comprehensive form.

I, hereby, take an opportunity to convey my gratitude for the generous assistance and cooperation, that I received from the **[Rajendra Sir]** and to all those who helped me directly and indirectly.

We are deeply indebted & thankful to our Department Faculties who helped and rendered their valuable time, knowledge and information and whose suggestion and guidance has enlightened on the subject.

We also thank “**Dr. Paavan Pandit**”, Director, CPC, GU for extending all the help and cooperation during our training period.

Finally, I am also indebted to my friends without whose help I would have had a hard time managing everything on my own.

**Dhattiwala Mohammadparvez Mehboob ilahi**  
**(202218100031)**

# **(Reference Purpose Only)**

## **Table of Contents**

---

ACKNOWLEDGEMENT .....	4
Table of Contents .....	6
List of Figures.....	7
ABSTRACT	8
Chapter 1 INTRODUCTION AND PROBLEM SUMMARY	9
1.1 Introduction	9
1.2 Problem Summary	11
1.3 Conclusion	12
Chapter 2 WEBSITE CONTENT	13
2.1 Login Page	13
2.2 Navigation Bar	20
2.3 Home Page	23
2.4 Menu Page	25
2.5 Delivery Detail Form	30
2.6 About Us Page	33
2.7 Contact Us Page	35
2.8 Footer Section	38
Chapter 3 SECURITY ISSUES AND FIXES	43
3.1 Security Problems in website	43
3.2 Security Problem Solving	44
3.3 Tools use	45
3.4 Summarized Example	46
CHAPTERS: BIBLOGRAPHY	47
REFERENCES	47

---

## **List Of Figures**

Figure 1.1 Login Page .....	19
Figure 1.2 Navigation Bar .....	23
Figure 1.3 Menu Page .....	29
Figure 1.4 Delivery Detail Form .....	32
Figure 1.5 About Us Page .....	34
Figure 1.6 Contact Us Page .....	38
Figure 1.7 Footer Section .....	41

## **(Reference Purpose Only)**

### **ABSTRACT**

- Hotel ParvezFood Menu is a website aimed at displaying the assorted types of food that they have at Hotel ParvezFood. The site is display the branches available inside the world. The site allows the user to browse and select from a menu, which includes decent descriptions of food items and price ranges, such as pizzas, burgers, dosa, pasta, etc.
- While the site does not allow for ordering online, it is effectively a digital menu for people to browse what is available for food when they arrive at the restaurant. The menu will conveniently help prospective diners select their dining options before arriving to dine. An "About Us" section on the site provides further information about the restaurant, its journey, and its mission, and there is a contact form to facilitate inquiries.
- Included in the menu is a Delivery Section that describes if the restaurant delivers to their home, delivery partners, approximate delivery times, and the phone number if they would like to place a phone order. This section also provides users who may be searching for something to eat to find delivery details at ease without the use of an online Ordering System.
- Links for social media engagement are also accessible at the bottom. Hotel ParvezFood Menu has a secure sign-in page, a clean and simple navigation bar, and has a Beautiful. Hotel ParvezFood Menu is designed to be a simple, accessible, and informative tool for the food options at the restaurant
- Hotel Parvez Food Menu Website in all the vulnerability Find and How to Fix all vulnerability or prevent from unauthorized access. The various Types of tools use like Let's Encrypt for assign SSL/TLS certificate, OWASP ZAP for website vulnerability scanning use and prevent unauthorized access of website after provide a simple example for understanding.

# **CHAPTER:1**

## **INTRODUCTION & PROBLEM SUMMARY**

### **1.1 Introduction**

**\*\*Welcome to ParvezFood – A Place to Eat!\*\***

- ParvezFood is an enjoyable eating experience with great food choices, drool-worthy spicy and savory food options to satisfy everyone's cravings. You can have juicy burgers, cheese-drenched pizza, crispy dosas, and spicy bhaji pav in our diverse menu. Founded in 2025, we're proud to deliver delicious, quality food that will have people coming back for more.
- Our goal is simple: provide a great experience online to find out what we sell and prices & provide feedback from their own home. Whether it's an old favorite or you're willing to try something new, ParvezFood wanted to share our delicious food with a full spectrum of savory flavors.
- Peruse our menu, learn more about our story or contact us, we look forward to hearing from you! Thank you for choosing ParvezFood, something different for your food adventures!

**\*\*Hungry? Visit Today!\*\***

At ParvezFood, we provide a wonderful gastronomic experience with all kinds of cravings covered on our menu. Juicy burgers, cheesy pizzas, crispy dosas, and spicy bhaji pav - we have something for everyone. Founded in 2025, we take pride in offering high-quality, tasty food made with fresh and locally sourced ingredients to provide you with as much flavor as possible.

**\*\*Why Choose ParvezFood?\*\***

- \*\*Real Taste, Modern Twist\*\*** - Enjoy the flavors of authentic recipes with a modern twist.

- \*\*Quick and Easy Check\*\*** - Quickly and easily check item available or not
- \*\*Affordable Prices\*\*** - Delicious food that won't break the bank.
- \*\*Delivered Food\*\*** - Food Delivered to your location as soon as possible.
- \*\*We guarantee your Satisfaction\*\*** - Your feedback allows us to keep improving.

**\*\*Check Out Our Menu\*\*** If you're looking for a cheesy Margherita Pizza, a Classic Burger, or a Crispy Dosa, all of our menu items are made to satisfy all taste buds. But, don't forget to check out our signature **\*\*Bhaji Pav\*\*** - a spicy, buttery goodness; it's one of the most popular items!

**\*\*Our Mission\*\*** We believe food is more than just a meal; it's an experience:

**\*\*Fresh Ingredients\*\*** - Only the best for our customers.

**\*\*Safe Preparation\*\*** - Clean kitchens, clean handling practices.

**\*\*Friendly Service\*\*** - A warm welcome every time you order!

**\*\*Delivery Service\*\*** - Delivery of food is fast and effective.

**\*\*Welcome to the ParvezFood Family!\*\***

Follow us on social media for exclusive offers, new menu launches, and everything food! Have a question or feedback? Get in touch; we would love to hear from you! 

● Security in this website breaches:

1. No Server-side CSRF Protection
2. Insecure Image Loading
3. Potential XSS (Cross-Site Scripting)
4. Weak CAPTCHA
5. Missing HTTPS/SSL
6. Plain Form SubmissionFile
7. Upload Path Disclosure

## 1.2 Problem Summary

⇒ Food delivery is currently inefficient because there is no food delivery website to make the process easier for both customers and restaurants. Customers are inconvenienced in getting restaurant menus, placing orders, and paying. Restaurants face order taking and order processing issues, which takes away potential sales, operational efficiency, and customer satisfaction.

⇒ If a food delivery website is not available, a number of problems and challenges arise for consumers and businesses alike. The lack of a food delivery website creates ordering inefficiencies on both the customer and restaurant sides. Ordering food becomes inconvenient for customers—accessing restaurant menus, placing orders, and making payments, or checking the status of their orders—and restaurants cannot efficiently take new orders, all due to the lack of availability. This costs them revenue, is inefficient to manage, and can leave customers dissatisfied.

- ▶ **No Server-side CSRF Protection:** CSRF token `{{{csrf_token}}}` placeholder suggests it's not dynamically generated server-side.
- ▶ **Insecure Image Loading:** Images are linked using **local file paths** like `C:\Users\...`, which won't work online and can lead to errors.
- ▶ **Potential XSS (Cross-Site Scripting):** No input sanitization/validation on user inputs like **contact forms** and **login forms**.
- ▶ **Weak CAPTCHA:** CAPTCHA is client-side only, easily bypassable with bots.
- ▶ **Missing HTTPS/SSL:** No evidence in the page that HTTPS is enforced.
- ▶ **Plain Form Submission:** Sensitive data (password, delivery details) are being transmitted possibly without encryption.
- ▶ **File Upload Path Disclosure:** Internal paths (`C:\...`) reveal the folder structure to the user, which is risky

Conclusion=> Customers and businesses alike may find it inconvenient, inefficient, and costly to not have a food delivery website. A recent, digital

solution to improve on the food ordering experience would bridge the gap for restaurant operations access, and provide competitive advantage over other companies that providing the food ordering process.

## **Website security is not optional — it is essential.**

- Protecting user data builds **trust** and **reputation**.
- Ignoring security issues can lead to **data theft**, **website hacking**, and **financial loss**.
- **Basic security steps** like HTTPS, input validation, CSRF protection, and CAPTCHA can prevent most common attacks.
- **Regular testing and updates** keep your website strong against new threats.
- A **secure website = safe customers + successful business**.

# **CHAPTER 2:**

## **WEBSITE CONTENT**

### **2.1 Login Page**

Login Page include Username and Password content and some security include related to this.This all includes using programming Languages.(HTML,CSS,JS).HTML used to create structure and CSS is Apply Style on our Login Page After The style section Java script using add security related alert message and use addeventlisner(),document.get elementbyid(),etc.. function use and create more Improvement Page.

#### **HTML Structure:**

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Hotel ParvezFood Menu</title>
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0/css/all.min.css">
    <link rel="stylesheet" href="styles.css">
</head>
<body>
    <!-- Login Section -->
    <div class="login">
<section id="Login" class="Login-section">
    <form id="Login">
        <h1>
            <b>Login Form</b>
        </h1>
        <div class="Login-Auth">
            <div class="Username">
                <label for="email">Username/e-mail:</label>
                <input type="email" id="email" name="email" required>
            </div>
        </div>
```

```

<div class="Login-password">
    <label for="password">Password:</label>
    <input type="password" id="password" name="password" required minlength="8">
</div>
<!-- CAPTCHA -->
<div class="captcha">
    <label for="captcha">Enter the code below:</label><br>
    <canvas id="captchaCanvas" width="100" height="40" style="border: 1px solid #ccc;">
    </canvas>
<br>
    <input type="text" id="captchaInput" name="captcha" required placeholder="Enter captcha">
</div>
<input type="hidden" id="csrf_token" name="csrf_token" value="{{csrf_token}}>
    <button type="submit" class="submit-bton">
        Submit
    </button>
</div>
<p id="error-message"></p>
</form>
</section>
</div>
<!-- Website Content (Hidden Initially) -->
<div id="content" style="display: none;">
<script src="script.js"></script>
</body>
</html>

```

## CSS:

```

/* Reset and Base Styles */
* {
    margin: 0px;
    padding: 0px;
    box-sizing: border-box;
}

```

```
}

body {
    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    line-height: 1.6;
    color: orangered;
    padding-top: 20px;
    width: 100%;
}

.login {
    display: flex;
    justify-content: center;
    align-items: center;
    width: 100%;
}

.Login-section {
    margin-top: 60px;
    width: 100%;
    max-width: 900px; /* Adjust for better responsiveness */
    background: yellow;
    padding: 45px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    border-radius: 10px;
}

.form input{
    width: 100%;
    align-items: center;
    padding: 0.8rem;
    border: 1px solid #ddd;
    border-radius: 5px;
}

h1{
    padding-top: 40px;
    text-align: center;
    padding-bottom: 10px;
}
```

```
.Login-Auth{  
    text-align: center;  
    line-height: 1.6;  
    height: 100%;  
}  
.Username{  
    display:block;  
    margin-bottom: 0.5rem;  
}  
.Login-password{  
    display: block;  
    margin-bottom: 0.5rem;  
}  
.Login-Auth input {width: 20%;  
    padding: 0.5rem;  
    border: 1px solid #ddd;  
    border-radius: 5px;  
    margin-bottom: 0.6rem;  
}  
.captcha {  
    margin-top: 10px;  
    margin-bottom: 15px;  
}  
.submit-bton {  
    width: 10%;  
    padding: 1rem;  
    background-color: #ff4757;  
    color: #fff;  
    border: none;  
    border-radius: 5px;  
    cursor: pointer;  
    transition: background-color 0.3s ease;  
}  
/* Hide content initially */  
#content {  
    display: none;  
    padding: 20px;
```

```
}
```

## JAVA SCRIPT:

```
//Login Function
function validateLogin() {
    const email = document.getElementById('email').value;
    const password = document.getElementById('password').value;
    const emailPattern = /^[^@\s]+@[^\s]+\.\[^@\s]+$/;

    if (!emailPattern.test(email)) {
        alert("Invalid email format");
        return false;
    }

    if (password.length < 8) {
        alert("Password must be at least 8 characters long");
        return false;
    }
    return true;
}

let captchaCode = "";

function generateCaptcha() {
    const canvas = document.getElementById('captchaCanvas');
    const ctx = canvas.getContext('2d');
    ctx.clearRect(0, 0, canvas.width, canvas.height);
    captchaCode = Math.random().toString(36).substring(2, 8);
    ctx.font = '24px Arial';
    ctx.fillStyle = '#000';
    ctx.fillText(captchaCode, 10, 30);
}

document.addEventListener('DOMContentLoaded', () => {
    generateCaptcha();
```

```

document.getElementById('Login').addEventListener('submit', function(e) {
    const input = document.getElementById('captchaInput').value;
    if (input !== captchaCode) {
        e.preventDefault();
        document.getElementById('error-message').textContent = 'Captcha
incorrect. Please try again.';
        generateCaptcha();
    }
});
});

//Login Form Validation
document.addEventListener("DOMContentLoaded", function () {
    const loginForm = document.getElementById("Login");
    const content = document.getElementById("content");
    const loginSection = document.getElementById("Login");
    const emailField = document.getElementById("email");
    const passwordField = document.getElementById("password");
    const errorMessage = document.getElementById("error-message");

    // Dummy user credentials
    const validEmail = "admin@example.com";
    const validPassword = "password123";

    // Check if user is already logged in
    if (localStorage.getItem("loggedIn") === "true") {
        loginSection.style.display = "none";
        content.style.display = "block";
        showLogoutButton() // Show logout button immediately
    }

    // Login form event listener
    loginForm.addEventListener("submit", function (event) {
        event.preventDefault() // Prevent form submission

        const email = emailField.value;

```

```

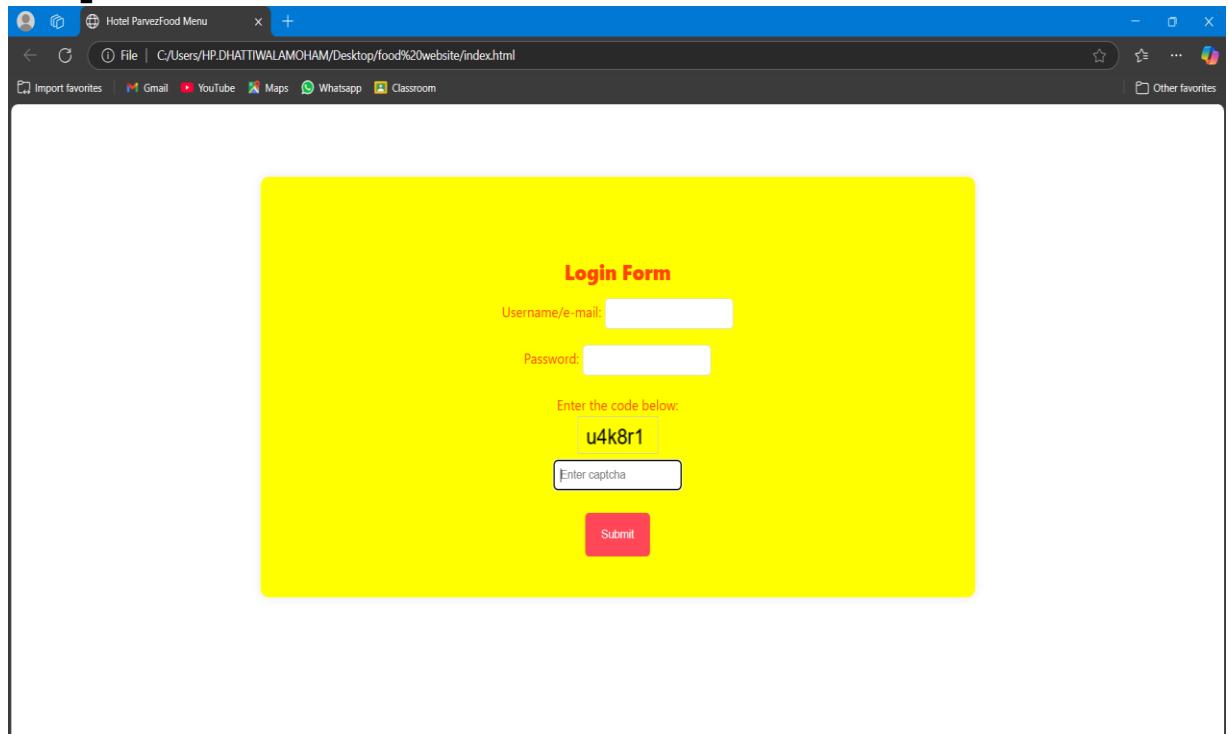
const password = passwordField.value;

// Check credentials
if (email === validEmail && password === validPassword) {
    localStorage.setItem("loggedIn", "true"); // Store login state
    loginSection.style.display = "none";
    content.style.display = "block";
    showLogoutButton(); // Show logout button immediately after login
} else {
    errorMessage.textContent = "Invalid login credentials!";
    errorMessage.style.color = "red";
}
});

});

```

## Output:



**Figure-1.1 Login Page**

Add Username:admin@example.com

Password:password123

Enter captcha show above line.

Use This Dummy Username And Password To Login in website.

## 2.2 Navigation Bar

=>Navigation Bar is provide the direct jump on specific tab and show the company's Logo.

-In this include the Home,menu,about us,contact us and cart button show and assigning style on this navigation bar to look attractive.

### HTML:

```
// Above HTML Structure same.head,body tags and css file.  
<nav class="navbar">  
    <div class="logo">ParvezFood</div>  
    <ul class="nav-links">  
        <li><a href="#Home">Home</a></li>  
        <li><a href="#Menu">Menu</a></li>  
        <li><a href="#About Us">About Us</a></li>  
        <li><a href="#Contact Us">Contact Us</a></li>&nbsp;  
        <div class="cart-container">  
            <span id="cart-count">0</span>  
            <i class="fas fa-shopping-cart"></i>  
        </div>  
    </ul>  
</nav>
```

### CSS:

```
/* Navigation Bar */  
.navbar {  
    position: fixed;
```

```
top: 0;
width: 99%;
background-color: green;
border-radius: 18px;
padding: 0rem 1rem;
display: flex;
justify-content: space-between;
align-items: center;
box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);
z-index: 1000;
}

.logo {
    font-size: 1.5rem;
    font-weight: bold;
    color: yellow;
}

.nav-links {
    display: flex;
    list-style: none;
}

.nav-links li {
    margin-left: 2rem;
}

.nav-links a {
    text-decoration: none;
    color: white;
    font-weight: 500;
    transition: color 0.3s ease;
}

.nav-links a:hover {
    color: #ff4757;
    background-color: yellow;
```

```

}

.cart-container {
    position: relative;
    font-size: 20px;
    cursor: pointer;
    margin-right: 20px;
}

.cart-container i {
    font-size: 24px;
    color: white;
}

#cart-count {
    position: absolute;
    top: -5px;
    right: -10px;
    background: red;
    color: white;
    font-size: 14px;
    width: 18px;
    height: 18px;
    text-align: center;
    border-radius: 50%;
    font-weight: bold;
}

```

## **JavaScript only for cart:**

```

//cart add
document.addEventListener("DOMContentLoaded", () => {
    let cartCount = 0;
    const cartCountDisplay = document.getElementById("cart-count");
    const addToCartButtons = document.querySelectorAll(".add-to-cart");

    addToCartButtons.forEach(button => {

```

```

button.addEventListener("click", () => {
    cartCount++;
    cartCountDisplay.textContent = cartCount;
});
});
);

```



**Figure-1.2 Navigation Bar**

## 2.3 Home Page

=>Our Home Page include a Tag Line “**From Juicy Burgers to Cheesy pizzas - We've Got It All! on The Doorstep**” and related to this one paragraph line add ‘Experience the best local restaurants food delivery on the Doorstep.’ After the side show our branch location that is exist in Ahmedabad Area And Background Image add and show the home in navigation bar when you click on Home to redirect on this page. This Page is created using HTML And CSS.

### HTML:

```

<!-- Home Section -->
<section id="Home" class="hero">
    <div class="hero-content">
        <h1>From Juicy Burgers to Cheesy pizzas - We've Got It All! on The
Doorstep</h1>

```

```

        <p>Experience the best local restaurants food delivery on the
Doorstep.</p>
        </div>
        <div class="home-grid">
            
        </div>
    </section>

```

## CSS:

```

/* Home Section */
.hero {
    height: 100vh;
    background: linear-gradient(rgba(0, 0, 0, 0.5), rgba(0, 0, 0, 0.5)),
        url('https://images.unsplash.com/photo-1504674900247-
0877df9cc836');
    background-size: cover;
    background-position: center;
    display: flex;
    align-items: center;
    justify-content: center;
    text-align: center;
    color: #fff;
    padding: 0 1rem;
}

.hero-content {
    max-width: 800px;
}

.hero h1 {
    font-size: 3rem;
    margin-bottom: 1rem;
}

.hero p {

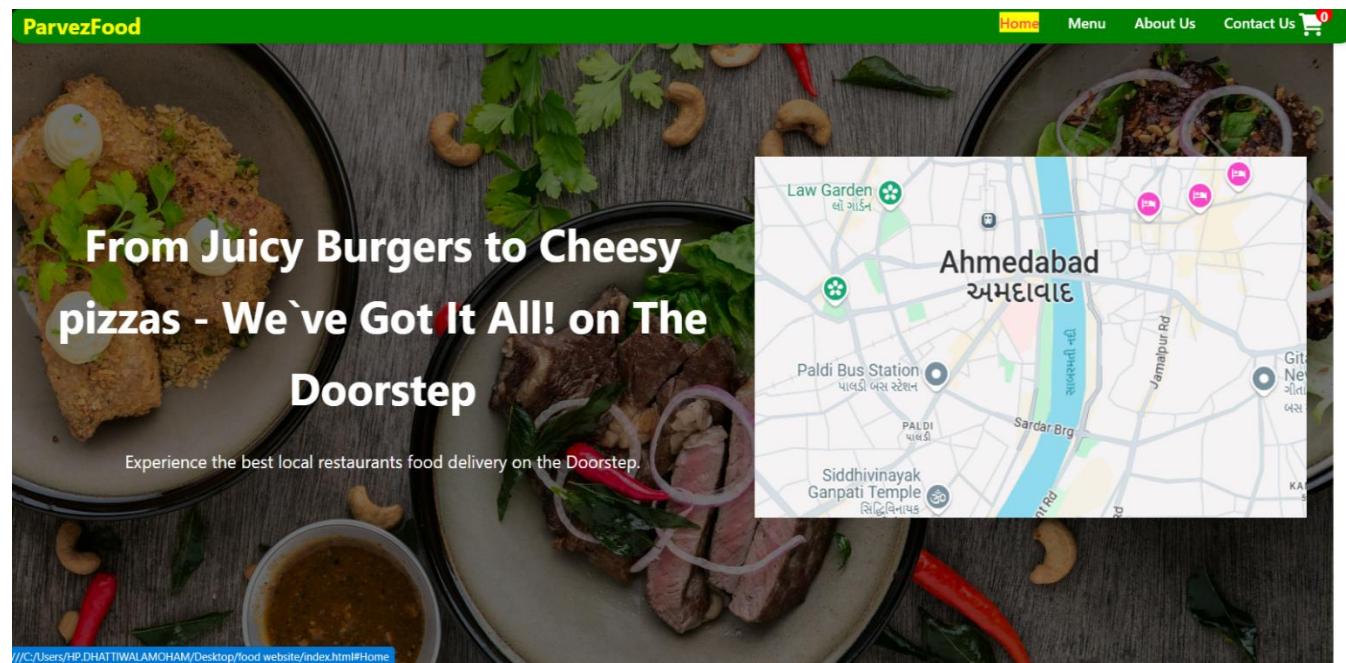
```

```

        font-size: 1.2rem;
        margin-bottom: 2rem;
    }

.home-grid {
    display: grid;
    grid-template-columns: repeat(auto-fit, minmax(280px, 1fr));
    gap: 2rem;
    max-width: 1200px;
    margin: 0 auto;
    border-radius: 30px;
}

```



**Figure-1.3 Home Page**

## 2.3 Menu Page

⇒ In This Page Include all food Items display easily for customer and best UI display because customer understand easily and order food items. All food item show in **card-UI- Effect** customer attract and card effect include image of the food item, description and best tag line after the customer like the item to display price after 'add to cart' button in every item.

## HTML

```
<!-- Menu Section -->
<section id="Menu" class="menu-section">
    <h2>Our Menu</h2>
    <div class="menu-grid">
        <div class="menu-item">
            
            <h3>Margherita Pizza</h3>
            <p>Fresh tomatoes, mozzarella, basil</p>
            <span class="price">₹250</span>
            <button class="add-to-cart" data-name="Margherita Pizza" data-price="250" style="background-color: red; text-align: center;">Add to Cart</button>
        </div>
        <div class="menu-item">
            
            <h3>Chocolety Cookies</h3>
            <p>A balanced diet is a cookie in each hand.</p>
            <span class="price">₹300</span>
            <button class="add-to-cart" data-name="Chocolety Cookies" data-price="250">Add to Cart</button>
        </div>
        <div class="menu-item">
            
            <h3>Classic Burger</h3>
            <p>Beef patty, lettuce, tomato, cheese</p>
            <span class="price">₹100</span>
        </div>
    </div>
</section>
```

```

        <button class="add-to-cart" data-name="Classic Burger" data-
price="250">Add to Cart</button>
    </div>
    <div class="menu-item">
        
        <h3>Fettuccine Alfredo</h3>
        <p>Creamy sauce, parmesan, garlic</p>
        <span class="price">₹300</span><br>
        <button class="add-to-cart" data-name="Fettuccine Alfredo" data-
price="250">Add to Cart</button>
    </div>
    <div class="menu-item">
        
        <h3>Crispy Dosa</h3>
        <p>Happiness is a hot dosa with chutney and sambar</p>
        <span class="price">₹150</span>
        <button class="add-to-cart" data-name="Crispy Dosa" data-
price="250">Add to Cart</button>
    </div>
    <div class="menu-item">
        
        <h3>Bhaji Pav</h3>
        <p>Hot,spicy bhaji with buttery pav-one bite,and you`ll taste pure
magic!</p>
        <span class="price">₹200</span>
        <button class="add-to-cart" data-name="Bhaji Pav" data-
price="250">Add to Cart</button>
    </div>
</div>
</section>

```

## CSS

```

/* Menu Section */
.menu-section {

```

```
padding: 5rem 2rem;
background-color: yellowgreen;
}

.menu-section h2 {
    text-align: center;
    margin-bottom: 3rem;
}

.menu-grid {
    display: grid;
    grid-template-columns: repeat(auto-fit, minmax(280px, 1fr));
    gap: 2rem;
    max-width: 1200px;
    margin: 0 auto;
}

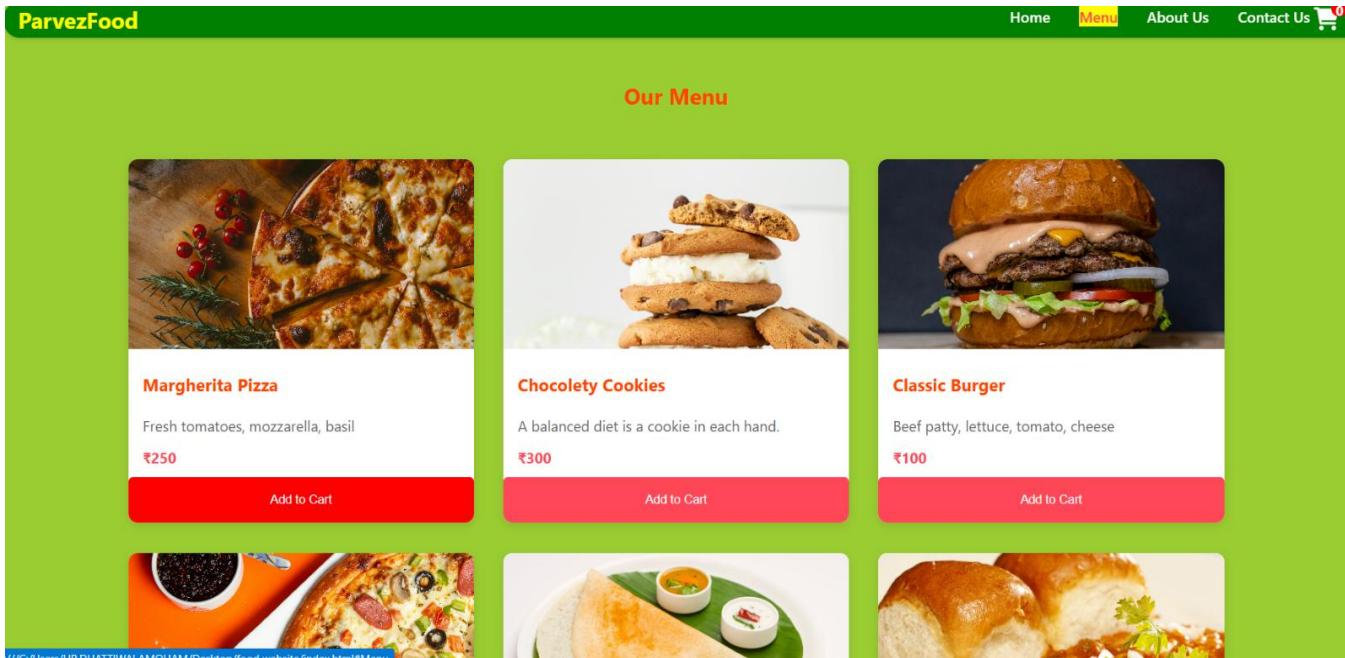
.menu-item {
    background-color: #fff;
    border-radius: 10px;
    overflow: hidden;
    box-shadow: 0 3px 10px rgba(0, 0, 0, 0.1);
    transition: transform 0.3s ease;
}

.menu-item:hover {
    transform: translateY(-5px);
}

.menu-item img {
    width: 100%;
    height: 200px;
    object-fit: cover;
}

.menu-item h3 {
    padding: 1rem;
```

```
margin: 0;  
}  
  
.menu-item p {  
    padding: 0 1rem;  
    color: #666;  
}  
  
.price {  
    display: block;  
    padding: 0.5rem 1rem;  
    font-weight: bold;  
    color: #ff4757;  
}  
.add-to-cart{  
    width: 100%;  
    padding: 1rem;  
    background-color: #ff4757;  
    color: #fff;  
    border: none;  
    border-radius: 5px;  
    cursor: pointer;  
    transition: background-color 0.3s ease;  
}
```



**Figure 1.4-Menu Page**

## 2.4 Delivery Detail Form:

In this form customer provide detail Like Full Name, Delivery Address, Phone Number to deliver food items on doorstep to the customer and any problem or return the order to message on mobile number that can provided by the customer. customer has provided detail accurate to deliver item on their location easily and timely. In this customer provide related data like phone number to enter only number, and all detail is required not skip etc... and when provide all detail perfect and click on place order button to show alert message your order placed successfully this is add using java script after the form is reset.

### HTML:

```
<section id="Delivery" class="delivery-section">
    <h2>Delivery Details</h2>
    <form id="delivery-form">
        <label for="name">Full Name:</label>
```

```

<input type="text" id="name" required>

<label for="address">Delivery Address:</label>
<textarea id="address" required></textarea>

<label for="phone">Phone Number:</label>
<input type="number" id="phone" required>

<button type="submit" class="submit">Place Order</button>
</form>
</section>
```

## CSS:

```

.delivery-section {
    padding: 20px;
    background: palegreen;
    border-radius: 10px;
    width: 50%;
    margin: auto;
}

.delivery-section form {
    display: flex;
    flex-direction: column;
}

.delivery-section input, .delivery-section textarea {
    margin-bottom: 10px;
    padding: 8px;
}

.submit{
    width: 100%;
    padding: 1rem;
    background-color: #ff4757;
    color: #fff;
```

```
border: none;  
border-radius: 5px;  
cursor: pointer;  
transition: background-color 0.3s ease;  
}
```

## **Java Script:**

```
//Delivery section  
document.addEventListener("DOMContentLoaded", () => {  
    let cartCount = 0;  
    const cartCountDisplay = document.getElementById("cart-count");  
    const addToCartButtons = document.querySelectorAll(".add-to-cart");  
  
    addToCartButtons.forEach(button => {  
        button.addEventListener("click", () => {  
            cartCount++;  
            cartCountDisplay.textContent = cartCount;  
        });  
    });  
  
    document.getElementById("delivery-form").addEventListener("submit",  
    function (event) {  
        event.preventDefault();  
        alert("Order placed successfully!");  
        this.reset();  
    });  
});
```

## Delivery Details

Full Name:

Delivery Address:

Phone Number:

Place Order

**Figure 1.5-Delivery Detail Form**

## 2.5 About Us Page

This Page shows our mission and story when our product launch date and location and mission to provide delivery soon as possible, all Ingredients fresh and healthy use to make cooking.

## **HTML:**

```
<!-- About Section -->
<section id="About Us" class="about-section">
    <h2>About Us</h2>
    <div class="about-content">
        <div class="about-text">
            <h3>Our Story</h3>
            <p>Founded in 2025, starting in the create a some product like Dosa,Pizza,pasta,Bhaji-pav etc... and we are most of successful of These products and today is very famous in Food Express And Provide The Delivery to Your Home</p>
            <h3>Our Mission</h3>
            <p>Customer Doesn't come to our location online check our menu items and related to price and order the items so increase the customer time value and after provide the feedback to our website. Provide Your Delivery On Doorstep in minimum time.</p>
        </div>
    </div>
</section>
```

## **CSS :**

```
/* About Section */
.about-section {
    padding: 5rem 2rem;
    background-color: salmon;
    color: brown;
}

.about-section h2 {
    text-align: center;
    margin-bottom: 3rem;
}

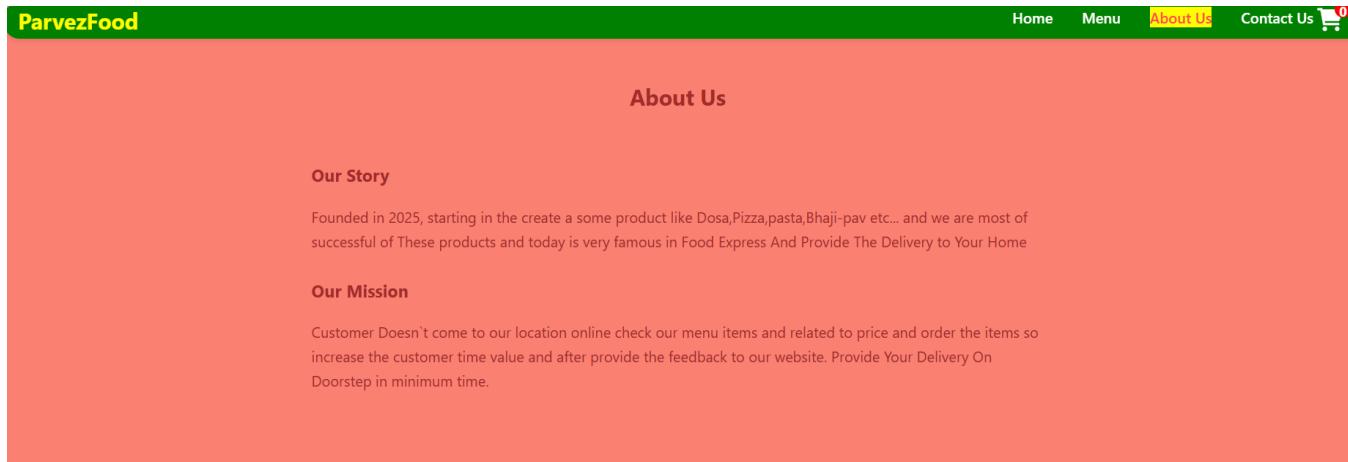
.about-content {
    max-width: 800px;
```

```

margin: 0 auto;
}

.about-text h3 {
    margin: 1.5rem 0 1rem 0;
}

```



**Figure 1.6-About Us Page**

## 2.6 Contact Us Page

Contact Us Page Include a Username, E-mail and Message. User Can Add Own Detail related to Field And Provide Your Feedback or suggestion which will be Improve or Any Changes Perform our support team as soon as possible.

### HTML:

```

<!-- Contact Section -->
<section id="Contact Us" class="contact-section">
    <h2>Contact Us</h2>

```

```

<div class="contact-container">
  <form id="contact-form">
    <div class="form-group">
      <label for="name">Name:</label>
      <input type="text" id="contact-name" name="name" required>
    </div>
    <div class="form-group">
      <label for="email">Email:</label>
      <input type="email" id="contact-email" name="email" required>
    </div>
    <div class="form-group">
      <label for="message">Message:</label>
      <textarea id="contact-message" name="message" required></textarea>
    </div>
    <button type="submit" class="submit-btn">Send Message</button>
  </form>
</div>
</section>

```

## CSS:

```

/* Contact Section */
.contact-section {
  padding: 5rem 2rem;
  background-color: blueviolet;
}

.contact-section h2 {
  text-align: center;
  margin-bottom: 3rem;
}

.contact-container {
  max-width: 600px;
  margin: 0 auto;
}

```

```
}

.form-group {
  margin-bottom: 1.5rem;
}

.form-group label {
  display: block;
  margin-bottom: 0.5rem;
}

.form-group input,
.form-group textarea {
  width: 100%;
  padding: 0.8rem;
  border: 1px solid #ddd;
  border-radius: 5px;
}

.form-group textarea {
  height: 150px;
  resize: vertical;
}

.submit-btn {
  width: 100%;
  padding: 1rem;
  background-color: #ff4757;
  color: #fff;
  border: none;
  border-radius: 5px;
  cursor: pointer;
  transition: background-color 0.3s ease;
}

.submit-btn:hover {
  background-color: #ff2e43;
```

}

## **JAVA SCRIPT:**

```
// Contact Form Validation
document.addEventListener("DOMContentLoaded", function () {
    document.getElementById("contact-form").addEventListener("submit", function (event) {
        event.preventDefault(); // Prevent form from reloading the page

        let name = document.getElementById("contact-name").value.trim();
        let email = document.getElementById("contact-email").value.trim();
        let message = document.getElementById("contact-message").value.trim();

        if (name !== "" && email !== "" && message !== "") {
            alert("Message sent successfully!"); // Display success message
            this.reset(); // Clear the form
        } else {
            alert("Please fill in all fields."); // Show validation message
        }
    });
});
```

The screenshot shows the 'Contact Us' page of the ParvezFood website. At the top, there's a green header bar with the logo 'ParvezFood'. To the right of the logo are navigation links: 'Home', 'Menu', 'About Us', and 'Contact Us'. The 'Contact Us' link is highlighted with a yellow background and a red border. Below the header, the page title 'Contact Us' is displayed in red. The form consists of three input fields: 'Name:' with a white input box, 'Email:' with a white input box, and 'Message:' with a larger white input box. A red button labeled 'Send Message' is located at the bottom of the form. The URL 'http://C:/Users/HP.DHATTIWAI/AMOHAM/Desktop/food website/index.html#Contact Us' is visible at the very bottom of the page.

**Figure 1.6–Contact Us Page**

## 2.7 Footer Section

Footer Page is Provide a Detail of company or Support Detail User Can contact to our company, add website Links and many social Media Links to follow customer to know new offers, discount, brouchers and any new Food items coming. In this section Insert a Copyright Tag and company Terms & condition Policies To follow the customers and company. After Add The Logout Button User Can Logged out.

### HTML:

```
<!-- Footer -->
<footer>
```

```

<div class="footer-content">
    <div class="footer-section">
        <h4>Contact Info</h4>
        <p><i class="fas fa-phone"></i> 701-607-9</p>
        <p><i class="fas fa-envelope"></i> parvez@foodexpress.com</p>
    </div>
    <div class="footer-section">
        <h4>Follow Us</h4>
        <div class="social-links">
            <a href="#"><i class="fab fa-facebook"></i></a>
            <a href="#"><i class="fab fa-twitter"></i></a>
            <a href="#"><i class="fab fa-instagram"></i></a>
        </div>
    </div>
</div>
<div class="footer-bottom">
    <p>&copy; 2025 FoodExpress. All rights reserved.</p>
</div>
</footer>

```

## CSS:

```

/* Footer */
footer {
    background-color: orange;
    color: black;
    padding: 3rem 2rem 1rem 2rem;
}

.footer-content {
    max-width: 1200px;
    margin: 0 auto;
    display: grid;
    grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));
    gap: 2rem;
}

```

```

.footer-section h4 {
    margin-bottom: 1rem;
}

.social-links a {
    color: #fff;
    margin-right: 1rem;
    font-size: 1.5rem;
    transition: color 0.3s ease;
}

.social-links a:hover {
    color: #ff4757;
}

.footer-bottom {
    text-align: center;
    margin-top: 2rem;
    padding-top: 2rem;
    border-top: 1px solid #555;
}

```

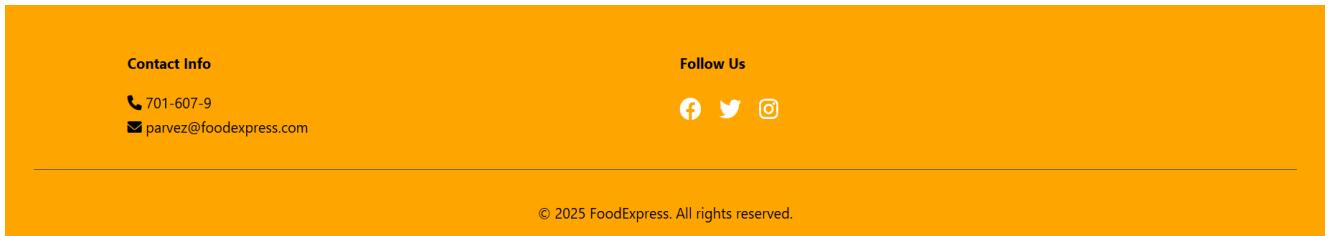
## JavaScript

```

// Logout button setup
const logoutButton = document.createElement("button");
logoutButton.textContent = "Logout";
logoutButton.style.margin = "20px";
logoutButton.style.padding = "10px";
logoutButton.style.cursor = "pointer";
logoutButton.addEventListener("click", function () {
    localStorage.removeItem("loggedIn");
    location.reload(); // Refresh the page to reset state
});

```

```
// Function to show logout button
function showLogoutButton() {
    if (!document.body.contains(logoutButton)) {
        document.body.appendChild(logoutButton);
    }
}
```



**Figure 1.8-Footer Section**

# **Chapter: 3**

## **Security Issues And Fixes**

### **3.1 Security Problems In website**

- **No Server-side CSRF Protection**
  - ➔ The form has a placeholder for CSRF token (`{{csrf_token}}`), but no real dynamic value is generated.
  - ➔ Attackers can forge unauthorized requests from an authenticated user (CSRF attacks).
  - ➔ Example: A malicious site could trick a logged-in user to submit unwanted requests to your site.
- **Insecure Image Loading**
  - ➔ Many images are linked via local computer paths like `C:\Users\....`
  - ➔ This won't work on live websites.
  - ➔ **Risk:** When users access the site, broken links occur, and worse, if local paths are mistakenly exposed, it reveals internal computer structure which attackers can exploit (example: trying path traversal).
- **Potential XSS (Cross-Site Scripting)**
  - ➔ User inputs (like name, email, message) are directly accepted without **sanitization**.
  - ➔ An attacker could inject malicious scripts (like `<script>alert('hacked');`) into input fields.
  - ➔ When another user views that input, the script will execute inside their browser.
- **Weak CAPTCHA**
  - ➔ CAPTCHA is generated only on the client side using a `<canvas>`.
  - ➔ Bots can bypass it easily because they can read the canvas without human intervention.
  - ➔ No server-side validation makes it useless against advanced spam bots.
- **Missing HTTPS/SSL**
  - ➔ No indication that the form submissions or page accesses are enforced over HTTPS.
  - ➔ Without HTTPS:

- ➔ **Passwords, emails, addresses, etc.** can be intercepted during transmission (man-in-the-middle attacks).
  - **Plain Form Submission Without Encryption**
    - ➔ Login, delivery details, contact forms — all submit user data.
    - ➔ If no encryption, sensitive information (passwords, addresses, phone numbers) can be stolen.
  - **File Upload Path Disclosure**
    - ➔ You're showing full local paths like C:\Users\HP.DHATTIWALA MOHAM\Desktop\....
    - ➔ Reveals:
      - Your username
      - File structure
      - Potential data leakage risk
    - ➔ Hackers could use this to craft specific attacks (e.g., guessing sensitive files).
- ## 3.2 Security Problem Solving
- **CSRF Protection**
    - ➔ Use a server-side framework to generate and verify CSRF tokens dynamically on each form. Example: Django or Express.js middleware.
  - **Insecure Image Loading**
    - ➔ Upload all images to your website directory (/images/) or a CDN. Reference them via relative or absolute web paths.
  - **XSS Risk**
    - ➔ Sanitize every user input using libraries like DOMPurify on the client-side and validate inputs again server-side. Escape special characters in outputs.
  - **Weak CAPTCHA**
    - ➔ Replace canvas-based CAPTCHA with Google reCAPTCHA v2/v3 which challenges the user invisibly and protects against bots.
  - **Missing SSL**

➔ Obtain a free SSL certificate from Let's Encrypt or use Cloudflare SSL. Force redirect HTTP → HTTPS.

- **Plain Form Submission**

➔ Always submit forms using HTTPS. Validate form inputs on the server before processing. Encrypt sensitive fields.

- **File Path Disclosure**

➔ Never use local system paths in web pages. Host images properly, and reference only URLs accessible to the web server.

### 3.3 Tools Use

#### 1. OWASP ZAP

**Purpose:** Vulnerability Scanner

**How it works:** Scans your website for security flaws like CSRF, XSS, SQL Injection. Provides detailed reports and fixes.

#### 2. Burp Suite

**Purpose:** Advanced Web Security Testing

**How it works:** Intercepts HTTP requests, simulates attacks (XSS, CSRF) to find vulnerabilities. Great for deep pen-testing.

#### 3. Google reCAPTCHA

**Purpose:** Bot Protection

**How it works:** Adds a “I'm not a robot” or invisible verification to forms. Ensures the submitter is human.

#### 4. Let's Encrypt

**Purpose:** Free SSL Certificates

**How it works:** Provides HTTPS encryption for your website, automatically renewable and trusted by all browsers.

## 5. DOMPurify

**Purpose:** XSS Sanitization

**How it Works:** Cleans HTML, removes any executable scripts, making inputs safe before inserting into the page.

## 6. SonarQube

**Purpose:** Code Security Scanner

**How it Works:** Static code analysis tool that finds security vulnerabilities in your codebase before deployment.

## 7. Cloudflare

**Purpose:** SSL, DDoS Protection, CDN

**How it Works:** Adds a layer between visitors and server. Automatically redirects to HTTPS, provides free SSL, and protects from attacks.

### 3.4 Summarized Example:

#### 1. If you have a login form...

=> Use CSRF Token + HTTPS + Input Validation

#### 2. If you have a contact form...

=> Use reCAPTCHA + Sanitize Input

#### 3. If you upload images...

=> Use server URLs only, not local paths

#### 4. Hosting the website...

=> Install SSL + enforce HTTPS-only access

# **BIBLOGRAPHY:**

## **REFERENCES =>**

### **► Reading material Books**

- HTML & CSS By. Jon Duckett
- JAVASCRIPT & JQUERY By. Jon Duckett

### **► Websites For researching purpose use**

- |                       |  |
|-----------------------|--|
| Login Page =>         | w3school   |
| Navigation Bar =>     | w3school   |
| Home Page =>          | crazzyegg blog refer, mycodeless website   |
| Menu Page =>          | Microsoft template, Pinterest for menu Design  |
| DeliveryDetailForm => | Forms.app check blog, jotiform, tutorials point for java script.                           |
| AboutUsPage =>        | Remontion.com for elements clear voice Resource for difference between mission and vision. |
| Contact Us Page =>    | involve.me blog, medfluenceadvisor blog read form.   |
| Footer Section =>     | Log-rocket for UI-UX design, beetlebeetle.com for Copyright blog refer.                    |
| Security Analysis =>  | OWASPZAP, BurpSuite, Google re`captcha, Let`s encrypt, cloudflare.                         |

### **► YouTube Videos Watching**

- |              |  |
|--------------|--|
| Home Page => | codewith harry tutorial for design home page   |
| Menu Page => | Lun Dev tutorial watch for designing menu Page |