Software engineering project

Restaurant website

By:

Seif Eldin Adel 221002144

Alaa atef Ahmed 221001916

Mohamed Hesham 221000991

Mohamed Helmy 221000212

Under supervision of:

Dr. Mourad Rafaat Abdelghani

TA Rahma Mohamed Ahmed

## **Introduction**

This website is designed to make online food ordering and delivery easy and user-friendly. It serves customers, managers, and delivery staff, providing tools for managing orders, real-time communication, and efficient delivery. The design focuses on simplicity, accessibility, and real-time interaction to create a smooth experience for everyone.

### **Problem Statement**

Many restaurants have outdated websites that are difficult to use. They often rely on delivery apps that take a large percentage of profits. This project aims to modernize restaurants by creating an easy-to-use, secure website that integrates delivery and provides helpful statistics for managers.

### **Proposed Solution**

The website includes these main features:

1. **Easy Order Placement:** Customers can browse menus, customize orders, and make payments easily.
2. **Real-Time Tracking:** Customers can track their orders from preparation to delivery.
3. **Manager Tools:** Managers can monitor orders, update menus, and manage delivery staff efficiently.

## **Purpose**

The website helps customers order food easily and provides restaurant staff with tools to manage orders and communicate effectively. Key features include browsing menus, adding items to a cart, customizing orders, and secure messaging.

## **Scope**

The website focuses on creating a responsive and easy-to-use interface for customers and staff. It includes a dynamic menu, order customization, secure communication, and payment options. The project scope may adjust based on feedback and updates.

## **Technologies Used**

* HTML
* CSS
* JavaScript
* Django (back-end)
* MySQL (Database)

## **Intended Audience**

* **Customers:** To explore menus and place orders.
* **Developers:** To build and improve the website.
* **Testers:** To ensure everything works correctly.
* **Managers:** To oversee orders and operations.
* **Project Leaders:** To manage the development process.

## **Overview**

This document explains the website’s features and how it works. It covers menu management, order processing, customer communication, and technical details.

## **Features**

### **Homepage**

* Overview of the restaurant, including promotions and featured dishes.
* Links to pages like "Menu," "Contact Us," and "Services."

### **Services Page**

1. **Menu Display:** Shows appetizers, main courses, desserts, and drinks with prices and images.
2. **Add to Cart:** Customers can select items and customize them (e.g., size, extras).
3. **Cart Management:** Adjust item quantities and see updated totals.
4. **Order Summary:** Review items and costs before checkout.

### **Contact Us Page**

1. **Customer Support:** Customers can send messages for inquiries or feedback.
2. **Contact Info:** Displays phone number, email, and address with a map.

### **Payment and Billing**

1. **Invoices:** Shows a breakdown of ordered items, tax, and discounts.
2. **Secure Payments:** Supports multiple payment methods like cards and digital wallets.

### **Accessibility and Mobile-Friendly Design**

* Works well on mobile and desktop devices.
* Accessible for users with disabilities.

## **Functional Requirements**

### **User Accounts**

* Customers, managers, and delivery staff can create accounts and log in.
* Users can update profiles and reset passwords securely.

### **Order Management**

* Customers can view menus, place orders, and pay securely.
* Orders can be cancelled within a set time.

### **Delivery Tools**

* Delivery staff can see and update order statuses (e.g., picked up, delivered).
* GPS integration helps optimize routes.

### **Communication**

* Chat or call options for customers, delivery staff, and managers.
* AI chatbot for common questions.

## **Non-Functional Requirements**

### **Performance**

* Orders confirmed within 5 seconds.
* Real-time updates for delivery.
* Handles up to 50 users at the same time.

### **Security**

* Data is encrypted.
* Strong password requirements.
* Payments are processed securely.

### **Usability**

* Works on web and mobile devices.
* Supports voice commands for accessibility.

## **Goal**

To create an efficient online food ordering system that improves customer satisfaction, streamlines delivery, and helps managers oversee operations.

## **Environmental Impact**

By using digital menus and optimizing delivery routes, the website reduces paper use and fuel consumption, helping the environment.

## **Conclusion**

This project provides a modern, easy-to-use platform for restaurants. It helps customers order food quickly, gives managers tools to manage operations, and ensures smooth delivery. With a focus on simplicity and security, this website is a great solution for restaurants in today’s digital world.

**Section 7: interface**

**Registration page:**

**A screenshot of a login form

Description automatically generated**

**Visuals:**

* **Clean, structured form with fields for Username, Email, Phone Number, Address, Password, and Password Confirmation.**
* **Prominent "Sign Up" button with a contrasting color for visibility.**

**Content:**

* **Clear password requirements are displayed to assist users.**
* **User-friendly and intuitive design for a seamless registration process.**

**Login page:**

**A screenshot of a login page

Description automatically generated**

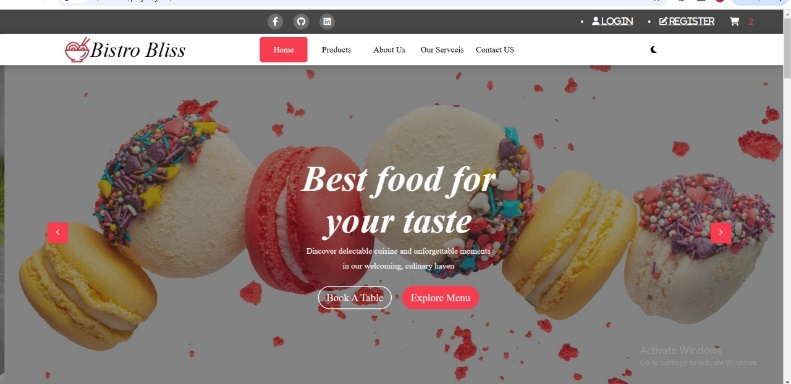
**Visuals:**

* **Minimalistic design with fields for Username and Password.**
* **Prominent black "Login" button for high visibility.**
* **Includes a "Sign Up" link for users without an account.**

**Content:**

* **Simple and functional interface.**
* **Easy navigation for new users to access the registration page via the "Sign Up" link.**

Home:



Focuses on welcoming and attracting visitors. It is a main image that contains a captivating logo, clear calls to action and search on the rest of the pages such as (products, services, contacts, about us), a slider or carousel to display images, social media links, and user login/registration options.

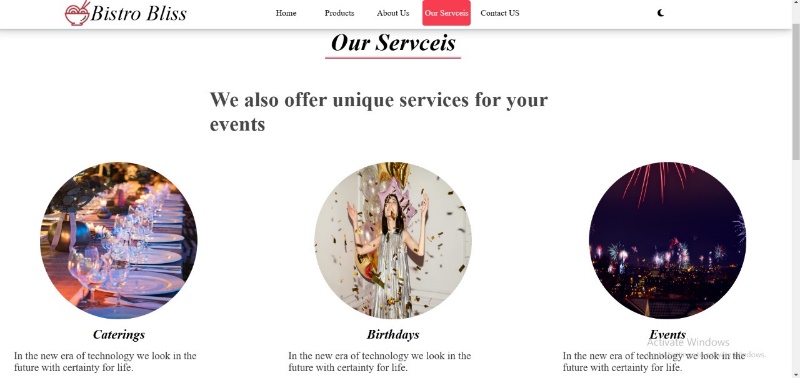
**About us:**

A screenshot of a website

Description automatically generated

The “About Us” page captures the attention of visitors with a well-taken photograph of a meal that perfectly corresponds to the restaurant’s setting. It includes a synopsis of the restaurant’s background, quest, principles, competitive advantages (CA), and character. The page can as well include information regarding the chef, the team, some history, among others with a “Read More” option which encourages users to navigate more.

Services:



The “Services” page utilizes a beautiful set of round images that accentuate different offered services such as and not limited to catering, birthdays and events. It also gives short but fascinating and clear explanations of every service provided as well as particulars like the hours of service, delivery, and specials. Such a design very well introduces the restaurant services to the clients in a simple and appealing manner.

Contact us:

A screenshot of a contact us page

Description automatically generated

The Contact Us page is designed simply and easy to use. It provides a very clear and simple direct contact form with fields for name, phone, email, and a message and has an added emphasis with a very visible send button in contrasting colors. This page will contain contact details (phone and email), social media links, and possibly online chat as some of the ways for customers to reach the restaurant. The model was ensured in very accessible and easy interaction.

Products:

A screenshot of a food website

Description automatically generated

The "Products" page is artfully designed and user-friendly, showcasing a grid layout with excellently captured images for each item, supplemented by short captions describing names, prices, ingredients, and allergens. It gathers product ratings and reviews for social proof and thus contributes to a user's disposition about trust. Navigation tools include category tabs, a search bar, and filtering/sorting options for easy discovery. The page even has features like a "More Products" section, an easy-to-access cart icon, and customer reviews to make the visit a little more motivating. The overall design features an aesthetic and a functional balance that invites customers to browse and engage.

Software interface:

The software interface uses **HTML**, **CSS**, and **JavaScript** for the front-end to create a responsive, interactive user experience. HTML structures content, CSS handles design and layout, and JavaScript enables dynamic features like form validation, menu filtering, and real-time updates. The back-end uses **MySQL** to store and manage data, including users, menu items, orders, and reviews. The database interacts with the front-end via APIs or secure queries for seamless data retrieval and updates. Together, these tools provide a secure and efficient platform for managing restaurant operations online.

Section 8: diagrams

Use case diagram:

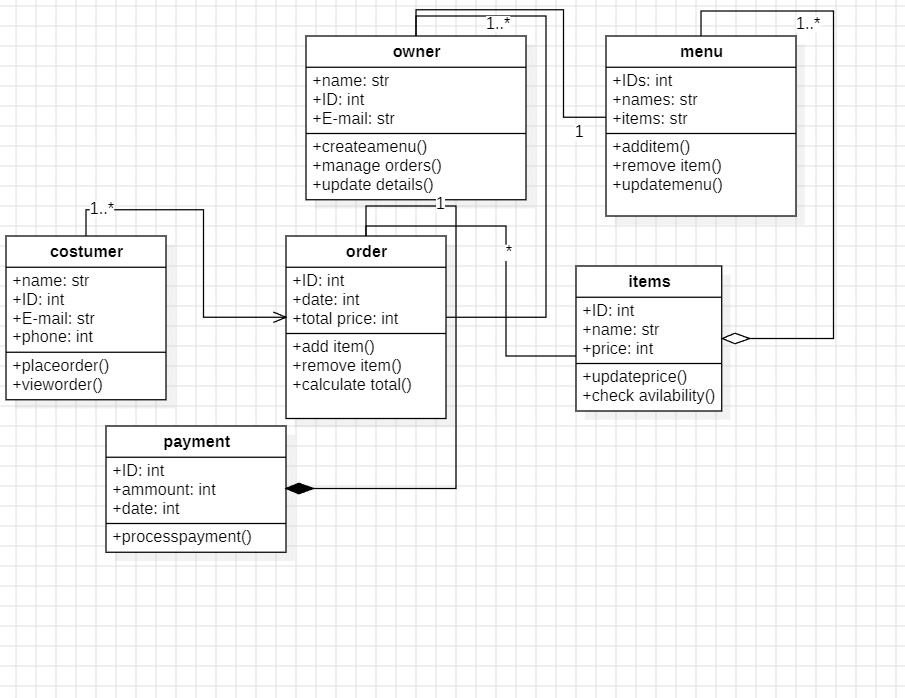
A diagram of a diagram

Description automatically generated

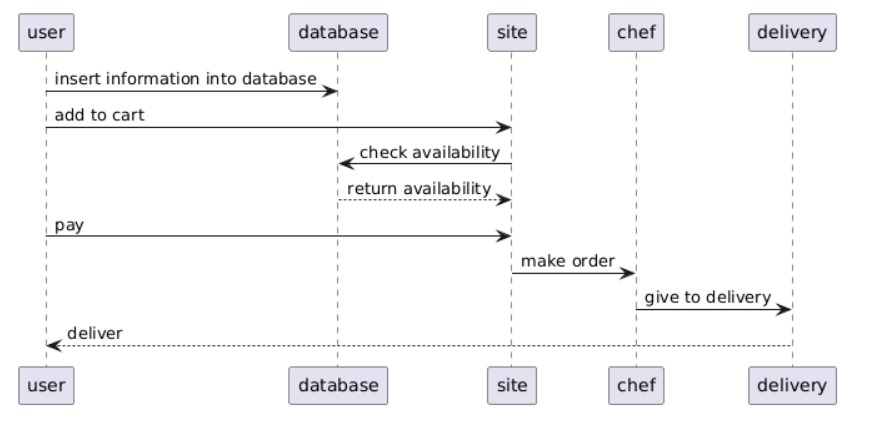
Use case scenario:

|  |  |
| --- | --- |
| Attribute | Details |
| Name | Place Online Order |
| Actor | Customer |
| Entry condition | The customer accesses the restaurant website and views the menu. |
| Exit condition | The customer's order is successfully placed, and they receive an order confirmation. |
| Flow of events | 1. The customer logs in or accesses the website as a guest. |
|  | 2. The customer views the menu. |
|  | 3. The customer selects items from the menu and adds them to the cart. |
|  | 4. The customer proceeds to checkout, reviews the order, and enters payment and delivery information. |
|  | 5. The system processes the payment and displays an order confirmation message. |
|  |  |
| Alternative events | 1. If the payment fails, the system prompts the customer to re-enter payment details or choose a different payment method. |
|  | 2. If the customer cancels the order during checkout, the system discards the current order and redirects them to the menu. |

Class diagram:



Sequence diagram:



Section 9:

Test cases:

**Test Case 1: Verify Login Failure (Backend Not Integrated)**

Test ID:TC01

Priority: High

Module Name: Login

Test Title: Verify Login Failure due to Backend Integration Issues

Test Description: Test the login functionality when the backend is not integrated.

Preconditions: User has valid login credentials.

Test Steps:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Steps | Test Data | Expected Result | Actual Result | Status (Pass/Fail) |
| 1- Open the website |  | Website is open | It open successfully | Pass |
| 2-Navigate to login page |  | Login screen appears | It does not appear because of the failing of integration between backend and frontend | Fail |
| 3-Enter valid username and password | Username:  alaa Password:  12345678 | Data appears in the fields | Data appears in the fields, but the page is alone it doesn't integrate with the website | Fails |
| 4-Click on Login button |  | Error message: 'Unable to connect to server' | It doesn't appear | Fail |

Post-Condition: User remains on the login page and cannot proceed without successful integration.

**Test Case 2: Add Items to Cart**

Test ID: TC02

Priority: Medium

Module Name: Cart

Test Title: Verify Adding Items to Cart

Test Description: Verify that items can be added to the cart.

Preconditions: User is logged in, and menu items are available.

Test Steps:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Steps | Test Data | Expected Result | Actual Result | Status (Pass/Fail) |
| 1-Open the website |  | Website is open | It works successfully | Pass |
| 2-Navigate to menu page |  | Menu page is displayed | It displayed with all menu items | Pass |
| 3-Click 'Add to Cart' |  | Item successfully added to cart | It added successfully | Pass |

Post-Condition: Cart updates with the new item.

---

**Test Case 3: Modify Cart Item Quantity**

Test ID: TC03

Priority: Medium

Module Name: Cart

Test Title: Verify Item Quantity Modification in Cart

Test Description: Verify increasing and decreasing item quantities in the cart.

Preconditions: Items are added to the cart.

Test Steps:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Steps | Test Data | Expected Result | Actual Result | Status (Pass/Fail) |
| 1-Open cart |  | Cart page displays all added items | All items that i added have existed in the cart | Pass |
| 2-Increase quantity |  | Quantity updates correctly | When i click on + it increases | Pass |
| 3-Decrease quantity |  | Quantity updates correctly | When i click on + it decreases | Pass |

Post-Condition: Quantity changes are reflected in the cart and the total is updated.

**Test Case 4: Calculate Total Amount in Cart**

Test ID: TC04

Priority: High

Module Name: Cart

Test Title: Verify Total Amount Calculation

Test Description: Verify that the total amount is calculated correctly.

Preconditions: Cart has items with known prices.

Test Steps:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Steps | Test Data | Expected Result | Actual Result | Status (Pass/Fail) |
| 1-Open cart |  | Cart page displays all added items and prices | All items that i added have existed in the cart | Pass |
| 2-Review total |  | Total matches the sum of item prices | I works successfully and sum the total price | Pass |

Post-Condition: Total amount reflects item quantities and prices.

**Test Case 5: Verify Dark Mode**

Test ID: TC05

Priority: Low

Module Name: UI Theme

Test Title: Verify Dark Mode Functionality

Test Description: Verify the dark mode functionality.

Preconditions: User has access to the dark mode toggle.

Test Steps:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Steps | Test Data | Expected Result | Actual Result | Status (Pass/Fail) |
| 1-Open website |  | Website is open | It opens successfully | Pass |
| 2-Toggle dark mode |  | Website theme switches to dark mode | It is a toggle when i click on the moon it become dark mood, and it change to sun when i click on it again it changes to light mood | Pass |

Post-Condition: Dark mode UI elements adjust correctly.

**Test Case 6: Verify Customer Messaging with Database Issue (Message Sent but Not Inserted)**

- Test ID: TC06  
- Priority: High  
- Module Name: Contact Us  
- Test Title: Verify Customer Message Sent but Not Inserted into Database  
- Test Description: Test the customer messaging functionality where the message is successfully sent but fails to be stored in the database due to a backend issue.  
- Preconditions: The website's "Contact Us" page is accessible.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Step | Test Step Description | Test Data | Expected Result | Actual Result | Status (Pass/Fail) |
| 1 | Open the website |  | Website opens successfully. | It opens as expected | Pass |
| 2 | Navigate to "Contact Us" page |  | The "Contact Us" page loads correctly. | Successfully open | Pass |
| 3 | Enter message details | Name:  Alaa Atef Email: alaaatef@gmail.com Message: "Looking forward to your response." | Input fields accept the provided details. | It works as i can add my data successfully | Pass |
| 4 | Click the "Send" button |  | The information disappears as it sent successfully | The information disappear as it sent | Pass |
| 5 | Verify backend database |  | The message is not inserted into the database due to an identified issue. | Didn't work because the failed integration | Fail |

Post-Condition: The user receives confirmation of the message being sent, but the database is missing the record.

Github: <https://github.com/m0hamedhelmy/Bistro-Bliss-V2.git>