Cafe Bliss – Business Website

Project Overview

This project is a **responsive business website for a café**, built using **HTML, CSS, and Vanilla JavaScript**.
The website showcases the café's brand, menu offerings, business information, and contact details in a visually appealing and user-friendly manner.

Project Goals and Objectives

- Create a professional multi-section business website
- Present menu items with pricing and visuals
- Provide clear navigation across sections
- Ensure mobile and desktop responsiveness
- Add basic interactivity for navigation and form handling
- Maintain clean, readable, and maintainable code

Setup Instructions

This is a frontend-only project and does not require any external frameworks or backend services.

Installation and Configuration Steps

1. Download or clone the project files.
2. Ensure the following structure is maintained:
    - index.html
    - style.css
    - script.js
    - assets/ (images used in the website)
3. Open index.html in any modern web browser:
    - Google Chrome
    - Mozilla Firefox
    - Microsoft Edge

No additional setup or configuration is required.

Code Structure

```
cafe-bliss/
|
├── index.html # Main HTML file (structure & content)
├── style.css # Styling, layout, and responsiveness
├── script.js # JavaScript for interactivity
├── assets/ # Images used across the website
└── README.md # Project documentation
```

File Responsibilities

- **index.html**
  - Defines page sections: Navbar, Hero, Menu, About, Contact, Footer
  - Links external CSS and JavaScript files
  - Uses semantic HTML elements for better structure
- **style.css**
  - Handles layout, colors, typography, and spacing
  - Implements responsive design using media queries
  - Uses Flexbox and Grid for layout management
- **script.js**
  - Manages mobile navigation toggle
  - Handles contact form validation and feedback
  - Prevents page reload on form submission

Technical Details

Architecture

- **Client-side static architecture**
- Separation of concerns:
  - HTML → Structure
  - CSS → Presentation
  - JavaScript → Behavior

Algorithms and Logic

- **Navigation Toggle Logic**
  - Toggles visibility of navigation links on small screens
  - Uses conditional checks on element display state
- **Form Validation Logic**
  - Checks for empty input fields on submission
  - Displays success or error messages dynamically
  - Clears input fields after successful submission

Data Structures

- No complex data structures are used
- User input and UI state are handled through DOM elements

Testing Evidence

Testing was conducted using **manual functional testing**.

Sample Test Cases

Test Case	Description	Expected Result
Navigation Toggle	Click menu icon on mobile	Menu opens/closes
Responsive Layout	Resize browser window	Layout adapts correctly
Menu Section	Scroll to menu	All items display correctly
Empty Form Submit	Submit without filling fields	Error message shown
Valid Form Submit	Submit filled form	Success message shown
Cross-browser Test	Open in different browsers	Consistent behavior

Validation Results

- No JavaScript console errors
- Responsive layout verified on mobile and desktop

- Navigation and form interactions work as expected