**Professional and comprehensive documentation plan**

**Name : Hugo Lima Marerua**

**1. Executive Summary**

* **Objective**: Develop a responsive, user-friendly web-based restaurant menu that aligns with the restaurant’s branding, provides seamless navigation, and enhances customer engagement.
* **Scope**: The project involves designing and developing a fully functional restaurant menu with features like categorization, online ordering, and dynamic updates.
* **Target Audience**: Restaurant customers accessing the menu on desktops, tablets, or mobile devices.

**2. Functional Requirements**

**Core Features:**

1. **Menu Display**:
   * Categorized menu sections (e.g., Appetizers, Mains, Desserts).
   * High-resolution food images and descriptions.
   * Prices displayed for each item.
2. **Search and Filters**:
   * Search bar for specific dishes.
   * Filters for dietary preferences (e.g., vegetarian, gluten-free).
3. **Online Ordering**:
   * Integration with a payment gateway for online orders.
   * Add-to-cart and checkout functionality.
4. **Accessibility**:
   * WCAG-compliant design for inclusivity (e.g., alt text for images, readable contrast).
5. **Content Management**:
   * Admin dashboard for updating menu items, prices, and specials.

**3. Non-Functional Requirements**

* **Performance**: Page load time should not exceed 2 seconds.
* **Scalability**: The system should accommodate increased user traffic during peak hours.
* **Security**:
  + SSL encryption for secure transactions.
  + Protection against vulnerabilities like SQL injection.
* **Cross-Browser Compatibility**: Ensure the menu works on major browsers (e.g., Chrome, Safari, Firefox).

**4. Technologies and Tools**

**Frontend Development:**

* **Languages**: HTML, CSS, JavaScript.
* **Frameworks**: React.js or Vue.js for dynamic and responsive UI.
* **Design Tools**: Figma, Adobe XD for wireframes and UI/UX prototyping.

**Backend Development:**

* **Languages**: Node.js, Django, or Flask.
* **Database**: MySQL, PostgreSQL, or MongoDB for storing menu data.
* **Server**: Nginx or Apache.

**Hosting and Deployment:**

* Hosting on platforms like AWS, Azure, or Netlify.
* Git for version control and GitHub for repository management.

**5. Design Process**

**Wireframing:**

* Create layout designs for the home page and menu sections using Figma or Sketch.

**User Interface (UI):**

* Use brand-specific colors, logos, and fonts.
* Employ visual hierarchy to prioritize key sections like "Specials" or "Order Now."

**User Experience (UX):**

* Optimize navigation for ease of use.
* Add animations for interactive elements (e.g., hover effects, transitions).

**6. Development Plan**

**Frontend Development:**

1. Build menu structure using HTML.
2. Style the layout with CSS for a visually appealing design.
3. Implement interactivity (e.g., search and filters) using JavaScript.

**Backend Development:**

1. Set up the database to store menu data (e.g., items, prices, images).
2. Develop API endpoints for retrieving and updating menu details.

**Integration:**

* Combine the frontend and backend for seamless functionality.
* Integrate payment gateways for online orders.

**7. Quality Assurance**

* **Testing Tools**: Selenium, BrowserStack for cross-browser and device testing.
* **Performance Testing**: Optimize load times using Google Lighthouse and GTmetrix.
* **Security Testing**: Use tools like OWASP ZAP to identify and fix vulnerabilities.

**8. Deployment**

* Push the final build to the hosting platform (e.g., Netlify or AWS).
* Secure a custom domain and SSL certificate for the website.

**9. Post-Deployment Maintenance**

* Monitor website analytics using Google Analytics.
* Regularly update content (e.g., seasonal items, pricing changes).
* Address bugs and security issues promptly.

**10. Timeline and Milestones**

* Week 1–2: Requirements gathering and wireframing.
* Week 3–4: Frontend and backend development.
* Week 5: Testing and quality assurance.
* Week 6: Deployment and live testing.
* Ongoing: Maintenance and updates.