Baikadi Vinay

12216019

Abstract

[Draw your reader in with an engaging abstract. It is typically a short summary of the document.   
When you’re ready to add your content, just click here and start typing.]

BookItNow

A food ordering website

**Software Requirements Specification (SRS) Document**

**1. Introduction**

**1.1 Purpose**

The purpose of this document is to provide a detailed overview of the requirements for the development of a restaurant management website, allowing restaurant owners to upload their menu items, manage orders, and provide contact information for customers.

**1.2 Scope**

The restaurant management website will include features such as:

* Restaurant owner login to manage menu items.
* Ability for restaurant owners to upload menu items, including images, names, descriptions, and prices.
* Display of uploaded menu items for customers to view.
* Customer order management, including viewing current orders and order history.
* Contact information section for customers to reach out to the restaurant.

**1.3 Definitions, Acronyms, and Abbreviations**

* SRS: Software Requirements Specification.
* HTML: HyperText Markup Language.
* CSS: Cascading Style Sheets.
* JS: JavaScript.
* API: Application Programming Interface.
* UI: User Interface.

**2. Overall Description**

**2.1 Product Perspective**

The restaurant management website will be a standalone system that interacts with restaurant owners and customers through a web interface. It will be designed to be user-friendly and accessible on various devices such as desktops, tablets, and smartphones.

**2.2 User Classes and Characteristics**

* **Restaurant Owners**: They will be able to log in to the system and manage their restaurant's menu items.
* **Customers**: They will access the website to view restaurant menus, place orders, and contact the restaurant.

**2.3 Operating Environment**

The website will be hosted on a web server and accessed through a web browser. It will be compatible with modern web browsers such as Google Chrome, Mozilla Firefox, and Safari.

**2.4 Design and Implementation Constraints**

* The website will be developed using HTML, CSS, and JavaScript for the frontend.
* Backend development will be done using Node.js for server-side logic.
* MongoDB will be used as the database management system.

**2.5 User Documentation**

User documentation, including user guides and tutorials, will be provided to assist restaurant owners and customers in using the website effectively.

**3. System Features**

**3.1 Restaurant Owner Features**

1. **Login**: Restaurant owners can log in to the system using their credentials.
2. **Upload Menu Items**: Restaurant owners can upload menu items with details such as name, image, description, and price.
3. **Edit Menu Items**: Restaurant owners can edit existing menu items, including updating details and images.
4. **Delete Menu Items**: Restaurant owners can delete menu items that are no longer offered.
5. **View Orders**: Restaurant owners can view incoming orders placed by customers.

**3.2 Customer Features**

1. **View Menu**: Customers can view the menu items uploaded by restaurants, including images, descriptions, and prices.
2. **Place Order**: Customers can place orders by selecting menu items and providing necessary information.
3. **View Order Status**: Customers can view the status of their current orders, including confirmation and delivery details.
4. **Contact Restaurant**: Customers can access contact information for the restaurant to inquire about reservations, menu items, or other queries.

**4. External Interface Requirements**

**4.1 User Interfaces**

The user interface will consist of:

* **Restaurant Owner Dashboard**: A dashboard for restaurant owners to manage menu items and view orders.
* **Customer Interface**: A user-friendly interface for customers to view menus, place orders, and contact the restaurant.

**4.2 Hardware Interfaces**

The website will be accessible on devices with internet connectivity, including desktop computers, laptops, tablets, and smartphones.

**4.3 Software Interfaces**

The website will interact with:

* Web browsers (Google Chrome, Mozilla Firefox, Safari, etc.).
* Node.js for server-side development.
* MongoDB for database management.

**4.4 Communication Interfaces**

Communication between the client and server will be handled using HTTP requests and responses.

**5. Non-functional Requirements**

**5.1 Performance Requirements**

* The website should load quickly, with minimal latency.
* It should be able to handle multiple concurrent user sessions without crashing or slowing down.

**5.2 Security Requirements**

* User authentication and authorization mechanisms should be in place to ensure data privacy and prevent unauthorized access.
* Data encryption should be used to protect sensitive information such as user credentials and payment details.

**5.3 Reliability Requirements**

* The website should be available and accessible 24/7 with minimal downtime for maintenance.
* Data backups should be performed regularly to prevent data loss in case of system failure.

**5.4 Usability Requirements**

* The user interface should be intuitive and easy to navigate for both restaurant owners and customers.
* Proper error handling and informative error messages should be provided to guide users in case of mistakes or issues.

**6. Other Requirements**

**6.1 Legal and Regulatory Requirements**

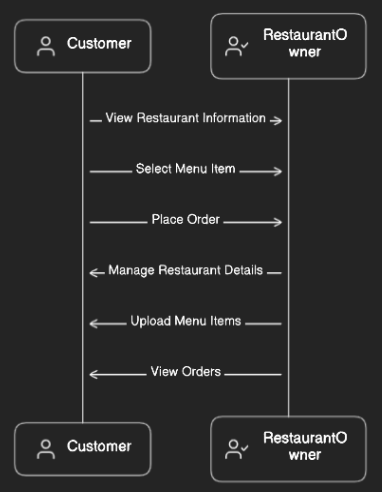
* The website should comply with relevant data protection and privacy laws, such as GDPR (General Data Protection Regulation).
* It should also adhere to any local regulations regarding online food ordering and restaurant management.

**6.2 Documentation Requirements**

* Comprehensive documentation should be provided for system installation, configuration, and usage.
* User manuals and guides should be available to assist users in operating the website effectively.

**7. Appendices**

* Use case diagrams



* Entity-relationship diagrams

