Software Requirements Specification

(SRS)

Hotel Management System

Submitting to:

Prof. TIRUMALARAJU BHAVANI
SCOPE
VIT-AP University

Submitted by:

Govindgari Sai Srushik
21BCE8659
Team-7
VIT-AP University

1. Introduction

1.1 Purpose of the Document

The purpose of this document is to outline the requirements for the development of a Hotel Management System (HMS) with a focus on automating food serving processes.

1.2 Scope of the Project

The scope includes modules for menu management, order processing, kitchen management, inventory tracking, and payment processing within the hotel.

1.3 Definitions, Acronyms, and Abbreviations

- HMS: Hotel Management System
- SRS: Software Requirements Specification

2. System Overview

2.1 System Description

The Hotel Management System is designed to streamline hotel operations, with a specific emphasis on automating food serving processes, enhancing order accuracy, and improving customer satisfaction.

2.2 System Architecture

The system architecture comprises a web-based front end for users, a backend server for processing, and a database for storing relevant data. Integration with point-of-sale systems will also be implemented.

3. Functional Requirements

3.1 User Stories

- As a customer, I should be able to browse the menu, place orders, and make payments online.
- As a chef, I should receive real-time order notifications and be able to update order status.
- As an administrator, I should be able to manage menus, view sales reports, and monitor inventory.

3.2 Use Cases

- Place Order:
  - Actors: Customer, System
  - Preconditions: Customer logged in, menu selected
  - Main Flow:
    1. Customer selects items from the menu.
    2. Customer adds items to the cart.
    3. Customer proceeds to checkout.
    4. System processes payment.
    5. System notifies the kitchen.

3.3 Functional Requirements

- Menu Management:
  - The system should allow administrators to add, update, and delete menu items.

- Order Processing:
  - The system should process customer orders in real-time and update the kitchen staff accordingly.

- Inventory Management:
  - The system should automatically update inventory levels based on processed orders.

4. Non-functional Requirements

4.1 Performance Requirements

- The system should handle a minimum of 1000 concurrent users.
- Response time for order processing should be under 3 seconds.

4.2 Reliability and Availability

- The system should have 99.9% uptime.
- Daily backups of the database should be performed.

4.3 Security Requirements

- User data should be encrypted during transmission.
- Only authorized personnel should have access to the administrative functions.

5. User Interface Requirements

5.1 User Interface Design

- The user interface should be intuitive and mobile-responsive.
- Visual elements should adhere to the hotel's branding.

5.2 Navigation

- Customers should be able to easily navigate through the menu, cart, and checkout pages.

6. System Constraints

6.1 Hardware Constraints

- The system should be compatible with modern web browsers.
- Minimum server requirements: 8GB RAM, Quad-core processor.

6.2 Software Constraints

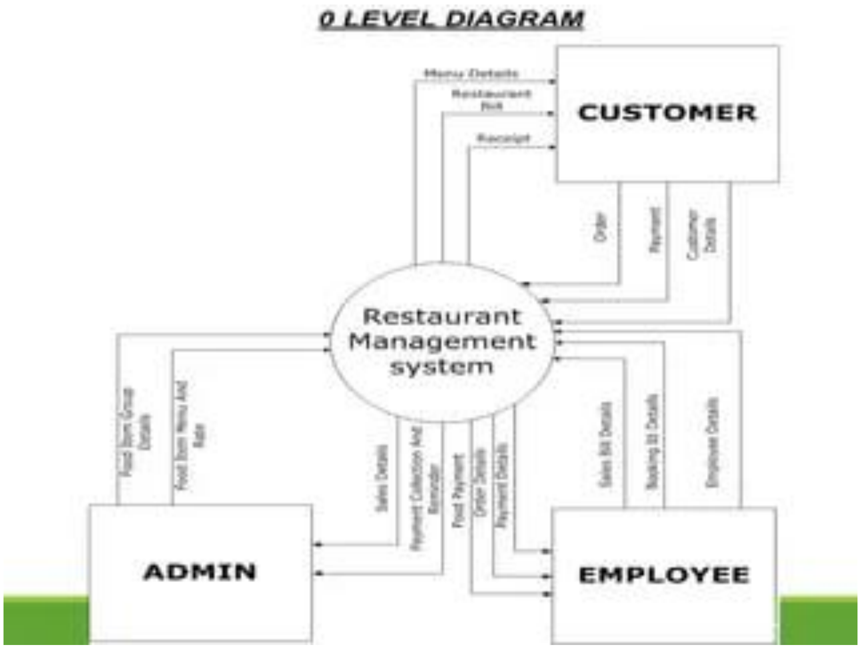
- The system should be compatible with NoSQL database.
- Frontend development should use HTML5, CSS3, JavaScript.

7. Data Requirements

7.1 Database Design

- The database should include tables for customers, orders, menu items, and inventory.
- Relationships should be established to maintain data integrity.

7.2 Data Flow Diagrams



8. Assumptions and Dependencies

- It is assumed that the hotel staff will be trained to use the new system.
- The system is dependent on a reliable internet connection for real-time order processing.

9. Risks and Mitigation Strategies

- Risk: Potential delays in system development.
  - Mitigation: Regular progress meetings and a contingency plan for critical features.

10. Appendix

Documentation Reference: https://www.slideshare.net/smitpatel10192/restaurent-management-system