DATE: 22/8/25 PAGE:

SRS Document

Week-1

A7 Hotel Management System.

1) Introduction

1.1) Purpose of Document -

The purpose of this document is to define the requirements for the Hotel Management system (HMS). The system will provide an efficient way to manage hotel operations such as reservations, check-in/check-out room allocation, billing, customer information, staff information and restaurant services, aiming to reduce manual work and improve efficiency.

1.2) Scope of the Document -

Hotel management system will -

- Allow customers to book rooms online or at the front desk.
- Support restaurant services (menu, orders, bills).
- Enable staff to manage reservations, cancellations and availability.
- Maintain customer records and history.
- Generate reports related to occupancy, revenue and customer statistics.

1.3) Overview -

The HMS includes modules for Customer management, Room management, Booking, Staff management, Restaurant services, billing and reporting.

DATE:

PAGE:

2) General Description.

HMS automates hotel operations: room booking, check-in/check-out, billing, staff, and restaurant services.

→ Users:-

> Guest - Book rooms, order food
> Staff - Manage bookings, orders, housekeeping
> Admin - Manage staff, reports.

→ Modules:-

> ~~Rooms~~

- Room and Booking management
- Customer Management
- Billing System.
- Staff Management
- Restaurant Services
- Reporting Module.

- Implemented as a user-friendly application ~~that~~ with secure database support.
- Ensures efficiency, security and scalability for hotel operations.

3) Functional Requirements.

a) Room Booking and Cancellation
b) Guest check-in/check-out
c) Staff Management (details, shift, salary)
d) Restaurant Management (Menu, orders, billing)
e) Customer Billing (Room + Restaurant charges)
f) Report Generation (Revenue, Occupancy, Staff, Restaurant)

DATE: PAGE:

4) Interface Requirements -
a) User Interface (UI)
b) Hardware Interface
c) Software Interface
d) Communication Interface.

5) Performance Requirements -
- Booking and billing must process within 4 seconds
- Restaurant orders should update instantly
- Should support 200+ simultaneous orders.

6) Design Constraints
- Must use object-oriented design.
- Must comply with hotel and staff privacy rules
- Web version must support standard browsers.

7) Non-Functional Requirements -
> Security - Only authorised staff can access admin and staff records.
> Reliability - 24/7 availability with backup and recovery.
> Usability - Simple interface for non-technical hotel staff.
> Scalability - Extendable to multiple hotel branches.

8) Preliminary Schedule and Budget -
Schedule
- Requirement analysis - 2 weeks —
- System Design - 2 weeks ← budget breakup
- Implementation - 4-6 weeks
- Testing - 2 weeks
- Deployment - 1 week

Budget -
The total budget is about $4500

DATE:

PAGE

6) Credit Card Processing System (CCPS)

1) Introduction -

1.1) Purpose of the document -

The purpose of this document is to outline the requirements and specifications for the development of a Credit Card Processing system. It will define objectives, scope and deliverables, ensuring secure, reliable and efficient handling of credit card transactions.

1.2) Scope of this Document -

The System will -

- Authenticate cardholder details
- Authorize and process credit card transactions
- ~~Staff~~ Support refunds and cancellations
- Maintain transactional history and logs
- Provide reporting tools for merchants and administrators

1.3) Overview -

The system is secure platform enabling customers, merchants and banks to complete credit card transactions seamlessly.

2) General Description -

The credit card processing system will handle authentication, settlement, and reporting of card payments.

Users - Customers, Merchants, Banks, Administrators

Features -

Secure communication, transaction logging, integration with payment gateways, real-time authorization.

3) Functional Requirements -

3.1) Authentication -

- Validate card no., expiry, CVV, and OTP/PIN.
- Prevent unauthorised use.

3.2) Transaction Processing -

- Approve or decline transactions
- Generate electronic receipts

3.3) Settlement and Refunds -

- Process settlements b/w banks and merchants
- Handle refunds and cancellations

3.4) Reporting -

- Generate transaction history
- Provide financial reports for auditing.

4) Interface Requirements

a) User Interface

- Simple and intuitive for merchants and customers
- Accessible via POS, mobile and web applications.

b) Integration Interface -

- Integration with banking API's
- Secure payment gateways.

5) Performance Requirements -
- Response time ≤ 3 seconds
- Handle 10,000+ transactions per day.
- Ensure 99.9% uptime.

6) Design Constraints -
- Most comply with PCI DSS standards
- Use encryption (AES, RSA)
- Object oriented design.

7) Non-functional attributes
> Security - Multi factor authentication.
> Reliability - Backup and recovery mechanisms
> Scalability - Support more merchants and transactions
> Usability - User-friendly for both technical and non-technical users.

8) Preliminary Schedule and Budget
- Estimated timeline : 5-6 months
- Budget : $120,000

Table