

Software Requirement Specification

i Hotel Management System.

1.1 Purpose of this document

The purpose of this document is provide a detailed SRS for hotel management system (HMS). It defines functionalities, features and constraints required for successful development and deployment of HMS.

1.2 Scope of this document

The HMS automates hotel operations such as room booking, check-in / check out, billing, housekeeping and report generation. It will reduce manual errors, improve efficiency and enhance customer experience.

1.3 Overview

The HMS consists of web-based User Interface accessible to customers, receptionists and administrators. Back end Systems stores guest details, reservations, payments and reports Admin Panel for managing hotel operations and monitoring staff.

2 General description

2.1 General functions

~~Room booking and availability check~~

~~Guest check in / check out~~

~~Billing and payment handling~~

~~Housekeeping scheduling~~

~~Report generation for revenue and occupancy.~~

3 Functional Requirements

3.1 User Registration and Authentication

Guests and staff must be able to register and log in

3.2 Room booking and Management

Admins / receptionists can add, update or cancel room bookings.

3.3 Billing and Payments

System generates bills, supports card / online payments and maintain records.

3.4 Housekeeping Management

Track room cleaning schedules and assign tasks.

4 Interface Requirement

- Database : Stores room, guests, booking, payments
- Web Interface : Accessible via HTTP / HTTPS
- API's : REST APIs for transaction.

5 Performance Requirements

Room search results within 2 seconds

- Handle 1000+ concurrent users
- Database optimized for booking queries

6 Design Constraints

Tech : HTML5, CSS3, javascript,

Hardware : Server with 8GB RAM, 1TB Storage

7 Non-functional attributes :-

Security : Encrypt payments, role-based access

Reliability : 99.9% uptime

Scalability : Support multiple hotel branches

8 Preliminary Schedule and Budget

Design : 1 month } week → budget breaking

Implementation : 3 months

Testing : 1 month

Deployment : 1 month

Budget : \$ 70,000.

A

SRS - 2

ii Credit Card Processing System

1.1 Purpose of this document

The purpose of this document is to define the requirements for a credit card Processing system that validates, authorizes and processes secure transaction between merchants and banks.

1.2 Scope of this document

The system will manage credit card validation, real-time transaction authorization, fraud detection, settlements, refunds and reporting

1.3 Overview

The CCPS consists of :

Secure Web Interface for merchants and banks
Transaction Engine for processing card operations
Fraud detection Module To monitor suspicious activity.

General description

a) Functional Requirement

Card Validation & Authorization - check, CVV, expiry date validation

Transaction Processing - Debit / credit authorization, refunds settlements

Fraud Detection - Monitor unusual activity

Reports & audits - Detailed log for compliance

b Interface requirement

Database: Store encrypted card / transaction data.

APIs : Define APIs for bank integration.

Communication: HTTPS, PCI DSS compliance.

c Performance Requirements

Process transaction within 3 seconds

Handle 2000+ concurrent transactions

Database optimized for high-volume writes

d Design Constraints

Tech : Java, Python, PostgreSQL

Security : Must comply with PCI DSS standards

e Non-functional attributes

Security : End-to-End encryption

Reliability : 99.99% uptime

Scalability : Cloud support for high-volume processing

f Preliminary Schedule and Budget

Design : 3 months

Implementation : 5 months

Testing : 2 months

Deployment : 1 month

Budget : \$150,000.

iii library Management System

1.1 Purpose of the document

The purpose of this document is to provide a clear, concise and detailed Requirements Specification for library Management system. This document defines the functionality, features, constraints and general system requirements of LMS to ensure successful development and implementation of the system.

1.2 Scope of the document

This document covers all the essential information related to the development of library Management system, including functional non-functional requirements, user interface, design constraints

1.3 Overview

The library Management System consists of

Web based User Interface : Accessible by library members, librarians & administrators

Backend System : A database for storing book and member details, along with transaction records.

Admin Panel : For library management & administrative functions.

General description.

a) Functional Requirements

2.1 User Registration and Authentication

- User must be able to register & log into the system
- Users will be able to access personalized services
- The system must authenticate users using secure login credentials.

2.2 Book Management

- Admins / librarians manage books
- Updated inventory in real time
- Book details include title, author, ISBN, genre, status

2.3 Borrowing and Returning

Track borrowing and returning transaction

Updated book status with due dates

Issued books marked, due dates set, returned books reset to available

2.4 Fine Management

Calculate and track overdue fines

Members fined for late return

Fines based on overdue days, automatic notifications

3) Interface Requirements

Database: Relational database for storing books, members and transaction.

Admin Interface: Web based panel for book / user / fine / report management.

Member Interface : Web based interface for searching, borrowing & account management.

Web Interface : Communication via HTTP / HTTPS

Data Exchange : REST APIs with JSON.

4) Performance Requirements

- Search results should appear within 2 seconds.
- Login, borrow and return actions should complete with 5 seconds.
- Support up to 1000 concurrent users.

5) Design Constraints

- Use web technologies (HTML5, CSS3, JavaScript, Python)
- Compatible with modern browsers
- Server with minimum 8GB RAM, 1TB storage.

6) Non-functional attributes

- Encrypt user data
- Support Role-based Access
- Ensure 99.9% uptime
- Support horizontal scaling.

7) Preliminary Schedule and Budget

Design : 1 month

Implementation : 3 months

Testing : 1 month

Deployment : 1 month

Budget : \$ 50,000

SRS - H

iv Stock Maintenance System

1 Introduction

1.1 Purpose of this document

The purpose of this document is to specify the requirements and functionalities of the stock Maintenance system. It ensures proper tracking, management and reporting of inventory levels, purchases and sales across the organisation.

1.2 Scope of this Document

The Stock Maintenance System aims to automate the process of monitoring stock level, issuing stock, generating stock reports and avoiding overstocking / stock-outs. The system will be used by store managers, employees, and administrators.

1.3 Overview

The stock Maintenance System is designed to provide real-time information about inventory items, including availability, movement history, supplier details and reorder alerts. It reduces manual effort and improves efficiency and minimizes error.

2 General Description:-

The system will handle day-to-day stock operations including stock entry, issue, update and report generation. It will also generate alerts for low stock & expired items.

3 Functional Requirements

3.1 Stock Entry & Update

- Add new items with details
- Update stock levels after purchase or issue

3.2 Stock Issue

- Allow issuing stock to departments with request tracking
- Deduct issued quantity automatically.

3.3 Reporting & Analysis

- Generate daily / weekly / monthly reports
- Provide inventory valuation and usage history

3.4 Alerts & Notifications

- Reorder alerts when stock falls below threshold
- Expiry alerts for perishable items.

4 Interface Requirements

4.1 User Interface

- Simple and user-friendly for staff
- Dashboard showing stock status, alerts and reports

4.2 Integration Interface

- Integration with accounting and supplier management systems.

5 Performance Requirements

- System should update stock within 2 sec of transaction
- Handle atleast 50 concurrent transaction
- Ensure real-time stock accuracy

6 Design Constraints

- Should run on standard business hardware
- Database: MySQL or PostgreSQL
- Development: Java/Spring boot or Python/Django

7 Non-Functional Attributes

- Security: Authentication for all users, role-based access
- Reliability: 24/7 availability with backup
- Usability: Easy navigation for non-technical user
- Scalability: Handle future branches and warehouse

8 Preliminary Schedule and Budget

Estimated development time: 5 months

Estimated budget: \$ 80,000.

A

SRS - 5

V Passport automation System

1. Introduction

1.1 Purpose of this document

The purpose of this document is to define requirements for the passport Automation System, which aims to simplify and digitize the passport application and issuance process.

1.2 Scope of this Document

The PAS provides services including passport application, renewal, appointment booking, fee payment, and status tracking. It is intended for use by applicants, passport officers and administration.

1.3 Overview

The system reduces manual paperwork by offering an online platform for applying, verifying and delivering passports. It also ensures better tracking and transparency.

2 General Description

The system automates the passport lifecycle from application to delivery. It integrates with police verification payment gateway, and government database.

3 Functional Requirements

3.1 User Registration & application

- Applicants register and fill out passport forms
- Upload required documents online

3.2 Appointment & Scheduling

- Book appointment for document verification
- Reschedule or cancel appointments

3.3 Payments & Fee management

- Online fee payment via integrated payment gateway
- Auto-generation of receipts.

3.4 Verification & Approval

- Passport officer reviews application
- Integration with police database for verification

3.5 Status Tracking & Delivery

- Application tracking their application status
- Notification via SMS/Email after passport dispatch

4 Interface Requirements

4.1 User Interface

- Applicant portal for form submission & status track
- Admin Officer dashboard for verification & approval

4.2 Integration Interface

Integration with national ID database, police records & payment status.

5. Performance Requirements

- Support 10,000+ concurrent users
- Response time < 3 seconds
- Ensure data accuracy and synchronization across system

6 Design Constraints

- Government security and compliance standards
- Must support regional languages
- Database: Oracle / MySQL

7 Non-Functional Attributes

Security: Strong encryption, multi-factor authentication.

Reliability: High uptime with disaster recovery.

Usability: Accessible via desktop & mobile.

Scalability: Handle nationwide traffic growth.

8) Preliminary Schedule and Budget.

Estimated development time: 9 months

Estimated budget: \$ 200,000.

C
D
20/21