

LAB - 1

SRS Documents

(a) Hotel managements Systems

(i) Introduction:

Purpose of the documents:

The purpose of this document is to define the requirements for the development of a Hotel management system. This system is intended to automate hotel operations such as room booking, check-in/check-out, billing, staff management and restaurant or bar services. The SRS provides a detailed description of the system's functionality, performance and constraints, serving as a reference for developers, testers and stakeholders.

(ii) Scope of the documents:

The HMS is designed to improve efficiency in hotel operations by reducing manual effort and minimizing errors. The system will:

- (a) Manage room reservations, availability and cancellations
- (b) Handle customer check in and check out
- (c) Generate billing and invoices automatically
- (d) Maintain customer records and history
- (e) Provide tools for staff scheduling and managing
- (f) Manage restaurant orders, integrate restaurant and bills with room bills, and maintains a record of restaurant sales
- (g) Generate reports such as occupancy rates, revenue and food sales

The system will be web based and accessible by administrator, receptionist, waiters, kitchen staff and customers. Features such as integration with third party travel booking system or advance payment gateway are out of scope for the initial versions.

> Overview:

The HMS will provide a centralised platform where hotel staff can manage daily activities seamlessly. The system will have multiple actors:

(a) customer (guests)

(b) Receptionist

(c) Administration

(d) Waiter / Restaurant Staff

(e) Kitchen Staff

(2) General Description:-

(a) System Perspective: HMS is an integrated system that may be deployed on a hotel server or hosted online. It interfaces with backend database for storing customer, bookings, billing and restaurant details.

(b) User characteristics:-

> Administrator :- Technically proficient, responsible for configuration, menu management and reporting.

> Receptionist : Moderate technical knowledge requires easy to use UI

> Waiter / Kitchen staff : minimal technical knowledge requires simple order entry and order tracking interface.

> customer : minimal technical knowledge, restaurant booking and ordering interface

(1) Assumptions and Dependencies:

- > Stable internet or LAN connection is required for online user
- > Each room and menu has an unique identification
- > Database Server must be reliable to prevent data loss

(2) Functional Requirements:

User Management

Room Management

Reservation Management

Check-in / check out management

Billing and Payment

Staff management

Reporting

Restaurant Management

(3) Interface Requirements

User Interface (UI)

Hardware Interface

Software Interface

Communication Interface

(4) Performance Requirements

- > Systems must support at least 50 concurrent users including hotel staff
- > Average response time for queries and orders should not exceed 5 seconds
- > The system should handle at least 10,000 customer records and 5000 foods ordered
- > Backup and records should be available in case of database failure

(6) Design constraints

- > the system should be developed using open source technologies
- > must comply with related data privacy regulations
- > should be compatible with commonly used browsers
- > Deployment cost must remain minimal due to the budget constraints

(7) Non-functional requirements

- (a) security: user authentication and role-based access control must be enforced
- (b) reliability: the system should have 99.9% uptime during hotel operations
- (c) maintainability: system updates should be easy to deploy without affecting existing data
- (d) scalability: should support additional hotel or browser in future upgrades

(8) Preliminary schedule and Budget

Schedule

Requirement analysis (1 unit / week)

System Design (2 units / week)

Development (8 units / week)

Testing 2 weeks / units

Deployment - 1 week

✓ \$

SRS Document

Credit Card Processing

1 Introduction

1.1 Purpose of this Document

The purpose of this document is to define the requirements and specifications for the development of a credit card processing system. This document aims to provide clarity on the objectives, scope, and deliverables of the system.

1.2 Scope of this Document

This document outlines the design and workings of a credit card processing system which allows secure transactions handling payment validation, fraud detection, and reporting. The scope also includes the estimated time and budget for development.

1.3 Operation

The credit card processing system is designed to handle secure authorization, billing and settlement of credit card transactions. It will facilitate merchants, banks, and customers by ensuring seamless and reliable payment processing.

General Description

The system will provide a secure platform for handling credit card transactions. It will cater to customers, merchants, and financial institutions by enabling features such as

transaction validation, fraud detection, payment authorization, and reporting.

3 Functional Requirements

3.1 Transaction processing

- process credit card payments securely
- validate card details and transaction amount
- provide real-time authorization responses

3.2 Fraud Detection

- Monitor unusual transaction patterns
- Flag and alert suspicious activities
- provide risk scores for transactions

3.3 Merchant management

- Allow merchants to register and integrate with the system
- provide transaction history and settlement reports

3.4 Customer Management

- maintain customer profiles with encrypted card details
- provide customers with transaction receipts and dispute management.

4 Interface Requirements

4.1 User Interface

- intuitive dashboards for merchants and admins
- secure payment interface for customers

1.2 Integration Interfaces

- Integration with banking APIs for authorizations
- Compatibility with payment gateway and POS terminal

6 Performance Requirements

6.1 Response Time

- The system should process transaction within 2 seconds

5.2 Scalability

- Handle a minimum of 10,000 concurrent transactions

5.3 Data Integrity

- Ensure accuracy and consistency of transactions

6 Design Constraints

6.1 Hardware Limitations

- The system should be deployable on cloud servers and support merchant POS devices

6.2 Software Dependencies

- Use secure relational databases (eg. PostgreSQL, MySQL)
- Implement with secure framework (eg Java, Spring Boot)

7 Non Functional Attributes

7.1 Security

- Use encryption standards (e.g. AES, SSL/TLS) for sensitive data
- Implement multi-factor authentication for merchants and admins

7.2 Reliability

- a Ensure 99.99% uptime

7.3 Scalability

- Supports growing number of merchants & transactions

7.4 Portability

- Compatibility with mobile, web, and POS systems

7.5 Usability

- Easy navigation for customers and merchants

8 Preliminary Schedule and Budget

The development of credit card processing system is estimated to take 8 months with a budget of ₹ 2 crores. This includes planning, design, development, testing, and deployment phases.

SRS Document

Library Management System

1.1 Introduction

Purpose of this Document

The purpose of this document is to outline the requirements and specifications for the development of a Library Management System. It defines objectives, scope, and deliverables of the project.

1.2 Scope of this Document

The Library Management System will automate book cataloging, borrowing, returning, and fine calculation process. The system will benefit librarians, students, and faculty members, improving efficiency and reducing manual effort.

1.3 Overview

The system provides a software solution for managing library resources. It allows catalog management users registration, borrowing/returning of books, fine calculation, and report generation.

2 General Descriptions

The Library Management System will serve Librarians, library members, and administrators. It will include book management, member management, transaction tracking and reporting modules.

3 Functional Requirements

3.1 Book Management

- Add, update and remove Books from the catalog.
- Maintains details such as title, author, ISBN and availability.

3.2 Member Management

- Register new members and maintain profiles.
- Track borrowing history and fines.

3.3 Borrowing & Returning

- Allow members to borrow available books.
- Track due dates and calculate fine for late returns.

3.4 Reporting

- Generate reports on borrowed books, overdue items and fines.
- Provide library usage statistics.

4 Interface Requirements

4.1 User Interface

- User-friendly interface for librarians and members.
- Accessible via web and desktop applications.

4.2 Integration Interfaces

- 1. Integration with university database for student information
- 2. Support for barcode scanner and RFID devices
- 3. Performance Requirements.
- 4.1 Response Time
 - The system should respond to user queries within 2 seconds
- 4.2 Scalability
 - Support at least 500 concurrent users
- 4.3 Data Integrity
 - Ensure accurate updates in real-time
- 5. Design Constraints
- 6.1 Hardware Limitations
 - Compatible with standard PCs, barcode scanners and printers.
- 6.2 Software Dependencies
 - Use relational databases eg PostgreSQL, MySQL
 - Implementations with Java or Python frameworks

7 Non Functional Attributes

7.1 Security

- Role based access control for librarians and members

7.2 Reliability

- Ensure minimal downtime

7.3 Scalability

- Allow expansion for more users and larger book inventories

7.4 Portability

- Support access via multiple devices (PC, tablet, mobile)

7.5 Data Integrity

- Ensure accurate book and transaction records

8 Preliminary Schedule and Budget

The development of the Library Management System is estimated to take 5 months with a budget of ₹ 65 Lakhs. This includes design, development, testing and deployment.

✓✓✓

SRS Document

Stock Maintenance System

Introduction

1.1 Purpose of this Document

The purpose of this document is to define the requirements and specifications for the development of Stock Maintenance System. It provides a clear understanding of the system's goals, scope and deliverables.

1.2 Scope of this Document

The Stock Maintenance System will automate inventory tracking, stock updates, supplier management and reporting. It will benefit store managers, employees, customers, and business owners by maintaining accurate stock levels and reducing losses.

1.3 Overview

The system ensures efficient management of stock by providing functionalities like item tracking, purchase / sales record management, stock alerts and reporting.

General Description

The Stock Maintenance System will serve administrators, warehouse staff, managers etc. It will include modules for inventory control, supplier management, transaction records, and reporting.

3 Functional Requirements

3.1 Inventory management

- Add, update and remove stock items,
- Track item details like ID, name, quantity and price
- Generate alerts for low stock

3.2 Supplier management

- Register suppliers and maintain contact details
- Track purchase history with suppliers

3.3 Transaction Records

- Record purchases, sales and stock transactions
- Generate invoices and receipts

3.4 Reporting

- Generate reports on stock levels, sales report trends, and supplier performance
- Provide analytics for inventory optimization

4 Integrative Requirements

4.1 User Interfaces

- Easy to use interface for staff and managers
- Accessible on desktop, tablet and mobile devices

4.2 Integration Interfaces

- Integration with accounting system for financial reporting
- Support for barcode and RFID scanning devices

5 Performance Requirements

5.1 Response Time

- The system should update stock data within 2 seconds of transaction

5.2 Scalability

- Support management of at least 100000 stock items

5.3 Data Integrity

- Ensure accurate update for all stock operations

6 Design Constraints

6.1 Hardware Limitations

- Compatible with warehouse computers, barcode scanners and printers

6.2 Software Dependencies

- Use relational database system like MySQL, PostgreSQL etc
- Implementations using Java or Python frameworks

7 Non - Functional Attributes

7.1 Security

- Role based access control for staff and managers.

7.2 Reliability

- Ensure reliable stock data with backup mechanism

7.3 Scalability

- Allow system expansion for larger businesses.

7.4 Reusability

- Modular design for integration with other business tools

7.5 Data Integrity

- Ensure real-time and accurate stock record

8 Preliminary Schedule and Budget

The development of the Stock Maintenance System is estimated to take 6 months with a budget of £10,000. This includes planning, development, testing and deployment.

✓
✓

SRS Document

Passport Automation System

1 Introduction

1.1 Purpose of this Document

The purpose of this document is to specify the requirements and functionalities of a passport automation system. It defines the objectives, scope and deliverables of the project.

1.2 Scope of this Document

The passport automation system will streamline the process of applying, verifying and issuing passports. It aims to benefit citizens, passport officers and government authorities by reducing manual effort, ensuring accuracy and improving service efficiency.

1.3 - Overview

The system provides an end-to-end digital platform for passport applications, verification, document submission, and status tracking. It ensures secure data management and seamless integration with government databases.

2 General Description

The passport automation system will serve applicants, passport officers, and administrators. It will include modules for user registration, application submission, document verification,

appointment scheduling and report generation.

Functional Requirements

3.1 Application Submission

- Allow users to register and submit passport applications online.
- Upload required documents in digital format.
- Generate an application reference number.

3.2 Appointment Management

- Provide applicants with appointment scheduling options.
- Notify applicants of confirmed appointments via email/SMS.

3.3 Document Verification

- Verify submitted documents against government databases.
- Flag discrepancies for manual review.

3.4 Payment Processing

- Allow secure online payments for passport fees.
- Generate payment receipts.

3.5 Passport Issuance & Tracking

- Update application status at each stage.

- Notify applicants when passports are dispatched
- Allow applicants to track passport delivery

4 Interface Requirements

4.1 User Interface

- Simple multilingual and user-friendly interface
- Accessible via web browser and mobile applications

4.2 Integration Interfaces

- Integration with national ID and criminal record databases
- Integration with payment gateways

5 Performance Requirements

5.1 Response Time

- System should respond to user actions within 3 seconds

5.2 Scalability

- Support at least 1 million users

5.3 Data Integrity

- Ensure accuracy and consistency of all application data

6 Design Constraints

6.1 Hardware Limitations

- Compatiable with standard government infrastructure and biometric devices

6.2 Software Dependencies

- Use secure relational databases (e.g. PostgreSQL, MySQL)
- Implement with framework supporting high security (e.g. Java, Spring Boot)

7 Non-Functional Attributes

7.1 Security

- Use strong encryption for personal data
- Implement role-based access and biometric authentication

7.2 Scalability

- Expand system to handle growing application demands

7.3 Portability

- Accessible across devices including PC, tablet and smartphones

7.4 Reusability

- Modular System design for integration with other e-governance applications

7.5 Data Integrity

- Maintain tamper-proof logs of applications and approvals

8 Preliminary schedule Budget

The development of the Passport Automation System is estimated to take 9 months with a budget of ₹ 2.5 crores. This includes system analysis, development, testing, and deployment.

