19/8/25

OOM LAB

Software requirement specification (SRS)

Introduction
- Purpose of this document
- Scope of this document
- Overview

General description
Functional requirements
Interface requirements
Performance requirements
Design constraints
Non functional attributes
Preliminary schedule and budget

Problem statement
(Manual hotel operations are ~~thereon~~ inefficient, creating the need for an automated system to streamline bookings & cust mang)

→ Hotel management system

(Title)

Problem statement: Manual management of hotel operations, room allocation, billing and customer records is inefficient and prone to errors. It often causes issues such as double booking, delays in check-in / check out. With the growing demand for quick and online services hotels need an automated system to streamline operations, improve accuracy and enhance customer satisfaction.

Purpose →

1. SRS for HMS

1.1 Purpose of HMS

The purpose of this document is to define the requirements for the HMS. It aims to automate hotel operations such as reservations, room allocation, billing, check in / check out. It also ensures better customer service, accuracy and efficiency.

1.2 Scope of HMS

The system will provide func like room bookings, customer information management, billing payment processing, house keeping management and report generation. It will help the hotel staff reduce manual errors. The doc also highlights the cost, dev efforts and benifits of the system.

1.3 Overview

The HMS is designed to streamline hotel operations and enhance customer satisfaction.

2. General description

The HMS allows users to interact with the system for room reservations, cancellations, checkins checkouts and bill payments. The system provides features like

- online/offline room booking
- customer data management
- Billing and payment processing
- Employee management
- Reports & analytics

3. Functional Requirements

User registration & login : secure login for admin, staff & custo.
Room management : Add, Update or remove room details
Booking management : Reserve rooms, cancel, check availability
Customer management : Check in/out records, store records
Billing management : generating invoices, discounted / taxes
Payment integration : support for multiple payment methods
Housekeeping management : cleaning & track housekeeping status
Report generation : daily, weekly reports for occupancy, revenu & staff performance

4. Interface Requirements

User interfaces : Guest portal, staff & admin portals
Hardware interfaces : Desktop, laptop or mobile devices
Software interfaces : Database (MYSQL), Payment gateway
Communication interfaces : Internet connectivity

5. Performance requirements

System should handel at least 500 concurrent users.
average response time should be less than 3 sec
maximum downtime should not exceed 1% per year
Database must support thousands of booking records efficiently.

6. Design constraints

The system should be web based and mobile feriendly
Must comply with payment gateway security standards
Limited to supported browsers
must work with hotel's existing hardware infrastructure

7. Non functional attributes

Security : encryped passwords
Reliability : ensures 24/7 availability
Portability : compatible with windows, mac & android
Scalability : should support multiple hotels in future expansion
Usability : Simple interface for both staff and customers
Data integrity : Prevent data loss with regular backups.

8. Preliminary Schedule & Budget

Development time : 4-6 monthes (17 - 26 weeks)
Estimated cost : Depends on time size ₹ 8 - 12 Lakhs
Resource requirement : Developers, UI/UX designer, Database administrator, QA team.

Table ,

→ Credit card processing.

Problem statement:
Manual handling of credit card transaction is slow error prone an insecure. Issue like delayed payments fraudulent and inaccurate record keeping affect both customers and business. With the growing need for secure and instant processing, an automated credit card processing system is required to ensure fast transaction, accurate billing, fraud detection and improved customer trust.

1 Introduction

1.1 Purpose

Purpose of doc is to define the req for a credit card processing system. The system will automate and secure credit card transactions for customers, merchants and banks. It ensures accurate billing, fast payments, fraud detection and compliance with security standards.

1.2 Scope

The system will allow merchants to accept customers to make purchases and banks to authorize transactions securely It will handel authorization authentikation. The scope also includes fraud detection, secure data storage and compliance with financial regulations

1.3 Overview

The CCP system acts as a bridge between merchants, customer and banks ensure seamless and secure transactions.

store
67

2. General description

The system enables processing of credit card transactions in real time ensuring secure communication between customer, merchant and bank. It provides features like

- Secure card verification
- Realtime authorization and settlement
- Fraud detection and prevention
- Transaction history and reporting
- Refunds and chargebacks handling.

3. Functional requirements.

- User authentication
- Card validation
- Transaction processing
- Authentication
- Settlement
- Fraud detection
- Transaction history
- Notifications
- Reports & analytics.

4. Interface requirements

- User interface
- Hardface Interfaces
- Software Interfaces
- Communication Interfaces

5. Performance requirements

- must process transactions within 2-3 seconds
- Should support at least 10,000 concurrent transactions
- maximum error rate < 0.1%
- System uptime should be 99.9%.

6. Design constraints
must comply with PCI DSS security standards
use encryption for data transmission
should be compatible with major operating systems eg brow
limited by network latency and banking API's response time

7. Non functional attributes
Security
reliability
scalability
Portability
Usability
Data integrity

8. Preminiary schedule and budget
Development time 26 - 34 weeks
Estimated cost ₹ 20 - 30 Lakhs depending on security and compliance features.
Resources requirement : Developers, security specitists, Database administrators, QA team.

store
67 /

→ Liblaery management system.

Perolblem statement

manual management of liblaery records including book cataloging issuing and member tracking is time consuming and perone to erreors. This often leads to misplaced books, difficultys in tracking due dates, inefficiency in serving students & staffs highliting the needs for an automated liberaery management system.

1. Introduction

1.1 Perpouse

The perpose of this document is to outline the requirements for a liblaery management system. The system will automate liblaery operations such as book cataloging, issuing / returning fine calculations and member management.

1.2 Scope

The system will allow librarians to manage books, members & transactions efficiently students & staffs can search, borrow & reserve books while administrations can generate reports and manage inventory It eleminates manual record keeping, reduces error and impeoves acessibility.

1.3 Overview

The LMS integrates book management member service and reporting into a single system through desktop & web interfaces

2. General description

→ adding, updating and deleting book records
→ managing member registrations & perofiles
→ issuing returning & reserving books
→ Calculating & collecting fines
→ generating reports for books, members and transactions

3. Function requirements

User management → register, update, delete member details
Book catalog management → add, search, update & remove books
Issue & return → issue books to members, track due dates & process
reservation system → allow members to reserve book in advance
fine calculation → automatically calculate overdue fines
search & filter → search books by title, author, ISBN.
reports → generate reports for issued books overdue fines
notifications → send reminders for due dates & over due fines
admin controls → manage staff configure policies.

4. Interface requirement

User interface : librarian portal
member portal
Hardware interfaces : desktop computers, barcode scanner
Software interfaces : database, email
Communication interfaces : Internet / intranet access.

5. Performance requirements

Should handel at least 500 concurrent users
book search should return result in <2 seconds
system uptime must be 99% or higher
must support 10,000+ book records efficiently

6. Design Constraints

- Should be web based with mobile support.
- database must prevent duplicate book entries
- must comply with institutional IT security policies
- limited by network speed and server capacity.

7. Non functional attributes

Security : role based access for librain, admin and members

reliability : regular backup to prevent data loss

Usability : simple interface for both librarians and students

Scalability : expandable to multiple branches of the library

maintainability : easy to update book records & user details.

8. Preliminary schedule & Budget.

Phase	Duration	Cost	Resources Required
Requirement analysis	2-3 weeks	1-1.5L	Business analyst, Project manag
System design	3-4 weeks	1.5-2L	system arch, UI/UX designer
Development	6-8 weeks	2.5-3L	developers, database admin
Testing & QA	3-4 weeks	0.8-1.2L	QA engineers, testers
Deployment	1-2 weeks	0.5-0.8L	devops engineer, system admin
Training & main-tenance	ongoing	0.5-1.5L	trainer, support staff & maintenance staff & team.

→ Stock maintenance system

Problem statement:
manual tracking of stock records, purchase and sales often leads to errors, delays and difficulty in monitoring inventory levels.

1. Introduction:

1.1 Perpouse
is to specify the requirements for a stock maintenance system The system will help business track inventory, manage stock levels, process orders and generate reports to ensure smooth supply chain and warehouse operations.

1.2 Scope
The SMS will allow administrators and staff to add, update and monitor stock items in real time. It will track purchase & sales transactions trigger alerts for low stock, generate invoices and maintenance supplier and customer details.

1.3 Overview
This system integrates stock management, purchase/sales tracking and reporting into one centralized application that supports both small and large businesses.

2. General description
- adding, updating and deleting stock items
- monitoring stock quantities and availability
- processing purchase and sales orders.
- managing suppliers and customer details
- generating stocks, sales and purchase reports.
- sending alerts for low or excess stock.

3. Functional requirements.

User management: admin and staffs login with role based.

Stock management: add, update, delete and search stock item with details.

Purchase management: record purchases, supplier details and update stock levels.

Sales management: record sales transactions, customer details & deduct from stock.

Low stock alerts: notify admin/staffs when item reach reorder level.

Reports & analytics: generate daily, weekly or monthly reports for stocks.

Invoice / billing: generate bills for sales and purchases

audit trail: maintain transaction history for accountability.

4. Interface requirements

User interface: admin dashboard (stock overview, reports, alerts)
Staffs panel (add / update stock, record purchase

Hardware interface: barcode scanner printers, desktop / laptops

Software interface: database integration with accounting tools.

Communication interface: lan / internet access for real time updates.

5. Performance requirements.

→ should handel thousands of stock records efficiently
→ response time for stock search <2 seconds
→ support 100+ concurrent users
→ system uptime must be 99% or higher.

6. Design Constraints

→ should be scalable for small and large businesses.
→ must prevent duplicate stock entries
→ limited by database capacity & network speed.
→ should comply with organizational data security policies.

7. Non functional attributes

security : role based access, data encryption, audit log

reliability : ensure 24/7 availability with regular backups

usability : simple dashboard for staff with minimal training

scalability : support multiple warehouse branches in the future

maintainability : easy update for stock categories and transaction

8. Preliminary schedule & budget.

Phase	Duration	Estimated Cost	Resource req.
Requirement analysis	2 weeks	0.8 - 1 Lakh	Business analyst, project man
System design	3 weeks	1 - 1.5 Lakh	Sys architect, UI/UX designer
Development	6 - 8 weeks	2.5 - 3 Lakh	developers (Front, back DBA)
Testing & QA	3 weeks	0.8 - 1 Lakh	QA engineers, testers
Deployment	1 - 2 weeks	0.5 - 0.8 Lakh	DevOps Engineers, Sadmin
Training & maintenance	Ongoing	0.5 - 1 Lakh Per year	Trainer, support staff.

→ Passport automation system.

Problem statement:
An automated passport automation system is needed to streamline applications, verifications and issuance with improved efficiency transparency and security.

1. Introduction

1.1 Purpose
is to outline the requirements for a passport automation system the system will simply and automate the process for passport application, verification, approval and issuance, reducing manual error and processing delays.

1.2 Scope
The PAS will allow applicants to apply for new passports, renewal and reissue online. It will enable government staff to verify applicant details, process police policies and approve passports efficiently. The system also approves appointment scheduling fee payment and status tracking.

1.3 Overview.
This system integrates applicants, passport offices & police departments into a centralized platform to ensure faster, transparent and secure passport processing.

2. General description
→ Online application form submission
→ appointment upload and verification
→ fee payment through secure gateway
→ police verification tracking
→ application approval rejection
→ passport generation & delivery
→ applicant status through notifications.

3. Functional requirements.

User registeration & login: secure login for applicants and passport officers.

application form: online submission for new renewal or reissue applications.

document management: upload, store & verify identity.

appointment scheduling: book slots at nearest passport office

fee payment: integration with payment for online payments.

verification process: police and office verification & document and details.

application tracking: applicants can check real time status of applications.

Passport generation: system generates passport records for approval applications

Notification: SMS / email alerts for applications updates.

4. Interface requirements.

User interface:

application portal

Passport office portal.

admin portal.

Hardware: desktop, biometric devices, printers

Software: database, payment gateway, police verification API

Communication: internet access with secure SSL / TLS encryption

5. Performace requirements

Must support 5000 + concurrent users

transaction should complete < 5 seconds

system uptime must be 99.9%.

database should support millions of records with high reliability

6. Design constraints.

- Must comply with government IT security and data protection policies.
- Strong encryption required for sensitive data.
- Limited by network speed and third party API response time.
- System should be web based and mobile friendly.

7. Non functional attributes

Security: end to end encryption, two factor authentication, secure data storage

Reliability: backup and disaster recovery to ensure data integrity

Scalability: support nationwide usage across multiple passport offices

usability: simple and accessible interface for applicants.

maintainability: easy to update workflow as per government policies changes

Transparency: clear status to applicants.

8. Preliminary schedule & budget.

Phase	Duration	Estimated Cost	Resources requirements
Requirement analysis	3 weeks	1-1.5 Lakh	Business analyst, gov Consultant, Project manager
System design	4 weeks	2 Lakh	System architect, UI/UX disig
development	10-12 weeks	5-6 Lakh	developers, database admin
testing & QA	4 weeks	1.5-2 Lakh	QA engineers, GOV IT sec
Deployment	2 weeks	1 lakh	Devops engineers & staff
Training & maintenance	ongoing	1-2 Lakh/year	Trainers, support team.