

SRS Document for Hotel Management System

1 Introduction

1.1 Purpose

Manages hotel operations including managing guest check-in/check-out, room service request, billing and reporting. This document is intended to guide developers, testers, project administrators and managers in designing, implementing and validating system.

1.2 Document Conventions

• Dates in DD-MM-YYYY format.

• Currency in Rupees (Rs)

• Time in 24 hour format.

1.3 Interested Audience & Reading Suggestions

• Developers: Focus on functional and non-functional requirements.

• Testers: Review acceptance criteria in functional requirements.

• Hotel Management: Review overall description & scope.

• Investors & Stakeholders: Read purpose, scope and key features for decision making.

1.4 Project Scope

HSMS will allow hotels to:

• Automate booking and room allocation.

- Track room status
- Integrate the system for restaurant and room billing
- Generate report for company, revenue & performance trends

The system will be cloud based with offline support. It will have access to million sized hotels and can scale for larger restaurants.

1.7 References

- IEEE Std Recommended practice for software measurement specification
- World Technology Report, 2004

2 Overall Description

2.1 Product Requirements

This will replace manual over-stocking with automated software solution, accurate & user friendly platform. It will integrate with payment gateways, key card management system, accounting software.

2.2 Product Functions

- Reservation Management i.e. manage room booking (new, modify, cancel)
- Guest check-in and check-out
- Room availability and status tracking
- Invoice generation and payment processing

- Staff and user access management
- Email, mobile and messaging alerts
- Configuration via excel sheet
- Customer feedback tracking and sentiment logging

2.3 User Goals and Constraints

- Administrator manages user access and system safety. They have full access
- Management handles logging, stock related issues, booking and food cost entry
- Manager view reports and monitor reservation. Read only access to all data
- Housekeeping updates room cleaning status. Limited access updates room
- Guest uses self service portal for booking. Access only booking & feedback

2.4 Operating Environment

Web App: Chrome, Firefox, Edge
 Mobile App: Android 10, iOS 14+
 Server: cloud server (AWS/Azure) with SQL database

2.5 Design and Implementation

- Uses low open source technologies
- Part comply with data protection laws
- Payment processing must use PCI DSS-compliant
- User interface must be accessible

- 2.4 User Documentation
- Installation guide
 - User manual
 - Online help system
 - API Documentation

3. Assumption and Dependencies

- Assume internet connection for full functionality
- Third party API for payment & logging system services

4. Specific Requirements

4.1 Functional Requirements

- System shall allow users to make, update and cancel subscriptions
- System shall process payments via integrated gateway
- System shall track active status of users

4.2 External Interface Requirement

- UI intuitive dashboard with drag & drop layout editor
- API REST API for integration with external services

4.3 Non-functional Requirements

- Performance: Must handle 100 concurrent users with response time under 2s
- Security: Data encryption in transit and at rest

5. Technical Requirements

5.1 System Features

- User registration and login
- Subscription management
- Payment processing
- Reporting and analytics

5.2 Architecture

- Frontend: React.js
- Backend: Node.js / Express.js
- Database: PostgreSQL
- API: REST API

2.1 Demand for credit card Processing

1. Introduction

1.1 Purpose
System will centrally authorize, process and settle credit card transactions for merchants, ensuring compliance with financial regulations and industry standards.

1.2 Business Objectives
Standard technical & financial process with less work with dependencies, OTO provided in the following way.

1.3 Stakeholder Analysis & Training Suggestions
Developers, Tester, Financial Analysts, Auditor, Customers and bank employees.

1.4 Project Scope

will automate billing, payments, receipts with OTO & integration, fraud detection and enable merchants to manage credit cards.

1.5 References

ISO banking guidelines

2 Overall Description

2.1 Product Purpose

System will work as part of bank's IT

information system to support payment processing.

2.2 Product Functions

2.2.1 Transaction Processing

- Generate bills and statements
- Fraud detection & alerts
- Card activation/deactivation

2.2.2 User Access

- Customers
- Bank employees
- Admins

2.2.3 Operating Environment

- Banking servers with high uptime
- Cloud services

2.2.4 Constraints

- PCI DSS, PSD2, etc. regulatory & security standards

2.2.5 User Documentation

- Admin, User & cardholder support

2.2.6 Assumptions & Dependencies

- Access to network & payment gateway

13/12/23

1. Overall Description

- 1.1 Product Perspective
It is a standalone system that works as a centralized database. Its modular design will be aligned with business.

1.2 Product Features

- Add / edit / delete / search books
- Register members
- Issue / return books
- Manage due date / fines
- Generate reports

1.3 User Classes

- Librarian: manages library database
- Member: borrow / return books

1.4 Operating Environment

- Windows Linux, macOS
- Chrome, Firefox
- MySQL
- Python / Java / PHP

1.5 Design and Implementation Constraints

- Must be web-based
- Support atleast 1000 concurrent users
- Use of open source technology

2. User Requirements

- User Manual
- Admin Guide
- PDF documents for reference

2.1 Nonfunctional and Dependencies

- User must have internet connection
- Proper authentication
- System deployed on secure cloud server

3. Functional Requirements

3.1 Functional Requirements

- Book management - The system shall allow Librarian to add new books with title, author & ISBN
- Member management - System shall allow registration for new members

3.2 Non functional Requirements

- System shall be available 24/7 of the time
- System shall respond to user queries within 2s

3.3 External Interface Requirements

- Responsive UI for desktop & mobile screen
- Page, book catalogue / member form

4. Appendices

- mobile app integration
- QR code scanning for issues/returns

5. Preliminary Schedule & Budget

- Requirement analysis & design (18 weeks)
- core development (40) and basic management modules (15 weeks)
- implementation of system (release & review for 12 weeks)
- testing & development prep (1 week)
- launch & support (1 week)
- 5 months estimated time
- Estimated budget 20 weeks approx

SRS Document for Stock Management System

1. Introduction

1.1 Purpose

Specify requirements for stock management system. It is intended for business, stock managers, sales and stock users.

1.2 Document Conventions

Text text used for main heading of interface labels. Filled list used for action options or text.

1.3 Intended Audience & Reading Guidelines

- Project manager to understand project scope, deliverables & system constraints
- Software engineers to design, implement & integrate system components.

1.4 Scope

It will not cover system with application used by customers. System will manage stock, sales, warehouse, retail, stock or individual settings.

1.5 References

IEEE 830-1998-1998 recommended practice for software requirement specification

2. Overall Description

- 2.1 Product Perspective
- This is web application based stock market with backend database. It can be used by investing managers, research staff and business experts.

2.2 Product Function

- User authentication and access control
- Add/updates/delete stock items
 - Show current stock levels
 - Generate call/purchase history
 - Alert minimum stocks
 - Reporting & analytics

2.3 User Classes & Characteristics

- Admin full system access
- Manager view reports & manage stocks
- Staff update stock list

2.4 Operating Environment

OS: Windows/MacOS

Browser: Chrome, Firefox

Database: MySQL

Server: Apache

2.5 Constraints

- System must be web based
- Use of open source technologies
- Support for atleast 1000 concurrent users

2.6 Other Documents

- User Manual
- Admin Guide
- API Documentation

2.7 Assumptions and Dependencies

- Users should have basic computer skills. System can be tested by manual testing. Provide backup integration.

3. Specification Requirements

3.1 Functional Requirements

- User management - System shall allow administrators to create and manage user accounts.
- Stock management - System shall allow users to add new stock items and delete stock in user name.

3.2 Non Functional Requirements

- System shall have 99.9% uptime availability.
- System shall respond to user queries within 5s.
- System shall have all data.

3.3 External Interface Requirements

- Require UI for admin & user login page, dashboard, ^{reporting} ~~data~~ analytics, home, referral forms.

4. Approach

- Initial brief, information
- All team working for consultation
- Subsequent work according to needs
- No fixed standard format

5. Preliminary Schedule & Endpoints

Week

- 1-2 Requirement analysis & system design
- 3-4 Low development of tech. working
- 5-6 Implementation of verification

- 7 System testing
- 8 User acceptance testing & subsequent preparation

- 9 Deployment & post-launch support

Schedule Budget - £15 weeks approx.

1.1. Document for Standard Submission

1.2 Introduction

1.3 Purpose

It is a specific functional and non-functional requirement for a project submission system. It is intended to maintain a record of applying for supply and using project.

1.4 Document Location

Each unit and unit for more handling a number of units that are used for using.

1.5 Functional Analysis & Testing Application

Project manager to understand project scope, determine the system environment, determine the design, implement & integrate the system requirements.

1.6 Scope

It is a user-based application to support and operate to include other application submission, application submission, document submission, data submission, application report, data and working.

1.7 References

IEEE Std 800-1047 - 1997 Recommended Practice for Software Requirements Analysis

2. General Description

2.1. Product Description

It is a distributed system consisting of
public facing web application, domain controller
database & admin panel for officials.
It also integrates with national services
database & payment gateway.

2.2. Product Features

User registration & login online application
form submission, updating & validation of
documents.
Appointment scheduling
Tracking application status
Admin approval/rejection
Verification system

2.3. User Classes & Characteristics

Applicants can apply, upload documents
book slots.
Admin manages applications, verifies
documents & issues passports.
Police officer verifies applicant information.

2.4. Operating Environment

Web server: Apache

Database: MySQL

Database: MySQL

2.5. Design & Implementation

Active government service portal
multilingual, multi language
support with user interface

2.6. User Requirements

Online User Manual - help by chat, email, phone
FAQ section

2.7. Assumptions & Dependencies

Internet access is required
user payment valid ID card
existing email gateway interface

3. Application Requirements

3.1. Functional Requirements

- User login & registration
- Application submission
- Appointment scheduling
- Document & photo upload
- Passport issuance
- Application tracking

3.2. Non-Functional Requirements

3.2.1. ID Requirements

- System shall support active users around
40000
- All data encrypted using AES-256

System shall be available 24/7 with
99.9% uptime

3.2 External Interface Requirement
web interface with form inputs, file upload
components and dashboard

3.4 Appendices

sample passport application form integration
with accounting software

5 Preliminary & Budget

week 1-3 Requirement gathering & system design

3-5 development of application submission &
tracking modules

6-7 development of verification interface &
administrative dashboards

8 testing & bug fixing

9 user acceptance testing & deployment

10 deployment & ongoing support

Estimated Budget 40 lakhs app including
development, security compliance & deployment

21/3/24