

# OOM - LAB 1

## Software Requirements Specification (SRS)

### Hotel Management System

#### 1. Introduction

##### 1.1 Purpose of the document :

The main purpose of this document is to present the requirements (functional and non-functional) of Hotel Management System (HMS). This document serves as a guide for the stakeholders, project manager, system architect and developers.

##### 1.2 Scope of the document

This document covers the overall functionality of the HMS, highlighting its goal and the value it provides to users for streamlined operations.

##### 1.3 Overview

The HMS is designed to automate most of the manual work done, including check-ins, reservations, billing, inventory management, etc.

## 2. General Description : ~~MIS~~

The HMS provides features which facilitates tasks for receptionists, managers & guests. Key feature includes reserving a room, and ~~Table Reservation~~.

It also provides some of the admin features such as billing, managing reservations and inventory.

## 3. Functional Requirements

- Login / Registration :- User must be able to login to the application or register and create a new account.

- Room Reservation :- User must be able to book a room on their preferred date.

- Table Reservation :- User must be able to book a table for their preferred time slot.

- Feedback :- customers must be able to provide feedback to the service provided by the Hotel.

## 4. Interface Requirements :-

- User interface - Windows & mobile

## 5. Performance Requirements

- Reliability: System must be able to perform bookings without any crash / server down during viral hours.
- Security: Should have a minimum uptime of 95%.
- Able to manage up to 10,000 records concurrently.

## 6. Design constraints

- Use React JS (JS framework) for frontend.
- MySQL or Other RDBMS for DB Management.

## 7. Non-Functional Requirements

- Response Time - System should have a maximum response time of 10ms.
- Scalability - It should support future expansion.

## 8. Preliminary Schedule and Budget

- Estimated Duration: 6 months for development and 4 months for testing.
- Project cost: Around \$7500 for approximately 10,000 lines of code.

### Split up

- |   |                      |         |
|---|----------------------|---------|
| 1 | * Requirements Stage | \$ 500  |
|   | * Development Stage  | \$ 2500 |
|   | * Design Stage       | \$ 3000 |
|   | * Evolution Stage    | \$ 1500 |

# Credit Card Processing System

## 1. Introduction

### 1.1 Purpose of the document

To outline the functional and non-functional requirements for a credit card processing system.

### 1.2 Scope of the document

The system will cater to merchants, customers and financial institutions, providing secure, fast, and reliable payment solutions. It will handle transaction processing, fraud detection and data encryption.

### 1.3 Overview

This system is designed to facilitate secure & efficient processing of credit card transactions for various businesses.

## 2. Functional Description

Transaction authorization : validate credit card information & authorize.

Transaction Capture : Process and record completed details, proof of origin.

Fraud & reporting : Generate reports and transaction data, including sales & refunds.

### 3. Functional Requirements :-

- System Shall allow merchant to input and process credit card information.
- Authorize transaction - verifies details.
- Should issue refunds for incomplete transactions.
- Fraud detection - identifying suspicious transactions.

### 4. Interface Requirements :-

- API integration - Integrating system with e-commerce platforms and payment gateway.

### 5. Design Constraints :-

- Should handle multiple transactions.
- Supporting multiple currencies and international payments.

### 6. Performance Requirements :-

- Processes transactions in less than two seconds.

### 7. Non-Functional Requirements :-

- High reliability with 99.99% uptime.
- Robust encryption standards for data protection.

## 8. Preliminary Schedule and Budget

Timeline - 4 months for development,  
Budget - ₹ 70,000 for dev and testing  
Split up → Stage 1 - ₹ 10,000, Stage 2 - ₹ 25,000, Stage 3 - ₹ 30,000, Stage 4 - ₹ 15,000  
Library Management System

### SRS Document

#### 1. Introduction

1.1 Purpose :- To automate library operations like cataloging, inventory management, and user tracking.

1.2 Scope :- Covers borrowing, return and fine collection, reducing manual work.

1.3 Overview :- Manages book inventory, and manages user data for streamlined library service.

#### 2. General Description

The system manages book inventories, lending processes and user data. Users include librarian, student, and faculty.

#### 3. Functional Requirements

- Borrowing and returning books
- Catalog search for users
- Overdue fine calculation and notification.
-

#### 4. Interface Requirements

- Web interface for login and book search
- Integration with student databases for user verification

#### 5. Performance Requirements

- Processes book loans within 3 seconds
- Manage upto 1000 concurrent users.

#### 6. Design Constraints

- Limited to the library's existing hardware infrastructure.
- Use of relational database to store book and user information.

#### 7. Non-Functional Requirements

- High reliability for uninterrupted library services.
- User-friendly interface for ease of use by non-technical staff.

#### 8. Preliminary Schedule

- Timeline : 5 months for development
- Budget : \$ 50,000 for full implementation.

Split up :-

Stage 1 - \$ 5,000

Stage 2 - \$ 10,000

Stage 3 - \$ 25,000

Stage 4 - \$ 10,000

## IV Stock Maintenance System SR5.1

### 1. Introduction

1.1 Purpose :- To automate stock tracking, reduce maintenance errors and optimize order management.

1.2 Scope :- Manages inventory levels, places orders, alerts and scope.

1.3 Overview :- Provides real-time stock tracking, helping business manage.

### 2. General Description

Tracks stock levels, order status, and generates reports.

Users include store managers and warehouse staff.

### 3. Functional Requirements

→ Real-time stock level update.

→ Automated low stock alert.

→ Order and supplier management.

### 4. Interface Requirements

→ Integration with suppliers for automated order requests.

→ API integration.

## 5. Performance Requirements

- Handle stock updates in real time.
- Concurrently manage up to 20,000 records.
- Support minimum of 500 users.

## 6. Design constraints

- Limited to use on company-provided hardware.
- Uses SQL-based database for stock records.

## 7. Non-functional Requirements

- High reliability and low error rate in inventory tracking.
- Fast and scalable to accommodate large inventory volumes.

## 8. Schedule and Budget

- Timeline: 3 months for Dev. & 1 month for testing.
- Budget: \$60,000 for dev.

Split-up:

Stage 1 - \$10,000

Stage 2 - \$20,000

Stage 3 - \$20,000

Stage 4 - \$10,000

## IV Passport Automation System :-

### 1.1 Introduction

#### 1.1 Purpose :

This document provides the software requirements for the development of the passport automation system.

#### 1.2 Scope

It will enable citizens to apply for passport online, track the status of their application, and schedule appointments.

#### 1.3 Overview

This system will provide a user-friendly platform for citizens to submit their passport application.

### 2. General Description:

- Supports Online passport application
- Document submission and verification
- Application status tracking.

### 3. Functional Requirements

- System will allow applicants to upload scanned documents
- It will allow the applicants to track the status of their application online.

#### 4. Interface Requirements

- Payment gateway for fee submission
- A web based interface / platform for citizens to submit their passport application

#### 5. Performance Requirements :-

- The system should respond to all user queries
- The system must support more concurrent users at a time.

#### 6. Design constraints :-

- The system must be compatible with major browsers
- The system must comply with data privacy & security regulations.

#### 7. Non-Functional attributes :-

- The system must be accessible on various devices
- Response Time :- Should have a maximum of 5ms response time
- Security :- Should provide robust security for user data

#### 8. Preliminary Schedule and Budget :-

- Development time :- Around 5 months for development and 3 months for testing.

→ Development Cost :- Estimated cost around \$ 50,000

Split - Up :-

1. Stage 1 - Requirement analysis - \$ 15,000
2. Stage 2 - System Design - \$ 15,000
3. Stage 3 - Development & testing - \$ 15,000
4. Stage 4 - Evolution - \$ 10,000

✓

7/10