

1. Hotel Management System

1. Introduction

1.1 Purpose of this Document: The purpose of this document of a Hotel Management System. It will provide a clear understanding of the project objectives, scope, and deliverables.

1.2 Scope of this Document: This document defines the overall working and main objectives of the Hotel Management System. It includes a description of the development cost and time required for the project.

1.3 Overview: The Hotel Management System is a software solution designed to streamline hotel operations, including reservation management, guest check-in/check-out, room assignment, billing, and reporting.

2 General Description: The Hotel Management System will cater to the needs of hotel staff and management, providing features such as room booking, guest profiles, inventory management, and financial reporting. It will be accessible to users with varying levels of technical expertise.

3 Functional Requirements

3.1 Reservation Management:

- Allow users to make room reservations

online or through the front desk.

- Generate reservation confirmations and send notifications to guests

3.2 Room Management:

- Assign rooms to guests based on availability and preferences
- Track room status (clean, occupied, vacant) in real-time.

3.3 Guest Management

- Maintain guest profiles with personal information, preferences, and booking history.
- Facilitate guest check-in and check-out processes.

3.4 Billing and Invoicing:

- Generate accurate bills for room charges, additional services, and taxes
- Accept various payment methods and generate invoices for corporate clients

4. Interface Requirements:

4.1 User Interface:

- Intuitive and user-friendly interface for hotel staff and guests
- Accessible via web browsers, mobile devices, and desktop applications

4.2 Integration Interfaces:

- Integration with payment gateways for secure transactions
- Integration with third-party

booking platforms for seamless reservation management.

5. Performance Requirements:

5.1 Response Time:

- The system should respond to user actions within 2 seconds.

5.2 Scalability:

- Handle a minimum of 1000 concurrent users during peak hours.

5.3 Data Integrity:

- Ensure data consistency and accuracy across all modules.

6. Design Constraints:

6.1 Hardware Limitations:

- The system should be compatible with standard hotel hardware (computers, printers, POS terminals).

6.2 Software Dependencies:

- Utilize a relational database management system (e.g., MySQL) for data storage.
- Use programming languages and frameworks conducive to CML modelling (e.g., Java, Spring Boot).

7. Non-Functional Attributes:

7.1 Security:

- Implement robust authentication and authorization.

Date : 10/10/2023

mechanisms to protect sensitive data

7.2 Reliability :

- Ensure high availability and fault tolerance to minimize system downtime

7.3 Scalability :

- Design the system to accommodate future growth and expansion

7.4 Portability :

- Support multiple platforms and devices for user accessibility

7.5 Usability :

- The system shall have a user-friendly interface with clear navigation

7.6 Reusability :

- The system shall use modular code design to facilitate future enhancements and maintenance

7.7 Compatibility :

- The system shall be compatible with common web browsers (Chrome ; Firefox, Safari).

7.8 Data Integrity :

- The system shall ensure accurate and consistent data storage and retrieval.

8 Primary Schedule and Budget :

The development of HMS is estimated

To take 6 months with a budget of \$100,000. This includes deployment phases

2 Credit Card Processing System

1 Introduction

1.1 Purpose: The purpose of this SRS document is to define the requirements for the credit card processing system. This system will securely handle credit card transactions including authorization, settlement, refunds and fraud detection.

1.2 Scope of this document: The credit card processing system document shall include and define the overall working and operations of the system.

1.3 Overview: The credit card processing system is a software solution that will function as an intermediary between merchants and financial institutions. It will validate card details, process payments securely and provide fraud prevention mechanisms.

2 General description: The credit card processing system will ~~cater~~ cater to the needs of customers, merchants and administrators providing a middleware solution connecting merchants to banks.

3 Functional requirements

3.1 Transaction Management

- Validate credit card details (number, cvv, expiry date).
- Authorize or decline transactions within 2 seconds.

- 3.2 Settlements and Refund management
- Batch settlements with banks at configurable intervals and generate reports of the same
 - Allow full and partial refunds

3.3 Fraud detection and Alerts

- Flag suspicious transactions using fraud detection
- Generate real time alerts for users
- Support integration with external fraud detection service

4 Interface Integration Interfaces

- Integration with bank and payment gateway
- Integration fraud detection APIs

5 Performance requirements:

5.1 Response time: transaction authorization must complete with 2 seconds under normal load 10000 concurrent transactions

5.2 Scalability: The system must support at least 10000 concurrent transactions

5.3 Data Integrity: All financial transactions must be recorded ~~automa~~ atomically to prevent duplication / loss.

6 Design constraint:

6.1 Hardware Limitations: Must operate on cloud hosted infrastructures with auto-scaling and require redundant services for high availability.

6.2 Software Limitations: Must support secure API integration and adequate frameworks which prohibit storage of CVV after transaction.

7 Non functional attributes:

7.1 Security: AES encryption for stored data and tokenization of card numbers.

7.2 Reliability: System must possess automatic failure and disaster recovery.

7.3 Scalability: Support future expansion to handle larger number of transactions.

7.4 Portability: Support multiple platforms and devices.

7.5 Usability: Merchant dashboard must be responsive and user friendly and ensure seamless checkout.

7.6 Reusability: API components must be responsible for different merchants.

7.7 Compatibility: Compatible with major browsers and devices.

7.8 Data Integrity: Transactions must maintain ACID properties in database

8 Preliminary Schedule and Budget:

The development of credit card processing system is estimated to take 20 weeks with a budget of \$180000 which includes project planning, development, testing and deployment phases.

3.

Library Management System

3

1 Introduction :

1.1 Purpose : The purpose of this document is to define the functional and non-functional requirements of the Library management system which will manage book records, user accounts, fine calculation and reporting.

1.2 Scope of this Document : This document defines the overall working and main objectives of the library management system. It includes a description of the development cost and the time required.

1.3 Overview : The library management system will automate the current manual system of book issuing, returning, catalogue management. The system will ensure that data integrity and security to ensure large numbers of users and book records.

4

2 General description :

The library management system will cater to the needs of library members (students, faculty), librarians and administrator providing an interface that replaces manual record-keeping with an automated digital platform.

3 Functional requirements

3.1 Book Catalog management

- Add, update, delete and search book records
- Maintain metadata (ISBN, title, author, category, availability)

3.2 Borrowing and Returning books

- Allow members to borrow available books
- Track due dates and calculate overdues

3.3 User Account Management

- Register new users with unique ID
- Maintain borrowing history for each user
- provide role based access

3.4 Reports and analytics management

- Generate reports on most borrowed books, overdue books and user activity
- Provide monthly/annual summaries for administrators

4 Interface Requirements

4.1 User Interface:

- Members: search, borrow, return, reserve and review books
- Librarians: Manage inventory, issue/return books
- Admin: view analytics, manage users

4.2 Integration Interfaces

- Barcode /QR scanner for physical books
- Integration with e-mail or SMS service for reminders

5 Performance Requirements

5.1 Response Time

- Search queries should return results within 2 seconds.

5.2 Scalability

- System must support 500 concurrent users and must have ability to handle 100000 book records.

5.3 Data Integrity

- Prevent duplicate records through unique identifiers.

6 Design constraints

6.1 Hardware Limitations:

- Requires standard systems/mobile device with internet access.
- Barcode scanners for physical book handling

6.2 Software Limitations:

- Web based system must support multiple browsers
- Must comply with data protection regulators

7 Non-Functional Attributes

7.1 Security:

Encrypted storage of user passwords and role based access control for different users.

7.2 Reliability:

Daily backups of database

7.3 Scalability : Horizontal scaling supported for database and application servers.

7.4 Portability : Compatible with major O's with responsive UI.

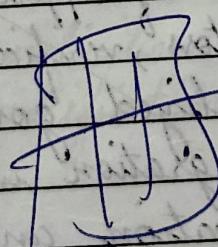
7.5 Usability : Intuitive UI for both members of librarians, multilingual support for accessibility.

7.6 Reusability : Reusable modules for catalog management & user authentication.

7.7 Compatibility : Compatible with standard barcode scanners.

7.8 Data Integrity : All transactions are logged with timestamps and user IDs.

8 Preliminary Budget & Schedule : The development of the system is estimated to take 4 months with a budget of \$90000, including deployment phases.



Requirement i. need 10000
data collection price

4. Passport Authentication System

2. Introduction

1.1 Purpose: The purpose of this document is to design software requirements for a Passport Authentication System. It will provide applicants, administrators and verification authorities with a secure, user friendly platform.

1.2 Scope of this document: This document defines the overall working and main objectives of the Passport Authentication System. It includes a description of the development cost and time required for the project.

1.3 Overview: The Passport Authentication System will centralize all passport-related service, integrating document verification, submission. The system will be accessible nationwide, insuring efficiency and reducing physical visits to passport offices.

2. General description:

The passport authentication system will cater to the needs of applicants, passport officials, administrators, verification offices providing a centralized government portal with optional integration into national ID and police verification and databases.

3. Functional Requirements

3.1 Application Management

- User registration & authentication
- Document upload & fee payment via payment gateway

3.2 Verification & approval

- Integration with police verification system
- Document verification by authorized officials
- Approval or rejection of applications with reasons.

3.3 Tracking & Issuance

- Real-time tracking of application status by applicants
- Notification via e-mail for each stage
- Online grievance redressal & customer support

4 Interface Requirements

4.1 User Interface

- Applicants: Online registration, application forms, payment and tracking
- Officials: Applications, dashboard, verification tools, aper approval panels.
- Admins: Manage users, monitor system logs, handle escalations.

4.2 Integration Interfaces

- Payment gateway for online interface
- National ID database / digilocker integration for ID verification
- Police verification system for background checks

5 Performance Requirements

5.1 Response Time

- Form submission & page loads within 3 secs
- Application status retrieval within 2 seconds

5.2 Scalability

- Support for 500000 concurrent users and handle lakhs of applications annually.

5.3 Data Integrity

- No unauthorized modification of records
- Auto Automated backups and check sum validation for accuracy

6 Design constraints

6.1 Hardware Limitations

- Standard web servers with scalable cloud infrastructures
- Biometric devices required at application centres

6.2 Software Limitations

- Must run on secure web servers and must comply and ~~not~~ must comply with government mandated encrypted standards
- Must integrate with government approved database

7 Non Functional attributes

7.1 Security : Multifactor authentication for officials and encrypted data transmission & storage with role based access control

7.2 Scalability : Cloud based architecture for future expansion

7.3 Reliability : Ensure high availability and fault tolerance to minimize system downtime

7.4 Portability : Accessible on desktops & mobile devices with cross browser compatibility

- 7.5 Usability: Simple, multilingual interface for applicants with accessibility compliance.
- 7.6 Reusability: Modular services like payment, verification, notification reusable in other e-governance projects.
- 7.7 Compatibility: Compatible with biometric scanners, IP verification APIs and e-payment systems.
- 7.8 Data Integrity: Full audit logs of every transactions and immutable record storage for legal compliance.

8 Preliminary Schedule and Budget

The development of Passport Authentication System is estimated to take 8 months at a budget of \$530,000 along with deployment phases.

5

Stock Maintenance System

1 Introduction

1.1 Purpose: The purpose of this document is to specify the functional and non-functional attributes of Stock Maintenance System. It aims to provide and replace manual methods of stock management with an automated, reliable and scalable digital solution.

1.2 Scope of this document: This document defines the overall working and main objectives of Stock Maintenance System. It includes a description of the development cost and time required for this project.

1.3 Overview: The Stock Management System will act as a centralized database of stock items. The system will enhance efficiency, reduce errors and provide real-time visibility of stock trends to ~~stock users~~ stakeholders.

2 General Description

The Stock Maintenance System will cater to the needs of auditors, administrators, etc providing a ~~standalone~~ application with optional integration into ERP systems where each stock item ~~must~~ must each have a unique identifier.

3 Functional Requirements

3.1 Stock Management

- Maintain stock item details - id, name, category by supplier, price and quantity

3.2 Stock Transitions

- Record stock Inflows and Outflows
- Generate alerts for low-stock and re-order levels

3.3 Reporting & Analysis

- Generate reports on current stock levels, stock movement and historical trends
- Provide graphical - candle bar - dashboards for managers and administrators

4 Interface Requirements

4.1 User Interface

- Managers : view dashboard, update stock levels, generate reports.
- Sales staff : Record stock usage and sales.
- Admin : Manage user accounts, audit logins and system settings

4.2 Integration Interfaces

- Barcode / QR code scanners for quick stock entry
- Database integrator for backend storage

5 Performance Requirements

5.1 Response time:

- Stock updates must reflect within 2 seconds
- Search queries must return results in 3 seconds

5.2 Scalability

- Initial support of 500+ concurrent users and a limit of 100000+ stock records with live updates

5.3 Data Integrity : All transactions must maintain

ACID properties

6. Design constraints

6.1 Hardware Limitations

- Runs on standard desktop or mobile devices
- Barcode scanners or module scanning devices required for just transactions and operations.

6.2 Software limitations

- Must support major browsers and backends related to relational databases
- Must comply with business data retention and audit policies

7 Non Functional Attributes

7.1 Security

- Role based authentication - admin, staff manager
- Passwords stored with encryption

7.2 Reliability

- Daily backups of database

7.3 Scalability

- System should allow horizontal scaling for large businesses and be accommodate future growth and expansion

7.4 Portability

- Accessible via Windows, Linux, macOS and possess mobile responsive design

7.5 Usability

- Simple and intuitive dashboard with minimal training required
- Multilingual support for global business

7.6 Reusability

- Reusable APIs for integration into ERP system

7.7 Compatibility

- Compatible with common web browsers

7.8 Data Integrity

- Maintain full transaction logs with timestamps and user IDs
- Automatic validation of supplier and sales records

8 Preliminary schedule and Budget

The development of the stock maintenance system is estimated to take 4.5 months with a budget of \$90000. This includes project planning, development and testing and deployment phase.

~~Project name +
development~~

~~Timeline
Week 1: Planning
Week 2: Requirements
Week 3: Design
Week 4: Development~~

~~Completion date:~~