SRS Document

Hotel Management System

Problem Statement

Hotels face inefficiencies in manual operations such as reservations, billing, and check in/out. Errors in bookings and delays affect customer satisfaction and staff productivity. A Hotel Management System (HMS) is needed to automate these processes, improve accuracy and provide real-time access to hotel operations.

1 Introduction

Purpose To automate hotel operations: reservations, room management, billing and reporting while improving efficiency and guest satisfaction

Scope The HMS will support online booking, check in/out, billing, staff tools and manager dashboards. It should be scalable for small to large hotels

Overview The system will include a customer portal, staff portal and admin dashboard

2 General Description

System Perspective Centralised client-server system with secure databases.

System functions Room booking, availability tracking, guest check in/out and billing

Users Guests, hotel staff and managers

Constraints 24/7 uptime, secure transactions, multi-platform access

Assumptions Reliable internet and trained hotel staff.

3 Specific Requirements

Functional

Allow online and in-person reservations

Update room availability in real time

Generate bills and accept payments

Generate reports.

Send booking and payment notifications

Performance

Support 1000+ concurrent users, booking response < 3 seconds

Design Constraints

Role-based access, SQL database, scalable to multi-hotel use

Non-Functional

Secure, reliable, user-friendly and maintainable

4 Schedule and Budget

Schedule: 6 months

Requirements and design (1), Core modules (2), housekeeping (1), testing (1), deployment and training (1)

Budget = $50000

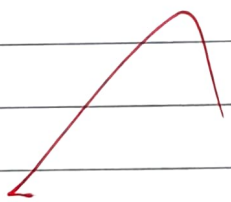
Requirement and Design: $5000

Development: $27000

Testing and QA: $5000

Training: $5000

Maintainence: $8000.



Credit Card Processing System

Problem Statement

Manual credit card handling increases fraud risks and delays processing. A digital system ensures fast, secure and reliable transactions for customers and banks.

1. Introduction

Purpose Automate credit card transactions, fraud detection and billing

Scope Handles secure payments, transaction logs and user authentication

Overview Real time payment authorisation, fraud detection module, transaction history

2. General Description

Users Customers, banks and merchants

Functions Transaction validation, billing and fraud alerts

Constraints Must comply with regulations

3. Specific Requirements

Functional Requirements

Authenticate users, process payments, generate transaction history and detect fraud

Interface Requirements

UI for mobile and web portal. System bank APIs, payment gateways

Performance Requirements

Process <2 seconds per transaction, 99.9% uptime

Non-functional Requirements

Reliable, scalable and secure

4. Schedule

Weeks 1-2: Requirements

Weeks 3-6: Development

Weeks 7-8: Testing

Week 9: Deployment.

Budget

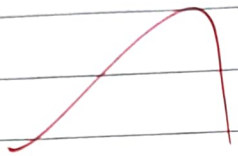
Development: 45%.

Security: 20%.

Testing: 15%.

Deployment: 10%.

Maintainance: 10%.



Library Management System

Problem Statement

Manual library operations cause errors in book tracking and delays in service. A centralised digital system improves efficiency and user experience.

1 Introduction

Purpose Digitize library operations like cataloging, lending and returns
Scope Supports staff and students for book search, issue/return and fines
Overview Online cataloging, member management, fine tracking

2 General Description

Users Students, librarians and admins
Functions Search, issue/return, fines
Constraints Internet access required

3 Specific Requirements

Functional

Manage members, search and catalog books, issue/return books, track due dates and fines and generate reports

Interface

Web and mobile UI. Barcode and RFID Integration system

Performance

Handle 500+ users/day. Search in <2 seconds

Design

Secure data handling, low maintainence cost

Non-Functional

Usable, scalable and reliable

4 Schedule

Weeks 1-2: Requirements

Weeks 3-5: Development

Weeks 6-7: Testing

Week 8: Deployment

5 Budget

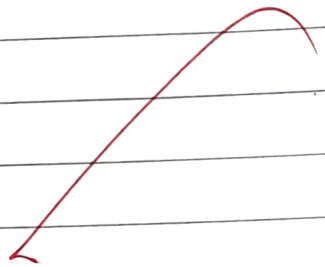
Development: 40%.

Testing: 15%.

Deployment: 15%.

Training: 10%.

Maintainance: 20%.



Stock Maintenance System

Problem Statement

Manual stock records often lead to shortages, overstocking or losses. A stock maintainance system provides accuracy and reduces operational risk.

1. Introduction

Purpose Track inventory levels, sales and stock replenishment

Scope Supports retailers and warhouses for efficient stock control

Overview Stock database alerts for low stock, sales reporting

2 General Description

Users Admin, staff, managers

Functions Stock update, reporting, alerts

Constraints Secure login required

3 Specific Requirements

Functional

Record stock entries, monitor sales, trigger low stock alerts, generate reports

Interface

Web UI dashboard, System POS integration

Performance

Handle 1000+ transactions/day. Generate reports <5s

Non-Functional

Reliable, scalable and high performance

Schedule

Weeks 1-2: Requirements

Weeks 3-6: Development

Weeks 7-8: Testing

Week 9 - Deployment

Budget

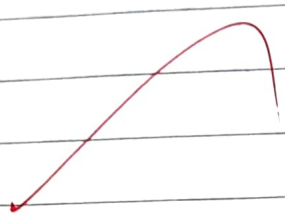
Development : 45%.

Testing : 15%.

Infrastructure : 20%.

Deployment : 10%.

Maintenance : 10%.



Passport Automation System

Problem System

Manual passport processing causes delays, paperwork errors and inefficiency. Automation ensures faster processing and transparency for applicants

1 Introduction

Purpose Automate passport application, verification and tracking

Scope Citizens can apply online, track status and recieve updates.

Overview Online application, document verification, application tracking.

2 General Description

Users Citizens, Passport Officers, Admin

Functions Apply, verify, approve and track

Constraints Must integrate with national ID systems

3 Specific Requirements

Functional

Submit applications, upload documents, verify identity and notify applicants

Interface

Web and mobile UI, government databases

Performance

Handle 10000+ applications/day, and respond <3 seconds

Design

Government security standards, multilingual support

Non-functional

Reliable, secure and accesible

Schedule

Weeks 1-3: Requirements and Design

Weeks 4-7: Development

Weeks 8-9: Testing

Week 10: Deployment

Budget

Development: 40%

Security and Compliance: 20%

Testing: 15%

Deployment: 15%

Maintenance: 10%