Week 1
19.08.2025

1 Hotel Management System. Develop to

Develop a problem statement. b

Software to ~~a~~ give solution to problems like fur- slow check-ins, poor communication between departments, inefficient room assignments and booking errors.

Develop a complete IEEE standard SRS document with several requirements.

Introduction

Purpose : Hotel management System is a software developed with the main purpose to overcome our problem statement. In this software their won't be communication gap between departments in the hotel. Similarly, during a ~~ek~~ check-in ~~the~~process the process will be smooth and no ~~ba~~ disturbances. The goal is to develop user-friendly and efficient system.

Scope : The hotel management system will manage the core of the hotel functioning.

Front office → Reservations, check~~s~~-ins /out, and guest services

Housekeeping → Room status and cleaning schedules

Management → Reports and analytics.

Overview : The system is built for performance, security and ease of use. It is built to replace manual processes and improve overall efficiency for hotel staffs and managers.

General description :
It is a software solution used by hotels to manage all of their core operations. It is a central hub for tasks like guest check-in / out, room assignments and communication between departments within the management. The system automates routine tasks, improve staff efficiency and provide real-time data to managers to run the buisness more ~~effes~~ effectively.

Functional requirements :
1. Reservation management
2. Check-in / Check-out
3. Billing and payments
4. Room Status management
5. Reports

nterface requirements :
1 User Interface (UI)
2. API
3. Hardware - Computers, tablets, barcode scanner
4. Software - Chrome, Microsoft edge.

formance requirements :
1. Response time
2. Availability
3. Throughput
4 Scalability

ign Constraints : built on a
1. System must be ~ secure & scalable web architecture.
2. System must be developed using open source technologies.

N
1 Security
2 Usability
3 Scalability
4 Reliability
5 Performance

Preliminary schedule and Budget :
Schedule : 5 - 10 months
Budget : Small team of 2-3 developers and a project manager = $150,000 (13 lakh).

↓

Table.

2. Credit Card Processing

Problem Statement - To develop a secure, fast, reliable credit card system that ensures real-time transactions and compliance. ~~[illegible]~~

Purpose - The purpose of the document is to define the requirements of credit card processing. It serves as a reference for developers, testers to build a secure system. It ensures all stakeholders have a clear understanding of the system's capabilities and objectives.

Scope - The credit card system is going to manage the following functions. Process credit card payments; support refunds, partial refunds, charge-backs; detect and prevent fraudulent transactions

Overview - The credit card processing system will operate as a middle-man between merchants and financial institutions. It will capture credit card data, encrypt it, communicate with payment gateways for authorization & settlement.

General description - Users: merchants, customers, banks, admins

Environment - web & mobile apps, secure APIs.

Constraints - Must comply with PCI DSS, GDRPR.

Functional requirements -

1. Transaction processing
2. Security
3. Fraud detection
4. Refunds and Chargebacks

Interface requirements -

1. User interface
2. Hardware interface
3. Software interface
4. Communication interface.

Performance requirements -

1. Supports upto 100000 transactions per second.
2. Ensure 99.99.1.
3. Process each transaction in < 3 seconds.

Design requirements -

1. System built on a secure & scalable web architecture.
2. System must comply with institutional IT policies for data security.

Non-functional requirements -

1. Reliability
2. Usability
3. Security
4. Scalability
5. Maintainability.

Preliminary Schedule -

Phase	Duration
Requirement analysis	2 weeks
System design	2 weeks
Development - core modules	4 weeks
Development - Additional modules	2 weeks
Integration	2 weeks
Testing	2 weeks
Deployment & Training	2 weeks

Total duration ≃16 weeks.

Budget -

Category	Estimated cost
Development team	$ 45,000
Hardware & Infra-structure	$ 10,000
Software licensing & APIs	$ 5,000
Security & Compliance	$ 7,000
Training & maintenance	$ 7,000
Total ≃	$ 74,000

3. Library management - system

Problem statement - To develop a automate book cataloguing, borrowing, returning, and record management to reduce manual errors & improve efficiency.

Introduction

Purpose - To define functional & non-functional requirements for the library management system. This helps administrators, librarians & students.

Scope - The system will maintain a digital catalog of all books and resources. Support members registration & authentication. Enable book search, issue & return. Automatically calculate & manage fines. Be accessible via web & mobile interfaces.

Overview - The system consists of admin module, user module, database, reports module, member activity.

General description - Users: Librarians, Students or members

Environment - Web-based & mobile-based interface.

Constraints - must work with limited internet bandwidth.

Assumptions - Users having valid membership.

Functional requirements -

1. User authentication
2. Book catalog management
3. Book issue / return
4. Search & browse.
5. Fine management
6. Notifications.

Interface requirements -

1. User interface
2. API interface
3. External interface - barcode scanner, student database.

Performance requirements -

1. Support 1000+ users
2. Book search results displayed in <1 sec.
3. 99% uptime requirement
4. Issue / return transactions.

Design requirements -

1. Must comply with privacy regulations.
2. Secure authentication.
3. Should run on standard platforms.

Non-functional requirements -

1. Security
2. User credentials
3. Reliability
4. Usability
5. Scalability

Preliminary schedule -

Phase	Duration
Requirement analysis	2 weeks
System design	2 weeks
Development	6 weeks
Testing	3 weeks
Deployment	2 weeks

Total duration = 15 weeks

Preliminary budget -

category	Estimation
Development team	$60.000
cloud infrastructure	$8.000
Database licensing	$5.000
Testing	$7.000
Total	$80.000

1. Stock Management System -

Problem statement - To develop an automate stock trading, monitor availability and generate alters.

Introduction

Purpose - To digitally manage stock items, ensuring real-time updates on availablility, incoming and outgoing stock and automated notifications for restocking.

Scope - Maintain a digital inventory of items, track incoming and outgoing stock. Generate alerts for low stock levels. Provide role-based access

Overview - It includes components like stock database, transaction module, altert system.

Stock database - stores item details, quantities

Transaction module - records incoming / outgoing stock

Alert system - triggers notifications for critical stock levels.

General description - It will be a centralized, web-based application accessible to different user roots. It will maintain a real-time database of all stock items, their quantities, transaction history

Functional requirements -

1. Stock management
2. Sales transaction
3. Supplier order
4. Stock alerts
5. Reporting.

Interface requirements -

1. User interface
2. Hardware interface
3. Software interface

Performance requirements -

1. Response time
2. Scalability
3. Concurrency

Design ~~se~~constraints -

1. Technology stack
2. Security

Non - functional requirements -

1. Usability
2. Reliability
3. Maintainability
4. Security.

Preliminary Schedule -

Phase	Duration
Requirement analysis	2 weeks
System design	2 weeks
Development	6 weeks
Testing	3 weeks
Deployment	2 weeks

Total duration ≃ 15 weeks

Preliminary budget -

Category	Estimation
Development team	$60,000
Cloud infrastructure	$8,000
Database licensing	$5,000
Testing	$7,000
Total ≃	$80,000

5. Passport Automation System

Problem Statement - To develop an automated system is required to digitize application submission, verification, approval & passport issue.

Introduction -

Purpose - To define requirements for a passport automation system that streamlines passport appl. verification, status, tracking & issurance

Scope - Allow applicants to apply online & upload documents.
Support documents verification & approval by officials

verview - Applicant module, admin/official module, database, notification module.

eneral description - Users - Applicants, passports officials, admins.
Environment - government web portal, secure servers.

unctional requirements -
User registration
Online application
Doument verification
Appointment booking
Status tracking
Passport inssuance.

Interface requirements -
1. UI responsive web portal
2. API interface
3. External interface.

Performance requirements -
1. Handle 10.000+ applications/day.
2. Application response time < 3 sec.

Design constraints -
1. must comply with e-governance, GDPB.
2. Strong authentication.

Non-functional requirements -
1. Security
2. Reliability
3. Scalability
4. Maintainability.

Preliminary Schedule -

Phase	Duration
Requirement	3 weeks
Design	3 weeks
Development	10 weeks
Testing	5 weeks
Deployment	3 weeks
Total	≃ 24 weeks.

Budget

Category	Estimation
Development team	$150,000
Secure infra.	$40,000
Testing	$20,000
maintenance	$30,000

Total = $2,40,000