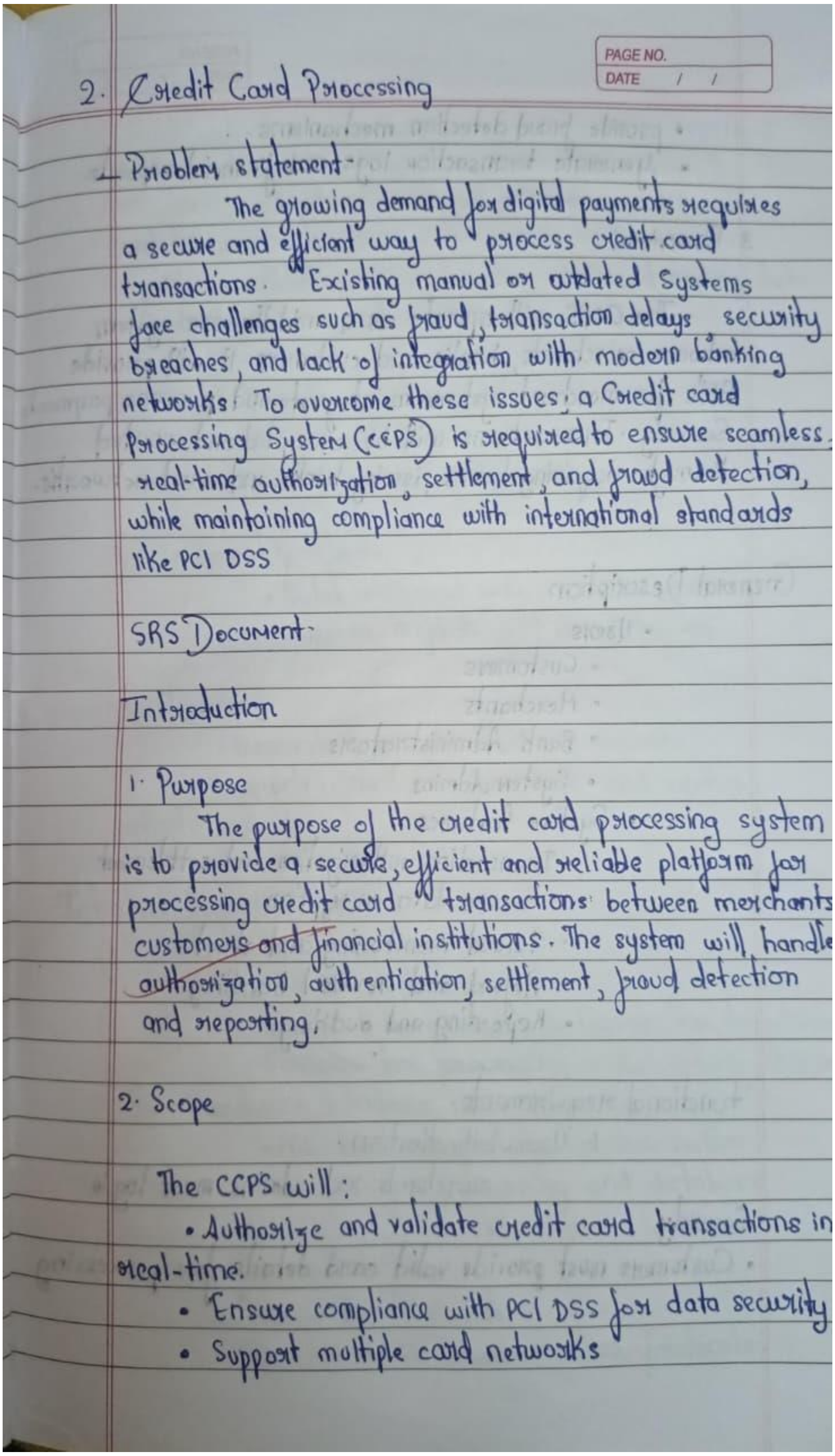
OOM LAB
SRS Document

Credit Card Processing



PAGE NO.
DATE / /

2. Credit Card Processing

Problem statement -

The growing demand for digital payments requires a secure and efficient way to process credit card transactions. Existing manual or outdated Systems face challenges such as fraud, transaction delays, security breaches, and lack of integration with modern banking networks. To overcome these issues, a Credit card Processing System (CCPS) is required to ensure seamless, real-time authorization, settlement, and fraud detection, while maintaining compliance with international standards like PCI DSS

SRS Document -

Introduction

1. Purpose

The purpose of the credit card processing system is to provide a secure, efficient and reliable platform for processing credit card transactions between merchants customers and financial institutions. The system will handle authorization, authentication, settlement, fraud detection and reporting.

2. Scope

The CCPS will :

- Authorize and validate credit card transactions in real-time.
- Ensure compliance with PCI DSS for data security
- Support multiple card networks

- provide fraud detection mechanisms.
- Generate transaction logs and financial reports.

3. Overview:-

The CCPS will operate as a middleware system between merchants, banks and customers. It will provide API's for merchant systems to integrate and process payments Securely. Transactions will be encrypted and routed through acquiring banks, issuing banks and card networks.

General Description

- Users
  - Customers
  - Merchants
  - Bank Administrators
  - Systen Admins
- System Features:
  - Transaction authorization and settlement
  - Secure data encryption
  - Fraud monitoring and alerts
  - Refund and reversal handling
  - Reporting and auditing

Functional requirements.

1. User Authentication
   - merchants and admins must log in Securely

- Customers must provide valid card details for processing

2. Transaction processing

- System must authorize transactions in real-time.
- perform validation of card number, CVV and expiry date
- Route transactions to appropriate banks.

3. Fraud Detection

- Flag suspicious activities
- Implement two-factor authentication

4. Settlement & refunds

- Batch settlement with acquiring banks.
- Support refunds and chargebacks

5. Reporting

- Generate transaction history reports
- provide fraud detection logs and system activity reports.

Interface Requirements.

- User interface
  - merchant dashboard for transaction trackin
  - Servors for processing and database stor
- Hardware Interfaces:
  - POS terminals for physical transaction.
  - Servers for processing and database storage.
- Software Interface.
  - Banking APIs
  - Secure communication protocols.

5. Performance Requirement
- must process atleast 2000 transactions per sec
- Transaction authorization must complete in < 2 seconds
- System uptime must be 99.99%.
- fraud detection latency should be < 500ms

6. Design Constraints.
- must Apply with PCI DSS for cardholder data
- Use strong encryption
- Must support multiple card networks
- Deployable on cloud and on premise infrastructure.

7. Non functional requirements.
- Reliability, security, scalability, maintainability, usability, portability

8. Preliminary Schedule and Budget

Schedule

1. Requirement Gathering - 4 weeks
2. System Design - 5 weeks
3. Deployment - 16 weeks
4. Integration with Banks - 4 weeks
5. Testing - 6 weeks
6. Deployment - 3 weeks
7. Maintanance & support - Ongoing

Budget

1. Software Development - $100,000
2. Security & Compliance - $40,000

3. Infrastructure & Hardware - $60,000
4. Testing & certification : $30,000
5. Maintenance (Yearly) : $25000
Total estimated cost : $255,000