27/8/25

classmate
Date
Page

Credit Card Processing System

Problem Statement -
Design and implement a secure and efficient Credit Card Processing System that enables merchants to authenticate, authorize and process credit card transactions in real-time. The system must support fraud detection, ensure compliance with financial security standards, and provide provide interfaces for users, administrators, and banking networks.

SRS Document -

1. Introduction

1.1 Purpose of the Document.
This document provides a detailed software requirements specification for the credit card processing system. It is intended to define the functional and non-functional requirements of the system for developers, testers and stakeholders.

1.2 Scope of the Document
The Credit Card Processing System will authorize, process and manage credit card transactions between merchants and customers. It will ensure secure communication, fraud detection, transaction logging and real-time processing and ensure compliance with security standards.

1.3 Overview
The System will allow processing of payments using credit card,

support transaction tracking and reporting, ensure high availability and secure communication, integrate seamlessly with existing merchant systems.

2. General Description

The system will handle credit card payment authorization, capture, settlement and reporting. Users include merchants, administrators and financial institutions. The ~~web~~ system will use encryption and decryption to protect cardholder data and will operate in real-time to ensure quick response times.

3. Functional Requirements

FR1: User Authentication: Merchants must log in using secure credentials or API keys. Multi-factor ~~auth~~ authentication will be supported.

FR2. Transaction Authorization: The system must validate card details with the issuing bank. Declined transactions must return clear error codes.

FR 3. Fraud Detection: Monitor transactions for suspicious activity (blacklist, velocity checks etc). Support integration with external fraud detection services.

FR 4. Transaction Settlement: Batch settlements at configurable intervals. Provide confirmation reports for processed batches.

FR 5. Refund and Chargeback Handling: Support partial or

full refunds. Track chargebacks and disputes with proper logs.

4. Interface Requirements

4.1 User Interface : Secure web and mobile interfaces for merchants and customers with MFA and clear payment forms.

2. Hardware Interface : POS terminals with EMV/NFC support, secure servers, and network devices.

3. Software Interfaces : Browser-based client, Linux/Windows servers, payment gateway APIs ~~etc~~ and PCI DSS-compliant middleware.

5. Performance Requirements

5.1 Response Time : Authorisation must complete within 2 seconds on Average.

5.2 Scalability : The system must handle up to 50,000 concurrent transactions.

5.3 Data Integrity : All transaction logs must be tamper-proof and consist.

6 Design Constraints.

- Must be deployable on standard cloud servers with automatic scaling.
- Compatible with major payment gateways.

7. Non-Functional Requirements

7.1 Security: End to end encryption for cardholder data.

7.2 Reliability: System uptime of 99.99%

7.3 Portability: Deployable across major cloud platforms.

7.4 Reusability: Core payment modules must be reusable for other payment modules.

8. Preliminary Schedule and Budget

8.1 Schedule

Phase	Timeline
Requirements Analysis	2 weeks
UI/UX	3 weeks
Development	10 weeks
Testing	4 weeks

8.2 Budget

estimated → $42,000