**Problem Statement:**

The goal is to design and develop a credit card processing system that meets the needs of all stakeholders, including customers, merchants, acquiring banks, issuing banks, payment processors, card networks, and regulatory bodies. The system should be easy to use, scalable, and secure, enabling fast and reliable transactions for all users.

For customers, the system should offer a seamless experience, enabling them to make purchases quickly and securely. Merchants should be able to manage transactions and track sales data easily, while acquiring banks should ensure that transactions are processed accurately, authorizing payments and transferring funds between merchants and issuing banks.

Issuing banks should be able to manage their customers' accounts efficiently, authorizing payments and preventing fraudulent activities. Payment processors should provide the technology and infrastructure to facilitate communication between acquiring banks and issuing banks, manage transaction routing, and ensure that payments are authorized and settled quickly and securely.

Card networks should operate the payment networks, ensuring that transactions are processed seamlessly and securely between all stakeholders. Regulatory bodies should oversee the credit card processing industry, ensuring compliance with regulations and protecting consumers from fraud and abuse.

**1. Introduction:**

* 1. Purpose of this Document: At first, main aim of why this document is necessary and what’s purpose of document is explained and described
  2. Scope of this document:

The credit card processing is usually done with a help of the swiper which scans all the details of the card. After each purchase, the details are recorded and document is made by the credit card issuer.

Accepting credit cards is an integral part of business today for merchants who want to be

competitive in their market and grow their business to its greatest potential. Software and Gateway processing helps reduce fraud losses, saves you time and money, and

provides powerful features and performance, including detailed transaction records and reports.Swiping credit cards ensures lower rates, resulting in potential savings of hundreds of dollars each month for your business.

* 1. Overview

Credit card processing through offline involves the merchant collecting order information (, storing this in a database on your site, and entering it using their on-site merchant credit card processing system. Takes time to manually enter credit card information for each order. This solution creates following cons: · Insecure – there is a possibility that a skilled hacker could break into the database and steal an entire list of credit card numbers, thereby damaging the merchant’s reputation with current client. · There is a higher risk of customer charge backs with no signature · Higher risk of fraud for using stolen credit cards · Many discerning online shoppers will not give their credit card to an “untrusted” online merchant

**2 General description:**

A credit card processing system is a software-based platform that facilitates the electronic authorization, verification, and settlement of credit card transactions. It enables merchants to process credit card payments securely and efficiently from customers using their credit or debit cards.

The system typically includes hardware and software components that work together to facilitate transactions. The hardware may include a point-of-sale (POS) terminal, card reader, or mobile device, while the software may include a payment gateway, merchant account, and fraud detection and prevention tools.

The credit card processing system works by securely transmitting customer payment information to the card issuer or the card network, which then approves or declines the transaction based on factors such as available credit, fraud risk, and other security checks. Once approved, the payment is settled and funds are transferred from the customer's account to the merchant's account.

Credit card processing systems are essential for businesses of all sizes, enabling them to accept credit card payments and expand their customer base. They provide a secure and efficient way to process transactions, reduce the risk of fraud and chargebacks, and simplify accounting and reconciliation processes.

**3 Functional Requirements:**

Payment processing: The system should allow merchants to accept credit and debit card payments from customers. This includes authorizing transactions, verifying cardholder information, and settling payments.

Payment gateway integration: The system should integrate with a payment gateway that securely transmits payment information between the merchant and the card issuer.

Fraud detection and prevention: The system should have measures in place to detect and prevent fraudulent transactions, including address verification, card verification, and real-time fraud screening.

Account management: The system should allow merchants to manage their accounts, view transaction histories, and generate reports on payment activity.

Cardholder data security: The system should be compliant with the Payment Card Industry Data Security Standards (PCI DSS) and other security protocols to protect cardholder data and prevent data breaches.

Refunds and chargebacks: The system should allow merchants to issue refunds and handle chargebacks in a timely and efficient manner.

Mobile payments: The system should support mobile payments, enabling customers to pay using their mobile devices.

Multi-currency support: The system should support transactions in multiple currencies, allowing merchants to sell to customers from different countries.

Settlement and reconciliation: The system should facilitate the settlement of payments and provide tools for reconciling transactions and managing accounting processes.

Customer service and support: The system should provide customer service and support to merchants, including technical assistance and help with billing and account management issues.

**4 Interface Requirements:**

User Interface: The system should have a user-friendly interface that makes it easy for merchants to access and manage their accounts, view transaction history, and generate reports.

API: The system should provide an API (Application Programming Interface) that allows merchants to integrate the payment processing functionality into their own software applications.

Payment Gateway Interface: The system should integrate with a payment gateway that securely transmits payment information between the merchant and the card issuer.

POS Integration: The system should integrate with point-of-sale (POS) systems, allowing merchants to process payments using their existing hardware and software.

Mobile Integration: The system should support mobile payments, enabling customers to pay using their mobile devices.

Multi-currency Support: The system should support transactions in multiple currencies and provide currency conversion functionality.

Reporting Interface: The system should provide a reporting interface that allows merchants to generate reports on transaction activity, settlement, and reconciliation.

Security Interface: The system should provide a security interface that enables merchants to configure security settings and monitor security events.

Third-Party Integration: The system should allow for integration with third-party services, such as accounting software or inventory management systems.

Help and Support Interface: The system should provide a help and support interface that allows merchants to access documentation, FAQs, and technical support.

**5 Performance Requirements:**

Response time: The system should respond quickly to user requests and actions, with minimal latency and delay.

Scalability: The system should be scalable, capable of handling a large number of users and transactions, without compromising performance or reliability.

Availability: The system should be available 24/7, with minimal downtime for maintenance or upgrades.

Reliability: The system should be reliable, with minimal errors or failures, and capable of recovering quickly from any disruptions.

Security: The system should be secure, protecting user data and transactions from unauthorized access or breaches.

Load testing: The system should undergo regular load testing, to ensure that it can handle peak loads and heavy traffic without compromising performance or availability.

Optimization: The system should be optimized for performance, with efficient algorithms, data structures, and processing techniques, to minimize resource usage and improve response time.

**6 Design Constraints:**

Hardware constraints: The system design may be constrained by the hardware resources available, such as servers, storage devices, and network equipment.

Software constraints: The system design may be constrained by the software resources available, such as the operating system, database management system, and programming languages.

Time constraints: The system design may be constrained by the project timeline, which may limit the scope, features, and functionality of the system.

Budget constraints: The system design may be constrained by the available budget, which may limit the investment in hardware, software, and personnel.

Regulatory constraints: The system design may be constrained by regulatory requirements, such as data privacy laws, security standards, and compliance regulations.

Compatibility constraints: The system design may be constrained by the need to integrate with existing systems, such as payment gateways, reservation systems, and loyalty programs.

Usability constraints: The system design may be constrained by the need to ensure usability and accessibility, such as designing for users with disabilities, elderly users, or users with limited technical skills.

**7 Non-Functional Attributes:**

Usability: The system should be user-friendly, easy to learn, and intuitive, with a well-designed user interface that enables users to perform tasks quickly and efficiently.

Reliability: The system should be reliable, with a low error rate and minimal downtime, ensuring that users can access and use the system at all times.

Security: The system should be secure, protecting user data and transactions from unauthorized access or breaches, and complying with regulatory requirements.

Performance: The system should be fast and responsive, with minimal latency and delay, enabling users to perform tasks quickly and efficiently.

Scalability: The system should be scalable, able to handle a large number of users and transactions, without compromising performance or reliability.

Maintainability: The system should be easy to maintain, with well-documented code, clear error messages, and easy-to-use debugging tools, enabling developers to identify and fix issues quickly.