Software Requirement Specification Document Of a Financial System Offering Virtual Cards To Customers

Lando Maheva
Simbock,Yaounde
30th/10/25

Introduction

The main aim of this system is to provide customers with a secure, efficient, and controllable digital payment tool for modern, primarily online and contactless transactions. Its scope includes enabling instant, on-demand issuance of unique, temporary card numbers for safer online purchases and subscription management for individuals, as well as providing businesses with granular spending controls, streamlined expense reporting, and automated accounts payable processes for enhanced financial oversight. The system also extends its utility by allowing these digital credentials to be integrated into mobile wallets for secure, in-store contactless payments, eliminating the need for a physical card.

General description

A financial system offers customers virtual cards by leveraging its technological infrastructure to generate unique, digital-only card numbers, expiration dates, and security codes that are linked to the user's primary bank account or a pre-funded balance. This provides users with enhanced security for online or app-based transactions, as their actual financial details are masked from merchants, significantly reducing the risk of fraud and data breaches. A key feature is the ability for a user to instantly generate single-use cards for one-off purchases or set up multi-use cards with specific spending limits, expiration dates, or merchant restrictions, giving them greater control over their expenses. This benefits both individual users, who gain a secure and convenient way to manage subscriptions and online shopping, and businesses, who can use them to streamline accounts payable, track employee spending, and automate expense management. The importance lies in creating a more secure and efficient digital payment experience, fostering greater financial control for customers, and building a stronger sense of trust and engagement within the user community.

Functional Requirements

1. User authentication (login feature)

User authentication is the process of verifying a user's identity to grant them secure access to the application.
Core functionality:

- **Registration:** A new user provides personal information (name, email, etc.) to create an account. This typically involves verifying their identity through email or phone.
- **Login:** Returning users securely access their accounts using a combination of a username and password. Modern applications often incorporate multi-factor authentication (MFA) to add an extra layer of security.
- **Password management:** This includes features for creating a strong password, resetting it if forgotten, and storing it securely on the system.

2. Card creation

This requirement involves the process by which a user creates a new virtual card. Virtual cards are digitally generated and used for online transactions, providing a secure alternative to physical card

Core functionality:

Request submission: The user initiates a request for a new virtual card within the app.

- **Issuance:** The financial institution's system generates a unique card number, expiration date, and CVV for the new card.
- **Card details:** The user can view and manage the virtual card's details within their profile.
-

3. Payment processing

This is the central feature that enables the application to handle financial transactions. It involves the entire flow of approving, processing, and settling payments.

Core functionality:

- **Transaction initiation:** The user initiates a payment, providing details such as the amount, recipient, and payment method.
- **Authorization:** The system verifies that the user has sufficient funds or credit to complete the transaction. The card issuer approves or denies the transaction.
- **Transaction completion:** Once authorized, the funds are securely transferred to the recipient, and the transaction is recorded

4. Notifications

Notifications keep users informed about their account activity and important security alerts. This is a key part of maintaining user trust and engagement.

Core functionality:

- **Transaction alerts:** The system sends real-time alerts for every transaction, including purchases, transfers, and deposits.
- **Security alerts:** The user is notified of any suspicious activity, such as a failed login attempt or a payment flagged for fraud.
- **Profile updates:** Notifications are sent for changes to a user's profile, such as a password change or a new device login

5. User management (example: profile management)

User management refers to the tools and features that allow users to control their own account information and settings.

Core functionality:Profile editing: Users can update personal details like their name, contact information, and address.

- **Security settings:** Users can manage security preferences, including setting up multi-factor authentication (MFA) or biometric login options.
- **Card management:** Users can manage all their associated cards, including virtual cards, and can report a card as lost or stolen.

6. Card creation (user request for virtual cards from the financial institution)
This is a more detailed version of the earlier card creation point, emphasizing the institutional aspect. It describes the technical process of requesting a virtual card directly from the financial institution.
Core functionality:

- **API integration:** The application's backend must integrate with the financial institution's APIs (Application Programming Interfaces) to submit and process card issuance requests.
- **Secure tokenization:** The process must securely handle sensitive card data, often using tokenization to prevent the actual card number from being exposed.
- **Virtual card delivery:** The virtual card details are securely delivered to the user and displayed in the application

7. Cashing
This requirement likely refers to cashing out funds from the user's account, either to another account or as a withdrawal of funds.
Core functionality:

- **Transfer to bank account:** A user can initiate a transfer to their linked bank account.
- **Instant withdrawal:** The system may offer options for instant cashing out, though this often comes with a fee.
- **Transaction history:** A record of all cash-out transactions is maintained for the user.

8. Fraud detection
Fraud detection is the crucial process of identifying and preventing fraudulent transactions in real-time. This protects both the user and the financial institution from losses.
Core functionality:

- **Machine learning models:** The system uses algorithms to analyze transaction patterns and identify anomalies that might indicate fraud. For example, a sudden large purchase in a new location could trigger an alert.
- **Rule-based systems:** The system is equipped with a set of predefined rules to flag suspicious activities, such as an excessive number of transactions in a short period.
- **Alerting and blocking:** When potential fraud is detected, the system can automatically block the transaction and notify the user and the fraud investigation team

Design Constraints

A financial system offering virtual cards, design constraints are strict regulatory compliance, robust security requirements for data protection and fraud detection and performance limitations for real time transactions . Other constraints involve integrating with existing banking infrastructure, maintaining a seamless user experience across different devices, adhering to budget and timeline limitations, and building a system that is highly reliable and scalable.
Regulatory and legal constraints

- **Data privacy:**
  Comply with data protection regulations like GDPR or CCPA, which dictate how customer data is collected, stored, and processed.
- **Payment card industry standards:**
  Adhere to the PCI DSS(Payment Card Industry Data Security Standard) to ensure the security of cardholder data.
- **Country-specific financial laws:**
  Follow all relevant financial regulations, which can include those for money laundering, sanctions, and consumer protection.

Security and reliability constraints

- **Authentication and authorization:**
  Used jwt (json web tokens) to prevent unauthorized use of cards.

- **Fraud prevention:**

Integrate robust fraud detection mechanisms, such as real-time transaction monitoring and machine learning models.

- **High availability:**

Design for 24/7 availability, as the system must be accessible for transactions at all times. This requires redundant systems and failover mechanisms.

- **Data integrity:**

Ensure the accuracy and immutability of transaction data to maintain trust and auditability.

Non -Functional Requirements

F or a financial system offering virtual cards, non-functional attributes like security (protecting user data and transactions), reliability (ensuring high availability for card use), scalability (handling a growing number of users and transactions), performance (fast response times for card creation and payments), portability (working across various devices and operating systems), and data integrity (preventing corruption and ensuring accuracy) are crucial. These attributes define how the system operates, which is critical for user trust and the system's success.
Security

- **Protecting user data:**
  This is paramount for a financial system. The system must use robust encryption for sensitive data like card numbers, CVVs, and personal information both in transit and at rest.

- **Preventing fraud:**

Multi-factor authentication, real-time transaction monitoring, and strong access control are essential to prevent unauthorized access and fraudulent use of virtual cards.

- **Compliance:**

The system must adhere to financial regulations and security standards, such as PCI DSS, to protect cardholder data and maintain customer trust.

Reliability

- **High availability:**
  Users expect to be able to access their virtual cards at any time. The system needs to be highly available, with minimal downtime, to allow for card creation, spending, and management.
- **Diasater recovery:**
  A plan for disaster recovery is necessary to ensure the system can recover quickly from failures and maintain service continuity.
- **Error handling:**
  The system must gracefully handle errors, such as failed transactions, and provide clear feedback to the user.
-

Scalability

- **Handling growth:**
  As the user base grows, the system must be able to handle a large increase in users and transaction volume without performance degradation.
- **Elasticity:**
  The infrastructure should be able to automatically scale up during peak loads and scale down during off-peak times to manage resources efficiently.

Performance

- **Fast response times:**
  Users expect near-instantaneous responses when creating a virtual card, checking a balance, or making a purchase. Response times for all operations should be within acceptable limits, even under heavy load.
- **Efficiency:**
  The system should be efficient, consuming minimal resources to perform its tasks, which contributes to lower operational costs and better performance.
-

Portability

- **Multi-device support:**
  The system's interface (e.g., a mobile app or web portal) should be accessible and function correctly across various devices (smartphones, tablets, desktops) and operating systems (iOS, Android, Windows, etc.).
- **Browser compatibility:**
  If a web interface is provided, it must be compatible with major web browsers.

Data integrity

- **Data accuracy:**

All financial data, including transaction history and card balances, must be accurate and consistent across the system.

- **Preventing data corruption:**
  Mechanisms must be in place to prevent data from being corrupted or lost, ensuring the reliability of all financial records.
-

Reusability

- **Modularity:** Key components, such as the payment processing module or the user authentication service, should be designed in a modular way so they can be reused in other applications or services within the company. This reduces development time and costs for future projects.

Application compatibility

- **Seamless integration:** The system should be compatible with other financial services, such as payment gateways, accounting software, and budgeting apps, to provide a complete financial experience for the customer.

Appendices