PRD - Procivis Banking Integration Prototype

Overview

This document outlines the requirements for a prototype banking system that integrates verifiable credentials (VC/VP) for secure payment processing. The system consists of three main components: Bank Account Management, Merchant Payment System, and Security Management.

System Architecture

The system includes:

- **Bank Interface**: Account creation and credential issuance
- **Merchant Interface**: Payment request and processing
- **Security System**: Credential suspension and revocation
- **Email Notification System**: User communications
- **ProCivIS Integration**: Verifiable credential management

Due to the simplicity of the prototype, the banking backend and merchant backend is the same component.

Core Features

1. Account Management

- Create new user accounts with card details
- Generate secure 6-digit PINs
- Display account information
- Issue verifiable credentials with QR codes

2. Credential Management

- Issue verifiable credentials for payment cards
- Generate QR codes for credential sharing
- Suspend credentials for security purposes
- Revoke credentials after multiple failed attempts

3. Payment Processing

- Create payment requests with QR codes
- Process credential-based payments
- Handle payment success/failure scenarios
- Store payment transaction history

4. Security & Notifications

- Send security alerts for failed payment attempts
- Provide security credential suspension screen
- Email notifications for account events
- Track consecutive PIN failures
- Automatic credential revocation after threshold

Use Cases & Sequence Flows

UC1: Account Registration & PIN Generation

Actor: Bank Employee
Flow: Employee creates account → System generates PIN → Email sent to customer → Account activated

Key Steps:

1. Bank Employee inputs customer data (email, PAN, expiry date, cardholder name, balance)
2. System generates random 6-digit PIN
3. System saves account data to database
4. System sends PIN notification email
5. System returns confirmation to bank employee

UC2: Credential Issuance

Actor: Bank Employee, Bank Customer
Flow: Customer requests credential → Bank employee initiates issuance → System generates VC → QR code provided → Customer scans QR → Credential stored in wallet

Key Steps:

1. Bank employee selects account for credential issuance
2. System retrieves account data from database
3. System requests credential creation from ProCivIS
4. Procivis creates and returns VC
5. System generates QR code for credential sharing
6. Customer scans QR code with wallet app
7. Credential securely stored in customer wallet

UC3: Payment Processing

Actor: Merchant, Bank Customer with Wallet
Flow: Merchant creates payment request → Customer scans QR → Credential verified → PIN required → Payment completed

Key Steps:

1. Merchant creates payment request with amount and description
2. System generates payment proof request via Procivis
3. Procivis returns QR code
4. Customer scans QR with wallet app
5. Customer shares payment proof
6. System polls for proof status until ACCEPTED
7. System verifies customer balance
8. Merchant prompted for PIN validation
9. If PIN correct and balance sufficient: payment completed
10. If PIN incorrect: increment failure counter and send security alert

UC4: Security Alert & Credential Suspension

Actor: Bank Customer
Flow: Payment fails → Security email sent → Customer chooses to suspend → Credential suspended

Key Steps:

1. Payment attempt fails (wrong PIN)
2. System sends security alert email with suspension option
3. Customer clicks suspension link in email
4. Customer confirms credential suspension
5. System suspends credential via ProCivIS
6. Confirmation email sent to customer
7. Credential blocked for 30 days

Technical Requirements

Data Models

Account

- ID, email, PAN, expiry date, CVC, cardholder name
- Balance, PIN, creation timestamp
- Credential ID (when issued)

Payment

- ID, amount, description, merchant ID
- Status (pending, processing, completed, failed)
- Customer information, timestamps
- Transaction ID, failure reasons

API Endpoints

Account Management

- POST /api/accounts - Create new account
- GET /api/accounts - List all accounts

Credential Management

- POST /api/credentials/issue - Issue new credential
- POST /api/security/suspend-credential - Suspend credential

Payment Processing

- POST /api/payments/request - Create payment request
- GET /api/payments/{id}/status - Check payment status
- POST /api/payments/{id}/process - Process payment
- POST /api/payments/{id}/verify-pin - Verify PIN
- GET /api/payments/all - Get payment history

Integration Requirements

Procivis Integration

- Credential creation and management
- Proof request generation
- QR code generation for sharing
- Status polling for proof acceptance

Email System

- PIN delivery notifications
- Security alert emails
- Credential suspension confirmations
- Credential revocation notifications

User Interface Requirements

Bank Interface

- Account creation form with validation
- Accounts table with masked card numbers
- Credential issuance buttons
- QR code modal for credential sharing

Merchant Interface

- Payment request form
- QR code display for customer scanning
- Real-time payment status updates
- PIN input interface for completion
- Payment history table

Security Interface

- Credential suspension page
- Clear warning messages
- Confirmation dialogs
- Status feedback

Security Requirements

- PIN masking and secure transmission
- Card number masking in UI (show first 4 and last 4 digits)
- Secure credential storage and transmission
- Rate limiting for PIN attempts
- Automatic credential revocation after 5 consecutive failures
- Email confirmation for security actions

Success Metrics

- Account creation success rate
- Credential issuance completion rate
- Payment processing success rate

Future Enhancements

- Multi-factor authentication
- Biometric validation
- Real-time fraud detection
- Mobile app integration
- Advanced reporting and analytics

Technical Constraints

- jQuery-based frontend (prototype constraint)
- Procivis integration for credential management
- In memory/file database for prototype
- Single-tenant architecture