

Software Requirements Specification (SRS) Document

1. Introduction

1.1 Purpose

The purpose of this document is to define the requirements for developing a Bank Application that allows users to manage their bank accounts, make transactions, and access banking services through a secure and user-friendly interface.

1.2 Scope

The Bank Application will provide customers and bank employees with features for account management, transaction processing, loan management, and reporting. It will utilize Java for the application's backend and MySQL as the database.

1.3 Definitions, Acronyms, and Abbreviations

- UI: User Interface
- DBMS: Database Management System
- API: Application Programming Interface

1.4 References

List any references to standards, documentation, or protocols that may be used.

1.5 Overview

This SRS document is organized by defining system requirements in terms of functionality, interface, and performance, as well as other constraints and dependencies.

2. Overall Description

2.1 Product Perspective

The Bank Application is an independent application designed to manage bank operations and client accounts. It will be a desktop/web-based application utilizing MySQL as the backend database and Java for backend development.

2.2 Product Functions

The key functions include:

- User authentication and access control
- Account management (e.g., opening, closing, and updating accounts)
- Transaction management (e.g., deposits, withdrawals, transfers)

- Loan management (e.g., loan application and status tracking)
- Report generation (e.g., transaction history, account summaries)

2.3 User Classes and Characteristics

- Customers: Perform operations on their accounts, view balances, and transaction history.
- Bank Staff: Manage customer accounts, approve loan applications, and generate reports.
- Admin: Configure and manage user roles and system settings.

2.4 Operating Environment

- Client Side: Desktop or web browser
- Server Side: Java-based server, MySQL database server

2.5 Design and Implementation Constraints

- Java and MySQL are mandatory technologies.
- The application must comply with relevant financial data protection laws and standards (e.g., PCI-DSS).

2.6 Assumptions and Dependencies

- Users will have an active internet connection.
- The bank's IT infrastructure can support Java and MySQL.

3. System Features

3.1 Authentication and Authorization

Description: Secure login and role-based access.

- Requirement: Only authorized users should access the system.
- Functional Requirements:
 - FR1: Allow customers to log in with account number and PIN.
 - FR2: Staff and admin login should require multi-factor authentication.

3.2 Account Management

Description: Manage customer accounts.

- Functional Requirements:
 - FR3: Allow staff to create, update, or delete customer accounts.
 - FR4: Enable customers to view account information.

3.3 Transaction Management

Description: Enable customers to perform transactions.

- Functional Requirements:
 - FR5: Customers can deposit and withdraw funds.
 - FR6: Funds can be transferred between accounts.
 - FR7: All transactions should be logged in the database.

3.4 Loan Management

Description: Handle loan applications and approvals.

- Functional Requirements:
 - FR8: Allow customers to apply for loans.
 - FR9: Staff can approve or reject loan applications.
 - FR10: Notify customers of loan status updates.

3.5 Customer Support and Help Desk

Description: Provide customer support functionality.

- Functional Requirements:
 - FR11: Provide a messaging system for customers to contact support.
 - FR12: Track support tickets.

3.6 Report Generation

Description: Generate reports for analysis.

- Functional Requirements:
 - FR13: Generate account and transaction reports for staff and admin.
 - FR14: Report generation should allow export in multiple formats (PDF, Excel).

4. External Interface Requirements

4.1 User Interfaces

- A login page for secure access
- Dashboard for customers to view account balances and transactions
- Admin dashboard with reporting and user management

4.2 Hardware Interfaces

- Compatible with standard computer hardware for desktop applications or web browsers.

4.3 Software Interfaces

- Database: MySQL for storing user and transaction data.
- API: REST API to handle operations between the client and server.

4.4 Communication Interfaces

- HTTPS for secure data transmission between client and server.
-

5. Non-Functional Requirements

5.1 Performance Requirements

- The application should process transactions within 2 seconds.

5.2 Security Requirements

- Passwords must be hashed and stored securely.
- Implement role-based access control.

5.3 Reliability and Availability

- The application should have 99.9% uptime.

5.4 Maintainability and Supportability

- Code should be modular and documented for easy maintenance.

5.5 Usability

- The interface should be user-friendly, with intuitive navigation and clear instructions.