Vamsy Kiran Aripaka

Integrated Banking System

Accounts and Statements Module

Contents

[Problem Statement: 2](#_Toc54162225)

[Identity Management 2](#_Toc54162226)

[Account Management 2](#_Toc54162227)

[Service Provider Management 2](#_Toc54162228)

[Beneficiary Management 2](#_Toc54162229)

[Expected Development Platform 3](#_Toc54162230)

[Mile Stones 3](#_Toc54162231)

# Problem Statement:

The **IBS (Integrated Banking System)** is going to be a web application to offer various banking services online. The below are the high level client expectations.

## Identity Management

1. IBS identifies each of its customer by a unique 16 digit system generated customer id(UCI),
2. A customer has to register himself/herself by providing all KYC details and has to upload all the scanned copies of the proof documents.
3. An Accounts Executive from IBS shall be able to view those docs and approve or decline the registration request.
4. In case of approval a system generated password and UCI shall be shared with the customer through email.
5. In case of decline the same shall be communicated to the customer through an email.
6. The customer should change the password on the first login.

## Account Management

1. IBS offers the below type of accounts like Savings Bank , Reoccurring Deposit, Fixed Deposit and the IBS may add a few in future.
2. A registered consumer can open an account of the above types at the time registration.
3. A customer can hold multiple accounts of various types.
4. A customer once registered and activated and open more accounts.
5. A customer can access (retrieving info or reports) any of his accounts using the UCI and a password and additional transaction password is required to execute any operation (withdraw, transfer, utility bill payments ..etc) .
6. Customer Level Account Operations
   1. Balance check
   2. Request Mini Statement
   3. Request for Monthly or Annual or Periodic Statements.
   4. Transfer Funds
   5. Pay utility bills from IBS recognized service providers.
7. Bank Representative Level Account Operations
   1. Request for Monthly or Annual or Periodic Statements.
   2. Funds Deposit Entry

## Service Provider Management

1. A service provider is one who wants to get his customer pay him from IBS portal.
2. A service provider must have an account with our bank or with any other bank.
3. A service provider shall register along with the KYC and the account details into which he wants to receive his payments.
4. A bank representative has to approve the service provider registration and assign him a SPI (service provider id.

## Beneficiary Management

1. Consumer can
   1. Add/modify his/her accounts in other banks as beneficiary.
   2. Add/modify his/her credit cards from other banks as beneficiary.
   3. Add/modify bank accounts of other people as beneficiary.
2. These beneficiary accounts must be available for fund transfer when attempted.

# Expected Development Platform

|  |  |
| --- | --- |
| Database | MySQL 5 or above |
| Backend | Java 8 or above |
| Persistence Layer | ORM – Spring Data JPA |
| Application Framework | Spring Web MVC on Spring Boot |
| Views | HTML 5, CSS 3, JSP with JSTL |
| Security | Spring Security - Role-based Authentication |
| SCM | GIT |
| Build Tool | Maven |
| CD/CI | Jenkins |
| Container | Docker |
| Testing | Spring Test |

# Mile Stones

|  |  |  |
| --- | --- | --- |
| Mile Stone | Expectation | Deliverables |
| Phase-1, Week-1 | 1. Database script to create all required tables for this project. 2. Class diagram of entities / models. 3. Entity / Model class source code. | Create a GitHub repo common to the group and upload the following and share that URL.   1. Dbscript.sql 2. modelsClassDiagram.jpg 3. Java Project containing the entity / model class source code. |
| Phase-1,  Week-2 | 1. JSP pages for the entire project. | A Maven Web Project to be uploaded in to the Git Repo. This project is expected to contain the models and JSP pages. |
| Phase-2,  Week-1 | 1. CSS styling for JSP views 2. Entities and Model split up 3. ORM repositories for entities | A Spring boot Web MVC project to be uploaded in to the Git Rpo. This project will have Entities, models, ORM based DAOs (repositories) and JSP pages |
| Phase-2,  Week-2 | 1. Controllers for first two modules 2. Services for first two modules 3. Exception Handling | A Spring boot Web MVC project to be uploaded in to the Git Rpo. This project will have Entities, models, ORM based DAOs (repositories), Services, Controllers and JSP pages |
| Phase-3,  Week-1 | 1. Controllers for let over modules 2. Services for let over modules 3. Exception Handling | A Spring boot Web MVC project to be uploaded in to the Git Rpo. This project will have Entities, models, ORM based DAOs (repositories), Services, Controllers and JSP pages |
| Final Review | 1. Spring Security Integration 2. Spring Test 3. Jenkins pipeline script 4. Docker script | A Spring boot Web MVC project to be uploaded in to the Git Rpo. This project will have Entities, models, ORM based DAOs (repositories), Services, Controllers and JSP pages with spring security integration, spring test cases, Jenkinsfile and Dockerfile |