|  |  |
| --- | --- |
| ONLINE BANKing system | Abstract  The Online Banking Application should be a Browser based Application to keep track of Account details, Fund Transfer, Transaction details and other events.    Java Full Stack Developer |

Contents

[Important Instructions 3](#_Toc496274043)

[Business-Requirement: An Overview 3](#_Toc496274044)

[Module Description 4](#_Toc496274044)

[Technical Spec – Solution Development Environment 5](#_Toc496274045)

[Front End Layer 5](#_Toc496274046)

[Middle Tier Layer 5](#_Toc496274047)

[Database & Integration Layer 5](#_Toc496274048)

[Ancillary Layer 5](#_Toc496274049)

[Deployment & Infrastructure 5](#_Toc496274050)

[Editors 5](#_Toc496274051)

[Wireframes 6](#_Toc496274052)

[Login 6](#_Toc496274053)

[Register 6](#_Toc496274054)

[Fund Transfer 7](#_Toc496274055)

[Open FD 8](#_Toc496274055)

[Database Tables 9](#_Toc496274056)

# Important Instructions

1. Follow the design specifications mentioned in the case study. You are free to improvise certain specifications mentioned in the case-study
2. You should stay **motivated** to initiate such and specific communications as it may have positive influence on the evaluation scores.
3. Please make sure that your code does not have any compilation errors while submitting your case study solution.
4. Implement the code using best design standards.
5. The **logo for the app should be in SVG format**. Use logo of your choice.
6. UI should be **responsive** across multiple devices.
7. It should be the **progressive web app** such that certain parts of the application are accessible in absence of connectivity.

# Business-Requirement: An Overview

The Online Banking System is an online responsive website, which is easily available to customers on laptops, desktops, mobile phones and tablets. Using this system people can apply to open new bank account, FD and request banking services like transfer funds, check balance, request for cheque book, ATM pin change. Also, people can use their social network credentials like Gmail and linked accounts to login and see different events and other services related to investment and currency trends, opt to attend the seminars and get notified when bank makes some announcements.

Below are the features of Online Banking System:

* Login
* Registration
* Forgot password and reset password
* Check Balance and view last 5 transactions
* Transfer Funds to another bank account
* Open New FD
* Login through social account and register for event
* Request for Cheque book and change ATM Pin password

# Module Description

**Login Page:**

User will be asked to login by providing the username and password, upon validation user will be taken to the accounts page where he can see his details like balance in the bank and links to other services for his account.

**Check Balance and transaction history.**

This page will display the current balance in the bank account of the user. And last 5 transactions which he did recently by default. User will also have option to see more transactions if needed

**Transfer Funds**

User will be able to transfer funds to another account within the banking system.

Successful transfer of available funds to another account with be managed by a making a call to API app.

Further WebJobs will email the transaction details to the user.

**Open New FD**

User also be able to open a new Fixed deposit for 3,6, 9 and 12 months. User should be able to choose from the option and select the FD product. This page communicates to API which will deduct the amount of FD from his bank account and generate a new FD, which will be emailed to his bank account.

**Login and Register for Events.**

User can register to attend the events organized by banks. For this user can login from his/her social media account like gmail, linkedin, twitter, Hotmail etc.

User does not need to hold the bank account with the bank. Thus this is an open invitation for the people who would want to get associated with the bank and explore the services provided by bank.

**Request for Cheque Book and Change ATM PIN**

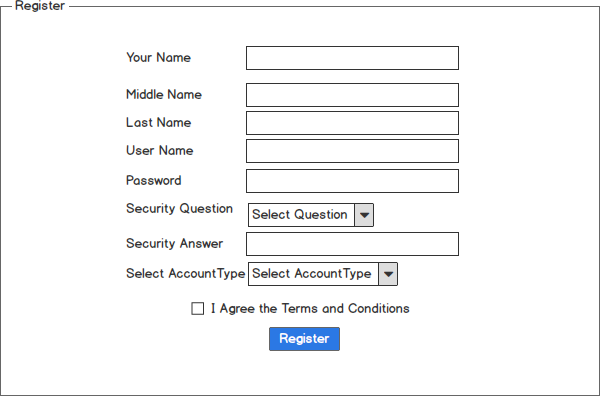
User should be able to make request for new cheque book and change/generate the ATM pin online. They interact with WebAPI which collects this information from the user .

# Wireframes

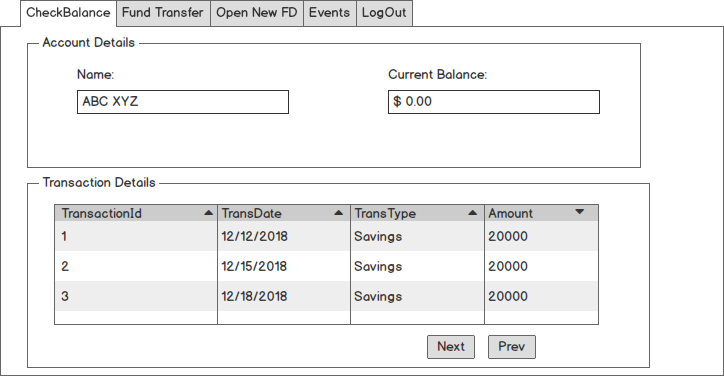
## LOGIN



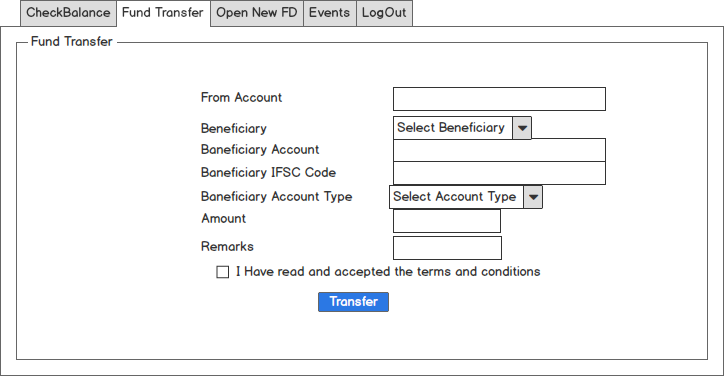
## Register



## Check Balance



## Fund Transfer



## Open New FD

# C:\Users\SANTHOSH\AppData\Local\Temp\flaC80C.tmp\Snapshot.png

## Change ATM PIN

# C:\Users\SANTHOSH\AppData\Local\Temp\flaD8E8.tmp\Snapshot.png

Based on the requirements, Front End need to be divided into multiple components to accommodate above Wireframes. React Routing can be used to create navigation Links. For Authentication, store JWT token in Local or Session Storage(on Client). REST APIs are invoked from the corresponding Services,

As known JWT token is generated on the Server side and received by Client on successful authentication.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

# Database Tables

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tbl\_Atmdtls** |  | **Tbl\_Accounts** |  | **Tbl\_checkbookdtls** |  | **Tbl\_FundTransfer** |
|  |  |  |  |  |  |  |
| ID |  | AccTypeID |  | Checkbook\_Id |  | TransferID |
| AccoutNumber |  | AccType |  | AccountNumber |  | SourceAccNumber |
| Atmpin |  |  |  | RequestedDate |  | DestAccNumber |
|  |  |  |  | IssueDate |  | DestAccTypeID |
|  |  |  |  | Priority |  | TransferAmount |

|  |  |  |
| --- | --- | --- |
| **Tbl\_Transactions** |  | **Tbl\_Users** |
|  |  |  |
| TransactionID |  | AccountNumber |
| AccountNumber |  | Name |
| AccountTypeID |  | MiddleName |
| TransactionType |  | LastName |
| TransDate |  | UserName |
| Amount |  | Password |
|  |  | SecurityQuestions |
|  |  | SecurityAnswers |
|  |  | AccountTypeID |
|  |  | MobileNumber |

# 

# Software Requirements

This case study assumes knowledge of programming and hands-on with below mentioned skills

The technologies included in Full Stack are not limited to following but may consist of:

* UI Layer (HTML5, CSS3, Bootstrap, JavaScript, jQuery, React)
* Middleware Restful API (Spring Boot REST API with MVC)
* Database & Database Persistence (MySQL, JPA)
* Ancillary skills (GIT & GITHUB)

To complete this case study, you should be comfortable with basic single page web application concepts including REST and CRUD.

# Technical Spec – Solution Development Environment

## Front End Layer

|  |  |
| --- | --- |
| **Framework(s)/SDK/Libraries** | **Version** |
| React Js | 5.0 or above |
| Bootstrap | 5.0 or above |
| CSS | 3 |
| HTML | 5 |

## Middle Tier Layer

|  |  |  |
| --- | --- | --- |
| **Technology** | **Framework(s)/SDK/Libraries** | **Version** |
| Java Stack | * Java * Spring Boot MVC * REST API |  |

## Database & Integration Layer

|  |  |  |
| --- | --- | --- |
| **Technology** | **Framework(s)/SDK/Libraries** | **Version** |
| Java Stack | * MySQL * JPA |  |

## 

## Ancillary Layer

|  |  |  |
| --- | --- | --- |
| **Technology** | **Framework(s)/SDK/Libraries** | **Version** |
| Source Code Management Tools | GITHUB |  |
| Dependency Management Tool | NPM | 4.x |

## Security

|  |  |
| --- | --- |
| **Name** | **Version** |
| JWT |  |

## Editors

|  |  |
| --- | --- |
| **Name** | **Version** |
| Visual Studio | 2022 |
| Eclipse Enterprise Edition | 2022 |

# Architecture

A physical architecture is an arrangement of physical elements, (system elements and physical interfaces) that provides the designed solution for a product, service, or enterprise. It is intended to satisfy logical architecture elements and system requirements. Workout Tracker follows a three layered architecture namely presentation layer, business logic layer and data access layer.

* **Presentation Tier** is the tier in which the users interact with an application. It is a single-page-application of responsive nature. Presentation Tier will consume restful API implemented in Business Tier to display content to the user.
* **Business Tier** is mainly working as the bridge between Data Tier and Presentation Tier. All the Data passes through the Business Tier before passing to the presentation Tier. Business Tier is the sum of Business Logic Layer, Data Access Layer and Value Object and other components used to add business logic. It exposes Rest API which can be called by Presentation Tier to display content to the user. It will also send the data from Presentation tier to Data Tier using Rest API.
* **Data Tier** is basically the server which stores all the application’s data. Data tier contents Database Tables, XML Files and other means of storing Application Data.

# 

# Deliverables of this Phase

1. Html templates along with React components

# Important Instructions

1. Consider using below Java8 features
2. Lambda Expressions
3. Collections
4. Generics
5. Sample Design provided is just for reference, Associates can make changes over it or follow their own Design.
6. Please make sure that your code does not have any compilation errors while submitting your case study solution.
7. The final solution should be a zipped code having solution. Solution code will be used to perform Static code evaluation.
8. Implement the code using best design standards/family Design Patterns.
9. Use Internationalization for all the labels and messages in Rest API Development.
10. Use appropriate logging methods for logging statements/variable/return values.
11. Write web service which takes input and return required details from database.
12. Use JSON format to transfer the results.