

---

# **Software Requirements Specification**

**for**

# **Bank Management System**

**Prepared by: Ankit Nehul 111903018 |**

**Harshal Chavan 111903036**

**TY Comp, Div 1, T1**

**College of Engineering, Pune.**

# Table of Contents

<b>Table of Contents .....</b>	<b>ii</b>
<b>Revision History .....</b>	<b>ii</b>
<b>1. Introduction .....</b>	<b>1</b>
1.1 Purpose .....	1
1.2 Intended Audience .....	1
1.3 Intended Use .....	1
1.4 Product Scope .....	1
1.5 References .....	1
<b>2. Overall Description .....</b>	<b>2</b>
2.1 Product Perspective .....	2
2.2 Product Functions .....	2
2.3 User Classes and Characteristics .....	2
2.4 Operating Environment .....	2
2.5 Design and Implementation Constraints .....	2
2.6 User Documentation .....	2
2.7 Assumptions and Dependencies .....	3
<b>3. External Interface Requirements .....</b>	<b>3</b>
3.1 User Interfaces .....	3
3.2 Hardware Interfaces .....	3
3.3 Software Interfaces .....	3
3.4 Communications Interfaces .....	3
<b>4. System Features .....</b>	<b>4</b>
4.1 Registration and Login .....	4
4.2 Transfer Funds .....	4
4.3 Transaction History .....	4
4.4 Edit Profile .....	4
<b>5. Other Nonfunctional Requirements .....</b>	<b>4</b>
5.1 Performance Requirements .....	4
5.2 Safety Requirements .....	5
5.3 Security Requirements .....	5
5.4 Software Quality Attributes .....	5
<b>Appendix A: Glossary .....</b>	<b>5</b>

# 1. Introduction

## 1.1 Purpose

- The main purpose of our project is to provide interface for secure and efficient net banking facilities.
- Reduce the manual work.
- This project will provide an interface wherein user can perform basic banking operations.

## 1.2 Intended Audience

- Developers
- End users

## 1.3 Intended Use

- **Developers:** The developers who can evaluate the project's capabilities and more easily determine where their effort should be focused to enhance or extend its capabilities. They may enhance this project by updating it.
- **End users:** This application's end-users who want to learn more about what this project can do, this is the user handbook for them to properly operate this application.

## 1.4 Product Scope

The scope of this project is limited to the activities of the operation units of the banking system which includes opening of account, login to an account, deposit of funds, withdrawal of funds and transfer.

- Bank can use this application to provide better interface for their customers.
- Customer can access his/her all accounts present in various branches of the same bank at one click.
- Manager can access all accounts present in the bank through this application.
- Reduction in manual work and workload will be possible through this application.

## 1.5 References

- 1 HTML - <https://developer.mozilla.org/en-US/docs/Web/HTML>
- 2 - CSS - <https://developer.mozilla.org/en-US/docs/Web/CSS>
- 3 - JavaScript - <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
- 4 - React Js - <https://reactjs.org/docs/getting-started.html>
- 5 Other resources from YouTube.

## 2 Overall Description

### a. Product Perspective

The client will have client interface in which he can interact with the banking system. It is web-based interface which will be the web page of the banking application. Starting a page is displayed asking the type of user he is whether customer or admin and the option for creating account. Then the page is redirected to the login page where use can enter the login details. If the login particulars are valid, then the user is taken to a home page. The admin will have admin interface. He will also have a login page where he can enter the login details.

### b. Product Functions

- Login process: This module allows valid customer to access the functionalities provided by the bank.
- Balance enquiry: This module maintains the balance details of a particular account.
- Update profile: This module allows customer to update profile of their account.
- Funds transfer: This module allows the customer to transfer funds from one account to another.
- Transaction history is visible.

### c. User Classes and Characteristics

- Customers: He is a normal user who will have account in the bank. He can transfer funds from one account to other accounts and view his transaction history.
- Admin: He can view and manage all accounts.

### d. Operating Environment

- Operating system: Windows, Linux, mac etc
- Platform: Web
- More than 1 GB RAM and free space in hard drive
- Development tools: VS code.

### e. Design and Implementation Constraints

- This system works only on web.
- Language will be using are JS, Mysql, HTML, CSS

- Time constraint: Completing the entire project in such a short period of time is a challenge

## **f. User Documentation**

A registered user can have following facilities:

- Accounts and account status.
- The balance enquiry.
- Fund transfer.
- Edit profile.
- Transaction history.
- Debit/credit card requests.

## **g. Assumptions and Dependencies**

- Every user should be comfortable of working with computer and net browsing.
- He should have aware of the banking system.
- He must have basic knowledge of English too.

# **3 External Interface Requirements**

## **a. User Interfaces**

There are three different ways for a user to interact with the system:

- New user: Who all visited the bank web page and are ready to start account in bank. They can register and create bank account.
- Existing user: The existing user is the most typical user of the Online Banking system. Each user have their own account and authorized login access. The existing user can login to their account and perform the operations.
- Admin: Admin maintain the database and the permissions to users. Admin is master user of system.

## **b. Hardware Interfaces**

- More than 1 GB RAM.
- Free space in hard drive.

## **c. Software Interfaces**

- Web browser: Chrome, firefox etc.
- Operating system: Any OS.

- Development tools: VS code

#### **d. Communications Interfaces**

This system will offer banking facilities anywhere anytime through internet facility.

- A web browser

## **4 System Features**

#### **a. Registration and Login**

- User can register and create account in the bank.
- If user already registered, then he can login using valid credentials.

#### **b. Transfer Funds**

- User can transfer money from one account to another account.
- User can deposit/withdraw money.

#### **c. Transaction History**

- User are able to see the previous transactions.
- User can search for the specific transaction.

#### **d. Edit Profile**

- User is able to edit his/her details.

## **5 Other Nonfunctional Requirements**

#### **a. Performance Requirements**

It is available during whole week for all 24 hours.

System can withstand even though many number of customers request the desired service. Access is given to only valid users of bank who requires the services such as balance enquiry, update profile, fund transfer etc.

#### **b. Safety Requirements**

- Secure databases
- Data should be backup.

- There should be power backup for server.

### **c. Security Requirements**

User authentication: Facility of unique user number and password in such a way that unauthorized user cannot log in.

Operational rights: Operational rights for each user can be defined. Thus, a user can have access to specific options only.

### **d. Software Quality Attributes**

- Usability: The user is assumed to have basic knowledge of the computer and internet browsing. The proper user interface, online help and guide to use and maintain the system must be sufficient to educate the user on how to use the system without any problems.
- Reliability: The system must be very reliable due to the importance of data and the damages incorrect or incomplete data can do.
- Availability: Banking system shall not experience any delay in the service response to their commands
- Security: All users of the system shall not be able to perform actions which will cause harm to any person or damage of the system.
- Maintainability: There shall be an access on the server for the purpose of upgrading the software.

## **Appendix A: Glossary**

Abbreviations:

OS: Operating System

VS code: Visual Studio Code