| | **Software Design Specifications**  **Bank Management System**  **Version: 1.30**   | Project Code |  | | --- | --- | | Supervisor | Ms. Syeda Rubab Jaffar | | Co Supervisor |  | | Project Team | Syed Yousif Ali Shah (22K-5174)  Huzaifa Kashif  (22K-5158)  Muhammad Hamza Hussain  (22K-5182) | | Submission Date | 05/05/2024 | | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |   **[Instructions]**   * *No section of template should be deleted. You can write ‘Not applicable’ if a section is not applicable to your project. But all sections must exist in the final document.* * *All comments/examples mentioned in square brackets ([]) are in the template for explanation purposes and must be replaced / removed in final document.* * *This’ Instruction’ section should also be removed in final document.* * *MS-Word Reviewing feature must be used to get the document reviewed by PMs or supervisors.*       **Document History**   | Version | Name of Person | Date | Description of change | | --- | --- | --- | --- | | v1.00 | Huzaifa Kashif | 02/05/2024 | [Created Document, And Added Diagrams] | | v1.20 | Syed Yousif Ali Shah | 04/05/2024 | [Added Diagrams] | | v1.30 | Muhammad Hamza Hussain | 05/05/2024 | [Added Theoretical Material] |         **Distribution List**  *[Following table will contain list of people whom the document will be distributed after every sign-off]*   | **Name** | **Role** | | --- | --- | |  | Supervisor | |  | Co Supervisor | |  |  |       **Document Sign-Off**  *[Following table will contain sign-off details of document. Once the document is prepared and revised, this should be signed-off by the sign-off authority.*  *Any subsequent changes in the document after the first sign-off should again get a formal sign-off by the authorities.]*   | **Version** | **Sign-off Authority** | **Project Role** | **Signature** | **Sign-off Date** | | --- | --- | --- | --- | --- | |  |  |  |  |  | |  |  |  |  |  | |  |  |  |  |  | |  |  |  |  |  | |  |  |  |  |  | |  |  |  |  |  | |  |  |  |  |  | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

**Document Information**

| **Category** | **Information** |
| --- | --- |
| Customer | FAST-NU |
| Project | <Project Title> |
| Document | Software Design Specification |
| Document Version | 1.0 |
| Status | Draft |
| Author(s) | <Names of all the authors of this document> |
| Approver(s) |  |
| Issue Date |  |
| Document Location |  |
| Distribution | Advisor  Project Coordinator’s Office (through Advisor) |

**Definition of Terms, Acronyms and Abbreviations**

*[This section should provide the definitions of all terms, acronyms, and abbreviations required to interpret the terms used in the document properly. ]*

| **Term** | **Description** |
| --- | --- |
| ASP | Active Server Pages |
| DD | Design Specification |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Table of Contents**

[**1**](#_1v1yuxt) **Introduction 8**

[*1.1*](#_4d34og8) *Purpose of Document 8*

[*1.2*](#_4f1mdlm) *Intended Audience 8*

[*1.3*](#_2u6wntf) *Document Convention 8*

[*1.4*](#_19c6y18) *Project Overview 8*

[*1.5*](#_3tbugp1) *Scope 8*

[**2**](#_28h4qwu) **Design Considerations 9**

[*2.1*](#_nmf14n) *Assumptions and Dependencies 9*

[*2.2*](#_37m2jsg) *Risks and Volatile Areas 9*

[**3**](#_2jxsxqh) **System Architecture 10**

[*3.1*](#_1mrcu09) *System Level Architecture 10*

[*3.2*](#_3j2qqm3) *Software Architecture 10*

[**4**](#_46r0co2) **Design Strategy 11**

[**5**](#_2lwamvv) **Detailed System Design 12**

[*5.1*](#_2xcytpi) *Database Design 12*

[5.1.1](#_1ci93xb) ER Diagram 12

[5.1.2](#_3whwml4) Data Dictionary 12

[5.1.2.1](#_2bn6wsx) Data 1 12

[5.1.2.2](#_qsh70q) Data 2 12

[5.1.2.3](#_3as4poj) Data n 12

[*5.2*](#_1pxezwc) *Application Design 14*

[5.2.1](#_49x2ik5) Sequence Diagram 14

[5.2.1.1](#_2p2csry) <Sequence Diagram 1> 14

[5.2.1.2](#_147n2zr) <Sequence Diagram 2> 14

[5.2.1.3](#_3o7alnk) <Sequence Diagram n> 14

[5.2.2](#_23ckvvd) State Diagram 14

[5.2.2.1](#_ihv636) <State Diagram 1> 14

[5.2.2.2](#_32hioqz) <State Diagram 2> 14

[5.2.2.3](#_1hmsyys) <State Diagram n> 14

[**6**](#_2grqrue) **References 15**

[**7**](#_3fwokq0) **Appendices 16**

# **Introduction**

## **Purpose of Document**

The purpose of this document is to provide a comprehensive design of the Bank Management System (BMS). Hence, it describes the intended functionality, design strategies, system architecture of the BMS project.

## **Intended Audience**

This document is intended for all the stakeholders involved in the designing, development, testing, deployment, maintenance, and ultimately utilization of our Banking System.

## **Document Convention**

**Font Type:** Arial

**Font Size:** 10-11

## **Project Overview**

The Bank Management System is a system specifically designed to automate various banking operations, including account creation, transaction processing, and account management.The basic design approach would involve implementing a modular (functional) architecture with distinct layers for user interface, system logic, and data access.

## **Scope**

* **Account Creation:** Users can create new accounts by providing personal and account details.
* **Transaction Processing:** Users can perform transactions such as deposits, withdrawals (including fast-cash withdrawals), and balance inquiries.
* **Account Management:** Users can manage their accounts by changing PINs, and viewing transaction histories.

# **Design Considerations**

## **Assumptions and Dependencies**

**Assumptions:**

We assume that:

* The system architecture follows a layered design approach, having separate layers for user interface, system logic, and data access.
* JavaSwing is chosen for developing the graphical user interface (GUI) components, assuming its compatibility with the overall system architecture.
* MySQL database is efficiently integrated into the system architecture, thereby, providing reliable data storage, and data retrieval.
* The system architecture is scalable, and robust enough to accommodate potential future enhancements.

**Dependencies:**

* The system’s design and architecture depends on adherence to design principles like modularity, and separation to ensure that it is maintainable and extensive..
* The architecture depends on the use of appropriate design methodologies, such as UML (Unified Modeling Language) Diagrams, for the modeling of system components, and their interactions to facilitate better communication and understanding among development teams, and the stakeholders.
* The architectural design also depends on effective communication and collaboration among teams of developers, architects and designers, as well as system stakeholders, for aligning architectural decisions with business requirements.

## **Risks and Volatile Areas**

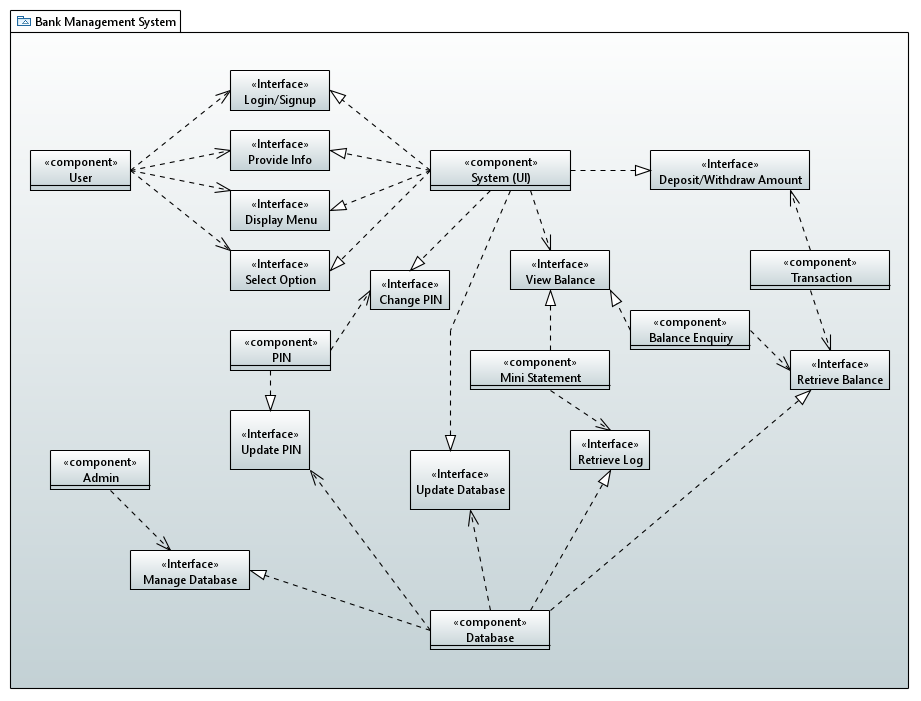
* Future changes in banking rules and regulations may influence the system design.
* Technological advancements may require updates for the system architecture and design.

As such, the system is being designed in a modular and layered manner, to ensure that it is maintainable, and extensive, as well as scalable to future changes.

## 3 System Architecture

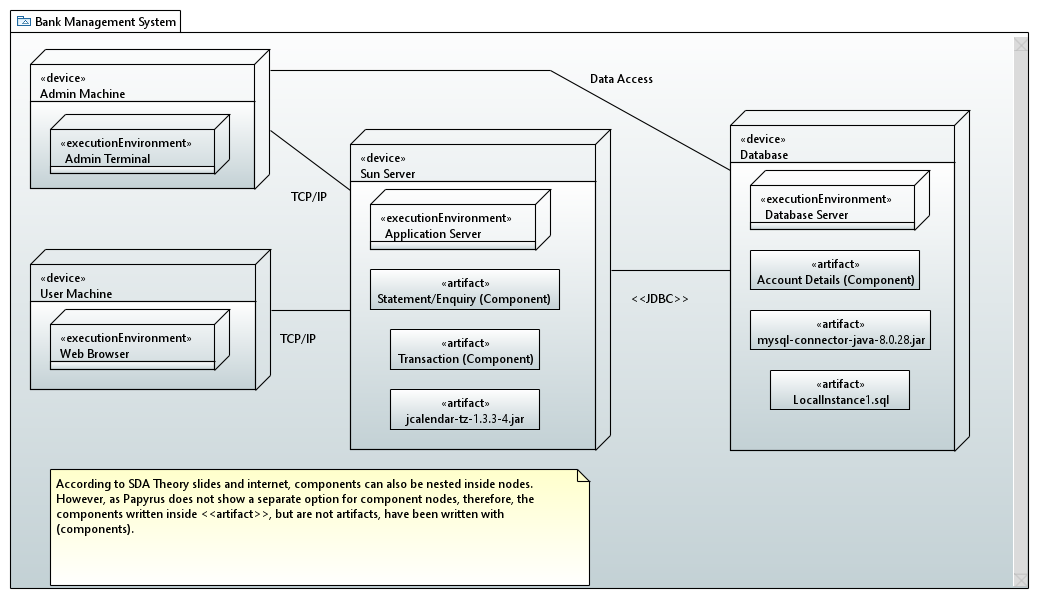
## 3.1 **System Level Architecture**

* The system is decomposed into three-tier architecture. The User Interface Tier, incorporates GUI components. The Middle Tier, where data is processed, and the Data Tier, where data is managed and stored in the database.
* The data entered by the user in UI Tier is sent to the Middle Tier for processing, and the Middle Tier manages, or retrieves the stored data from the system MySQL Database.
* The BMS interfaces with external systems such as IntelliJ Idea, MySQL Workbench Database, and System Operating System (such as MacOS, and Windows, etc).
* The system also incorporates robust error handling mechanisms to detect, report, and recover from errors. Moreover, the system incorporates security measures such as data encryption, and user authentication, to safeguard data, and prevent unauthorized access.



## 3.2 **Software Architecture**

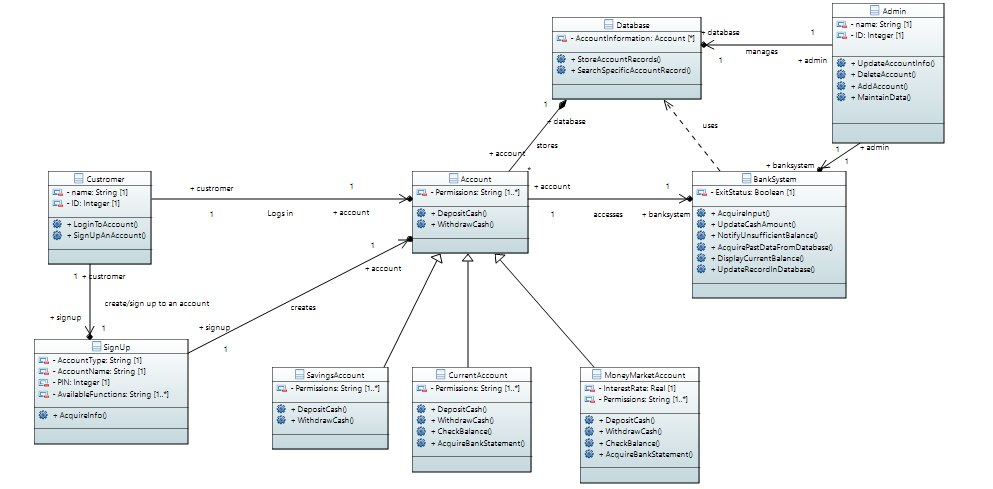
* **User Interface Layer:** JavaSwing is used for implementing Graphical User Interface (GUI) components.
* **Middle Tier:** Involves system logic for account management, transaction processing, and for viewing transaction history.
* **Data Access Tier:** Utilizes Java Database Connectivity (JDBC) for communicating with the MySQL database.



# 4 **Design Strategy**

* **Future system extension or enhancement:** As stated above, the project’s modular, and layered design allows for further modularization and scalability so as to accommodate future requirements, and changes.
* **System reuse:** Components are designed for reusability to minimize redundancy, thereby, improving maintainability.
* **User interface paradigms:** JavaSwing is used to create a user-friendly interface.
* **Data management:** MySQL database is utilized for secure and efficient data storage, and data retrieval.

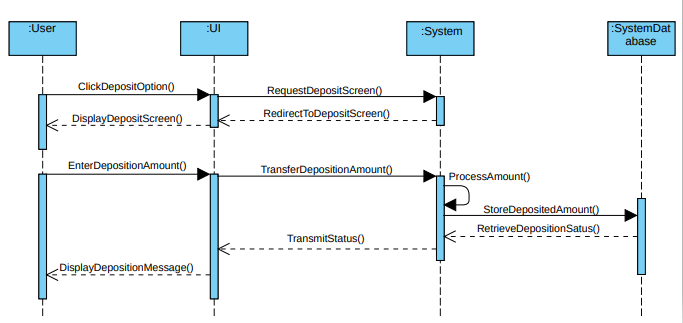
# 5 **Detailed System Design**

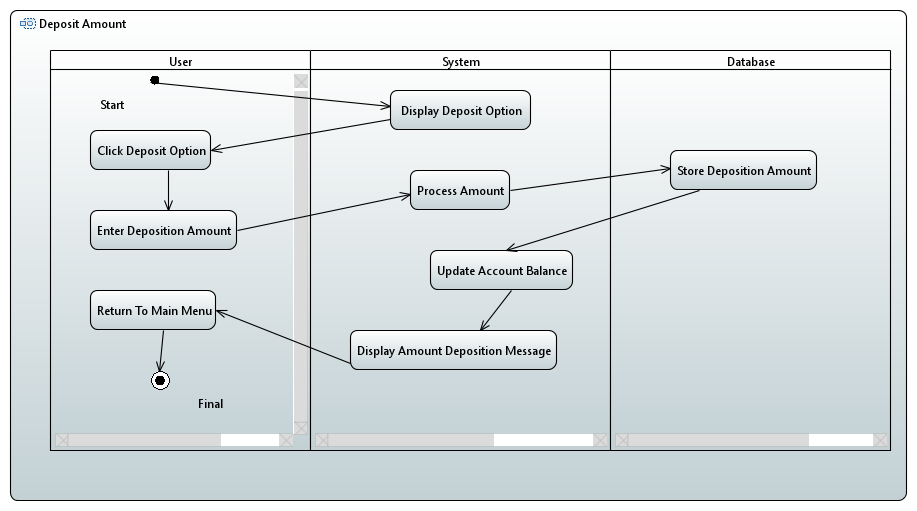


## 5.1 **Application Design**

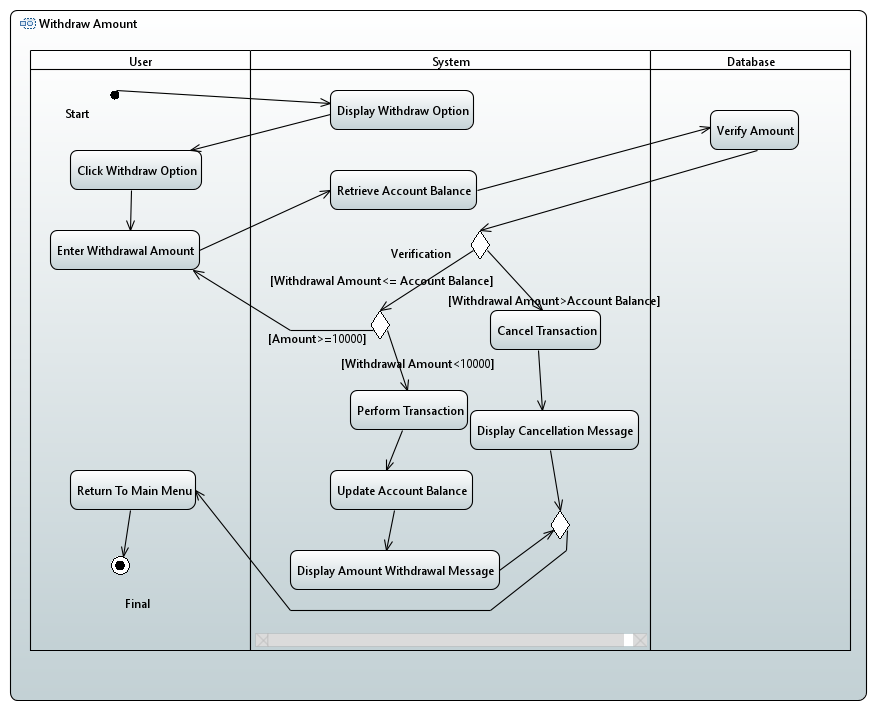
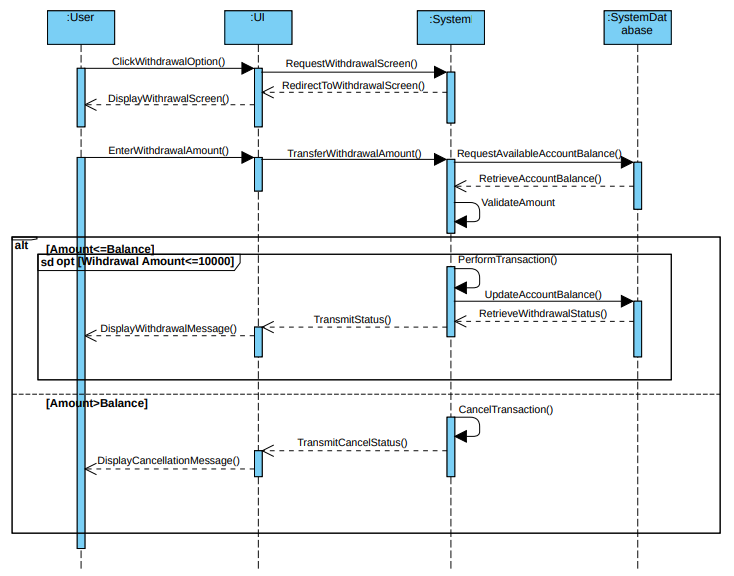
### 5.1.1 **Sequence Diagram And Activit**y Diagram

#### 5.1.1.1 Deposit Amount

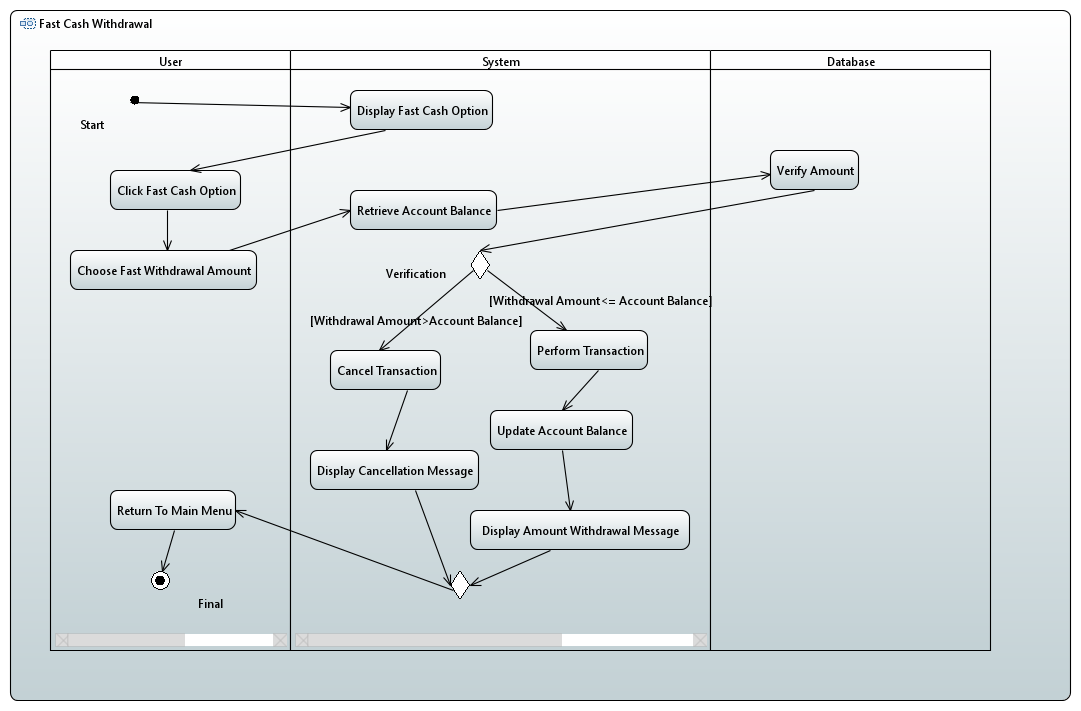
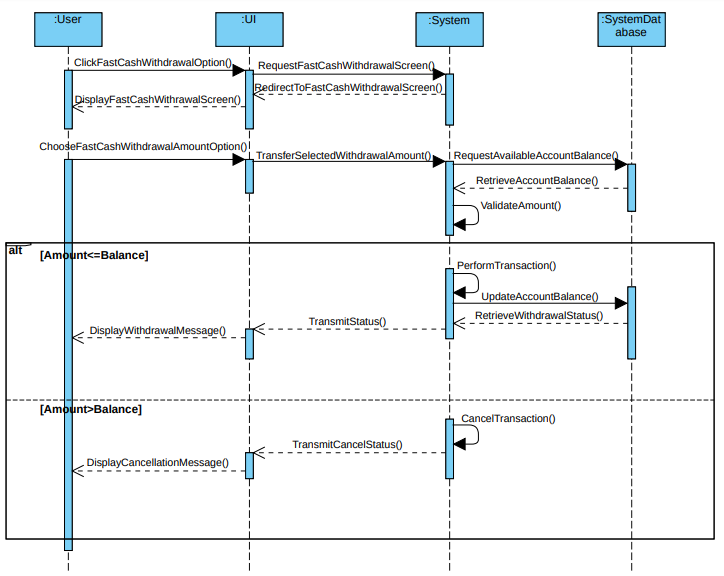
**

**

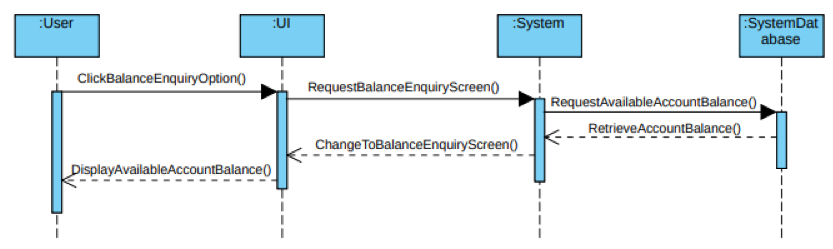
#### 5.1.1.2 Withdraw Amount

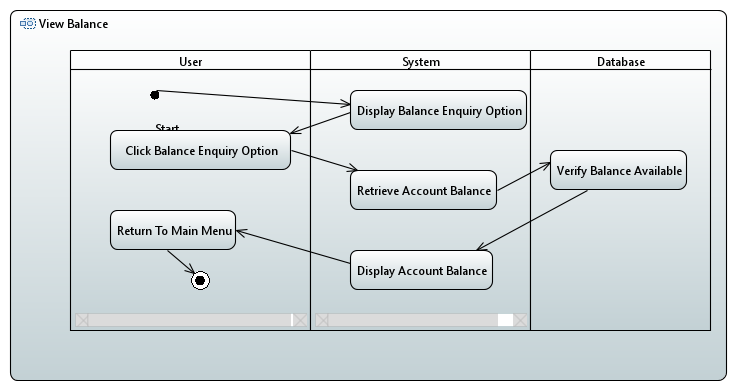


#### 5.1.1.3Withdraw Via Fast Cash

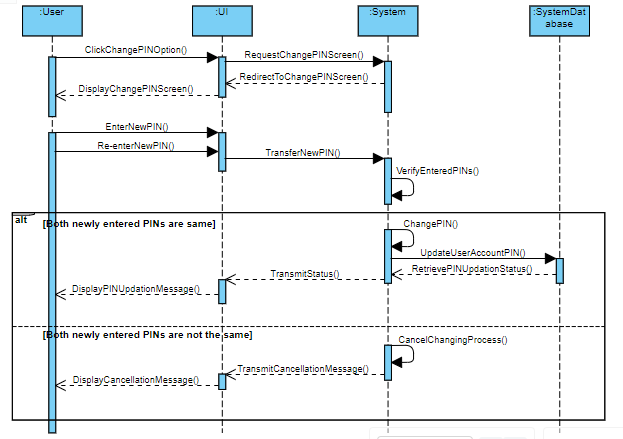


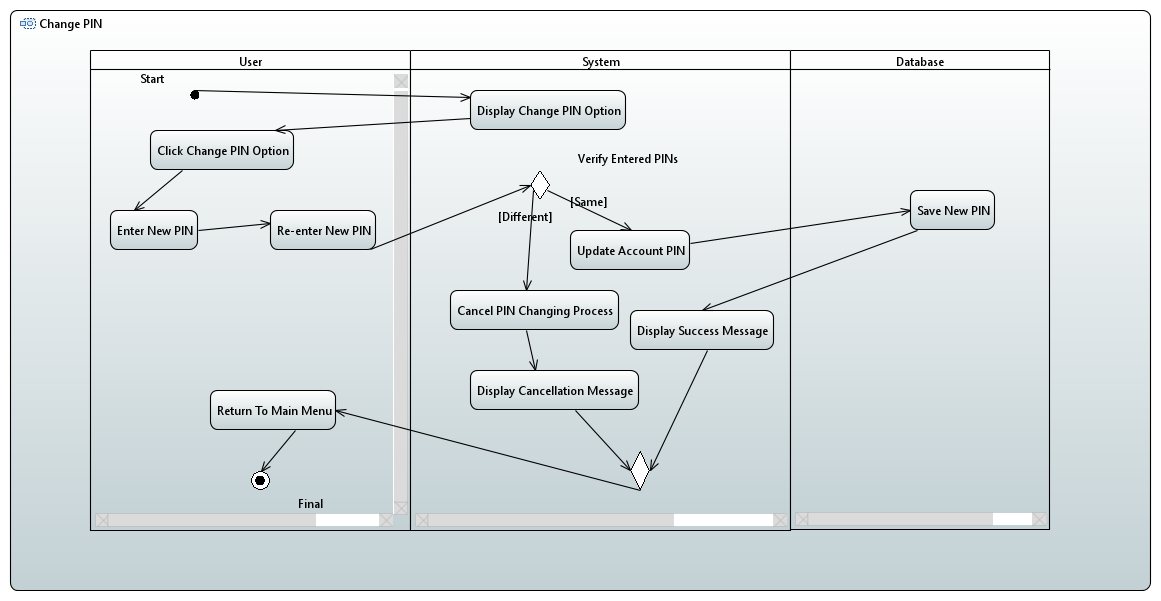
#### 5.1.1.4 View Balance





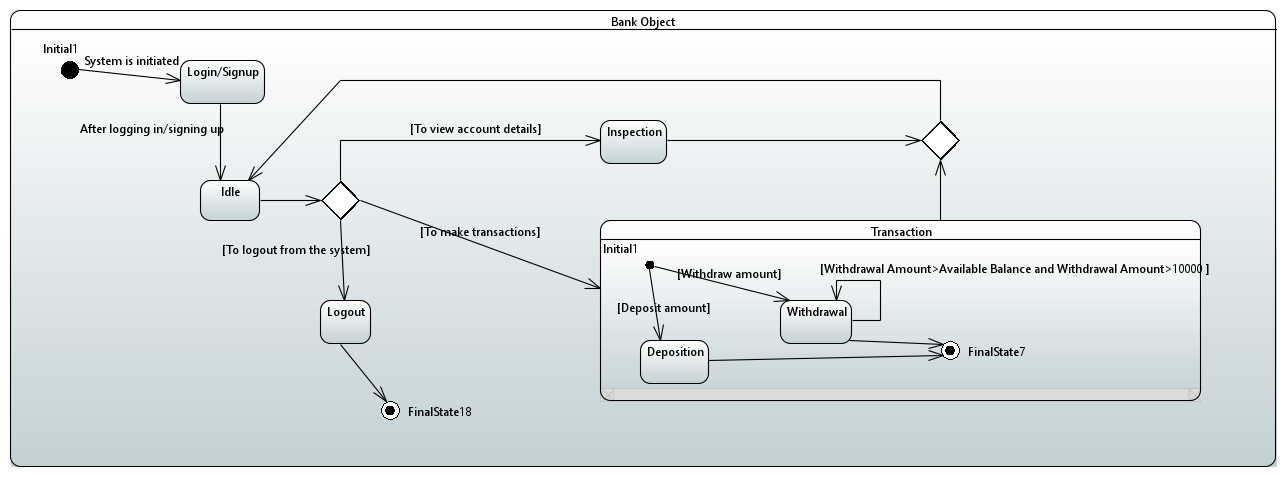
#### 5.1.1.5 Change PIN



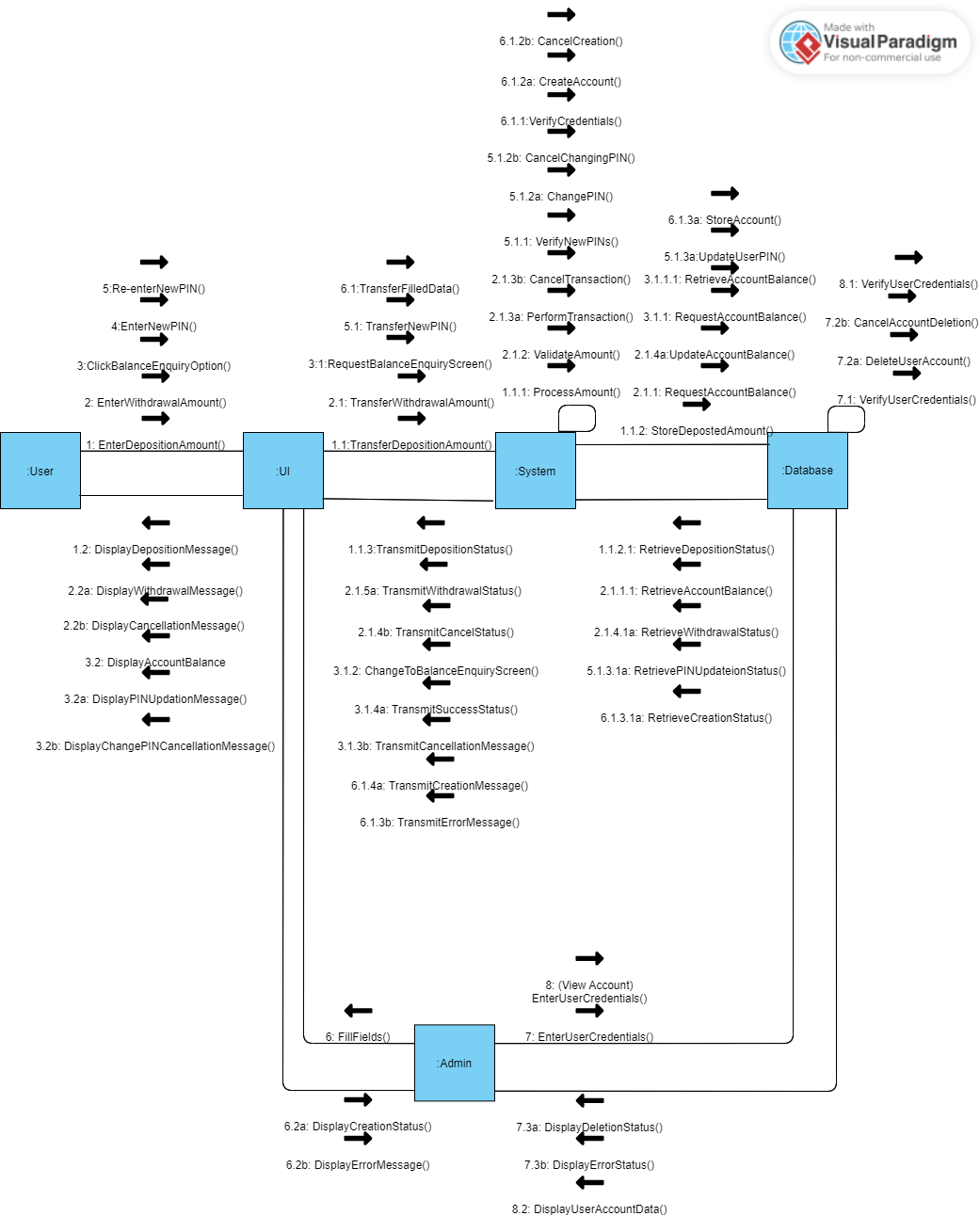


### 5.1.2 **State Diagram**

#### 5.1.2.1 Bank As An Object

**

### 5.1.3 Collaboration Diagram



# 6 **References**

* **Project References:**

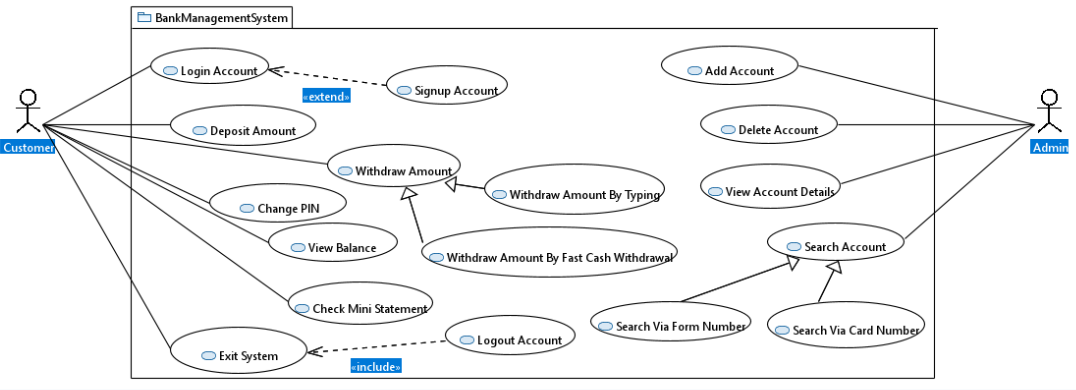
1. Code For Interview – <https://www.youtube.com/@codeforinterview>
2. Learning Never Ends – <https://www.youtube.com/@LearningNeverEnds786>

* **Designing References:**

1. Collaboration Diagram Reference –<https://www.newthinktank.com/wp-content/uploads/2012/11/UML-Communication-Diagram1.png>
2. Collaboration Diagram Designing – <https://online.visual-paradigm.com/app/diagrams/#diagram:proj=0&external=https://cdn-us-04.visual-paradigm.com/node/on/w/fkmbgvzs/rest/diagrams/shares/diagram/d19593ac-edbf-47d7-979b-27de653a055f/content&name=collaboration%20diagram%20shopping%20system>
3. Sequence Diagram Designing – https://online.visual-paradigm.com/diagrams/features/sequence-diagram-software/

# 7 **Appendices**

| BMS | Bank Management System |
| --- | --- |
| BS | Banking System |
| UI | User Interface |
| OS | Operating System |
| HDD / SSD | Hard-Disk Drive / Solid-State Drive |
| GUI | Graphical User Interface |
| JDBC | Java Database Connectivity |

******