



Software Engineering

SET 312: Agile software Engineering

Submitted For : Dr: Reham Abobeah

Target Date: 12/26/2023

Eng . Janna

Year : 2023-2024

Team Members	ID
Sherry Romany	192000008
Mahmoud Aref	192000012
George Fady	192000036
Mark Osama	192000060

# Table of content

INTRODUCTION
PROJECT OBJECTIVES
REQUIRMENTS
Activity diagram
Class Diagram

## INTRODUCTION

A desktop banking application is to provide customers with a convenient and efficient means of accessing a comprehensive range of banking services through a standalone desktop program. By leveraging the power of technology, this app aims to simplify financial interactions, enhance security measures, and deliver a personalized user experience that matches the evolving needs of the digital age.

The system will cover account management, transaction and other essential banking operations.

## PROJECT OBJECTIVES

### Enhanced User Experience:

Create a user-friendly interface that simplifies banking tasks, making it intuitive for users to navigate and perform transactions.

### Comprehensive Account Management:

Provide users with a comprehensive suite of account management tools, allowing them to view balances, transaction histories, and conduct account-related activities seamlessly.

### Efficient Fund Transfers:

Streamline the fund transfer process, both within the user's accounts and across different banks, minimizing transaction processing times.

# REQUIREMENTS

Requirements for a Desktop Banking Application typically encompass both functional and non-functional aspects. Below is a list of requirements that you might consider for such an application:

## Functional Requirements:

### User Authentication:

- Users must be able to register for an account.
- Secure login with username/password.
- Optional multi-factor authentication.

### Account Management:

- View account balances, transaction history, and account details.
- Ability to open, close, or modify account information.

### Fund Transfers:

- Transfer funds between the user's own accounts.
- Real-time balance updates.

### User Profile Management:

- Change login credentials securely.

### Search and Filtering:

- Search functionality for transactions.
- Filter transactions based on amount.

## Non-Functional Requirements:

### Performance:

- Response time for transactions should be within 2 seconds.

### Reliability:

- The application should have 99.9% uptime.
- Regular data backups to prevent data loss.

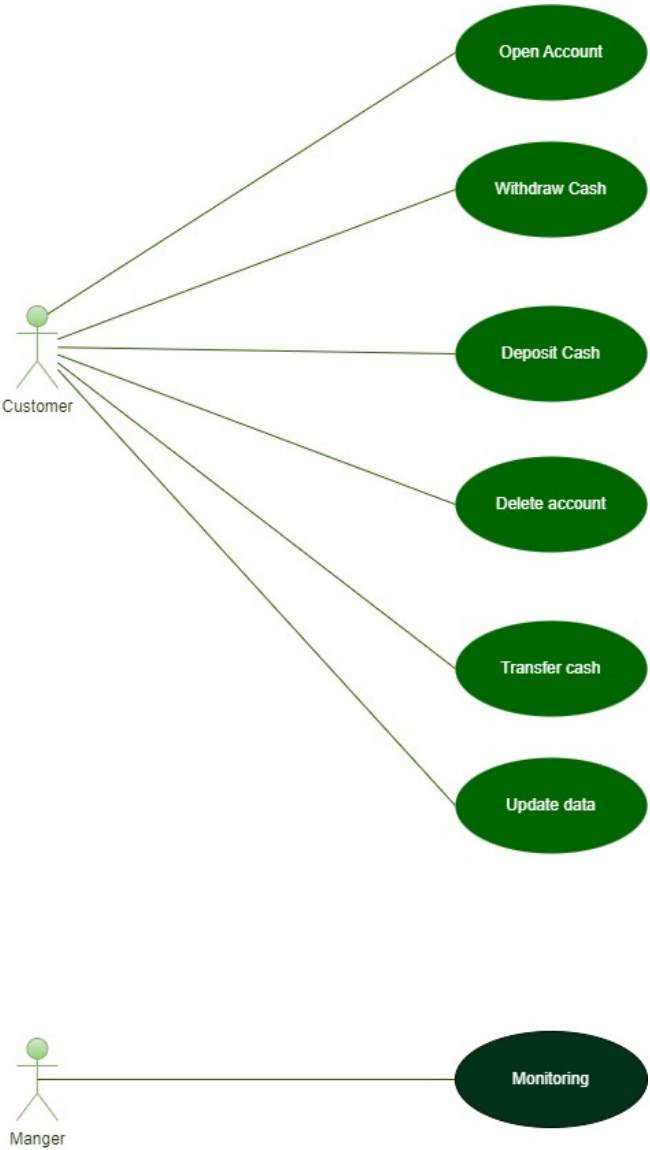
### Usability:

- The user interface should be intuitive and easy to navigate.
- Accessibility features for users with disabilities.

### Scalability:

- The system should be scalable to accommodate future growth in users and transactions.

# Activity diagram



# Class Diagram

