Software Engineering, CS385T- Project

Title: Banking management system

Section: 41S

Group no. 4

|  |  |  |
| --- | --- | --- |
|  | **Student Name** | **Student Number** |
| **1** | Nada Al-otaibi | 442003374 |
| **2** | Monerah Al-mobarak | 442002988 |
| **3** | Sarah Al-taweel | 442000786 |
| **4** | Sarah Al-juhani | 442005104 |

**First Semester 2022**

**Write your name next to section(s) you worked on, or state you all shared the work by ticking on the following statement box. (You will still share responsibilities even if you state your contribution)**

**🞏 We all share the work and responsibility of the project.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PLO** | **Student Name** | **FOR INSTRUCTOR’S USE ONLY** | | |
| **Max.**  **Mark** | **Mark**  **Obtain** | **Notes** |
| **Cover and Table of Contents** | Group work | **0.5** |  |  |
| **Project Description** | Group work |
| **Functional Requirements**  **(user and system requirements)** | **Sarah Al-juhani & Sarah Al-taweel** | **2** |  |  |
| **Non-Functional Requirement**  **(user and system requirements)** | **Monerah Al-mobarak &** **Nada Al-otaibi** |
| **Context Diagram with description** | **Sarah Al-Juhani** | **0.5** |  |  |
| **Use-case Diagram (for the whole system)** | **Monerah Al-mobarak & Sarah Al-Juhani** | **1** |  |  |
| **Scenario (MSS & Extension)**  **For Use Case 1 : [create account]** | **Monerah Al-mobarak** | **1** |  |  |
| **Scenario (MSS & Extension)**  **For Use Case 2 : [transfer money]** | **Sarah Al-Juhani** |
| **Scenario (MSS & Extension)**  **For Use Case 3 : [check balance and transactions]** | **Nada Al-otaibi** |
| **Scenario (MSS & Extension)**  **For Use Case 4 : [Name the use case]** | **Sarah Al-taweel** |
| **Scenario (MSS & Extension)**  **For Use Case 5 : [pay bills ]** | **Monerah Al-mobarak** |  |  |  |
| **Class Diagram (for the whole system)** | **Sarah Al-taweel** | **1** |  |  |
| **Sequence Diagram**  **For Function 1 : [create account]** | **Monerah Al-mobarak** | **1** |  |  |
| **Sequence Diagram**  **For Function 2 : [transfer money]** | **Sarah Al-Juhani** |
| **Sequence Diagram**  **For Function 3 : [check balance and transactions]** | **Nada Al-otaibi** |
| **Sequence Diagram**  **For Function 4 : [Name the function]** | **Sarah Al-taweel** |  |  |  |
| **Sequence Diagram**  **For Function 5 : [pay bills]** | **Monerah Al-mobarak** |  |  |  |
| **Activity Diagram**  **For Function 1 : [Name the function]** | **Sarah Al-taweel** | **1** |  |  |
| **Activity Diagram**  **For Function 2 : [transfer money]** | **Sarah Al-Juhani** |
| **Activity Diagram**  **For Function 3 : [check balance and transactions]** | **Nada Al-otaibi** |
| **Activity Diagram**  **For Function 4 : [create account]** | **Nada Al-otaibi** |  |  |  |
| **Activity Diagram**  **For Function 5 : [pays bills]** | **Monerah Al-mobarak** |  |  |  |
| **State Diagram (If any)**  **For Object 1 : [pays bills]** | **Monerah Al-mobarak** | **1** |  |  |
| **System Architecture** | **Nada Al-otaibi** |  |  |  |
| **Total Score** |  | **9** |  |  |

Contents

[1. Chapter 1 5](#_Toc92888693)

[System descriptions 5](#_Toc92888694)

[The development Process model used 5](#_Toc92888695)

[Requirements 5](#_Toc92888696)

[Functional Requirements ( User and system requirements) 5](#_Toc92888697)

[Non-Functional Requirements ( User and system requirements) 6](#_Toc92888698)

[2. Chapter 2 8](#_Toc92888699)

[Context Diagram 8](#_Toc92888700)

[System Use-case Diagram with description 9](#_Toc92888701)

[Use case scenario (MSS +Extentions) 9](#_Toc92888702)

[Use case 1[insert a title] **Error! Bookmark not defined.**](#_Toc92888703)

[Use case 2 [insert a title] **Error! Bookmark not defined.**](#_Toc92888704)

[Use case 3 [insert a title] 11](#_Toc92888705)

[Use case 4[insert a title] **Error! Bookmark not defined.**](#_Toc92888706)

[3. Chapter 3 11](#_Toc92888707)

[System Class Diagram 13](#_Toc92888708)

[Sequence Diagrams 13](#_Toc92888709)

[Sequence diagram for [insert a title] 13](#_Toc92888710)

[Sequence diagram for [insert a title] 13](#_Toc92888711)

[Sequence diagram for [insert a title] 15](#_Toc92888712)

[Sequence diagram for [insert a title] 15](#_Toc92888713)

[4. Chapter 4 16](#_Toc92888714)

[Activity Diagram 17](#_Toc92888715)

[Activity Diagram for [insert a title] 17](#_Toc92888716)

[Activity Diagram for [insert a title] 18](#_Toc92888717)

[Activity Diagram for [insert a title] 18](#_Toc92888718)

[Activity Diagram for [insert a title] 20](#_Toc92888719)

[State Diagram 21](#_Toc92888720)

[State Diagram for [insert a title] 21](#_Toc92888721)

[5. Chapter 5 22](#_Toc92888722)

[Architecture Diagram 22](#_Toc92888723)

# Chapter 1

## **System descriptions**

The bank management system is an application for maintaining a person's account in a bank. Customers can use the system to create accounts, make deposits and withdrawals, and review reports for all of their accounts. The basic aim of the system is to develop software for solving financial applications of a customer in a banking environment to nurture the needs of an end banking user by allowing them to do banking operations in a variety of ways. Additionally, to allow the user workspace to have additional functionalities not available in traditional banking software.

## **The development Process model used**

The development Process model used is Incremental model.

We chose this model because:

1. It will generate our system quickly and early.

2. It will reduce the cost of accommodating changing customer requirements.

3. It is more flexible than other models and work well with large projects like our project.

4. It will be easier for us to get customer feedback on the work that has been done.

## System Requirements

### Functional Requirements

#### **User Requirements**

1-The user should create an account in the system to log in into application at any time.

2- The user shall be able to transfer money to other bank accounts.

3-User can see their transaction report and balance enquiry too.

4-customer shall be able to pay bills.

5- The customer can check his cheque service.

#### **System Requirements**

### 1.1 To create a user account the user should write a unique username and password with a minimum of 8 characters that include at least one capital letter, one small letter, and one symbol.

### 1.2 Then the user should enter their number and the system will send a verification code, the user will enter the code to complete the registration.

### 1.3 If any of the information is incorrect the system will display an error message.

### 2.1 When the user wants to transfer money, first the user should choose from a list if the account is in a local or international bank, then enter the account number and name of the account they want to transfer the money into.

### 2.2 The system should check if the account number and name are correct and matches with a single account.

### 2.3 Then the user will enter the amount of money they want to transfer and the purpose of the transfer.

### 3.1 Balance inquiry ,the user must log into his bank account.

### 3.2 The user has to go to the accounts, and choose the account that he wants to inquire about his balance.

### 4.1 The customer most be logged into Banking System. With internet banking, the customers can use Online Pay Bill service to pay bills by debiting their account.

### 4.2 The customer needs to key in his/her bill account number each time you make a payment. In addition, the consumer makes a payment to his or her owns (up to the outstanding debt).

### 5.1 The customer most be logged into Banking System. The customer may enquiries cheque status, whether it is paid, unpaid, stopped or returned. It also allows customer to stop cheque payment and to request for a cheque book online.

### Non-Functional Requirements

**System Requirements**

**Security:**

1. The system should be secure and safe.

1.1 The system does not grant access until the user generates a strong password. For example, a strong password may contain a certain number of characters, an uppercase letter, and symbols.

1.2 When a user tries to login more than twice, their account will be locked to protect their information from hackers. To open his account, the user can contact the bank's support team to verify his identity and set a new password.

1.3 the Users should receive a confirmation code to access the account after logging in and receive a message after any action has been performed.

1.4 Passwords must never be seen when entering or at any other time.

1.5 An audit trail should be kept for each unsuccessful attempt by a user to access a piece of data.

**Maintainable:**

2. System should be able to maintain.

2.1 The system should be easy to maintain for any future issues and install new versions

2.2 It's important to keep track of the number of versions and updates by documenting it properly.

2.3 The system should keep a service log and should check if system service is due when it boots up.

2.4 The system shall not be shut down for maintenance more than once in a 24-hour period.

**Reliability :**

3 System should be reliable and available all the day

3.1 The system should be available at all times (24 hours a day, seven days a week) in order to provide better and faster service to clients at all times.

3.2 The system should be set up in a way that minimizes user errors.

Data Integrity:

4. deals with the integrity, consistency, and correctness of the data in the system.

4.1 All the data in the bank Management System should be accurate and reliable.

4.2 The internal audit system should examine the integrity of the system data area twice per second; if inconsistencies in the data are identified, the system should be alerted.

**User Requirements Usability:**

1. The system should be easy to use .

1.1 The system should be easy to understand and use, and customers should be able to simply enter the system and choose any service at any time.

1.2 the Users can quickly figure out what a feature is and what it can accomplish.

1.3 the system should support multiple languages such as Arabic and English.

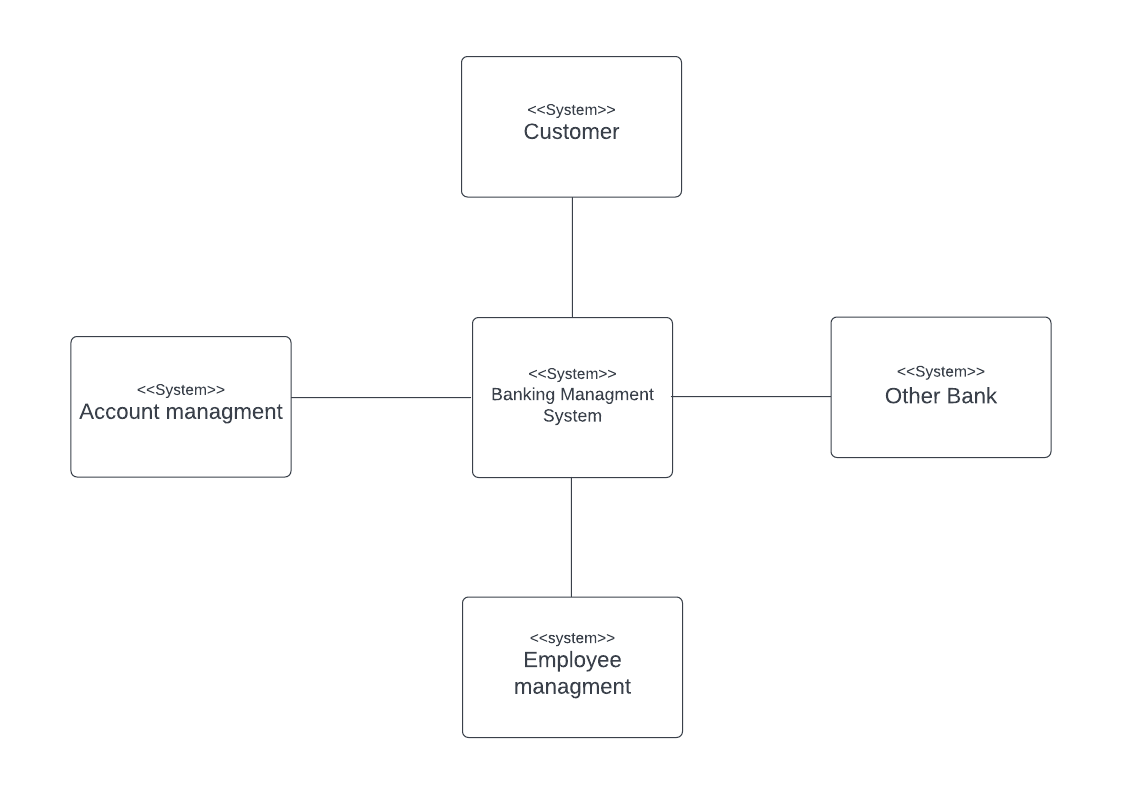
**Compatibility :**

2. how a system can co-exist with another system in the same environment.

2.1 The system should be able to work on both the web and in the application form.

# Chapter 2

## Context Diagram



## System Use-case Diagram with description

## 

|  |  |  |
| --- | --- | --- |
| Banking management system: create an account | | |
| Actor | Customer |
| Description | Customer must create an account (User name, Password, and phone number) for the first time, then they must login in each time via their username and password, then they shall receive one time password to their phone number to enter it in login page to verified the user. |
| Data | Username, password, phone number. |
| Stimulus | Press login button. |
| Response | Login in app or wep site . |
| Comments | The customer providers should create a unique username and password with minimum 8 characters' password including at least one capital letter, one small letter and one symbol. |

|  |  |
| --- | --- |
| Banking management system: transfer money | |
| Actor | Customer |
| Description | The user shall be able to transfer money to other bank accounts. |
| Data | Beneficiary Name, Amount. |
| Stimulus | Choose from a list if the account is in a local or international bank. |
| Response | A transfer message will appear with the amount and date. |
| Comments | - |

|  |  |
| --- | --- |
| Banking management system : Check balance & transaction  transactions | |
| Customer. | **Actors** |
| User can see their transaction report and balance enquiry too. | **Description** |
| Username, password to login. | **Data** |
| Pass view balance enquiry button | **stimulus** |
| Show the available balance in the account. | **response** |
| - | **comment** |

|  |  |
| --- | --- |
| Banking management system: check his cheque  service | |
| Customer. | **Actors** |
| The customer can check his cheque service. | **Description** |
| Username, password to login, received code. | **Data** |
| Go to services list – cheque service | **stimulus** |
| Show available cheque information | **response** |
| - | **comment** |

|  |  |
| --- | --- |
| Banking management system: Pay bills | |
| Customer. | **Actors** |
| customer shall be able to pay bills. | **Description** |
| Username, password to login. | **Data** |
| Go to payment – invoices – active invoices | **stimulus** |
| A message that the bills have been paid will appear | **response** |
| - | **comment** |

## Use case scenario (MSS +Extentions)

### Use case 1

The user should create an account in the system to log in into application at any time.

### Use case 1: create an account

### GOAL: To create an account in application/ web

### ACTORS: Customer, system

### Precondition: click create an account button

Main Success Scenario:

### The Customer fills all the necessary information

### The System checks the inserted information for any mistakes

### The System creates an account for the customer

1. The System will store customer login information

Postcondition: System confirms creating an account for the customer.

### Extensions:

### 1a-The Customer fills incorrect information

### 3a- The system fails to create an account

1. The customer doesn’t receive a verification message.

### Use case 2

The user shall be able to transfer money to other bank accounts.

Use case 2: transfer money.

GOAL: To transfer money to other bank accounts.

### ACTORS: Customer

### Precondition: Choose from a list if the account is in a local or international bank.

Main Success Scenario:

1. Customer should visit app
2. Customer should login to the account
3. Customer should select transfer money
4. Customer must choose whether the transfer is local or international bank.

### Postcondition: A transfer message will appear with the amount and date.

### Extensions:

1 insufficient balance for transfer money

### Use case 3: check balance and transactions

**GOAL**: Customer can see their transaction report and balance enquiry too

**ACTORS**: Customer

**Precondition**: customer should provide will login

**Main Success Scenario:**

1. Customer should visit app
2. Customer should login to the account
3. Customer should select account
4. System should check the account balance and if it is valid send report transaction

**Extensions:**

4a. System does not find available information

### Use case 4

customer shall be able to pay bills

Use case 4: pay bills.

### GOAL: the customer pay bills.

### ACTORS: Customer, system

### Precondition: login to the account, check sufficient balance

Main Success Scenario:

### The customer should Visit the app/web

1. The customer should Login into the account
2. The customer should Add the bill to the account.
3. The system should have check sufficient balance.

### The costumer pay the bill.

Postcondition: the bills are paid.

### Extensions:

### 3a- The system fails to add the bill because incorrect information

4a- insufficient balance for payment.

### Use case 5: check his cheque service.

**GOAL**: The customer can check his cheque service.

**ACTORS**: Customer , system

### Precondition: The customer is registered to the system.

**Main Success Scenario:**

### Customer login to their account.

1. customer will ask the system for cheque status
2. the system will provide the cheque status to the customer

**Extensions:**

3a. System does not find available information

# Chapter 3

## System Class Diagram Diagram Description automatically generated

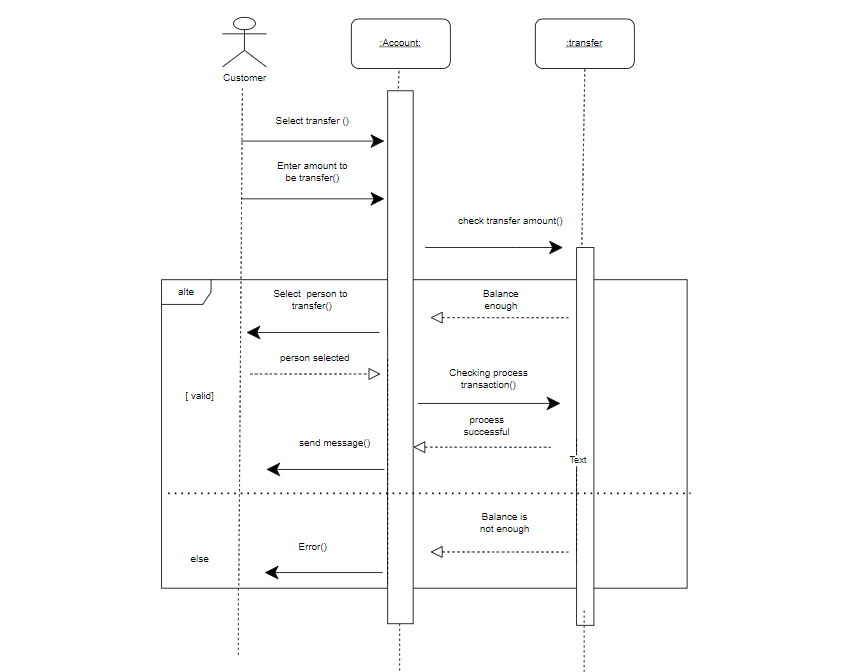
## Sequence Diagrams

### Sequence diagram for [create account]

### 



### Sequence diagram for [transfer money]



### Sequence diagram for [check balance and transactions]

### Chart Description automatically generatedSequence diagram for [check his cheque service.]

### Diagram Description automatically generatedSequence diagram for [pay bills]

# Chapter 4

## 

## Activity Diagram

### Diagram Description automatically generatedActivity Diagram for [check balance and transactions]

### Activity Diagram for [transfer money]

### 

### 

### Activity Diagram for [create account]

Diagram

Description automatically generated

### Activity Diagram for [check his cheque service.]

Diagram

Description automatically generated

### 

### Diagram Description automatically generatedActivity Diagram for [pays bills]

## 

## State Diagram

### State Diagram for [pay bills ]

***Diagram

Description automatically generated***

# Chapter 5

## Architecture Diagram

Diagram

Description automatically generated

The particular reason to used (client–server architecture) in our project , is that we can use it when data in a shared database has to be accessed from range of location because service can be a replicated may also be used for in the load on the system is variable.

This model server can be distributed across a network, In addition to that it can be available to all clients and does not need to be implemented by all services, also In a client-server network, the data is well protected due to it is centralized architecture.