

E-Banking System

**Software Requirement Specification**

– Danang, May 2023 –

Contents

[I. Record of Changes: 2](#_Toc135856982)

[II. Software Requirement Specification 2](#_Toc135856983)

[Product Overview 2](#_Toc135856984)

[1. User Requirements 3](#_Toc135856985)

[1.1. Actors 3](#_Toc135856986)

[1.2. Use cases 4](#_Toc135856987)

[1.2.1. Diagram(s) 4](#_Toc135856988)

[1.2.2. Descriptions 6](#_Toc135856989)

[2. Functional Requirements 6](#_Toc135856990)

[2.1. System Functional Overview 6](#_Toc135856991)

[2.1.1. Screens Flow 6](#_Toc135856992)

[2.1.2. Screen Descriptions 7](#_Toc135856993)

[2.1.3. Screen Authorization 7](#_Toc135856994)

[2.1.4. API Functions 7](#_Toc135856995)

[2.1.5. Non Screen Function 8](#_Toc135856996)

[2.1.6. Entity Relationship Diagram 8](#_Toc135856997)

[2.2. Use case specification 10](#_Toc135856998)

# Record of Changes:

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **A,M,D** | **In charge** | **Change description** |
| 23/May | A | CuongTNQ | Product overview |
| 23/May | A | All | Use case, System functional overview |
| 24/May | A | NhatTQ | Entity relationship diagram |
| 24/May | A | DungTA, TuHA | Use case specification |

A - Added M - Modified D - Deleted

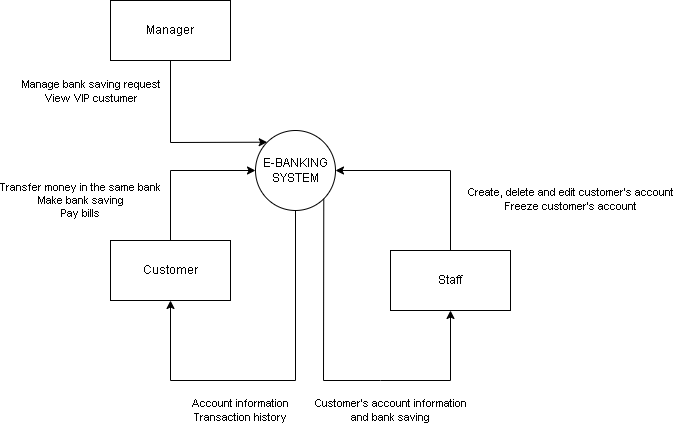
# Software Requirement Specification

### Product Overview

An eBanking Transaction System is a software application that enables customers to access and manage their bank accounts, perform various types of transactions, and receive real-time updates on their account balances and transaction history, with the assistance of staff and administrators behind the scene. eBanking is a critical component of modern banking systems, as it provides customers with a convenient and secure way to perform financial transactions.

The eBanking system is responsible for processing and managing a wide range of customer transactions, including transfers, bank savings and pay bills. It typically includes various features, such as online banking portals, which enable customers to perform various financial transactions remotely.

The goal of an eBanking Transaction System is to provide customers with a convenient and reliable platform for managing their bank accounts, performing financial transactions, and receiving real-time account updates. This can result in increased customer satisfaction, loyalty, and revenue for banks and other financial institutions. As a result, the development of an eBanking system is an important strategic initiative for many banks and financial institutions, and is expected to continue to grow in importance in the coming years.



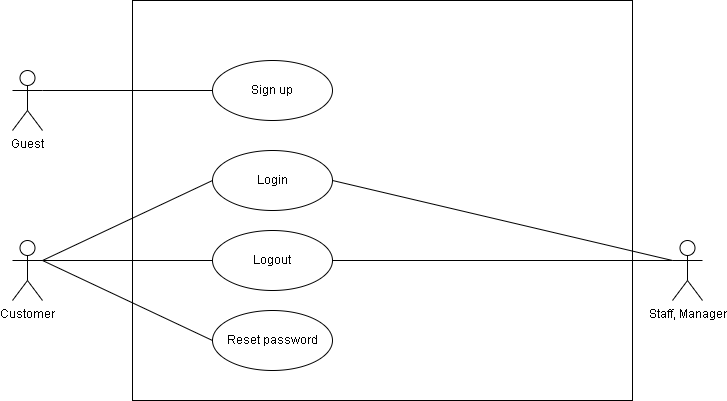
## 1. User Requirements

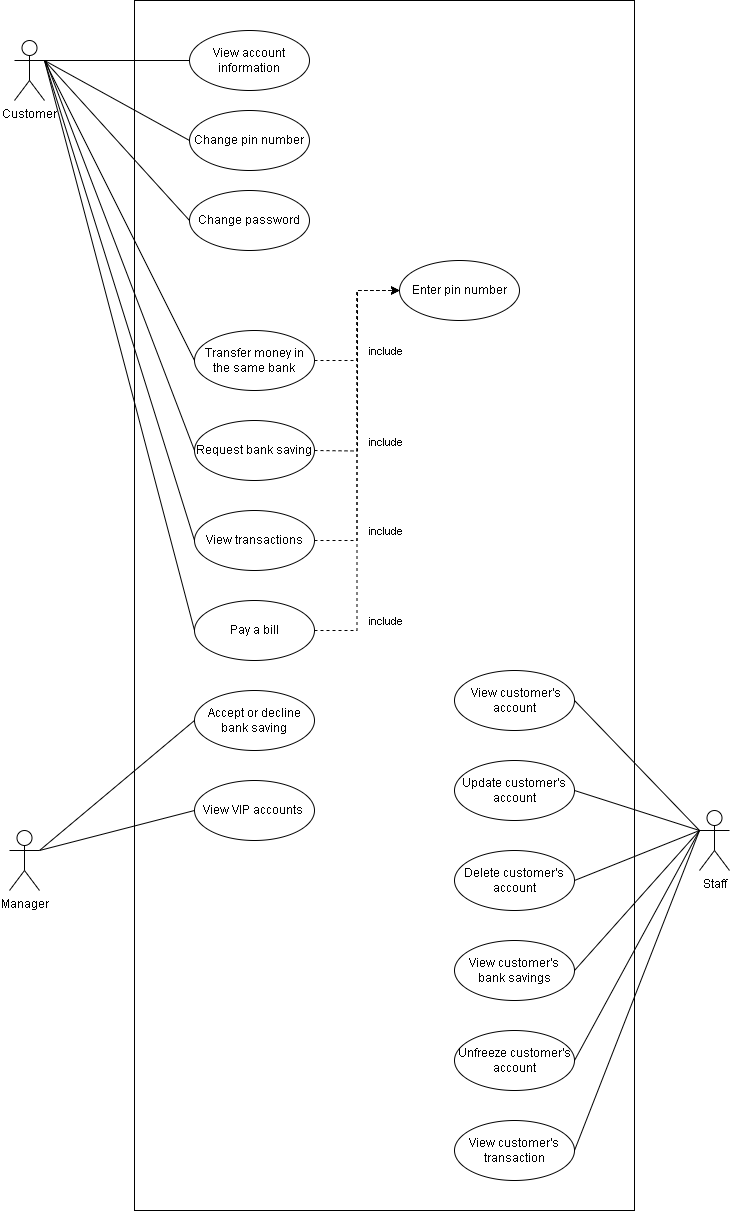
### 1.1. Actors

|  |  |  |
| --- | --- | --- |
| **#** | **Actor** | **Description** |
| 1 | Guest | Who use the web without signing in |
| 2 | Customer | Who use the web after signing in |
| 3 | Staff | Who use the web to monitor and help customer do their tasks |
| 4 | Manager | Who monitor the customer’s more delicate information |

### 1.2. Use cases

#### 1.2.1. Diagram(s)





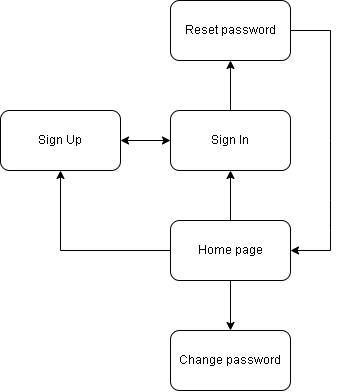
#### 1.2.2. Descriptions

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Use Case** | **Actors** | **Use Case Description** |
| UC-01 | Sign up | Guest | Sign up an account to access the system |
| UC-02 | Sign-in | User | Sign-in an account to access the system |
| UC-03 | Logout | User | User log out from the website |
| UC-04 | Change password | User | Let user change password |
| UC-05 | Forgot password | User | Let user reset their password |

## 2. Functional Requirements

### 2.1. System Functional Overview

#### 2.1.1. Screens Flow



#### 2.1.2. Screen Descriptions

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Feature** | **Screen** | **Description** |
| 1 | Authentication | Home page | An overview page for guests. |
| 2 | Authentication | Sign In | Allow guests to login into the system. |
| 3 | Authentication | Sign Up | Allow guests to create an account. |
| 4 | Authentication | Forgot password | Helps users to send a request to reset their password. |
| 5 | Profile | Change password | Helps users to create a new password. |
| 6 | Flashcard | Home Page | An overview page for users. |

#### 2.1.3. Screen Authorization

| **Screen** | **User** | | | **Guest** |
| --- | --- | --- | --- | --- |
| **Customer** | **Staff** | **Manager** |
| Landing page |  |  |  | X |
| Sign In |  |  |  | X |
| Sign Up |  |  |  | X |
| Forgot password | X | X | X |  |
| Change password | X | X | X |  |
| Home page | X | X | X |  |

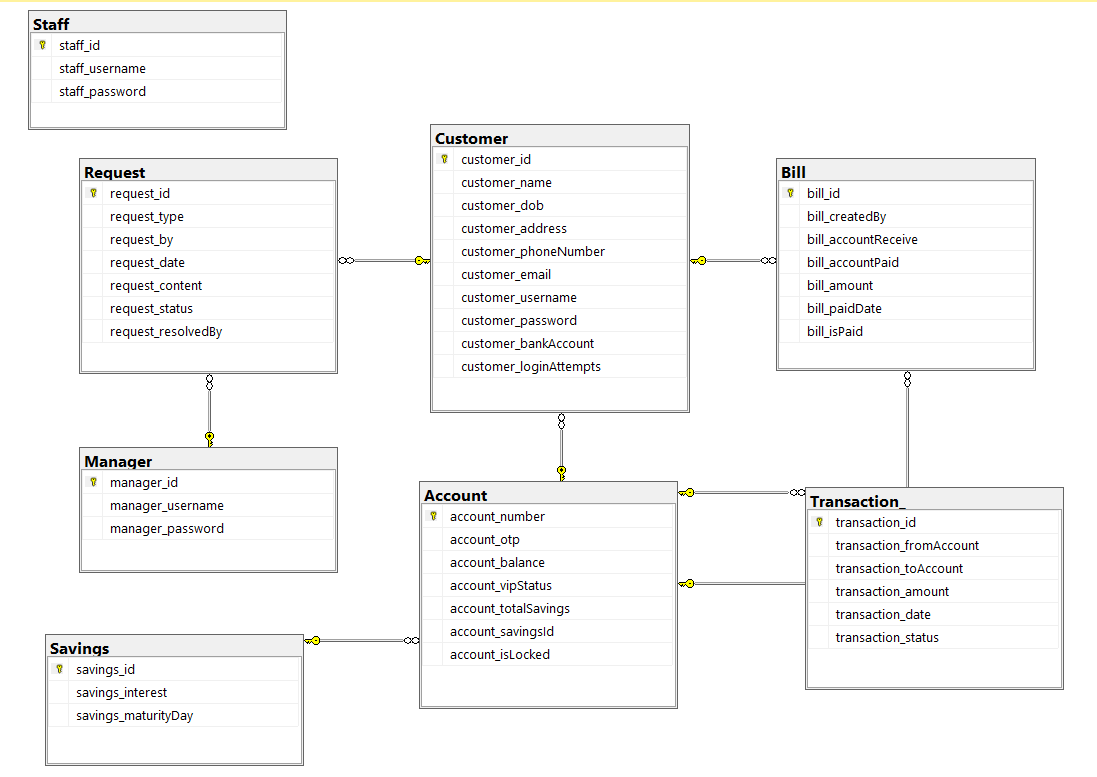
#### 2.1.4. API Functions

|  |  |  |
| --- | --- | --- |
| **API ID** | **Screen** | **Description** |
| API-001 | Sign up with email, password | Sign up |
| API-002 | Sign in with email, password | Sign in |

#### 2.1.5. Non Screen Function

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Feature** | ***System function*** | **Description** |
| 1 | Lock account | Auto lock account | Auto lock when input the wrong password 4 times |

#### 2.1.6. Entity Relationship Diagram



**Entities Description**

|  |  |  |
| --- | --- | --- |
| **#** | **Entity** | **Description** |
| 1 | Customer | Store customer’s information  Include attributes:   * PK: customer\_id * FK: customer\_bankAccount * customer\_name * customer\_dob * customer\_address * customer\_phoneNumber * customer\_email * customer\_username * customer\_password * customer\_loginAttempts |
| 2 | Transaction | Store transactions’ information  Include attributes:   * PK: transaction\_id * FK: transaction\_fromAccount,transaction\_toAccount * transaction\_amount * transaction\_date * transaction\_status |
| 3 | Bill | Store bills’ information  Include attributes:   * PK: bill\_id * FK: bill\_createdBy, bill\_accountReceive * bill\_accountPaid * bill\_amount * bill\_isPaid * bill\_paidDate |
| 4 | Account | Store customer’s account’s information  Include attributes:   * PK: account\_number * FK: account\_savingsId * account\_otp * account\_balance * account\_vipStatus * account\_totalSavings * account\_isLocked |
| 5 | Request | Store requests information  Include attributes:   * PK: request\_id * FK: request\_by, request\_resolvedBy * request\_type * request\_date * request\_content * request\_status |
| 6 | Manage | Store manager’s information  Include attributes:   * PK: manager\_id * manager\_username * manager\_password |
| 7 | Saving | Store customer’s bank saving information  Include attributes:   * PK: savings\_id * savings\_interest * savings\_maturityDay |
| 8 | Staff | Store staff’s information  Include attributes:   * PK: staff\_id * staff\_username * staff\_password |

### 2.2. Use case specification

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | **UC-01** | Use Case Name: | **Sign Up** |
| Created By: | TuHA | Date Created: | 2023/05/22 |
| Primary Actor: | Guest | Secondary Actor: |  |
| Description: | User want to sign up account on the system | | |
| Priority: | Must have | | |
| Trigger: | Guest click sign up button to create a new account on the system | | |
| Preconditions: | PRE-1.1 Guest has a valid email PRE-1.2 Guest has a valid password PRE-1.3 Guest has a valid name  PRE-1.4 Guest must be more than 18 years old | | |
| Post - Conditions: | When normal flow is completed, a new account is created on the system with given information, new user should be redirect to home page | | |
| Basic Flow: | 1. Guests visit home page website from browser 2. Waiting for the home page to be loaded 3. Guest click on the Sign Up button to register new account 4. Waiting for the register page to be loaded   5. Guest fill in email, password, confirm password,and name on the displayed register form   1. Guests click on the Signup button 2. New user is redirected to the home page | | |
| Exception Flow: | **Users enter invalid value in register form**   * System shows error messages on each invalid field and prevents guests from submitting the register form | | |
| Business rules: | None | | |
| Mockup | Không có mô tả.Không có mô tả. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | **UC-02** | Use Case Name: | **Sign In** |
| Created By: | NhatTQ | Date Created: | 2022/05/22 |
| Primary Actor: | Guest | Secondary Actor: |  |
| Description: | User want to sign in account on the system | | |
| Priority: | Must have | | |
| Trigger: | A guest clicks on the sign in button | | |
| Preconditions: | PRE-1. The guest has not logged in the system | | |
| Post - Conditions: | POST-1 The user sign in the system  POST-2. The user is provided access to his allowed systems. POST-3 The user accesses the system, makes a login request and is directed to the login page. | | |
| Basic Flow: | 1. Access the home page 2. Click on the sign in link to sign in to website 3. Fill in form with email and password 4. Enter or click on the sign in button to sign in website | | |
| Exception Flow: | **Account does not exist or the password is incorrect**   * System shows error messages | | |
| Business rules: | None | | |
| Mockup | Không có mô tả. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | **UC-03** | Use Case Name: | **Logout** |
| Created By: | HungND | Date Created: | 2023/05/22 |
| Primary Actor: | User | Secondary Actor: |  |
| Description: | The user logs out and is now determined as Guest. The guest is provided access to only see the homepage | | |
| Priority: | Must have | | |
| Trigger: | User click the button logout to logout account out of the system | | |
| Preconditions: | PRE-1. The user has signed in the system | | |
| Post - Conditions: | POST-1. The user logged out the system | | |
| Basic Flow: | 1. Go to the homepage 2. Click logout 3. System redirect the user to the home page | | |
| Alternative Flow: | N/A | | |
| Exception Flow: | N/A | | |
| Business rules: | None | | |
| Mockup | Không có mô tả. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | **UC-04** | Use Case Name: | **Change password** |
| Created By: | DungTA | Date Created: | 2023/05/23 |
| Primary Actor: | User | Secondary  Actor: |  |
| Description: | User want to change password for their account | | |
| Priority: | Must have | | |
| Trigger: | A user goes to the homepage, clicks the Change password button and fills all the required fields. After that, the user clicks change password button and now the password was changed | | |
| Preconditions: | PRE-1. User is signed into system | | |
| Post - Conditions: | POST-1. The user’s password is changed. | | |
| Basic Flow: | 1. User go to the homepage 2. User clicks the Change password button. 3. User fills all the required fields. 4. User clicks Change password button.(Cô yêu cầu đổi password cần 1 mã otp gửi đến mail, chắc để ô nhập otp dưới password và nút gửi mail bên cạnh) | | |
| Alternative Flow: | N/A | | |
| Exception Flow: | **Invalid input form**  - The user does not provide all required fields.  **New password and confirm password do not match**  - System shows an error message | | |
| Business rules: | None | | |
| Mockup | Không có mô tả. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | **UC-05** | Use Case Name: | **Forgot password** |
| Created By: | CuongTNQ | Date Created: | 2023/05/23 |
| Primary Actor: | User | Secondary Actor: |  |
| Description: | Users can reset their account password | | |
| Priority: | Must have | | |
| Trigger: | User click the send button | | |
| Preconditions: | PRE-4.1 Users has a valid email account which has been registered to the website | | |
| Post - Conditions: | POST-4.1. System sends an email which contains the randomly generated password to the provided email. | | |
| Basic Flow: | 1. Fill in the required email field. 2. Focus on the email field and press enter or click on the “Reset Password” button below the field. 3. System should show a notification to inform users to check their email. 4. System randomly generate and set a password for customer’s account 5. System send an email containing the password and advice user to change their password. | | |
| Alternative Flow: | N/A | | |
| Exception Flow: | N/A | | |
| Business rules: | None | | |
| Mockup | Không có mô tả. | | |