**Software Design Document**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Project Name** | **Version** | **Author** | **Date** | **Comment** |
|
| **Internet Banking System** | **1.0** | **Alaa Gamal** | **5-May-19** | **Initiate SDD** |
| **Internet Banking System** | **1.1** | **Alaa Gamal** | **8-May-19** | **Adding HLD , DD, DFD, ERD** |

Contents

[**1. Introduction** 3](#_Toc8128730)

[1.1. Purpose 3](#_Toc8128731)

[1.2 Scope 3](#_Toc8128732)

[1.3 Overview 3](#_Toc8128733)

[**2. System Overview** 3](#_Toc8128734)

[**3. System Architecture Design** 3](#_Toc8128735)

[**4. Data Design** 3](#_Toc8128736)

[4.2 Data Dictionary 3](#_Toc8128737)

[**5. Component Design** 4](#_Toc8128738)

[**6. Graphical User Interface Design** 4](#_Toc8128739)

[6.1 Overview of User Interface 4](#_Toc8128740)

[6.2 Screen Images 4](#_Toc8128741)

# **1. Introduction**

## 1.1. Purpose

Software design is a process by which the software requirements are translated into a representation of software components, interfaces, and data necessary for the implementation phase. It is the primary reference for code development and, therefore, it must contain all the information required by a programmer to write code.

## 1.2 Scope

This software design document describes the architecture and system design of internet Banking system, it shows how the software system will be structured to satisfy the requirements.

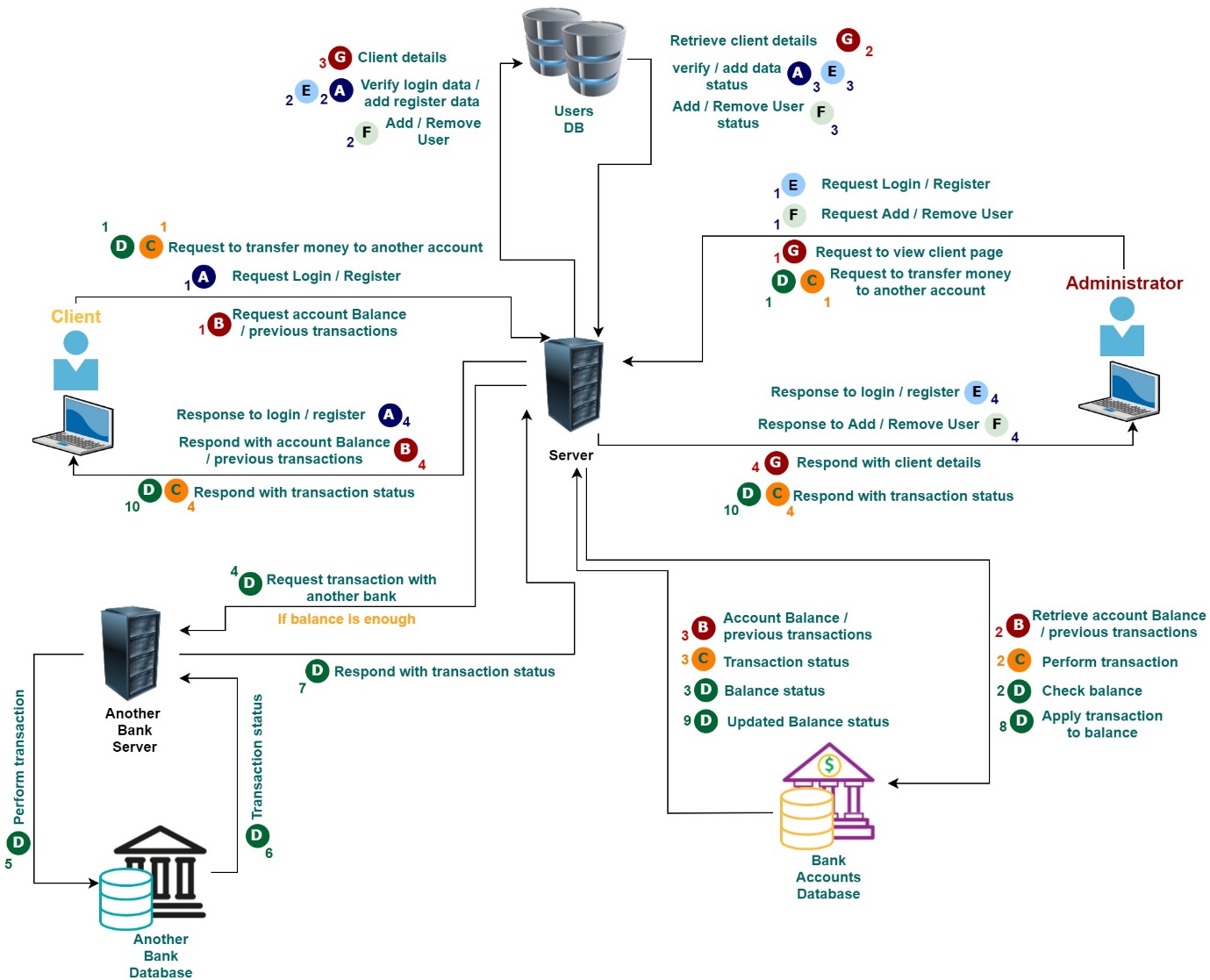
## 1.3 Overview

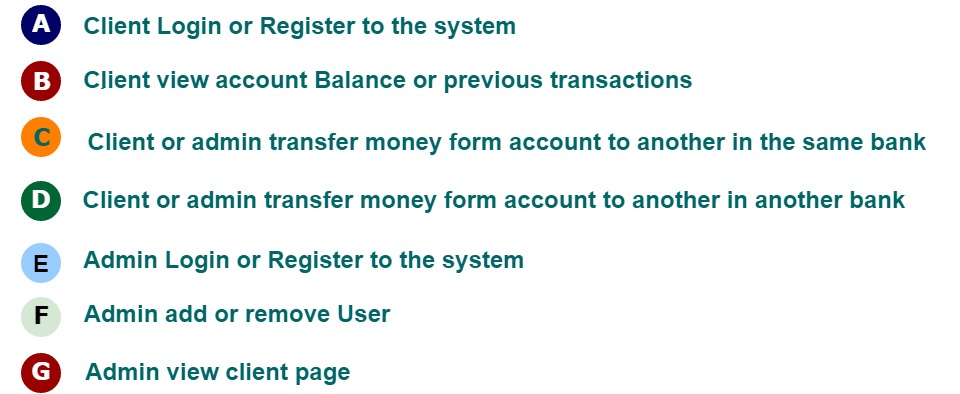
The SDD is performed in two stages. The first is a preliminary design in which the overall system architecture and data architecture is defined. In the second stage, i.e. the detailed design stage, more detailed data structures are defined and algorithms are developed for the defined architecture.

# **2. System Overview**

Internet Banking System gives to our clients some functionalities as login, registration, check their balance, check previous transaction and transfer from its account either to inter account or to external accounts. `

# **3. System Architecture Design**

The High level design describes what components will be in the system, how they interact with one another and what actions users can perform using the system.



## 3.1. Components:

The system consists of three main components as shown in the table below.

The database is split into two databases to separate bank accounts details from web application users’ data due to security concerns.

|  |  |
| --- | --- |
| Design ID | Description |
| Bank\_SYS\_HLD\_comp\_001 | The system consists of **User Database**, this database contains client and administrators data as users for the web application |
| Bank\_SYS\_HLD\_comp\_002 | The system consists of **Bank accounts database**, this database contains clients' accounts data including account number, balance, transactions, etc. |
| Bank\_SYS\_HLD\_comp\_003 | The system consists of **Web Server**, this server acts as a middle layer between the user and the database. It receives requests, fetches data and returns response to user to be displayed in the web application. It also sends requests to other banks servers to perform transactions with other accounts. |

## 3.2. Allowed Processes:

### **3.2.1. Client processes:**

The following requirements describes what processes client can perform on the system.

|  |  |
| --- | --- |
| Design ID | Description |
| Bank\_SYS\_HLD\_client\_001 | Client can register as a new user |
| Bank\_SYS\_HLD\_client\_002 | Client can login to the system |
| Bank\_SYS\_HLD\_client\_003 | Client can view his account balance |
| Bank\_SYS\_HLD\_client\_004 | Client can view his account previous transactions |
| Bank\_SYS\_HLD\_client\_005 | Client can transfer money from his account to another account in the same bank |
| Bank\_SYS\_HLD\_client\_006 | Client can transfer money from his account to another account in another bank |

### **3.2.2. Admin processes:**

The following requirements describes what processes administrator can perform on the system.

|  |  |
| --- | --- |
| Design ID | Description |
| Bank\_SYS\_HLD\_admin\_001 | Admin can register as a new user |
| Bank\_SYS\_HLD\_admin\_002 | Admin can login to the system |
| Bank\_SYS\_HLD\_admin\_003 | Admin can add a new client user |
| Bank\_SYS\_HLD\_admin\_004 | Admin can remove any client user |
| Bank\_SYS\_HLD\_admin\_005 | Admin can transfer money from any client account to another account in the same bank |
| Bank\_SYS\_HLD\_admin\_006 | Admin can transfer money from any client account to another account in another bank |

# **4. Data Design**

Internet Banking System is designed to allow our clients to manage and use his/her account/s and an admin to manage and control all bank accounts, as shown the following diagram illustrate the sequence and how the flow through our system.

## 4.1. Flow diagram of actors and their responsibilities

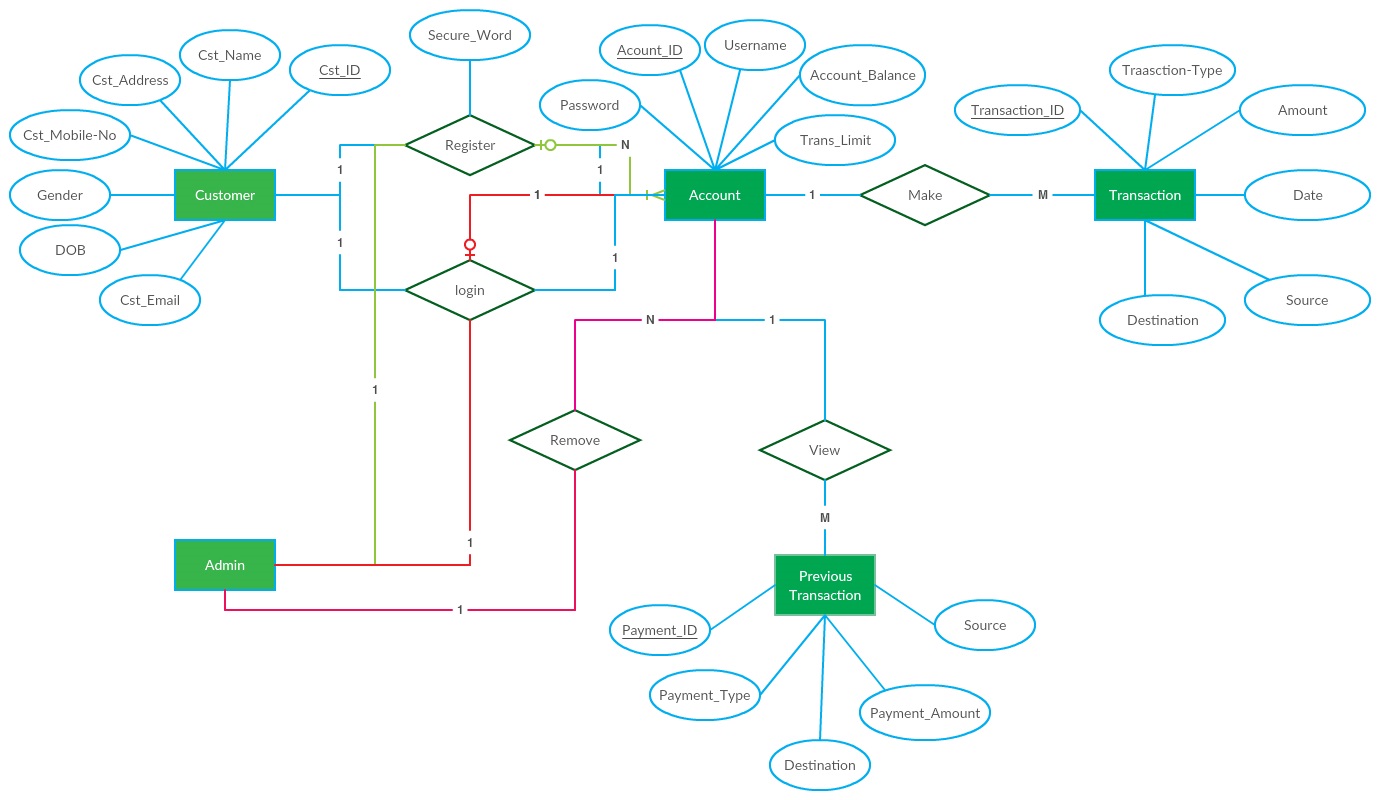
## 4.1.1. Admin

|  |  |
| --- | --- |
| Design ID | Description |
| Bank\_Sys\_DM\_ADMIN\_001 | Admin shall be able to login to his/her account using valid user name and valid password then a verification request is sent to the customer data base and a confirmation response is sent back to Admin login page, if his/her data is exist then redirecting to the main page. |
| Bank\_Sys\_DM\_ADMIN\_002 | Admin shall be able to register a new account using valid data then a storing request is sent to the customer data base and a confirmation response is sent back to Admin registration page without any verification code then redirecting to the main page |
| Bank\_Sys\_DM\_ADMIN\_003 | Admin shall be able to remove any account using customer ID for the targeted account then a removing request is sent to the customer data base and a confirmation response is sent back to Admin page |
| Bank\_Sys\_DM\_ADMIN\_004 | Admin shall be able to Manipulate any account settings; updating customer name and changing password then an updating request is sent to the customer data base and a confirmation response is sent back to Admin page |
| Bank\_Sys\_DM\_ADMIN\_005 | Admin shall be redirected to customer home page after login or register successfully, then the customer home page sends a request to the bank data base and a response is sent back to customer home page which contains all accounts for this customer. |
| Bank\_Sys\_DM\_ADMIN\_006 | Admin shall be able select specific account type then redirecting to the account page. |
| Bank\_Sys\_DM\_ADMIN\_007 | The account page sends a request to the bank data base and a response is sent back to account page which will contain the account balance and two options to select from:   1. Display previous transactions 2. Transfer money |
| Bank\_Sys\_DM\_ADMIN\_008 | The admin shall be able to select display previous transactions then a request is sent to the bank data base and a response is sent back to account page which will contain the account all previous transactions. |
| Bank\_Sys\_DM\_ADMIN\_009 | The admin shall be able to select transfer money; then selecting targeted bank after that enter the amount of money to transfer and finally the transfer is done with other banks and a request is sent to our bank data base and a response is sent back to account page to confirm the update of account balance. |

## 4.1.2. Customer

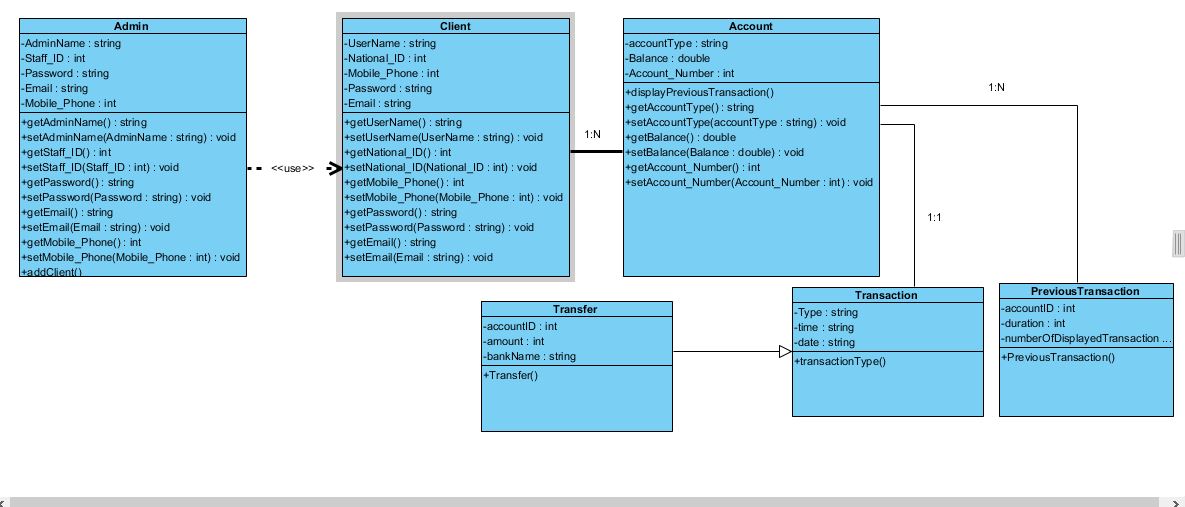
|  |  |
| --- | --- |
| Design Requirement ID | Description |
| Bank\_Sys\_DM\_CST\_001 | Customer shall be able to login to his/her account using valid user name and valid password then a verification request is sent to the customer data base and a confirmation response is sent back to Customer login page, if his/her data is exist then redirecting to the main page. |
| Bank\_Sys\_DM\_CST\_002 | Customer shall be able to register a new account using valid data and entering valid verification key then a storing request is sent to the customer data base and a confirmation response is sent back to Admin registration page without any verification code then redirecting to the customer home page. |
| Bank\_Sys\_DM\_CST\_003 | Customer shall be able to Manipulate his/her accounts settings; updating customer name and changing password then an updating request is sent to the customer data base and a confirmation response is sent back to customer home page |
| Bank\_Sys\_DM\_CST\_004 | Customer shall be redirected to customer home page after login or register successfully, then the customer home page sends a request to the bank data base and a response is sent back to customer home page which contains all accounts for this customer |
| Bank\_Sys\_DM\_CST\_005 | Customer shall be able select specific account type then redirecting to the account page |
| Bank\_Sys\_DM\_CST\_006 | The account page sends a request to the bank data base and a response is sent back to account page which will contain the account balance and two options to select from:   1. Display previous transactions 2. Transfer money |
| Bank\_Sys\_DM\_CST\_007 | The customer shall be able to select display previous transactions then a request is sent to the bank data base and a response is sent back to account page which will contain the account all previous transactions. |
| Bank\_Sys\_DM\_CST\_008 | The customer shall be able to select transfer money; then selecting targeted bank after that enter the amount of money to transfer and finally the transfer is done with other banks and a request is sent to our bank data base and a response is sent back to account page to confirm the update of account balance |

## 4.2. The Data model Entity Relationship Diagram description

Internet Banking System is designed to allow our clients to manage and use his/her account/s and an admin to Manage and control all bank accounts, as shown the following ERD identify the system entities and their attributes, the actions which can be executed by those entities and the relationship between them including the degree and cardinality of those relationships

# **5. Component Design**

In this section, we take a closer look at what each component does in a more systematic way.

We use class diagram to describe detailed design of our internet banking system as we have client and admin as a user of the system.

**Client class**:

Our client must have a username, password, national id , mobile phone , email .

That used in Login and register ration

|  |  |
| --- | --- |
| Design ID | Description |
| BANK\_SYS\_DD\_CST\_001 | Client class have username Attribute |
| BANK\_SYS\_DD\_ CST \_002 | Client class have password Attribute |
| BANK\_SYS\_DD\_ CST \_003 | Client class have mobile phone Attribute. |
| BANK\_SYS\_DD\_ CST \_004 | Client class have Email Attribute. |

**Admin class**:

|  |  |
| --- | --- |
| Design ID | Description |
| BANK\_SYS\_DD\_Admin\_001 | Admin class have admin name Attribute |
| BANK\_SYS\_DD\_ Admin \_002 | Admin class have password Attribute |
| BANK\_SYS\_DD\_ Admin \_003 | Admin class have staff id Attribute. |
| BANK\_SYS\_DD\_ Admin \_004 | Admin class have Email Attribute. |
| BANK\_SYS\_DD\_ Admin \_005 | Admin class have mobile phone Attribute. |
| BANK\_SYS\_DD\_ Admin \_006 | Admin can add user |
| BANK\_SYS\_DD\_ Admin \_007 | Admin can remove user |
| BANK\_SYS\_DD\_ Admin \_008 | Admin can make a transection for any user |

**Account class**:

|  |  |
| --- | --- |
| Design ID | Description |
| BANK\_SYS\_DD\_account\_001 | Account class have account type Attribute |
| BANK\_SYS\_DD\_ account \_002 | Account class have balance Attribute |
| BANK\_SYS\_DD\_ account \_003 | Account class have account number Attribute. |

**Transaction class:**

|  |  |
| --- | --- |
| Design ID | Description |
| BANK\_SYS\_DD\_TR\_001 | Transaction class have type of transaction Attribute |
| BANK\_SYS\_DD\_ TR \_002 | Transaction class have time of transaction Attribute |
| BANK\_SYS\_DD\_ TR \_003 | Transaction class have date of transaction Attribute. |

**Transfer class**:

|  |  |
| --- | --- |
| Design ID | Description |
| BANK\_SYS\_DD\_Trans\_001 | Transfer class have account id Attribute |
| BANK\_SYS\_DD\_ Trans \_002 | Transfer class have bank name Attribute |
| BANK\_SYS\_DD\_ Trans \_003 | Transfer class have amount of money Attribute. |

**Previous transaction class**:

|  |  |
| --- | --- |
| Design ID | Description |
| BANK\_SYS\_DD\_PR\_001 | Transfer class have account id Attribute |
| BANK\_SYS\_DD\_ PR \_002 | Transfer class have duration Attribute. |
| BANK\_SYS\_DD\_ PR \_003 | Transfer class have number of displayed transaction Attribute. |

# **6. Graphical User Interface Design**

## 6.1 Overview of User Interface

Describe the functionality of the system from the user’s perspective. Explain how the user will be able to use our system to complete all the expected features and the feedback information that will be displayed for the user.

## 

## 6.2 Screen Images

Display screenshots showing the interface from the user’s perspective. These can be hand drawn or you can use an automated drawing tool. Just make them as accurate as possible. (Graph paper works well.)

**7. REQUIREMENTS MATRIX**

Provide a cross reference that traces components and data structures to the requirements in your SRS document. Use a tabular format to show which system components satisfy each of the functional requirements from the SRS. Refer to the functional requirements by the numbers/codes that you gave them in the SRS.