

E-Banking System

**Software Design Specification**

– Danang, May 2023 –

**Record of changeS**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **A\* M, D** | **In charge** | **Change Description** |
| 07/June | A | NhatTQ | Code Packages, Database Design |
| 08/June | A | All | Code Design |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

\*A - Added M - Modified D - Deleted

**Table of Contents**

[I. Overview 4](#_heading=h.gjdgxs)

[1. Code Packages 4](#_heading=h.30j0zll)

[2. Database Design 4](#_heading=h.1fob9te)

[a. Database Schema 4](#_heading=h.3znysh7)

[b. Table Description 4](#_heading=h.2et92p0)

[II. Code Designs 5](#_heading=h.tyjcwt)

[1. <Feature/Function Name1> 5](#_heading=h.3dy6vkm)

[a. Class Diagram 5](#_heading=h.1t3h5sf)

[b. Class Specifications 5](#_heading=h.4d34og8)

[c. Sequence Diagram(s) 5](#_heading=h.2s8eyo1)

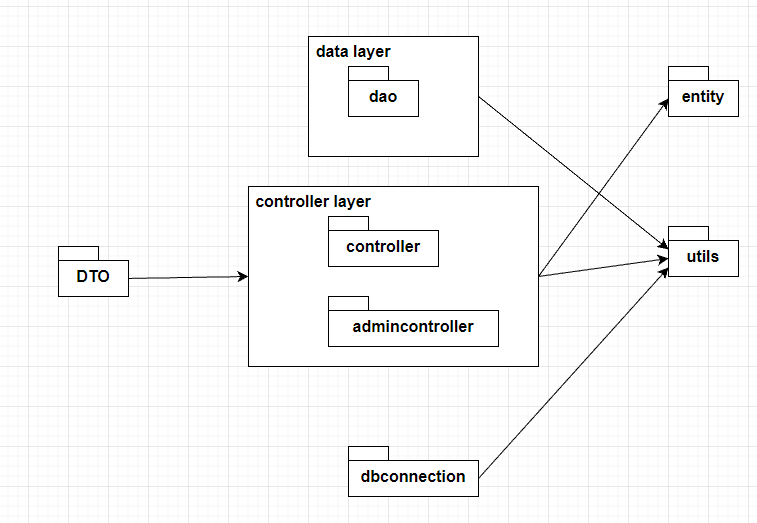
[d. Database queries 6](#_heading=h.17dp8vu)

[2. <Feature/Function Name2> 6](#_heading=h.3rdcrjn)

# I. Overview

## 1. Code Packages

*[Provide the package diagram for each sub-system. The content of this section including the overall package diagram, the explanation, package and class naming conventions in each package. Please see the sample & description table format below (****please note: package names don’t follow Java package naming convention yet****)]*



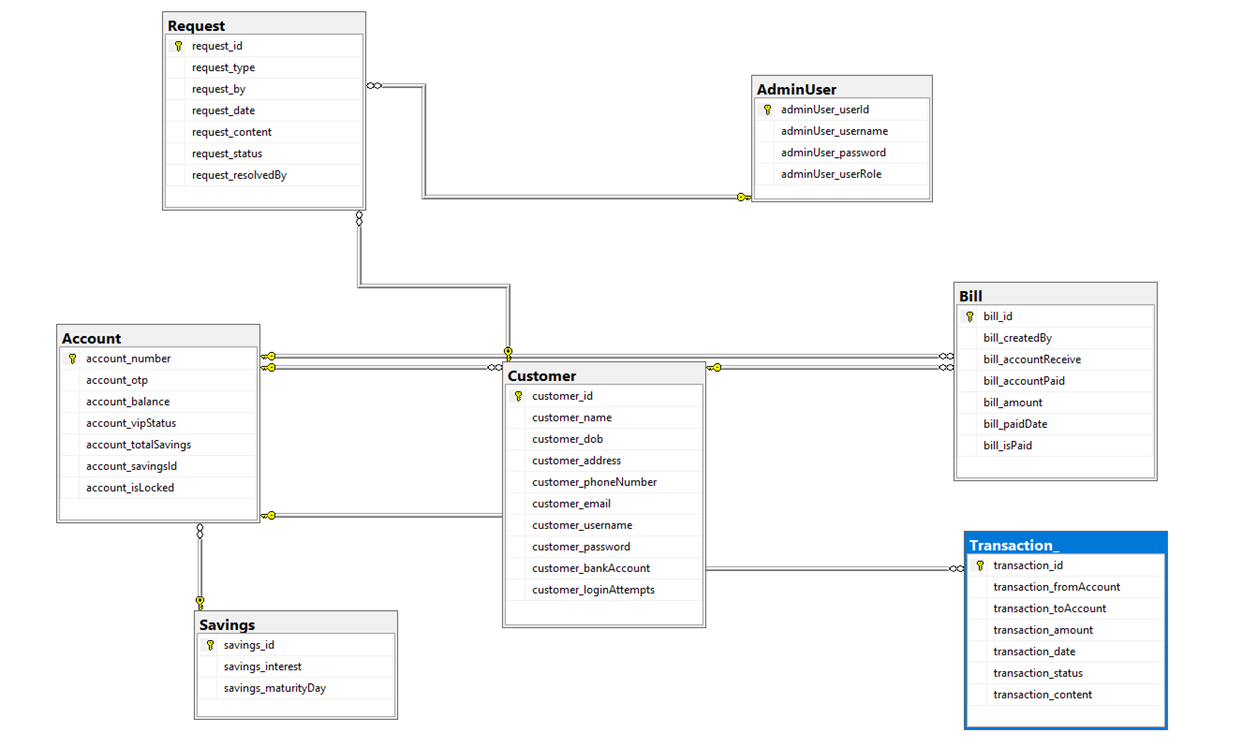
***Package descriptions***

|  |  |  |
| --- | --- | --- |
| **No** | **Package** | **Description** |
| *01* | *dbconnection* | *This package contains database information and connect to database.* |
| *02* | *DTO* | *This package contains data request from Frontend to Backend.* |
| *03* | *dao* | *This package contains the functions which are involved in data processing with the database.* |
| *04* | *controller* | *This package contains functions that perform tasks of customers.* |
| *05* | *Utils* | *This package contains some useful tools (functions) for handling controllers.* |
| *06* | *entity* | *<Description of the package>* |
| *07* | *admincontroller* | *This package contains functions that perform tasks of staffs and managers.* |

## 2. Database Design

### a. Database Schema

*[Provide the tables relationship like example below – following MySQL database naming convention]*



### b. Table Description

|  |  |  |
| --- | --- | --- |
| **#** | **Table** | **Description** |
| 1 | Customer | Store customer’s information |
| 2 | Transaction\_ | Store transactions’ information  Include attributes: |
| 3 | Bill | Store bills’ information |
| 4 | Account | Store customer’s account’s information |
| 5 | Request | Store requests information |
| 7 | Savings | Store customer’s bank saving information |
| 8 | AdminUser | Store staff’s information |

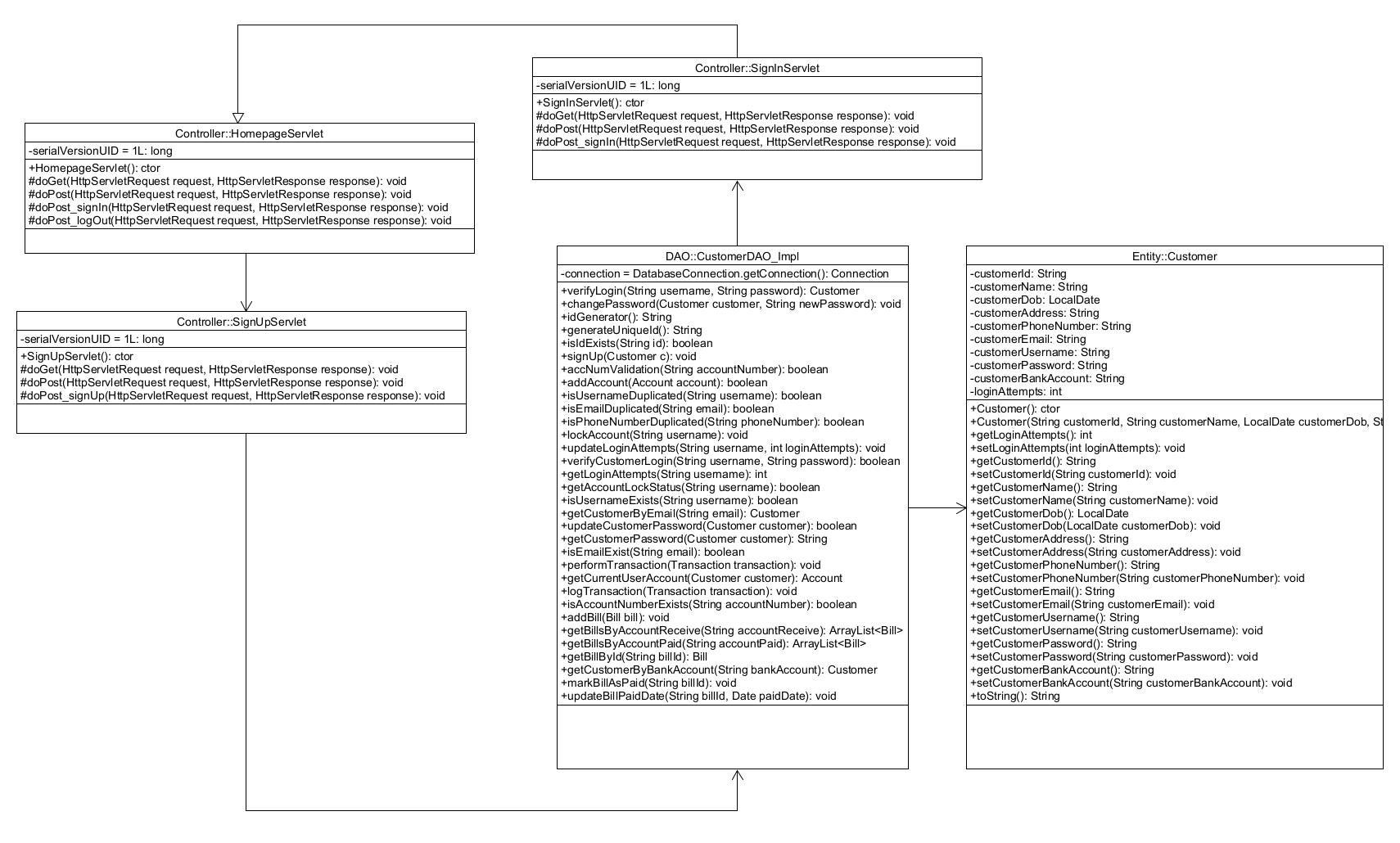
# II. Code Designs

## 1. <Sign up>

*[Provide the detailed design for the function <Feature/Function Name1>. It include Class Diagram, Class Specifications, and Sequence Diagram(s)]*

### a. Class Diagram

*[This part presents the class diagram for the relevant feature]*



### b. Class Specifications

*[Provide the description for each class and the methods in each class, following the table format as below]*

#### SignInServlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when client requests a GET request* |
| *02* | *doPost* | *This method will be called when client requests a POST request* |
| *03* | *doPost\_signIn* | *Allow user to redirect to Sign In page.* |

#### 

#### 

#### 

#### 

#### HomepageServlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when client requests a GET request* |
| *02* | *doPost* | *This method will be called when client requests a POST request* |
| *03* | *doPost\_signIn* | *Allow users to be redirected to the Sign In page.* |
| *04* | *doPost\_logOut* | *Allow users to log out.* |

#### 

#### SignUpServlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when client requests a GET request* |
| *02* | *doPost* | *This method will be called when client requests a POST request* |
| *03* | *doPost\_signUp* | *Allow users to redirect to the Sign Up page.* |

#### 

#### CustomerDAO\_Impl

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *accNumValidation* | *This method will check if account number exist in database* |
| *02* | *isUsernameDuplicated* | *This method will check if a parameter username duplicates with at least 1 username in the database.* |
| *03* | *isEmailDuplicated* | *This method will check if a parameter email duplicates with at least 1 email in the database.* |
| *04* | *isPhoneNumberDuplicated* | *This method will check if a parameter phone number duplicates with at least 1 phone number in the database.* |
| *05* | *addAccount* | *This method will insert an Account object to Account table in database* |
| *06* | *generateUniqueId* | *This method will create a unique ID for account* |
| *07* | *signUp* | *This method will Insert customer’s register information to Customer table in database* |

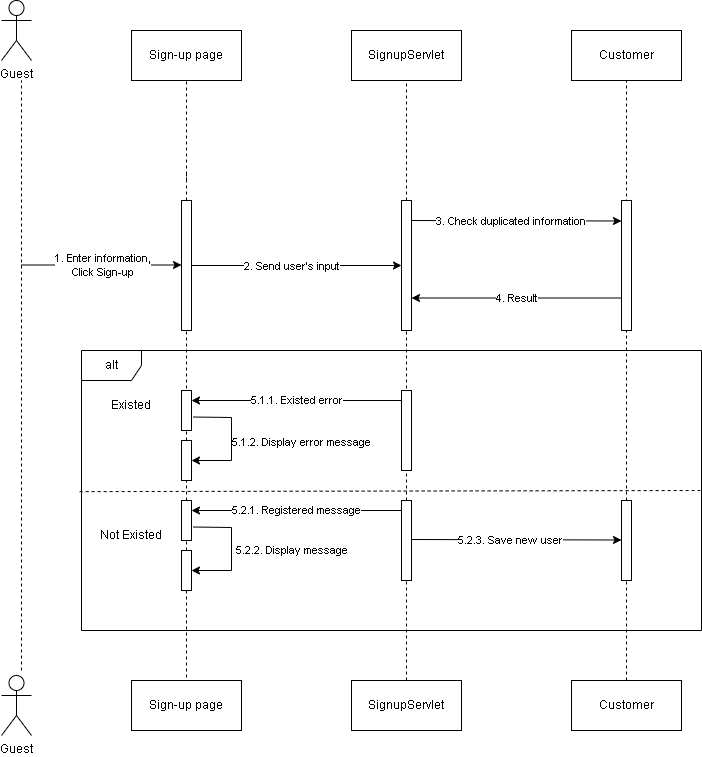
#### Customer

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Getters* | *Retrieve value of an attribute of a Customer object.* |
| *02* | *Setters* | *Set value to an attribute of a Customer object.* |

### c. Sequence Diagram(s)

*[Provide the sequence diagram(s) for the feature, see the sample below]*



### d. Database Queries

*[Provide the detailed SQL (select, insert, update...) which are used in implementing the function/screen]*

|  |
| --- |
| SELECT COUNT(\*) FROM Account WHERE account\_number = ? |

|  |
| --- |
| SELECT COUNT(\*) FROM Customer WHERE customer\_username = ? |

|  |
| --- |
| SELECT COUNT(\*) FROM Customer WHERE customer\_email = ? |

|  |
| --- |
| SELECT COUNT(\*) FROM Customer WHERE customer\_phoneNumber = ? |

|  |
| --- |
| SELECT COUNT(\*) FROM Customer WHERE customer\_id = ? |

|  |
| --- |
| INSERT INTO Account (account\_number, account\_otp) VALUES (?, ?) |

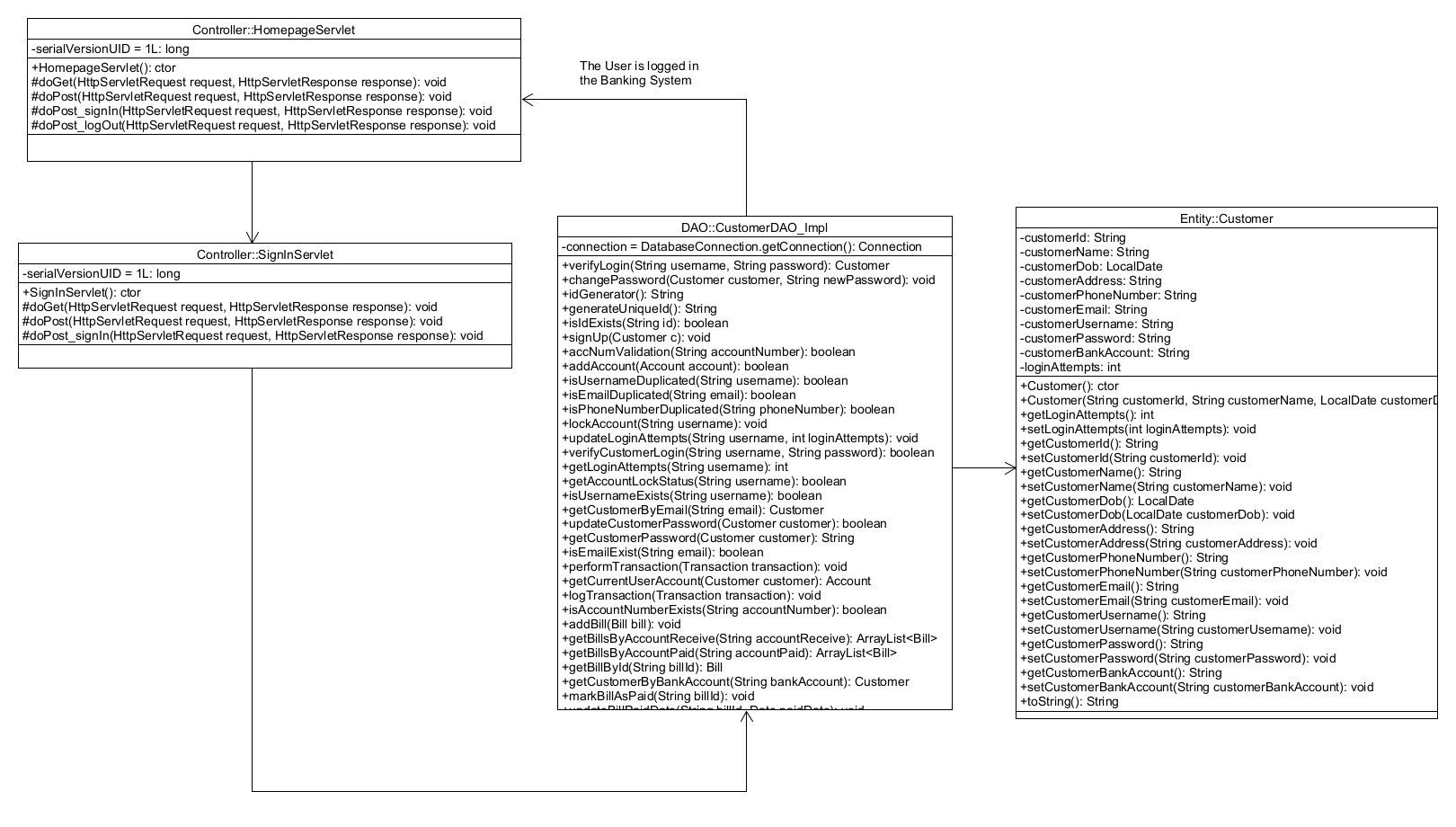
|  |
| --- |
| INSERT INTO Customer (customer\_id, customer\_name, customer\_dob, customer\_address, customer\_phoneNumber, customer\_email, customer\_username, customer\_password, customer\_bankAccount) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?) |

## 2. <Sign in>

*[Provide the detailed design for the function <Feature/Function Name1>. It include Class Diagram, Class Specifications, and Sequence Diagram(s)]*

### a. Class Diagram

*[This part presents the class diagram for the relevant feature]*



### b. Class Specifications

*[Provide the description for each class and the methods in each class, following the table format as below]*

#### Homepage Servlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when client requests a GET request* |
| *02* | *doPost* | *This method will be called when client requests a POST request* |
| *03* | *doPost\_signIn* | *Allow users to redirect to the Sign In page.* |
| *04* | *doPost\_logOut* | *Allow users to log out.* |

#### 

#### SignInServlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when a client requests a GET request.* |
| *02* | *doPost* | *This method will be called when a client requests a POST request.* |
| *03* | *doPost\_signIn* | *Receive User’s Username and Password, then call a method to retrieve the Customer object who has the inputted Username and Password, finally set Customer object to current Session.* |

#### 

#### CustomerDao\_Impl

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *getLoginAttempts* | *This method returns the number of attempts of failed login of this account.* |
| *02* | *getAccountLockStatus* | *This method returns the lock status of an account.* |
| *03* | *isUsernameExists* | *This method returns true if the username exists in the system.* |
| *04* | *verifyCustomerLogin* | *This method checks if the entered password is wrong 4 consecutive times.* |
| *05* | *verifyLogin* | *This method returns true if login successfully.* |
| *06* | *updateLoginAttempts* | *This method updates or reset failed login attempts.* |

#### 

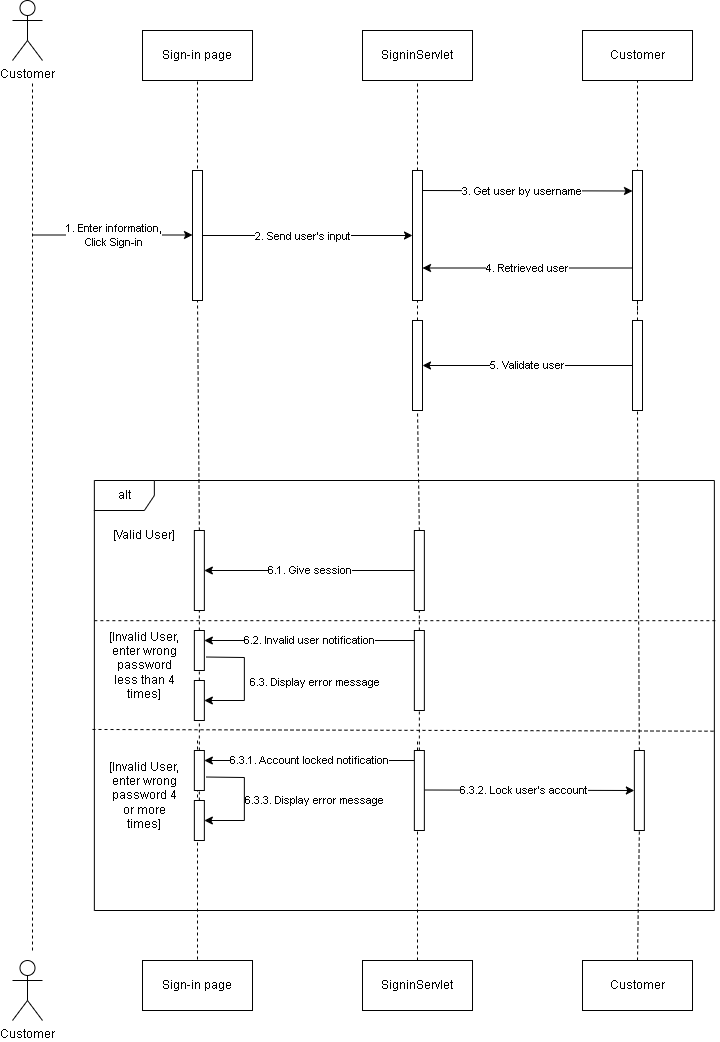
#### Customer

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Getters* | *Retrieve value of an attribute of a Customer object.* |
| *02* | *Setters* | *Set value to an attribute of a Customer object.* |

### c. Sequence Diagram(s)

*[Provide the sequence diagram(s) for the feature, see the sample below]*



### d. Database Queries

*[Provide the detailed SQL (select, insert, update...) which are used in implementing the function/screen]*

|  |
| --- |
| SELECT account\_isLocked FROM Account INNER JOIN Customer ON Account.account\_number = Customer.customer\_bankAccount WHERE Customer.customer\_username = ? |

|  |
| --- |
| SELECT customer\_username FROM Customer WHERE customer\_username = ? |

|  |
| --- |
| SELECT \* FROM Customer WHERE customer\_username = ? |

|  |
| --- |
| SELECT customer\_loginAttempts FROM Customer WHERE customer\_username = ? |

|  |
| --- |
| SELECT \* FROM Customer WHERE customer\_username = ? AND customer\_password = ? |

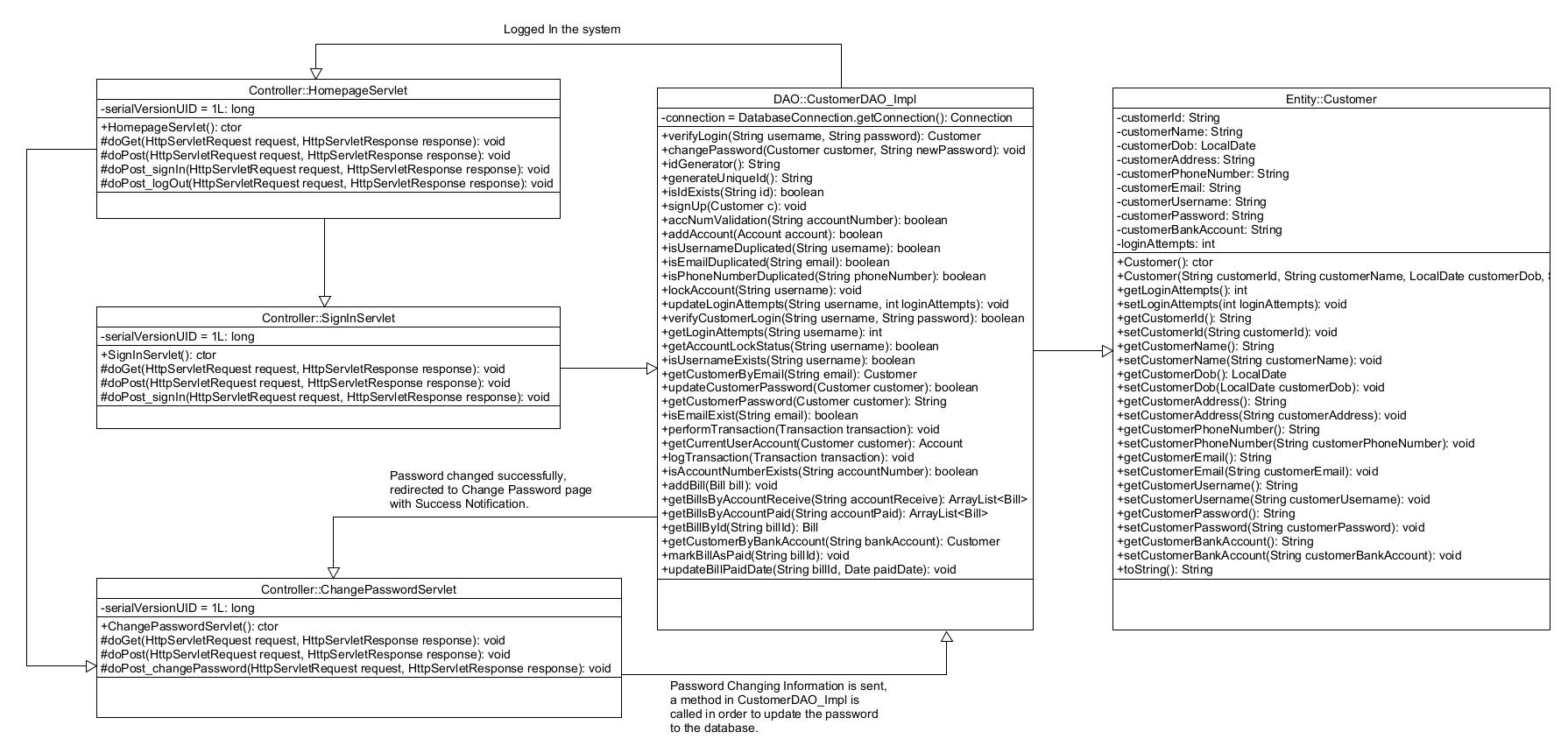
|  |
| --- |
| UPDATE Customer SET customer\_loginAttempts = ? WHERE customer\_username = ? |

## 3. <Change password>

*[Provide the detailed design for the function <Feature/Function Name1>. It include Class Diagram, Class Specifications, and Sequence Diagram(s)]*

### a. Class Diagram

*[This part presents the class diagram for the relevant feature]*



### b. Class Specifications

*[Provide the description for each class and the methods in each class, following the table format as below]*

#### Homepage Servlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when client requests a GET request* |
| *02* | *doPost* | *This method will be called when client requests a POST request* |
| *03* | *doPost\_signIn* | *Allow users to redirect to the Sign In page.* |
| *04* | *doPost\_logOut* | *Allow users to log out.* |

#### 

#### SignInServlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when a client requests a GET request.* |
| *02* | *doPost* | *This method will be called when a client requests a POST request.* |
| *03* | *doPost\_signIn* | *Receive User’s Username and Password, then call a method to retrieve the Customer object who has the inputted Username and Password, finally set Customer object to current Session.* |

#### 

#### ChangePasswordServlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when a client requests a GET request.* |
| *02* | *doPost* | *This method will be called when a client requests a POST request.* |
| *03* | *doPost\_changePassword* | *This method will change password of customer’s account* |

#### CustomerDao\_Impl

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *getCustomerPassword* | *This method will get the customer's password.* |
| *02* | *updateCustomerPassword* | *This method will update the customer's password.* |

#### 

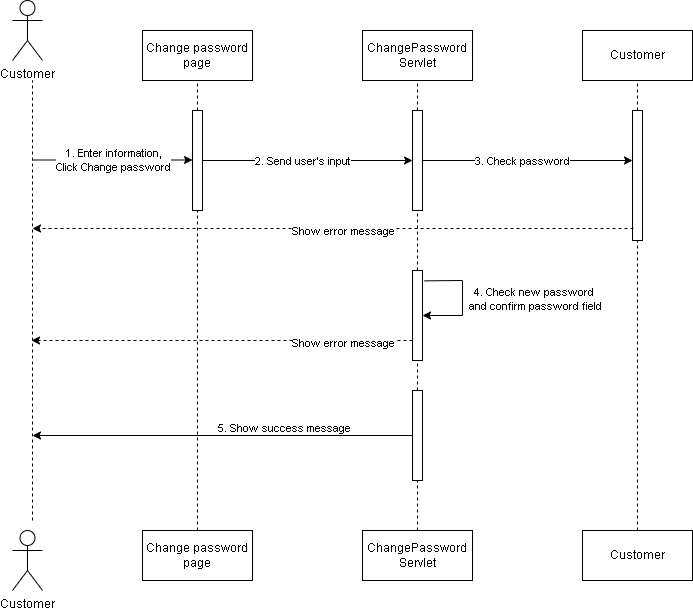
#### Customer

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Getters* | *Retrieve value of an attribute of a Customer object.* |
| *02* | *Setters* | *Set value to an attribute of a Customer object.* |

### c. Sequence Diagram(s)

*[Provide the sequence diagram(s) for the feature, see the sample below]*



### d. Database Queries

*[Provide the detailed SQL (select, insert, update...) which are used in implementing the function/screen]*

|  |
| --- |
| SELECT customer\_password FROM Customer WHERE customer\_id = ? |

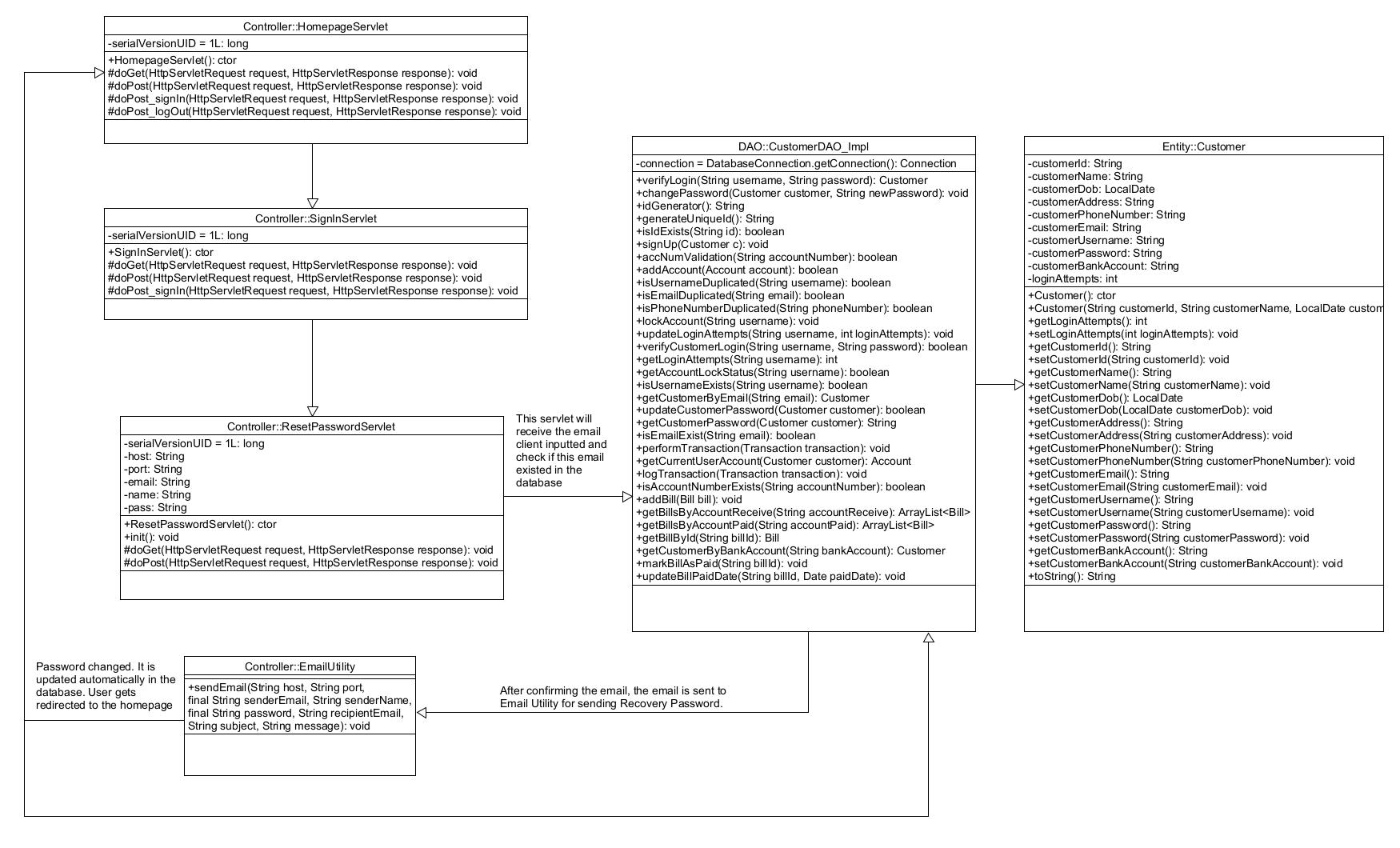
|  |
| --- |
| UPDATE Customer SET customer\_password = ? WHERE customer\_id = ? |

## 4. <Forgot password>

*[Provide the detailed design for the function <Feature/Function Name1>. It include Class Diagram, Class Specifications, and Sequence Diagram(s)]*

### a. Class Diagram

*[This part presents the class diagram for the relevant feature]*



### b. Class Specifications

*[Provide the description for each class and the methods in each class, following the table format as below]S*

#### Homepage Servlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when client requests a GET request* |
| *02* | *doPost* | *This method will be called when client requests a POST request* |
| *03* | *doPost\_signIn* | *Allow users to redirect to the Sign In page.* |
| *04* | *doPost\_logOut* | *Allow users to log out.* |

#### 

#### SignInServlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when a client requests a GET request.* |
| *02* | *doPost* | *This method will be called when a client requests a POST request.* |
| *03* | *doPost\_signIn* | *Receive User’s Username and Password, then call a method to retrieve the Customer object who has the inputted Username and Password, finally set Customer object to current Session.* |

#### 

#### ResetPasswordServlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when client requests a GET request* |
| *02* | *doPost* | *This method will let user reset their password* |

#### EmailUtility

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *sendEmail* | *This method will send email to user* |

#### 

#### CustomerDao\_impl

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *isEmailExist* | *This method will check if email exist* |

#### 

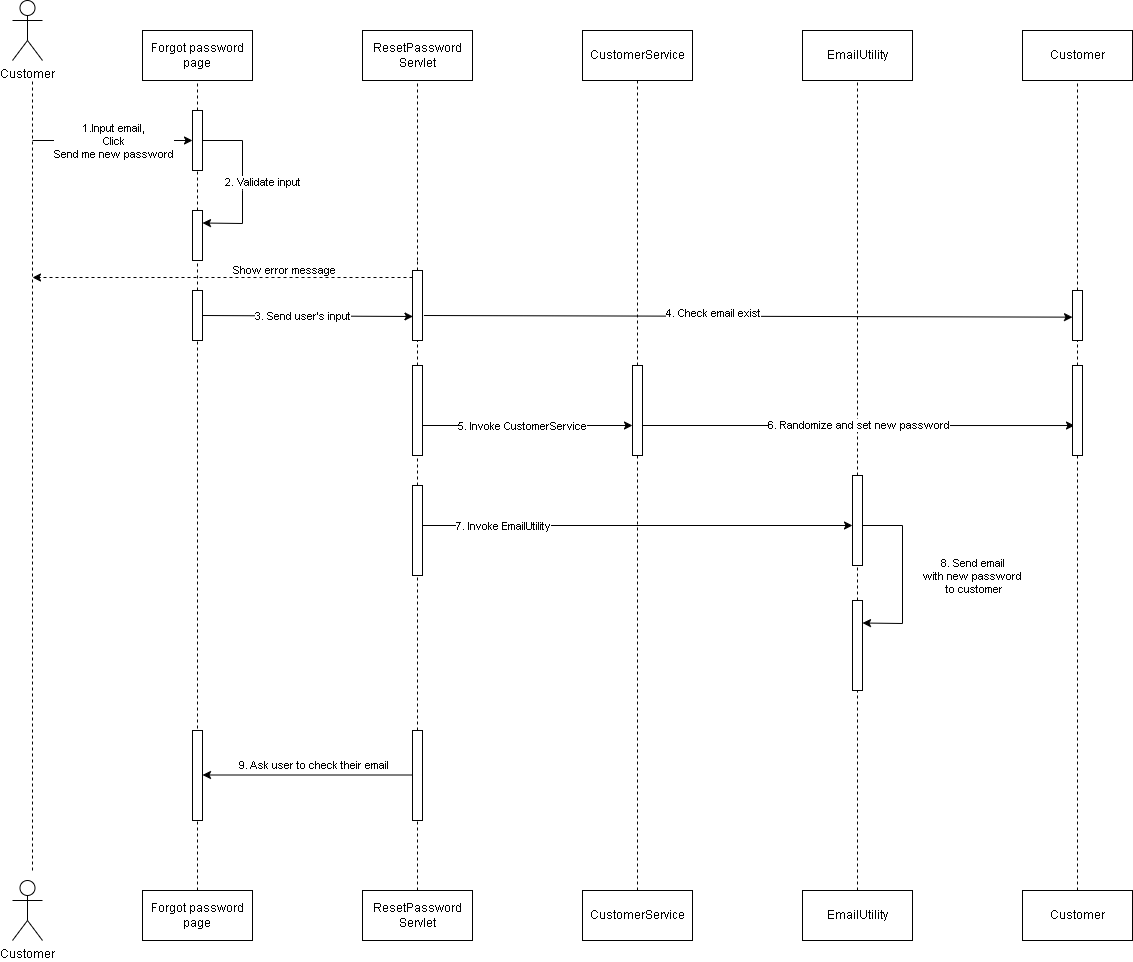
#### Customer

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Getters* | *Retrieve value of an attribute of a Customer object.* |
| *02* | *Setters* | *Set value to an attribute of a Customer object.* |

### c. Sequence Diagram(s)

*[Provide the sequence diagram(s) for the feature, see the sample below]*



### d. Database Queries

*[Provide the detailed SQL (select, insert, update...) which are used in implementing the function/screen]*

|  |
| --- |
| SELECT COUNT(\*) AS count FROM Customer WHERE customer\_email = ? |

|  |
| --- |
| SELECT \* FROM Customer WHERE customer\_email = ? |

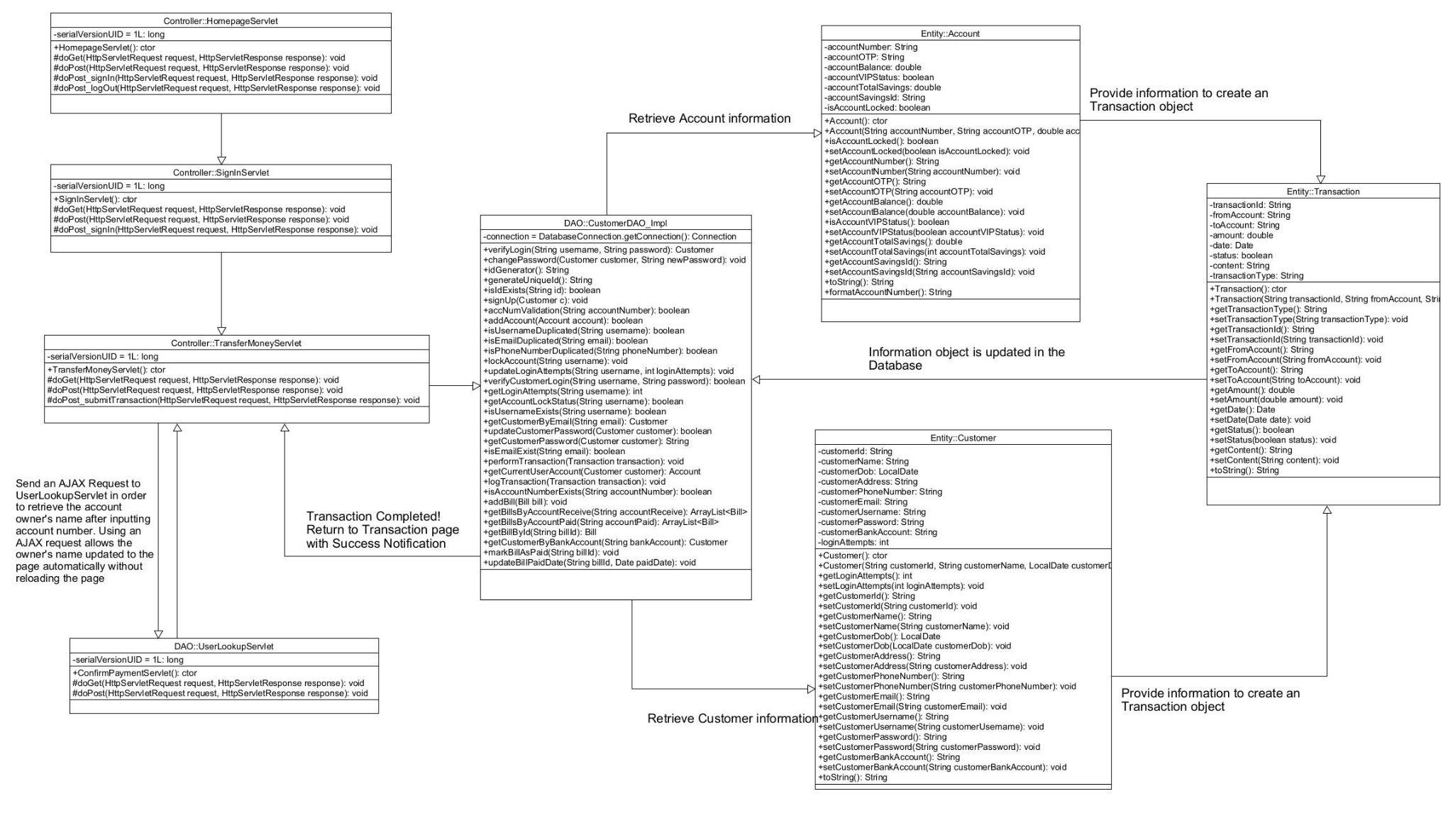
|  |
| --- |
| UPDATE Customer SET customer\_password = ? WHERE customer\_id = ? |

## 5. <Transfer>

*[Provide the detailed design for the function <Feature/Function Name1>. It include Class Diagram, Class Specifications, and Sequence Diagram(s)]*

### a. Class Diagram

*[This part presents the class diagram for the relevant feature]*



### b. Class Specifications

*[Provide the description for each class and the methods in each class, following the table format as below]S*

#### HomepageServlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when client requests a GET request* |
| *02* | *doPost* | *This method will be called when client requests a POST request* |
| *03* | *doPost\_signIn* | *Allow user to redirect to Sign In page.* |
| *04* | *doPost\_logOut* | *Allow user to log out.* |

#### 

#### SignInServlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when a client requests a GET request.* |
| *02* | *doPost* | *This method will be called when a client requests a POST request.* |
| *03* | *doPost\_signIn* | *Receive User’s Username and Password, then call a method to retrieve the Customer object who has the inputted Username and Password, finally set Customer object to current Session.* |

#### 

#### 

#### TransferMoney

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when client requests a GET request* |
| *02* | *doPost* | *This method will be called when client requests a POST request* |
| *03* | *doPost\_submitTransaction* | *Allow users to submit transaction.* |

#### 

#### UserLookupServlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when client requests a GET request* |
| *02* | *doPost* | *This method will handle the AJAX request from the Transfer Money Page. When a user fills in the Bank Account input with a valid bank account, the Account owner’s name will be displayed without reloading the page, else the display text will be “Unknown”.* |

#### 

#### CustomerDAO\_Impl

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *getCurrentUserAccount* | *This method will get customer’s account number* |
| *02* | *isAccountNumberExists* | *This method will check if account number exists* |
| *03* | *performTransaction* | *This method will update customers’ account balance* |
| *04* | *logTransaction* | *This method will log transaction* |

#### 

#### Customer

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Getters* | *Retrieve value of an attribute of a Customer object.* |
| *02* | *Setters* | *Set value to an attribute of a Customer object.* |

#### 

#### Account

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Getters* | *Retrieve value of an attribute of an Account object.* |
| *02* | *Setters* | *Set value to an attribute of an Account object.* |

#### 

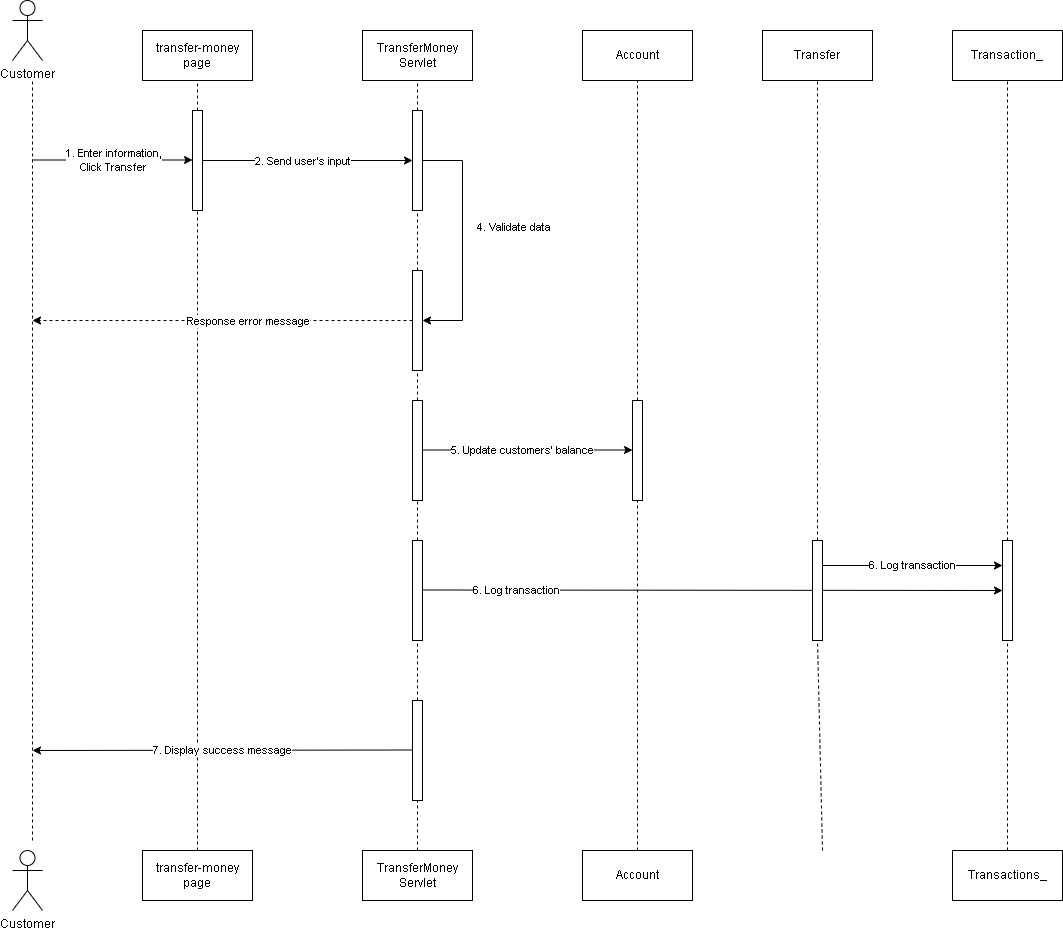
#### Transaction

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Getters* | *Retrieve value of an attribute of an Transaction object.* |
| *02* | *Setters* | *Set value to an attribute of an Transaction object.* |

### c. Sequence Diagram(s)

*[Provide the sequence diagram(s) for the feature, see the sample below]*



### d. Database Queries

*[Provide the detailed SQL (select, insert, update...) which are used in implementing the function/screen]*

|  |
| --- |
| SELECT \* FROM Account WHERE account\_number = ? |

|  |
| --- |
| UPDATE Account SET account\_balance = account\_balance - ? WHERE account\_number = ? |

|  |
| --- |
| UPDATE Account SET account\_balance = account\_balance + ? WHERE account\_number = ? |

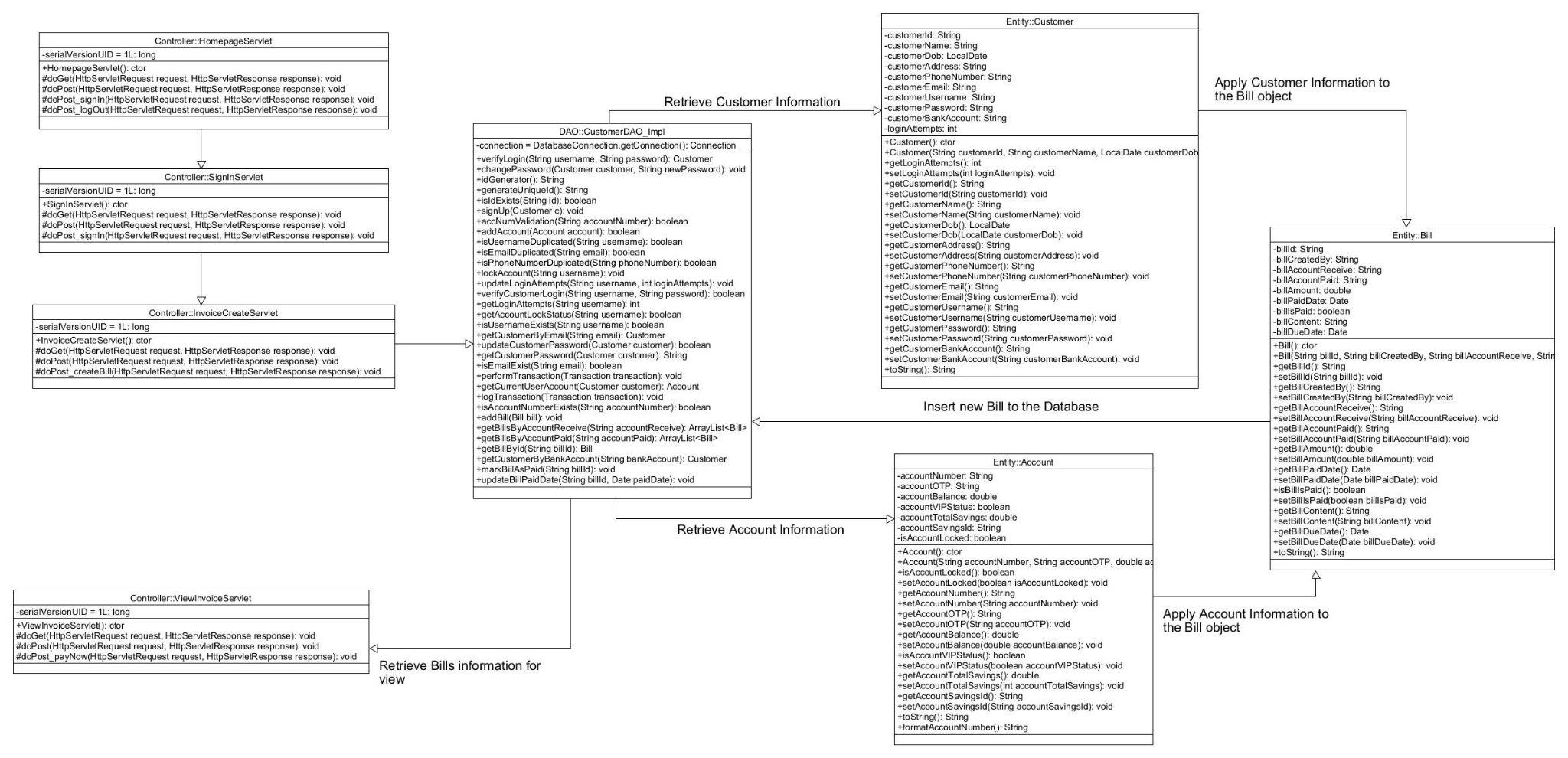
|  |
| --- |
| INSERT INTO Transaction\_ (transaction\_id, transaction\_fromAccount, transaction\_toAccount, transaction\_amount, transaction\_date, transaction\_status, transaction\_content) VALUES (?, ?, ?, ?, ?, ?, ?) |

## 6. <Generate invoice>

*[Provide the detailed design for the function <Feature/Function Name1>. It include Class Diagram, Class Specifications, and Sequence Diagram(s)]*

### a. Class Diagram

*[This part presents the class diagram for the relevant feature]*



### b. Class Specifications

*[Provide the description for each class and the methods in each class, following the table format as below]S*

#### Homepage Servlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when client requests a GET request* |
| *02* | *doPost* | *This method will be called when client requests a POST request* |
| *03* | *doPost\_signIn* | *Allow users to redirect to the Sign In page.* |
| *04* | *doPost\_logOut* | *Allow users to log out.* |

#### 

#### 

#### 

#### 

#### SignInServlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when a client requests a GET request.* |
| *02* | *doPost* | *This method will be called when a client requests a POST request.* |
| *03* | *doPost\_signIn* | *Receive User’s Username and Password, then call a method to retrieve the Customer object who has the inputted Username and Password, finally set Customer object to current Session.* |

#### 

#### InvoiceCreateServlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when client requests a GET request* |
| *02* | *doPost* | *This method will be called when client requests a POST request* |
| *03* | *doPost\_payNow* | *Allow users to redirect to pay invoice* |

#### 

#### CustomerDAO\_Impl

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *getCurrentUserAccount* | *This method will get customer’s account number* |
| *02* | *getBillsByAccountReceive* | *This method will get bills of bill receiver* |
| *03* | *getBillsByAccountPaid* | *This method will get bills of bill sender* |
| *04* | *isAccountNumberExists* | *This method will check if account number exists in the system* |
| *05* | *addBill* | *This method will add bill information into the system* |

#### 

#### Customer

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Getters* | *Retrieve value of an attribute of a Customer object.* |
| *02* | *Setters* | *Set value to an attribute of a Customer object.* |

#### Account

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Getters* | *Retrieve value of an attribute of an Account object.* |
| *02* | *Setters* | *Set value to an attribute of an Account object.* |

#### 

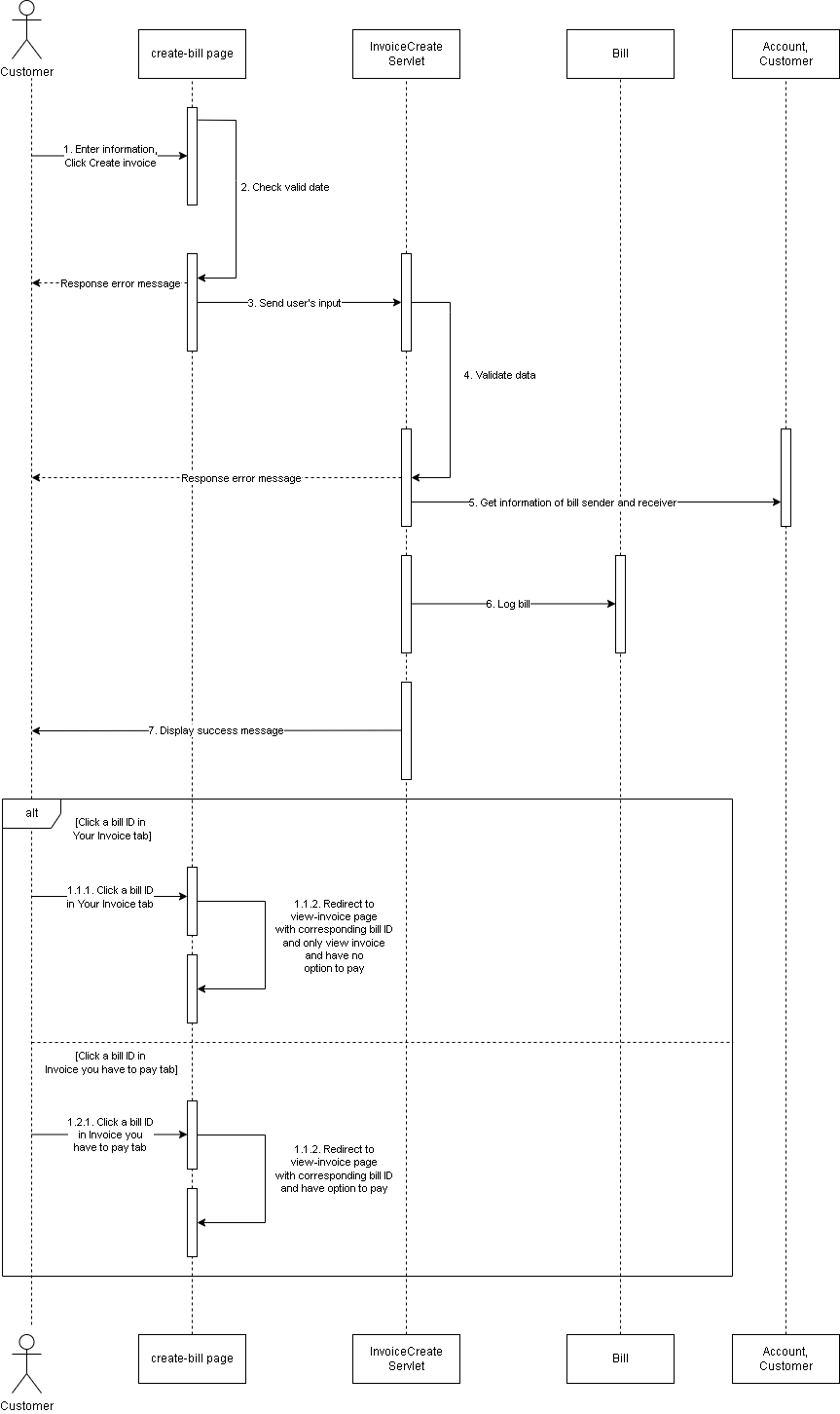
#### Bill

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Getters* | *Retrieve value of an attribute of a Bill object.* |
| *02* | *Setters* | *Set value to an attribute of a Bill object.* |

### c. Sequence Diagram(s)

*[Provide the sequence diagram(s) for the feature, see the sample below]*



### d. Database Queries

*[Provide the detailed SQL (select, insert, update...) which are used in implementing the function/screen]*

|  |
| --- |
| SELECT \* FROM Account WHERE account\_number = ? |

|  |
| --- |
| SELECT \* FROM Bill WHERE bill\_accountReceive = ? and bill\_isPaid = 0 |

|  |
| --- |
| SELECT \* FROM Bill WHERE bill\_accountPaid = ? and bill\_isPaid = 0 |

|  |
| --- |
| SELECT COUNT(\*) FROM Account WHERE account\_number = ? |

|  |
| --- |
| INSERT INTO Bill (bill\_id, bill\_createdBy, bill\_accountReceive, bill\_accountPaid, bill\_amount, bill\_paidDate, bill\_isPaid, bill\_billContent, bill\_dueDate) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?) |

|  |
| --- |
| SELECT \* FROM Bill WHERE bill\_id = ? |

|  |
| --- |
| SELECT \* FROM Customer WHERE customer\_bankAccount = ? |

|  |
| --- |
| UPDATE Account SET account\_balance = account\_balance - ? WHERE account\_number = ? |

|  |
| --- |
| UPDATE Account SET account\_balance = account\_balance + ? WHERE account\_number = ? |

|  |
| --- |
| INSERT INTO Transaction\_ (transaction\_id, transaction\_fromAccount, transaction\_toAccount, transaction\_amount, transaction\_date, transaction\_status, transaction\_content, transaction\_type) VALUES (?, ?, ?, ?, ?, ?, ?, ?) |

|  |
| --- |
| UPDATE Bill SET bill\_isPaid = 1 WHERE bill\_id = ? |

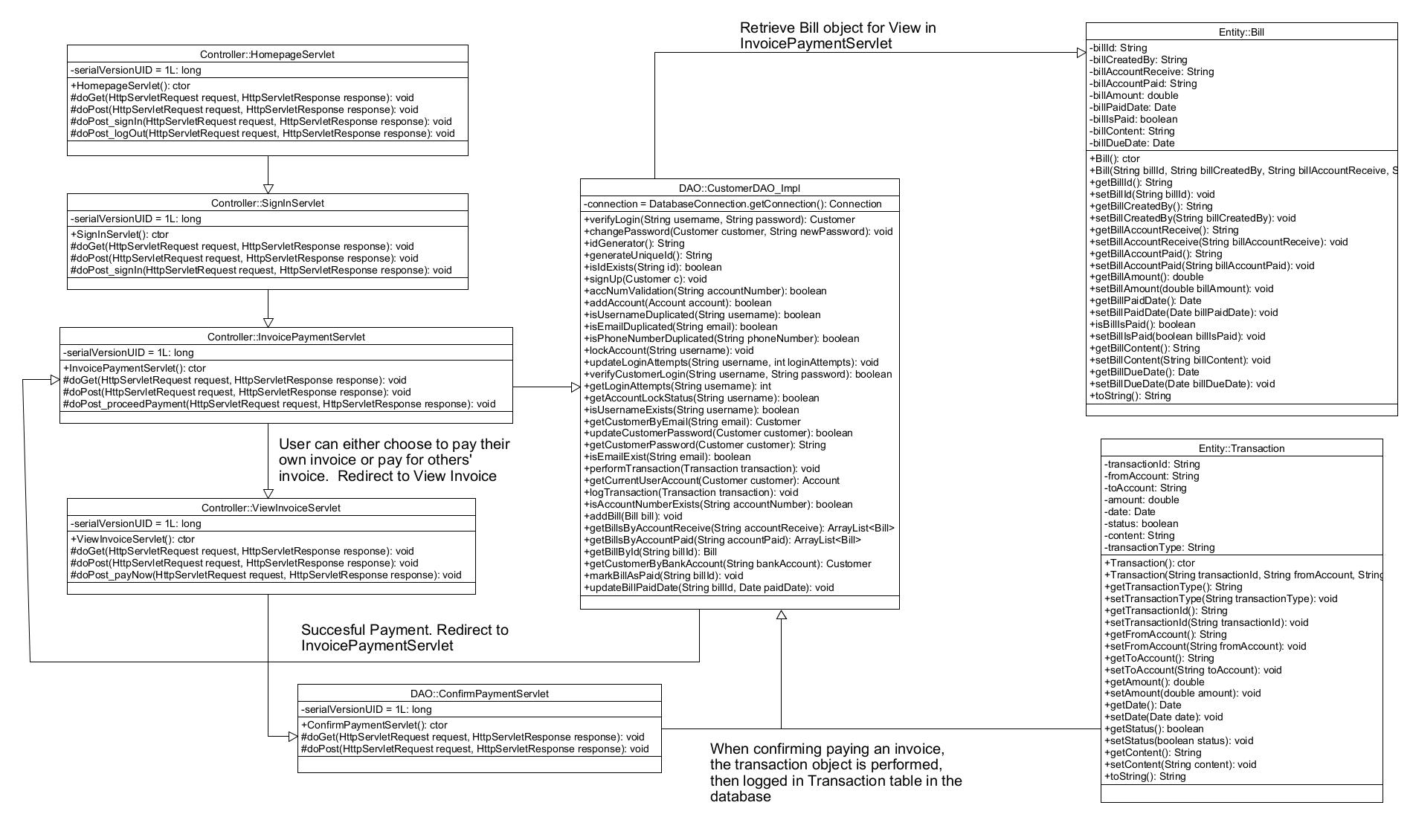
|  |
| --- |
| UPDATE Bill SET bill\_paidDate = ? WHERE bill\_id = ? |

## 7. <Pay invoice>

*[Provide the detailed design for the function <Feature/Function Name1>. It include Class Diagram, Class Specifications, and Sequence Diagram(s)]*

### a. Class Diagram

*[This part presents the class diagram for the relevant feature]*



### b. Class Specifications

*[Provide the description for each class and the methods in each class, following the table format as below]S*

#### HomepageServlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when client requests a GET request* |
| *02* | *doPost* | *This method will be called when client requests a POST request* |
| *03* | *doPost\_signIn* | *Allow user to redirect to Sign In page.* |
| *04* | *doPost\_logOut* | *Allow user to log out.* |

#### 

#### SignInServlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when a client requests a GET request.* |
| *02* | *doPost* | *This method will be called when a client requests a POST request.* |
| *03* | *doPost\_signIn* | *Receive User’s Username and Password, then call a method to retrieve the Customer object who has the inputted Username and Password, finally set Customer object to current Session.* |

#### 

#### InvoicePaymentServlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when a client requests a GET request.* |
| *02* | *doPost* | *This method will be called when a client requests a POST request.* |
| *03* | *doPost\_proceedPayment* | *This method will allow a user to check the invoice once again before proceeding to Invoice Payment.* |

#### 

#### ViewInvoiceServlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when a client requests a GET request.*  *It loads necessary objects and sets them to requestScope for use in the JSP page.* |
| *02* | *doPost* | *This method will be called when a client requests a POST request.* |

#### 

#### ConfirmPaymentServlet

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *doGet* | *This method will be called when a client requests a GET request.* |
| *02* | *doPost* | *This method will receive a AJAX request, handle the request by performing the Invoice payment, logging the Payment in Transaction table and marking the Invoice as paid when a user presses PAY NOW button in View Invoice.* |

#### 

#### CustomerDAO\_Impl

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *getCurrentUserAccount* | *This method will get customer’s account number* |
| *02* | *getBillsByAccountPaid* | *This method will get bills of bill sender* |

#### 

#### Bill

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Getters* | *Retrieve value of an attribute of a Bill object.* |
| *02* | *Setters* | *Set value to an attribute of a Bill object.* |

#### 

#### 

#### 

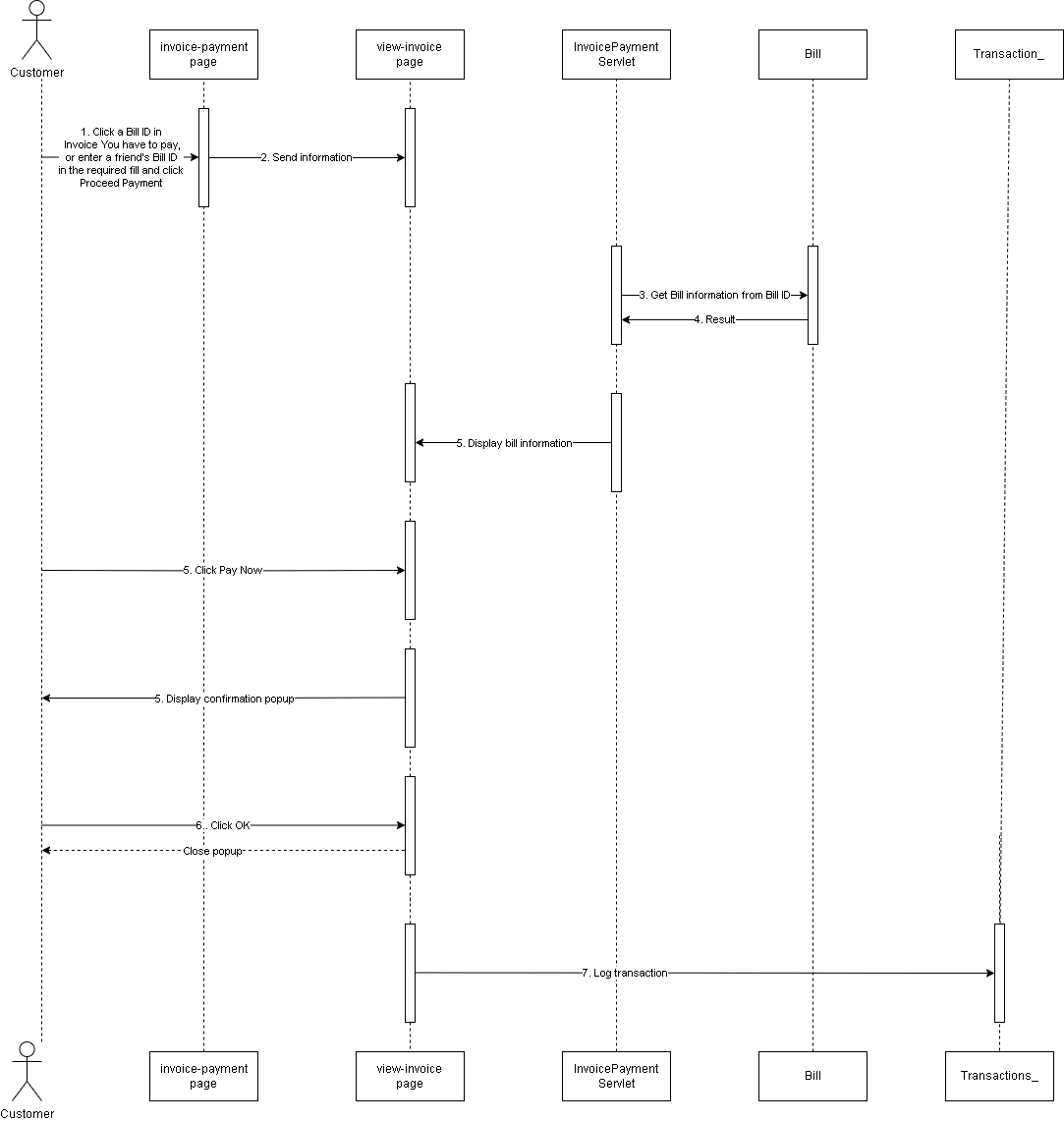
#### Transaction

*[Provide the detailed description for the class methods]*

|  |  |  |
| --- | --- | --- |
| **No** | **Method** | **Description** |
| *01* | *Getters* | *Retrieve value of an attribute of a Customer object.* |
| *02* | *Setters* | *Set value to an attribute of a Customer object.* |

### c. Sequence Diagram(s)

*[Provide the sequence diagram(s) for the feature, see the sample below]*



### d. Database Queries

*[Provide the detailed SQL (select, insert, update...) which are used in implementing the function/screen]*

|  |
| --- |
| SELECT \* FROM Account WHERE account\_number = ? |

|  |
| --- |
| SELECT \* FROM Bill WHERE bill\_accountPaid = ? and bill\_isPaid = 0 |

|  |
| --- |
| SELECT \* FROM Bill WHERE bill\_id = ? |

|  |
| --- |
| SELECT \* FROM Customer WHERE customer\_bankAccount = ? |

|  |
| --- |
| UPDATE Account SET account\_balance = account\_balance - ? WHERE account\_number = ? |

|  |
| --- |
| UPDATE Account SET account\_balance = account\_balance + ? WHERE account\_number = ? |

|  |
| --- |
| INSERT INTO Transaction\_ (transaction\_id, transaction\_fromAccount, transaction\_toAccount, transaction\_amount, transaction\_date, transaction\_status, transaction\_content, transaction\_type) VALUES (?, ?, ?, ?, ?, ?, ?, ?) |

|  |
| --- |
| UPDATE Bill SET bill\_isPaid = 1 WHERE bill\_id = ? |

|  |
| --- |
| UPDATE Bill SET bill\_paidDate = ? WHERE bill\_id = ? |

## 