# 1. Problem Statement

## Objective

To create an online payment wallet application that is linked with an existing bank account, allowing a user to perform various activities such as creating an account, money transfer and performing transactions.

## Abstract of the project

This project is aimed at developing an online payment wallet system that can allow a user to link their bank accounts and perform various transactions on the go. Users would also be able to deposit their money in the wallet, check the balance, withdraw money, print transactions and perform fund transfers.

## CLASSES AND DEPENDENCIES

Following are the list of packages, classes and their respective dependencies.

* **Bean package**

It contains

* *PaymentWallet.java class*

This class includes all the private declared variables along with their respective getters and setter methods.

* **UserInterface package**

It contains

* *MainUI.java class*

This class includes the menu, the input/output functionality and main method. This is the core class that a user will interact with.

* **Service package**

It contains

* *IPaymentService.java Interface*

This interface declares the abstract methods of the service class.

* *ServiceValidation.java*

This class is responsible for the validation process of the input entered by the user. It contains various validation methods for individual inputs such as username, password, email etc.

* *PaymentService.java class*

This class is responsible for holding the methods that forward the object to the Data Access Object (DAO) class.

* **DAO package**

It contains

* *IPaymentDAO.java Interface*

This interface contains the declaration of the abstract methods of the PaymentDAO class.

* *PaymentDAO.java class*

This class contains the methods that have the activity’s logic in them. For example, if a user wishes to create an account, the process’s logic will be written inside the createAccount() method of the DAO class. This class receives the object from the service class’s methods.

* **PaymentExceptions package**

It contains

* *NoUserFoundException.java class*

This exception class is called when a user tries to sign in but his data is unavailable inside the database.

* **Test package**

It uses JUnit framework and well written test code to check the authenticity of code.

## Functional components of the project

1. Following are the list of functionalities of the **PaymentDAO** class in the **com.capgemini.DAO** package.
   1. Create an Account.

public boolean createAccount (PaymentWallet obj);

* Access Modifier: public
* Return Type: Boolean
* Arguments: Bean Class Object
  1. Show Balance of an existing account.

public PaymentWallet showBalance (PaymentWallet obj);

* Access Modifier: public
* Return Type: Bean Class Object
* Arguments: Bean Class Object
  1. Deposit Money in an existing account.

public PaymentWallet depositBalance (double deposit, PaymentWallet obj);

* Access Modifier: public
* Return Type: Bean Class Object
* Arguments: Primitive double variable. Bean Class Object
  1. Withdrawing money from an existing account.

public PaymentWallet withdrawBalance (double withdraw, PaymentWallet obj);

* Access Modifier: public
* Return Type: Bean Class Object
* Arguments: Primitive double variable, Bean Class Object
  1. Performing Transactions.

public PaymentWallet transferFund (long bankAccount2, double amount, PaymentWallet pw);

* Access Modifier: public
* Return Type: Bean Class Object
* Arguments: Primitive long variable, Primitive double variable. Bean Class Object
  1. Print transaction for an existing transaction ID.

public PaymentWallet printTransaction (long transactionId, PaymentWallet pw);

* Access Modifier: public
* Return Type: Bean Class Object
* Arguments: Primitive long variable, Bean Class Object

1. The following are the list of functionalities in the **ServiceValidation** class of the **com.capgemini.service** package:

* public boolean validateConsumerName(String consumerName);

Access Modifier: Public

Return Type: Boolean

Arguments: String variable

* public PaymentWallet validateLogIn(String username, String password);

Access Modifier: Public

Return Type: Bean Object

Arguments: String variable, String variable

* public boolean validateUserName(String a);

Access Modifier: Public

Return Type: Boolean

Arguments: String variable

* public boolean validatePassword(String password, String repassword);

Access Modifier: Public

Return Type: Boolean

Arguments: String variable, String variable

* public boolean validateEmailID(String email);

Access Modifier: Public

Return Type: Boolean

Arguments: String variable

* public boolean validatePhoneNo(String phoneNo);

Access Modifier: Public

Return Type: Boolean

Arguments: String variable

* public boolean validateAge(int age);

Access Modifier: Public

Return Type: Boolean

Arguments: int variable

* public boolean validateGender(String gender);

Access Modifier: Public

Return Type: Boolean

Arguments: String variable