**PAYMENT WALLET APPLICATION**

**1 Problem Statement**

* 1. **Objective**

To develop a payment wallet application where customers can check balance and deposit amount etc.

* 1. **Abstract of the project**

This project is aimed at developing a payment wallet application. This application can be used to enable customers to park their money in the wallet. There are features like balance checking and perform transactions like deposit into and withdraw from the account.

* 1. **Functional components of the project**

Following is a list of functionalities of the application. There are six modules in the application. Every module performs a set of operations as per the action performed by the customer. These modules allow the customer to interact with their wallet and perform various transactions and queries regarding their wallet.

Customer should be able to:

1. Create Account
2. View balance
3. Deposit amount
4. Withdraw amount
5. Transfer fund
6. Print Transactions

Layered Architecture is used to implement the modules in this application.

This is as follows:

1. **ui package**

This is the presentation layer which will be visible to the customer from where options can be selected.

It contains display class and main method with the menu to take input from the user.

* **UI package:**

Application class:

public static void main(String[] args){…}

1. **Service package**

Business/service layer that contains the interfaces and service class with abstract implementation of the modules.

* **Service package**:

Interface IApplicationService

void createAccount();

double showBalance();

double showBalance();

double withdraw();

void fundTransfer();

void printStatement();

class ApplicationService extends IApplucationService

void createAccount(){…}

double showBalance(){…}

double showBalance(){…}

double withdraw(){…}

void fundTransfer(){…}

void printStatement(){…}

1. **DAO package**

Business logic is implemented in the DAO class and is accessed by the service class through dao interface. It accesses data of the document object.

* **DAO package**:

Interface IApplicationDAO

void createAccount();

double showBalance();

double showBalance();

double withdraw();

void fundTransfer();

void printStatement();

class ApplicationDAO extends IApplucationDAO

void createAccount(){…}

double showBalance(){…}

double showBalance(){…}

double withdraw(){…}

void fundTransfer(){…}

void printStatement(){…}

1. **Data Transfer Object package(POJO)**

The data related to the customer is stored privately in the variables and is accessed by other classes through getter and setter methods.

* **Bean(POJO) Package**:

Class Customer

Customer Id

Customer name

Phone number

Balance

Email-id

Getters and setters

1. **Exception handling package**

Contains exception class that would be called when any exception raises because of the input given by the customer and has the functionality to handle the exceptions.

1. **Test package**

Contains test class with test cases for dao class to test various possible valid input scenarios.