Parallel Project Documentation

Capgemini

**PAYMENT WALLET SYSTEM**

# PAYMENT WALLET APPLICATION

## AIM

This project aimed to create a Payment Wallet Application for any bank xyz.

## Abstract of the project

Our aim is to create payment wallet application for XYZ bank. This application allowed its customer to park their money in the bank wallet. And this application provide the various services to the customer like to check the balance in an account, deposit and withdraw the money.

Functional components of the project

Services provided by this application is as follows:

* Customer should be able to:
  1. Create Account
  2. LogIn to Account
  3. Check the balance.
  4. Deposit money.
  5. Withdraw money.
  6. Print transactions.
  7. Fund transfer.

## CLASSES AND DEPENDENCIES

Following is a list of packages, classes and their dependency.

* **Bean package**
* PaymentApp.java class

Attributes :

private String customerName;

private long accountNumber;

private String address;

private String gender;

private String phoneNumber

private double balance;

private LocalDate aod;

private int age;

private String user\_ID;

private String password

This class contains getter and setter for all the attributes. Setter methods have return type as void, while getter method have return type same as attribute data type. The toString() method is also used.

* **UI package**
* Customer.java class

This includes methods that displays the menu to consumer and calls

appropriate methods for bank application

* public boolean createAccount(PaymentApp paymentApp);
* public double showBalance();
* public boolean logIn(String user\_ID,String password);
* public boolean deposit(double amount);
* public boolean withdraw(double amount);
* public boolean fundTransfer(long receiverAccountNumber,double amount);
* public void printTranscation();
* **Service package**
* PaymentAppValidate.java (validation class)
* public boolean validatePhoneNo (String phoneNo.)
* public boolean validateCustomerName (String customerName)
* public boolean validateGenderName(String genderName)
* IPaymentAppService.java (interface)
* public boolean createAccount(PaymentApp paymentapp);
* public double showBalance();
* public boolean logIn(String user\_ID,String password);
* public boolean deposite(double amount)
* public boolean withdraw( double amount);
* public boolean fundTransfer( long receiverAccountNumber,double amount);
* public void printTranscation();
* PaymentappService.java (class)

This class implements all the methods of interface of this package, also contains validation methods.

* **DAO package**
* IPaymentAppDao.java (Interface)
* public boolean createAccount(PaymentApp paymentapp);
* public double showBalance();
* public boolean logIn(String user\_ID,String password);
* public boolean deposite(double amount)/
* public boolean withdraw( double amount);
* public boolean fundTransfer( long receiverAccountNumber,double amount);
* public void printTranscation();
* PaymentAppDao.java (Implementation class)

This class implements all the methods of interface of this package, also contains validation methods.

* **Exception package**
* InsufficientBalanceExp.java (Exception class)
* InvalidInputExp.java (Exception class)

(These classes are used to handle Exceptions that disrupts the normal flow of the program’s instructions)

* **Test package**

* PaymentAppDaoTest.java (JUnit Test Cases)
* PaymentAppServiceTest.java (JUnit Test Cases)
* PaymentAppTest.java (JUnit Test Cases)
* PaymentAppValidateTest.java (JUnit Test Cases)

These classes uses Junit framework and well written test code to check the authenticity of code.