NNU

Faculty of Engineering and IT

Computer Engineering Department



Software Engineering Project

**E-wallet (PayMe)**

Fall Semester 2021

**Group(2)**

Supervision :

**Dr. Amjad Abu Hassan**

* **Project Name**
* **PayMe**
* **Team members**

1. Yousef Hanbali
2. Mustafa habroumman
3. Mustafa Soker
4. Momen odeh
5. Hosney ishtaya
6. Raoof Fares
7. Ahmad Droobi

* **Accounts**
* **Git repository**: [ahmaddroobi99/SoftwareProject\_2021\_Agile\_Trello (github.com)](https://github.com/ahmaddroobi99/SoftwareProject_2021_Agile_Trello)
* **Slack account** : [Slack | softwareproject\_2021\_agile\_trello | Droobi Team | 2 new items](https://app.slack.com/client/T02J9SFPCSU/C02H5JT1JVD)
* **Trello account** :[ahmad droobi's Workspace (ahmaddroobisworkspace) | Trello](https://trello.com/ahmaddroobisworkspace)

**Table of Contents**

1. [Introduction 1](#_bookmark0)
   1. Purpose 1
   2. [Scope 1](#_bookmark2)
   3. [Definitions, Acronyms, and Abbreviations 1](#_bookmark3)
   4. [References 1](#_bookmark4)
   5. [Glossary 2](#_bookmark5)
2. [Overall Description 5](#_bookmark6)
   1. [Product Perspective 5](#_bookmark7)
   2. [Product Functions 5](#_bookmark8)
      1. [Account Registration 5](#_bookmark9)
      2. [Account Login 5](#_bookmark10)
      3. Search 6
      4. [Add to Shopping Cart 6](#_bookmark12)
      5. [Delete from Shopping Cart 6](#_bookmark13)
      6. [Reserve 6](#_bookmark14)
      7. [Checkout 6](#_bookmark15)
      8. [Contact Us 6](#_bookmark16)
      9. [Update Account Information 6](#_bookmark17)
      10. [View Shipping Status 7](#_bookmark18)
      11. [View Account Purchase History 7](#_bookmark19)
      12. [Account Logout 7](#_bookmark20)
      13. [Help 7](#_bookmark21)
   3. [User Characteristics 7](#_bookmark22)
   4. [Constraints 7](#_bookmark23)
3. [Specific Requirements 7](#_bookmark26)
   1. [External Interface Requirements 7](#_bookmark27)
      1. [System Interfaces: 7](#_bookmark28)
      2. [User Interfaces: 7](#_bookmark29)
   2. [Functional Requirements 8-14](#_bookmark33)
   3. [Logical Structure of the Data 15](#_bookmark48)
      1. [Types of Information Used 15](#_bookmark49)
      2. [Data Model 15](#_bookmark50)
   4. [Software System Attributes 16](#_bookmark53)
      1. [Reliability 16](#_bookmark54)
      2. [Availability 16](#_bookmark55)
      3. [Security 16](#_bookmark56)
      4. [Maintainability 16](#_bookmark57)
      5. [Portability 16](#_bookmark58)

**Software Requirements Specifications (SRS) for Group 2—( PayMe Project)**

**1.1.Purpose:**

The main objective of e-Wallet is to make money transaction easier so through the e-wallet service , a range of transactions and transfers can be carried out directly in an easy ,quickly and secure manner

in addition of introducing services for specified Clients like university students and business men , distinguishing itthat its easy to use ,and its well organized design that make sense for different users from different backgrounds .

**1.2. Scope:**

The possibility of paying fees and subscriptions to municipalities, government departments, insurance companies and billing.

In addition, payment service and payment of the price of purchases (goods) and payment of university fees as well as money transfer service between customers.

**1.3.Definitions, Acronyms, and Abbreviations:**

|  |  |
| --- | --- |
| Electronic Funds Transfer | EFT |
| E-Money Directive | EMD |
| Bank Account Number | BAN |
| Payment Service | PS |

**1.4. References:**

* [**https://economictimes.indiatimes.com/definition/e-wallets**](https://economictimes.indiatimes.com/definition/e-wallets)
* [**https://www.investopedia.com/terms/d/digital-wallet.asp**](https://www.investopedia.com/terms/d/digital-wallet.asp)
* [**https://business.ebanx.com/en/resources/payments-explained/e-wallets**](https://business.ebanx.com/en/resources/payments-explained/e-wallets)

# Stackoverflow and Youtube Overview

**1.5 Glossary** :

|  |  |
| --- | --- |
| **Term** | **Definition** |
| e-Wallet | The software component that handles the payment and associated data and provides the user( in our case, a student) with options to make payments, recharge and avail loyalty benefits provided by the vendors. |
| Student | Refers to the students at Bits Goa. More specifically, any person with access to the outlets at Bits Goa who has an SWD account maintained at the campus and is identified by a unique ID number provided by the campus. |
| Database | Collection of all the information, i.e., payment logs, credit balances and dues, account balances and registered student information is monitored by this system. |
| Vendor | Person/outlet on campus who(which) receives the payment from the students when a purchase is made by them. |
| Payment tracker | A system which facilitates the secure payment process by interacting with the student and the database and the vendor. |
| Loyalty benefit | Special offers made by outlets on campus to students who make purchases from their outlets using the e- Wallet system. |
| Logs | Refers to data that indicates the proceedings of various transactions between the student and the vendor. In a way, they behave as a sort of proof of payment. |
| Account | Refers to the unique account maintained for each registered student which comprises a security pin, balance amounts, unique name and id. |
| Recharge | Recharge here, is used to refer to the transfer/translation of money from the student’s swd- account to e-Wallet cash, which the student can use at the various on-campus outlets. |
| User | The term user has been used along with the term Student in this document. In all cases both terms refer to a student as described above. |

**2.Overall Description**

**2.1 Product Perspective**

This product is a new application released in the market with new features and improved characteristics in comparison with another product versions available. The PayMe e-wallet is an application that gives the any user the possibility to make payments and transfers in addition to another transactions using his own stored money or debit/credit card in a secure mode. The system will interact with a complete maintained database with all necessary information and a number of different servers and websites.

The e-Wallet system has four active actors.

The **User (student ,business man, teacher …etc), Vendor, Payment Tracker System and the Database.**

**2.2 Product Functions**

This numbered list gives the main features and functions of the PayMe system.

**2.2.1 Account Registration**

The account registration function shall allow users to register in the application by entering their information and a password.

Rationale: This function make users get the users registered in the system in a safe way with all details stored for later use and a strong checked password. In addition to that there is an authentication message with a code to create the account in a verified way

**2.2.2 Account Login**

The account login function shall allow the registered users to enter the application in order to use it

Rationale: This function make the user navigate into the application in a safe mode, because he/she can only access entering the correct username and password.

**2.2.3 Add Debit/Credit Card/s**

This function shall allow users to add an existing credit or debit card linked with a personal bank account.

Rationale: This function allow users to add any necessary card and money source information manually or via scanning to make their payments and pay all the necessary fees in a easily way.

**2.2.4 Set Pin**

The Set Pin function shall allow users to set a pin representing the serial number of the debit card/s of each account.

Rationale: This is a very important feature that allows users to make their investments and pays in a very straight forward form without filling all information of the payment method each time.

**2.2.5 Make Payment**

The make payment function shall allow users to make necessary payments using the wallet.

Rationale: This function is the very main facility that is given by the PayMe application, because allows to pay in a safely and comfortable way.

**2.2.6 Municipality Services Pay**

The function allows users to pay the municipality fees like electrical fees, water fees, etc…

Rationale: This function is very useful and allows user to pay all this types of fees in a very fast way.

**2.2.7 Trusted Websites Pay**

The trusted websites pay function shall allow users to pay to a number of trusted websites of the system without needed to authentication and confirmation.

**2.2.8 University Services**

This function shall allow university students to pay their fees of any semester by choosing the number of hours and their university name.

**2.2.9 Transfer Between Users**

This function allows very safe and fast money transfer between 2 users of the application.

**2.2.10 Exchange Currency**

The exchange currency function shall allow users to exchange the currency of the amount of money to pay in order to choose the suitable account or debit card.

**2.2.11 Check Balance**

The check balance function shall allow users to check the remaining amount of money stored in the wallet and the record of all payments in the last 12 month.

**2.2.12 Disable Account**

The function allows a user to disable an non used account or delete one account by answering a security question.

**2.2.13 Recharge Points**

The function allows a user to recharge money in the wallet in a number of accredited offices.

**2.2.14 Contact us**

The user can contact the customer support team via the provided function in case of any problem.

**2.3 User Characteristics**

The user of the PayMe system can be any person including students and workers and any type of category with an amount of money and number of credit cards or money sources that want to achieve their payments in a easily way. They must have a minimum knowledge of transfers and payment methods.

**2.4 Constraints**

The system have a number of constraints like the balance check(The wallet must be charged with money before pay) and the confirmation process before paying. Another constraint is that the user can only charge the wallet from accredited points. Debit cards cancelled by the owning bank cannot be used definitely by the user.

# External Interface Requirements:

**The project consists of many GUIs that makes communication with the user is easy.**

1. Registration interface: that work on making an E-wallet account for the user, and this interface contains information like name, email, phone number, number of bank account, type of account,... etc.
2. Login interface: to make a user enable login to the e-wallet system by account number and password, and we have the option restore the password if forgotten
3. Add card interface: by this interface, we can add credit card to the e-wallet be entering card number, card holder name, expiration date, and CSV.

1. Pay checks interface: by this interface, we can pay the checks to another user by putting the photo of checks, check number, the user number who will receipt check, Check's exchange date.
2. University registration interface: the system provides features that allow to the student pay the university fee by entering the number of our that will be registered, price of the hour, university name, student id, and on this page, we can Submit student loan applications and know the response.
3. Trust site interface: by this interface, we can make some places reliable for pay (we do not need the confirm the payment information in each pay operation), so we enter the number of site, name of the site.

# 3.2.1Functional Requirements:

# 1.Register:

- The system should allow a non-registered user to create an account (one account).

- The system should ask the user to enter username, password, first name, last name, email and mobile phone.

- The system should confirm that the password is valid, and the username is available and email and mobile phone is not used by another account.

- The system should send a confirmation email or SMS Message for the account activation.

- After the account is confirmed, the system should store the information in its database.

# 2.Login:

- The system should ask the user for the username and password.

- The system will show a pop-up message if one of the credentials is not correct.

- After the system verifies the username and password, the user should proceed to the main dashboard and will be considered “Online”.

# 3.Add card:

- The system should allow the user to add credit card from the bank to use it for top up.

- The user should enter the information for the credit card, like the card number, card holder name, expiration date, and CSV.

- The system should check if the card number entered exist or doesn’t exist, and then check if the credentials is valid or not valid.

- The system should link the card with the account.

# 4.Pay checks:

- The system should allow the users to schedule a checks for credit withdrawn.

- The user should enter the date for the check and the value required, and the account id for the destination account.

- The system should check if the user has enough credit for the check, and the destination account is valid.

- When everything is correct, the check is stored in the database of the system, and the credit is transferred to the destination account.

# 5.Trust sites:

- The system should allow the user to add trusted web sites so he can pay without confirmation every time.

- The system should ask the user to enter the website he needs to trust.

- The system should confirm that the user enters a website and he is responsible for this trust link.

- The system now should allow payment without confirmation every time.

# 6.University Registration:

- The system should allow the users to pay for university fees.

- The user is asked to select the university and he logs in his student account.

- The user is asked to enter number of hours he want to sign up, and then a complete invoice is generated and he is asked to confirm the operation.

- The user gets a confirmation message.

- The system handles the transfer operation from the account credit to the university account and assures that the hours is assigned correctly.

# 7.Sub Accounts:

- The system should allow the user to add sub accounts with different currencies.

- The user should enter the currency he wants to add a sub account for.

- The system should check whether the account has a sub account with that currency already, if it exists a pop-up message will show up.

- The system should add the account if it doesn’t exist before, and the user can top it up with credit.

# 8. Transfer Credit:

- The system should allow the users to transfer credit between accounts.

- The user is asked to enter the amount he wants to transfer, the account id for the destination account.

- The user is given a confirmation number.

- The system should check if the user is a student, and if he is a student it takes a lower fee.

- When the user enters the confirmation number, it verifies it and either gives an error message about the information entered or the confirmation number, or if the everything is correct, the operation is done and the system transfers credit.

# 

# To make it more clear I will represent some of them as Tables :

* + 1. **User Registration**

|  |  |
| --- | --- |
| **Use Case Name** | Registration/Initial Identification |
| **Trigger** | --- |
| **Precondition** | Student has not registered for an e-Wallet account |
| **Basic Path** | 1. The User (Student) registers for the service using his/her student ID. The rest of the details will be then imported from their SWD accounts 2. On input of a valid student ID the program will then prompt the User to set a 4-digit pin for security purposes. 3. On Successful registration the User is asked to set the initial balance and credit limits. 4. The Entered amount for the initial balance is then   added to his/her respective e-Wallet. |
| **Alternative Paths** | In step 1, if the User enters a non-existent ID number the program prompts the User to re-enter.  If Valid then return to step 2.  In step 3, if the User enters an amount greater than that allowed by the system.  The system prompts the user to re-enter the amount under the limit. |
| **Postcondition** | The e-Wallet account is initiated and ready to use. |
| **Exception Paths** | The User may abandon the registration process at any time. |
| **Other** | NA |

* + 1. **Recharging e-Wallet**

|  |  |
| --- | --- |
| **Use Case Name** | Recharge e-Wallet a/c |
| **Trigger** | The user selects a *Recharge A/C* option. |
| **Precondition** | Student has registered successfully and possesses a unique 4-digit pin |
| **Basic Path** | 1. The User enter the amount required in the e-Wallet account. 2. The e-Wallet is credited with requested amount from the SWD account. |
| **Alternative Paths** | In Step 2 if the entered amount is greater than what is available in the account, the User is then prompted to avail the credit option.  1. If amount entered is more than the credit limit the user is prompted to enter again. |
| **Post condition** | The e-Wallet is recharged with the aforementioned balance. |
| **Exception Paths** | The attempt may be abandoned at any time. |
| **Other** | The Credit amount is then billed to the availing User in the next semester. Logs are then updated |

### Viewing Logs

|  |  |
| --- | --- |
| **Use Case Name** | View Logs |
| **Trigger** | User selects the option to view logs. |
| **Precondition** | Student possesses an e-Wallet a/c with a unique 4-digit pin |
| **Basic Path** | 1. The System requests the database for logs. 2. The database then acquires logs from the Payment tracker. 3. The logs are then presented by the system to the User. |
| **Alternative Paths** | NA |
| **Post condition** | The Logs have been displayed to the User. |
| **Exception Paths** | The User may abandon the operation at any time. |
| **Other** | NA |

* + 1. **Make Payments**

|  |  |
| --- | --- |
| **Use Case Name** | Make Payments |
| **Trigger** | The User purchases something and wishes to use the e-Wallet |
| **Precondition** | The payments are authorized by the payment tracker |
| **Basic Path** | 1. Payment request is sent to the payment tracker. 2. Based on various criteria the tracker approves/disapproves the payment request. 3. On approval the payment is then received by the   vendor. |
| **Alternative Paths** | In step 2, if the criterions are not met then the payment is not authorized. In that case the system reverts to step 1. |
| **Post condition** | The Payment is received by the concerned system. |
| **Exception Paths** | The User may abandon the operation at any time. |
| **Other** | The User information includes ID number and the 4-digit security pin. |

# Performance Requirements

The performance requirements are as follows:

* + - System login/logout shall take less than 5 seconds.
    - Searches shall return results within 10 seconds.
    - Orders shall be processed within 10 seconds.
    - System shall support 10,00 simultaneous users.
    - Verification the phone number shall take less than 10 seconds

# Logical Structure of the Data

The two sections below show the different types of information used by various functions and the overall data model, respectively.

# Types of Information Used

The types of information used by various functions of the website:

|  |  |
| --- | --- |
| **Function** | **Types of Information Used** |
| Account Registration | User information (name, mailing address, credit card type, number, expiration date, and user  name,phone number and password) |
| Account login | User information (user name, and password) |
| Search | Book information (title, author, course number,  professor, ISBN, price, and quantity) |
| Checkout | Credit card information (credit card type, credit card number), account purchase history (see below, *View account purchase history*), book  information (number on hand) |
| Update Account info. | User information (name, mailing and billing address, credit card type, number, and expiration  date, user name and password. |
| View account purchase history | Purchase history information (quantity, confirmation number, date of purchase, credit card type, credit card expiration date, title of  book(s), price of book(s), total of purchases |

# 

# Data Model

The following diagram describes the attributes and the types of data that represented :

Registration/Initial Identification

Recharge e-wallet a/c

Studen

Make Payment

View logs

Receive

credit

Avail Loyalty benefits

# Design Constraints

The PAyMe shall conform to the following design constraints:

* Able to support PC, Mac platforms.
* System logs out user after a ten minute inactivity period.
* System supports all web browsers (i.e. graphical, non-graphical).

# 3.5.1. Standards Compliance

The PayMe E-Wallet will follow existing standards and regulations, which are stated in the monetary authority disclaimer policy.

# Software System Attributes

* + 1. **Reliability**

The average time to failure shall be 30 days. In the event that a server does crash, a backup server will be up and running within the hour.

# Availability

The PayMe shall be available to users 24 hours a day, 7 days a week, with the exception of being down for maintenance no more than one hour a week. If the system crashes, it should be back up within one hour.

# Security

Users will be able to access only their own personal information and not that of other users. Purchases will be handled through a secure server to ensure the protection of user’s credit card and personal information.

# Maintainability

Any updates or defect fixes shall be able to be made on server-side computers only without any patches required by the user.

# Portability

Nothing required