

Use Case Details

Case Use Model

Version • Proposed



Date/Time
Generated:
Author:

25/04/2018 08:54:00 p.m.

TOSHIBA

EA Repository : D:\d_carpet\EAProjects\SPFG_v2\SPFG_cd\Documentation\SPGF.eap

Table of Contents

Case Use Model	3
Case Use diagram	3
Actors	4
Actors diagram	4
Use Case	5
SPFG diagram	5
Sign In	6
Sign Up	7
Modify User Profile	8
Chose Type of User	8
Technical Request Pool	9
Historical Data Overview	10
Technical Service Request	10
Categorized	11
Search a User	11
Send a Message	12
Delete User Account	13
Account Overview	13
Delete Account	14
Generate a report	15
Log Out	15
Add Goal	16
Recover Account	16
Personalize	17
Incomes Managment	17
Expenses Managment	18
Payment Queue	19

Case Use Model

Package in package 'Model'

Case Use Model

Version Phase 1.0 Proposed

Dave created on 31/03/2018. Last modified 24/04/2018

Case Use diagram

Use Case diagram in package 'Case Use Model '

The Use Case model is a catalogue of system functionality described using UML Use Cases. Each Use Case represents a single, repeatable interaction that a user or "actor" experiences when using the system.

A Use Case typically includes one or more "scenarios" which describe the interactions that go on between the Actor and the System, and documents the results and exceptions that occur from the user's perspective.

Use Cases may include other Use Cases as part of a larger pattern of interaction and may also be extended by other use cases to handle exceptional conditions.

Case Use

Version 1.0

Dave created on 31/03/2018. Last modified 25/04/2018

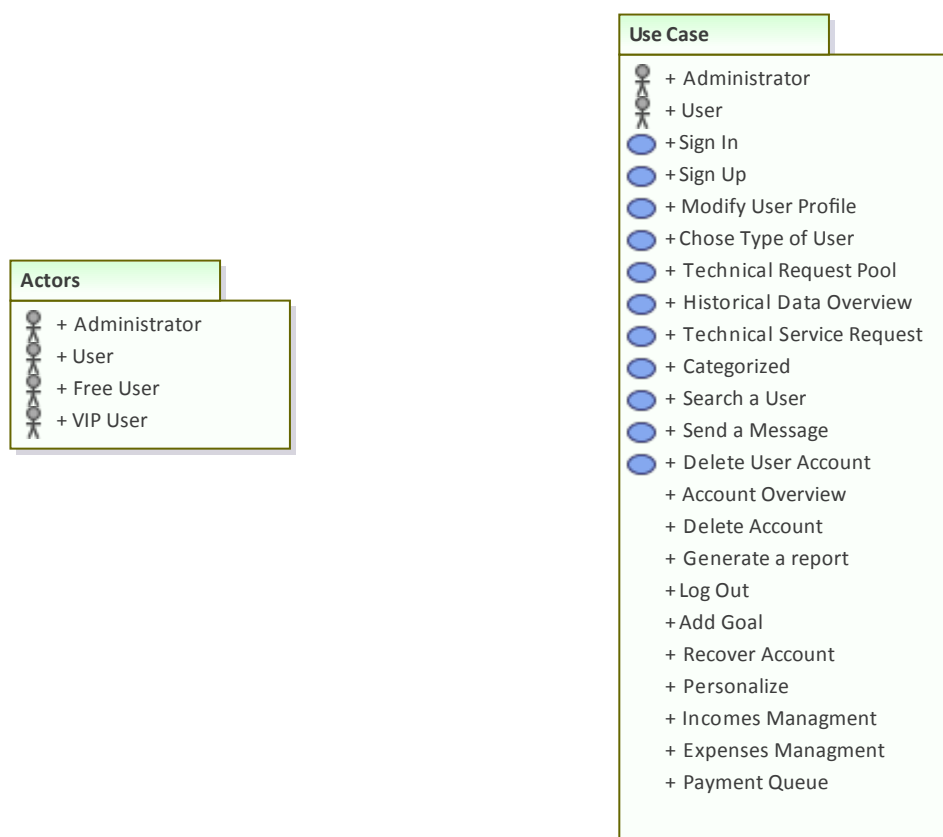


Figure 1: Case Use

Actors

Package in package 'Case Use Model '

Actors

Version 1.0 Phase 1.0 Mandatory

Dave created on 31/03/2018. Last modified 17/04/2018

Actors diagram

Use Case diagram in package 'Actors'

Actors are the users of the system being modeled. Each Actor will have a well-defined role, and in the context of that role have useful interactions with the system.

A person may perform the role of more than one Actor, although they will only assume one role during one use case interaction.

An Actor role may be performed by a non-human system, such as another computer program.

Actors

Version 1.0

Dave created on 31/03/2018. Last modified 17/04/2018

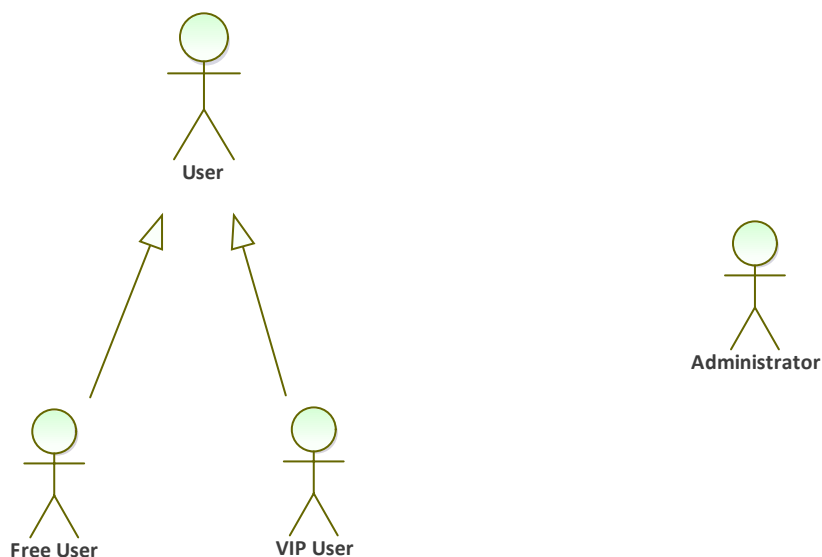


Figure 2: Actors

Use Case

Package in package 'Case Use Model '

Use Case

Version 1.0 Phase 1.0 Mandatory

Dave created on 31/03/2018. Last modified 17/04/2018

SPFG diagram

Use Case diagram in package 'Use Case'

This package contains use cases which define how an Actor will interact with the proposed system.

Each interaction may be specified using scenarios, sequence diagrams, communication diagrams and other dynamic diagrams or textual descriptions which together describe how the system, when viewed as a "black-box", interacts with a user.

SPFG

Version 1.0

TOSHIBA created on 31/03/2018. Last modified 25/04/2018

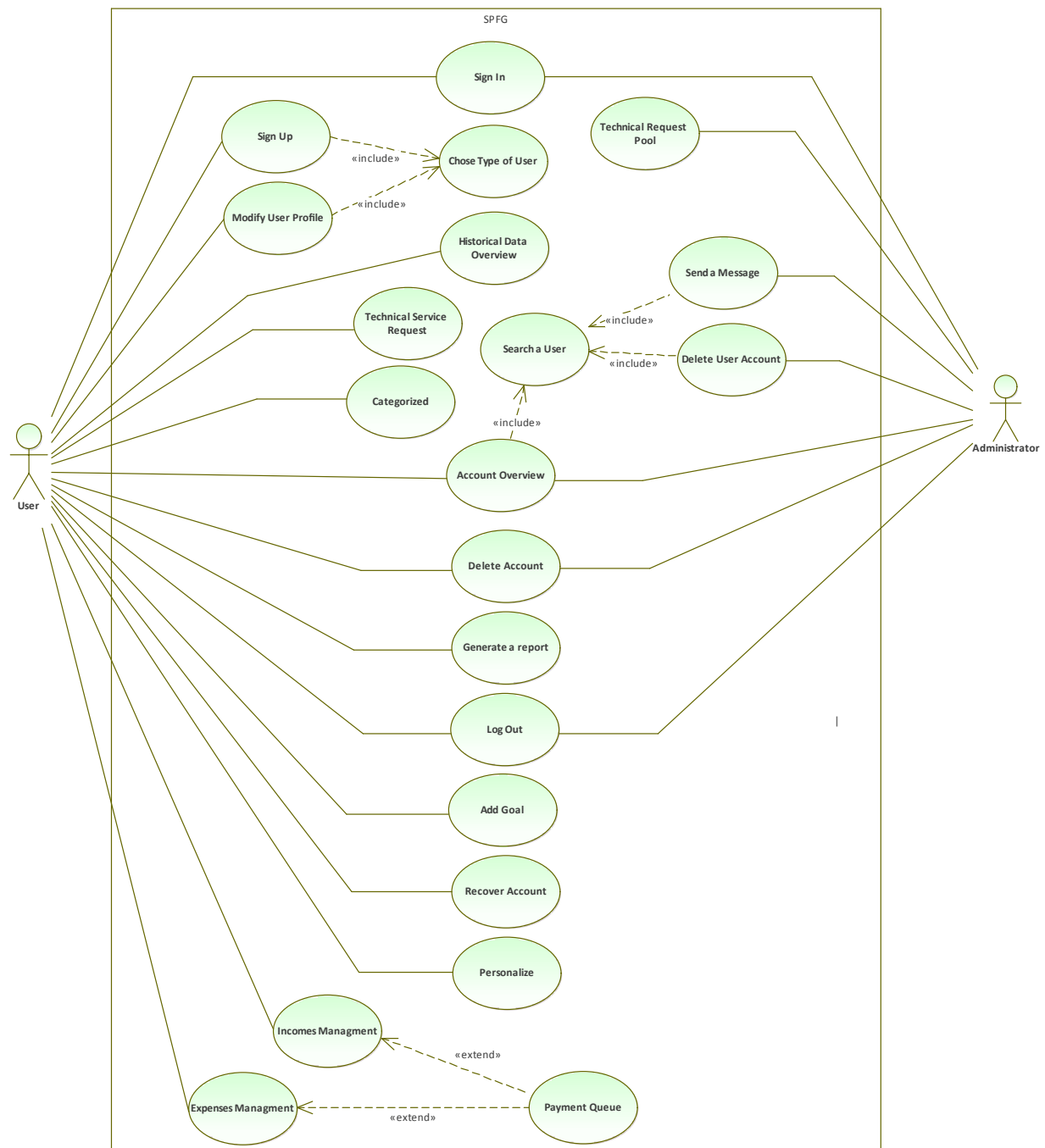


Figure 3: SPFG

Sign In

UseCase in package 'Use Case'

Objective:

Allows Users and Administrators to get into their account. Also, make a differentiation between roles.

Main flow:

- 1: A form is display.
- 2: The user writes his/her email and password.
- 3: The system designs a SQL request and send it to the database.
- 4: The database returns a positive answer.
- 5: The account can be accessed.

Secondary flow:


- 6: If the database returns a negative answer, then an error message is shown next to the form.


Sign In

Version 1.0 Phase 1.0 Mandatory

Dave created on 07/04/2018. Last modified 25/04/2018

CONNECTORS

 **UseCaseLink** Source -> Destination
From: User : Actor, Public
To: Sign In : UseCase, Public

 **UseCaseLink** Source -> Destination
From: Administrator : Actor, Public
To: Sign In : UseCase, Public

Sign Up

UseCase in package 'Use Case'

Objective:

Registers and saves data from new users. Besides, it allows you to choose a pricing plan.

Main Flow:

- 1: A form is display to acquire necessary information.
- 2: The user selects a pricing plan.
- 3: The use types his/her personal information.
 - 3.1: In case of free accounts, only name, email and password are necessary.
 - 3.2: Otherwise, a VIP user must add an extra email and a phone number.
- 4: In order to avoid SQL injection, the system should validate incoming data.
- 5: Once validation process has ended, the system design a SQL query.
- 6: The database looks for accounts with the same email.
- 7: If the database returns a positive answer; then new data is accepted.
- 8: The system displays a success message.

Secondary Flow:

- 4.1 If data validation is failed, then an error message is shown.
- 7.1 In case of a negative answer, incoming data is not accepted and the user must write a new account with different information.


Sign Up


Version 1.0 Phase 1.0 Mandatory

Dave created on 07/04/2018. Last modified 25/04/2018

CONNECTORS

CONNECTORS

 **Include** «include» Source -> Destination
 From: Sign Up : UseCase, Public
 To: Chose Type of User : UseCase, Public

 **UseCaseLink** Source -> Destination
 From: User : Actor, Public
 To: Sign Up : UseCase, Public

Modify User Profile

UseCase in package 'Use Case'

Objective:

Allows any kind of user to modify his/her personal data. This functionality includes an option to choose a pricing plan.

Main Flow:

- 1: A form is displayed to receive new information. Independently of the type of user, there are five fields to complete: Name, Email, Password are obligatory. Extra email and phone number are optional.
- 2: The user types his/her new personal information
- 3: In order to avoid SQL injection, the system should validate incoming data.
- 4: Once validation process has ended, the system design a SQL query.
- 6: The database looks for accounts with the same email.
- 7: If a positive answer is returned, then the new data is saved.
- 8: A success message is displayed.

Secondary Flow:


- 7.1 In case of a negative answer, a message error is displayed and the user should write new information.


Modify User Profile

Version 1.0 Phase 1.0 Mandatory

Dave created on 09/04/2018. Last modified 25/04/2018

CONNECTORS

 **Include** «include» Source -> Destination
 From: Modify User Profile : UseCase, Public
 To: Chose Type of User : UseCase, Public

 **UseCaseLink** Source -> Destination
 From: User : Actor, Public
 To: Modify User Profile : UseCase, Public

Chose Type of User

UseCase in package 'Use Case'

Objective:

Grants interpolation between free and VIP users. Once done, this option will be unable for a week.

a) For free user**Main Flow:**

- 1: A modal appear to receive a ticket number.
- 2: The users types the ticket number.
- 3: The system validates the ticket.
- 4: If exists an extra email and a phone number, the free account is upgraded.
- 4.1: Otherwise, the system will display a modal to receive extra information. After that, the free account is upgraded.
- 5: A success message is displayed.

b) For VIP user:**Main Flow:**

- 1: A modal appears to confirm a free account version.
- 2: In case of a positive answer, the system will return that account to a free version. VIP function will be deactivate automatically.
- 3: Finally a success message is displayed.

Secondary Flow:


- 4: This functionality will be applied automatically if a payment plan is canceled by unspecific situation.


Chose Type of User

Version 1.0 Phase 1.0 Mandatory

Dave created on 09/04/2018. Last modified 25/04/2018

CONNECTORS

 **Include** «include» Source -> Destination
From: Modify User Profile : UseCase, Public
To: Chose Type of User : UseCase, Public

 **Include** «include» Source -> Destination
From: Sign Up : UseCase, Public
To: Chose Type of User : UseCase, Public

Technical Request Pool

UseCase in package 'Use Case'

Objective:

The request pool is a place where all technical request are stored. There is just one request pool and only administrators will have the access to it.

Main Flow:


- 1: An administrator reviews the last n-technical request stored in the pool.
- 2: Any new request is marked as "Unsolved" until an administrator takes care of it.
- 3: If a solution is found for any technical request, then this is marked as "solved" by the administrator.
- 4: The system adds a solution date and writes administrator id next to the request.

Secondary Flow:

- 5: Also, the administrator can click in a option box to send a message for users.

Technical Request Pool
Version 1.0 Phase 1.0 Proposed
Dave created on 09/04/2018. Last modified 25/04/2018

CONNECTORS

 **UseCaseLink** Source -> Destination
From: Administrator : Actor, Public
To: Technical Request Pool : UseCase, Public

Historical Data Overview

UseCase in package 'Use Case'

Objective:


Makes an exhaustive analysis from the last weeks, months and even years.

Main Flow:

- 1: Each time an expense or income is added, deleted or modified, the daily balance is worked again and added in that day.
- 2: Subsequently and according to the daily balance, it's define as a "green day" or "red day".
- 3: When a user wants to see an historical data overview, the system designs a diagram based in every daily balance. This diagram is organized by years and months.
- 4: If a user selects an specific day, a modal appears and shows every incomes and expenses in that day.

Historical Data Overview
Version 1.0 Phase 1.0 Mandatory
Dave created on 16/04/2018. Last modified 25/04/2018

CONNECTORS

 **UseCaseLink** Source -> Destination
From: User : Actor, Public
To: Historical Data Overview : UseCase, Public

Technical Service Request

UseCase in package 'Use Case'

Objective:

Sends a request to the administrator in case of any problem.


Main Flow:

- 1: The user accesses to the "Report a Fail" modal.
- 2: The user types the description of the fault [Required].
- 3: The user selects an option:
 - 3.1: Send Solicitude
 - 3.2: Cancel
- 4: In case of "Send Solicitude", the system adds user information and designs a SQL query to send the

technical request to the pool. Otherwise the "Report Fail" modal is closed.

Technical Service Request
Version 1.0 Phase 1.0 Proposed
Percy created on 09/04/2018. Last modified 25/04/2018

CONNECTORS

 **UseCaseLink** Source -> Destination
From: User : Actor, Public
To: Technical Service Request : UseCase, Public

Categorized

UseCase in package 'Use Case'

Allows users to organize any kind of incomes and expenses in a category. A category is defined under an ID that represent a list from different Incomes/Expenses ID's. A category can be added, modified or deleted. Also, the system won't allow categories with the same name.


By default the system will provide you with a Category called "Others"*. This category can't be deleted since it's the pool for uncategorized incomes/expense. Also, categories could have the same name if they're not in the same environment**.

* Obviously, there would be a default category for Expenses and a default category for Incomes.

**Environments: Expenses and Incomes.

Categorized
Version 1.0 Phase 1.0 Mandatory
Dave created on 16/04/2018. Last modified 25/04/2018

CONNECTORS

 **UseCaseLink** Source -> Destination
From: User : Actor, Public
To: Categorized : UseCase, Public

Search a User

UseCase in package 'Use Case'

Objective:


Searches a user.


Main Flow:


- 1: A form is displayed with different fields to make a better search.
- 2: The administrator types the information in the fields.
- 3: The system accesses to the database.
- 4: The system returns the records founded.

Search a User
Version 1.0 Phase 1.0 Mandatory
Percy created on 09/04/2018. Last modified 25/04/2018

CONNECTORS

 **Include** «include» Source -> Destination
From: Delete User Account : UseCase, Public
To: Search a User : UseCase, Public

 **Include** «include» Source -> Destination
From: Account Overview : UseCase, Public
To: Search a User : UseCase, Public

 **Include** «include» Source -> Destination
From: Send a Message : UseCase, Public
To: Search a User : UseCase, Public

Send a Message

UseCase in package 'Use Case'

Objective:


The administrator can send messages to the users.

Main flow:


- 1: The administrator presses the button to see users.
- 2: A list of users by license is shown.
 - 2.1: Free users.
 - 2.2: VIP users.
- 3: Write the message you want to be sent [Required].
- 4: The administrator can:
 - 4.1: Search for a specific user. [ID, Name, Mail]
 - 4.2: Select the users to send the message.
- 5: Finally, select:
 - 5.1 Submit.
 - 5.2 Cancel.

Send a Message
Version 1.0 Phase 1.0 Proposed
Percy created on 09/04/2018. Last modified 25/04/2018

CONNECTORS

 **Include** «include» Source -> Destination
From: Send a Message : UseCase, Public
To: Search a User : UseCase, Public

CONNECTORS

 **UseCaseLink** Source -> Destination
 From: Administrator : Actor, Public
 To: Send a Message : UseCase, Public

Delete User Account

UseCase in package 'Use Case'

Objective:

Modify the status of a user.

Main flow:


- 1: The administrator enters the panel to delete accounts.
- 2: Search an User.
- 3: A text box is displayed.
- 4: The administrator types the reason why the user is banned.
- 5: The administrator modifies the status of the account.
- 6: Select:
 - 6.1: Save.
 - 6.2: Cancel.


Delete User Account

Version 1.0 Phase 1.0 Mandatory

maldonado created on 21/04/2018. Last modified 25/04/2018

CONNECTORS

 **Include** «include» Source -> Destination
 From: Delete User Account : UseCase, Public
 To: Search a User : UseCase, Public

 **UseCaseLink** Source -> Destination
 From: Administrator : Actor, Public
 To: Delete User Account : UseCase, Public

Account Overview

UseCase in package 'Use Case'

Objective:

Shows the organized information of the user's money management.

Main flow:

- 1: The user accesses Overview.
- 2: The user information is obtained [income / expenses - Objectives].
- 3: The defined objectives and their progress are shown in a segment.
- 4: The organized information of the user is displayed:
 - 4.1: Frequency: Monthly.

4.2: Type of Graph: Linear.

5: The user can:

5.1 Change the type of Graphic.

- bars.
- linear
- circular.
- area.
- histogram.

5.2 Change the Frequency.


- weekly
- monthly.
- annual.


Account Overview


Version 1.0 Phase 1.0 Mandatory

maldonado created on 09/04/2018. Last modified 25/04/2018

CONNECTORS

 **Include** «include» Source -> Destination
From: Account Overview : UseCase, Public
To: Search a User : UseCase, Public

 **UseCaseLink** Source -> Destination
From: Administrator : Actor, Public
To: Account Overview : UseCase, Public

 **UseCaseLink** Source -> Destination
From: User : Actor, Public
To: Account Overview : UseCase, Public

Delete Account

UseCase in package 'Use Case'

Objective:

Cancels any payment plan and inactivates your account.

Main flow:

1: Inside the profile page you click the button "Delete account".

2: Appears a form that asks you to enter your password in order to delete your account.

2.1: If the password was correct, inside the database your account now has a type: inactive, all the payment plans are canceled and the system automatically logs out and you are redirected to the login page.

2.2: Otherwise, it shows you a message "Invalid password, enter it again".


Delete Account


Version 1.0 Phase 1.0 Proposed

Alonso created on 09/04/2018. Last modified 25/04/2018

CONNECTORS

CONNECTORS

 **UseCaseLink** Source -> Destination
 From: User : Actor, Public
 To: Delete Account : UseCase, Public

 **UseCaseLink** Source -> Destination
 From: Administrator : Actor, Public
 To: Delete Account : UseCase, Public

Generate a report

UseCase in package 'Use Case'

Objective:

Generates a pdf file with a summary of your expenses and income (including graphics and numbers/text).

Main flow:


- 1: Inside the Overview page, you click the button "Generate report".
- 2: Appears a form that asks you for the number of the last months that you want your report to be generated based on that.
3. If you enter a correct number and you click "Generate", it is downloaded a pdf file with a summary of your expenses and income of the last N months.

Generate a report

Version 1.0 Phase 1.0 Proposed

Dave created on 09/04/2018. Last modified 25/04/2018

CONNECTORS

 **UseCaseLink** Source -> Destination
 From: User : Actor, Public
 To: Generate a report : UseCase, Public

Log Out

UseCase in package 'Use Case'

Objective:

Takes you out of session and the next time you want enter to the system it will ask you for your password.

Main Flow:


- 1: If you click in your profile bar, a frame will appear with summed information of your profile and also a "Log out" button.
- 2: If you click in the "Log out" button, you will be redirected to the login page and the next time you want to enter to the system it will ask you for your password.


Log Out

Version 1.0 Phase 1.0 Mandatory

Alonso created on 09/04/2018. Last modified 25/04/2018

CONNECTORS

 **UseCaseLink** Source -> Destination
 From: User : Actor, Public
 To: Log Out : UseCase, Public

 **UseCaseLink** | Source -> Destination
 From: Administrator : Actor, Public
 To: Log Out : UseCase, Public

Add Goal

UseCase in package 'Use Case'

Objective:

Show you a list of bars that indicate how much you goals are completed. This is a good way to indicate you what kind of things you can buy.

Main Flow:


- 1: In the overview page there will be a frame with the current goals and below it a button "Add Goal".
- 2: If you click that button it will appear a form to add a new goal (name, percentage of the total money you want to designate for that goal, price of the goal), and a button "Add".
- 3: If the form was entered correctly, your goal was added, and you will see it in the overview page, inside the list of goals.

Add Goal

Version 1.0 Phase 1.0 Mandatory

Alonso created on 16/04/2018. Last modified 25/04/2018

CONNECTORS

 **UseCaseLink** Source -> Destination
 From: User : Actor, Public
 To: Add Goal : UseCase, Public

Recover Account

UseCase in package 'Use Case'

Objective:

Makes the type of your account active so that you can use it again.

Main Flow:

- 1: You try to login as a user that deleted his/her account.
- 2: If the data entered was correct.
 - 2.1 An email is sent with a link so you can login properly.
 - 2.2: You will be redirected to a page that says that you are recovering your account and you must go to your respective email to confirm the recovering.
 - 2.3 If you click on the link in the email, the database will change the type of your account to active, and


you will go the overview page of your account as a normal login, and you recovered your account.

Recover Account

Version 1.0 Phase 1.0 Mandatory

Alonso created on 22/04/2018. Last modified 25/04/2018

CONNECTORS

 **UseCaseLink** Source -> Destination
From: User : Actor, Public
To: Recover Account : UseCase, Public

Personalize

UseCase in package 'Use Case'

Objective:

Change the colors and the fonts of the design of the system.

Main Flow:


- 1: Inside your profile settings, you click on the button "Personalize".
- 2: There will appear a box with colors to choose so you can change the colors of the design of the system, a list of font so you can change the letter fonts of all the system, and 2 buttons "Accept" and "Cancel".
3. If you click on "Accept" the changes will be done, or if you click on "Cancel" the changes will be discarded,

Personalize

Version 1.0 Phase 1.0 Proposed

Alonso created on 22/04/2018. Last modified 25/04/2018

CONNECTORS

 **UseCaseLink** Source -> Destination
From: User : Actor, Public
To: Personalize : UseCase, Public

Incomes Managment

UseCase in package 'Use Case'

Objective:

Complex functionality to add or delete incomes.

a) Creates an Income:

Main Flow:

- 1: A form is display to acquired necessary information.
- 2: The user must type in the next fields: Name[Obligatory], Amount[Obligatory] , Payment Plan[Not obligatory], Category[Not obligatory] and Date[Automatic].
- 3: Selects "Create" option. It will send the new income to the database.

b) Deletes an Income:

Main Flow:


- 1: The user selects an existing income.
- 2: The user presses the "delete" button.
- 3: The user confirms the action.
- 4: A success message is displayed.


Incomes Managment

Version 1.0 Phase 1.0 Mandatory

Dave created on 09/04/2018. Last modified 25/04/2018

CONNECTORS

 **Extend** «extend» Source -> Destination
From: Payment Queue : UseCase, Public
To: Incomes Managment : UseCase, Public

 **UseCaseLink** Source -> Destination
From: User : Actor, Public
To: Incomes Managment : UseCase, Public

Expenses Managment

UseCase in package 'Use Case'

Objective:

Complex functionality to add or delete expenses.

a) Creates an Expense:

Main Flow:

- 1: A form is display to acquired necessary information.
- 2: The user must type in the next fields: Name[Obligatory], Amount[Obligatory] , Payment Plan[Not obligatory], Category[Not obligatory] and Date[Automatic].
- 3: Selects "Create" option. It will send the new income to the database.

b) Deletes an Expense:

Main Flow:

- 1: The user selects an existing expense.
- 2: The user presses the "delete" button.
- 3: The user confirms the action.
- 4: A success message is displayed.


Expenses Managment


Version 1.0 Phase 1.0 Mandatory

Dave created on 09/04/2018. Last modified 25/04/2018

CONNECTORS

CONNECTORS

 **Extend** «extend» Source -> Destination
 From: Payment Queue : UseCase, Public
 To: Expenses Managment : UseCase, Public

 **UseCaseLink** Source -> Destination
 From: User : Actor, Public
 To: Expenses Managment : UseCase, Public

Payment Queue

UseCase in package 'Use Case'

Objective:

Implement a queue to establish a payment plan.

Main Flow:

- 1: When a user saves an income or expense, an option to create a payment plan is display.
- 2: The user can create a payment plan with those days in which an expense or income is gonna be effected.

By default each expense or income is unique.

- 3: Once done, the user press the button "Create"

4: The system add the new payment plan to the user account. In case of VIP user, an alarm function will activate to send notifications.

Secondary Flow:


2.1 By default and only if a user do not want to create a payment plan, the income or expense will be saved as unique.


Payment Queue

Version 1.0 Phase 1.0 Mandatory

Dave created on 13/04/2018. Last modified 25/04/2018

CONNECTORS

 **Extend** «extend» Source -> Destination
 From: Payment Queue : UseCase, Public
 To: Expenses Managment : UseCase, Public

 **Extend** «extend» Source -> Destination
 From: Payment Queue : UseCase, Public
 To: Incomes Managment : UseCase, Public