

# Requirements Document

## Team Alpha Finance

17 February 2015

Table 1: Team

| Name                     | ID Number |
|--------------------------|-----------|
| Patrick Cristofaro       | 5410282   |
| Aymeric Grail            | 6810020   |
| Richard Kallos           | 6939325   |
| Khodayar Jeirroodi       | 7580525   |
| Luis M. Saravia Patron   | 6800505   |
| Victoria Zaytseva        | 7367821   |
| Ahmed Shaheen            | 9736190   |
| Andre-Philippe Cianflone | 5239214   |
| Khaled El-Badawi         | 27182643  |
| David Abran-Cote         | 7142587   |

# Contents

|           |                                      |          |
|-----------|--------------------------------------|----------|
| <b>1</b>  | <b>Purpose of the document</b>       | <b>3</b> |
| <b>2</b>  | <b>Business goals</b>                | <b>3</b> |
| <b>3</b>  | <b>Main domain concepts</b>          | <b>3</b> |
| <b>4</b>  | <b>System overview</b>               | <b>4</b> |
| <b>5</b>  | <b>User groups</b>                   | <b>4</b> |
| <b>6</b>  | <b>Functional requirements</b>       | <b>4</b> |
| <b>7</b>  | <b>Non-functional requirements</b>   | <b>4</b> |
| <b>8</b>  | <b>User requirements (Use Cases)</b> | <b>4</b> |
|           | Overview . . . . .                   | 4        |
|           | Use Case 1 . . . . .                 | 5        |
|           | Use Case 2 . . . . .                 | 6        |
|           | Use Case 3 . . . . .                 | 6        |
|           | Use Case 4 . . . . .                 | 7        |
|           | Use Case 5 . . . . .                 | 8        |
|           | Use Case 6 . . . . .                 | 8        |
| <b>9</b>  | <b>Constraints</b>                   | <b>9</b> |
| <b>10</b> | <b>Solution ideas</b>                | <b>9</b> |
| <b>11</b> | <b>Acronyms and abbreviations</b>    | <b>9</b> |
| <b>12</b> | <b>References</b>                    | <b>9</b> |

# 1 Purpose of the document

The purpose of this document is to define the user requirement of the Personal Budget Manager Application, which is the project for COMP 5541. It will provide information on the subjects such as main concepts, user groups and functional and nonfunctional requirements of the project. The intended audiences of this document are described in table 2. Audience.

Table 2: Audiences

| Audience               | Purpose   |
|------------------------|---|
| Users and Customers    | To be familiar with the application and its features, also to give feedback about the requirements and confirm them |
| Application Developers | To fully understand what functions and properties the system must contain from the customer point of view           |
| Application Testers    | To test the system against the requirements   |
| Documentation Team     | To document the different steps of the project and make the required documents (this document, user manuals, etc.)  |

# 2 Business goals

The Personal Budget Manager Application provides users with an execution environment that can be used to control personal expenses by entering the expenses and allowing the user to review or edit the entered data. It is distinguished between two kind of expenses: Purchase expenses and Bill expenses. Also there are some planned future extensions of the Personal Budget Manager such as addition of new expense types, support for multiple user interfaces (e.g., Web UI, Mobile UI) and multiple presentations of the expense list (e.g., using trees or lists).

# 3 Main domain concepts

pc-This chapter gives a brief introduction to the problem domain. It might include Textual definitions of the most important domain concepts  
A conceptual data model defining the relationship between the most important domain concepts

## 4 System overview

pc-This section is a high level description of the intended solution (=the system). It might include

a list of essential features of the system

a graph (for example a use case diagram) that defines the users and the main functions of the system

## 5 User groups

## 6 Functional requirements

pc-This section defines the functionality (services) that the system must provide in order to meet the business goals.

Functional requirements can be organized in several ways, e.g. by feature or by use case. The layout can be determined by each group the example layout below can be used.

## 7 Non-functional requirements

## 8 User requirements (Use Cases)

### Overview

A Use Case is a piece of functionality in the system. Those pieces will return a value or perform a service to a user, in this application we have two actors : *User and System*. In table 3 the use cases of this Application are mentioned, also Figure 1, depicts a diagram for use cases.

Figure 1: Use Case Diagram

### Use Case 1

#### Name

Login

#### Summary

User must login into the software in order to enter or view the data. The characters in the password field are masked.

Table 3: Use case table

| Use case ID | Name       | Primary Actor |
|-------------|------------|---------------|
| 1           | Login      | User          |
| 2           | Data Entry | User          |
| 3           | Edit       | User          |
| 4           | Delete     | User          |
| 5           | View       | User          |
| 6           | Logout     | User          |

### Actors

User

### Precondition

User name and password must be predefined in the system

### Main Scenario

1. User runs the application
2. User enters username (ID) and the related password and press the *login* button
3. If the User enters the valid username and password she/he will have access to the application main dashboard

### Exceptions

If any of the username or password fields be empty or not be valid after clicking the login button an error message should warn the user to enter or correct the data. The system will remain in the login form state.

### Postcondition

after entering the correct username and password the application main interface will be shown.

### Priority

Must

### Traces to Test Cases

Add when test cases done.

## Use Case 2

### Name

DataEntry

### Summary

User fills required field to enter a new expense into the software

**Actors**

User

**Precondition**

login

**Main Scenario**

1. User selects among the two types of expenses (bills or purchases) for a new entry
2. User enters (selects?) the required information in the fields: description, amount, date and then status and repetition interval based on the type of the expense determined in the 1st step
3. User adding the new expense entry by clicking on a enter button

**Exceptions**

If any of the mentioned filed be empty or not be in a proper format after clicking the enter button an error message should warn the user to complete or correct the data.

**Postcondition**

After entering the data the new record should be appear in the table part of the UI

**Priority**

-

**Traces to Test Cases**

Add when test cases done.

**Use Case 3****Name**

Edit

**Summary**

User finds an entry and changes the filed paid/unpaid

**Actors**

User

**Precondition**

login

**Main Scenario**

1. User searches (scrolls?) among the entered expenses
2. User finds the desired entry and clicks on it
3. That entrys data appears in the data entry area of the UI
4. User changes the status Paid/Unpaid then submits

**Exceptions**

-

**Postcondition**

after submitting the data the payment status of the record must be changed to new status in the table part of the UI

**Priority**

-

**Traces to Test Cases**

Add when test cases done.

**Use Case 4****Name**

view

**Summary**

User can view/ scroll or search among the records

**Actors**

User

**Precondition**

login

**Main Scenario**

1. User searches (scrolls?) among the entered expenses
2. User searches for a desired record ??

**Exceptions**

If there has not been any data entry the user will see the empty table.

**Postcondition**

-

**Priority**

-

**Traces to Test Cases**

Add when test cases done.

**Use Case 5****Name**

Delete

**Summary**

User finds an entry and deletes it

**Actors**

User

**Precondition**

login

**Main Scenario**

1. User searches (scrolls?) among the entered expenses
2. User finds the desired entry and clicks on the delete icon beside it (or click on it and in the entry menu click on delete button?)
3. That entry's data doesn't appear in the data entry area of the UI anymore

**Exceptions**

If there has not been any data entry the user will see the empty table.

**Postcondition**

After deleting a record, its content must no longer exist in the table

**Priority**

-

**Traces to Test Cases**

Add when test cases done.

**Use Case 6****Name**

Logout

**Summary**

User logs out of the application

**Actors**

User

**Precondition**

login

**Main Scenario**

1. User clicks on the button *logout*

**Exceptions**

-

**Postcondition**

After logging out the user can't do any of the use cases 2-5. User has to login in order to use the application.



**Priority**

-

**Traces to Test Cases**

Add when test cases done.

## **9 Constraints**

## **10 Solution ideas**

## **11 Acronyms and abbreviations**

## **12 References**