# 

# 

# 

# 

# CSC 431 FundsBook Software Requirements Specification (SRS)

**Team 16**

|  |  |
| --- | --- |
| Luis Diaz | Developer |
| Julio Ojalvo | Developer |
| Temuulen Ganbold | Developer |

# Version History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author(s) | Change Comments |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Table of Contents

[1.](#_3dy6vkm) System Requirements 6

[1.1](#_1t3h5sf) Functional Requirements 6

[1.1.1](#_4d34og8) Requirement Title 6

[1.2](#_2s8eyo1) Non-Functional Requirements 6

[1.2.1](#_17dp8vu) Requirement Title 6

[2.](#_3rdcrjn) System Constraints 7

[2.1](#_26in1rg) Tool Constraints 7

[2.1.1](#_lnxbz9) Requirement Title 7

[2.2](#_35nkun2) Language Constraints 7

[2.2.1](#_1ksv4uv) Requirement Title 7

[2.3](#_44sinio) Platform Constraints 7

[2.3.1](#_2jxsxqh) Requirement Title 7

[2.4](#_z337ya) Hardware Constraints 7

[2.4.1](#_3j2qqm3) Requirement Title 7

[2.5](#_1y810tw) Network Constraints 7

[2.5.1](#_4i7ojhp) Requirement Title 8

[2.6](#_2xcytpi) Deployment Constraints 8

[2.6.1](#_1ci93xb) Requirement Title 8

[2.7](#_3whwml4) Transition & Support Constraints 8

[2.7.1](#_2bn6wsx) Requirement Title 8

[2.8](#_qsh70q) Budget & Schedule Constraints 8

[2.8.1](#_3as4poj) Requirement Title 8

[2.9](#_1pxezwc) Miscellaneous Constraints 8

[2.9.1](#_49x2ik5) Requirement Title 8

[3.](#_2p2csry) Requirements Modeling 10

[3.1.1](#_147n2zr) Requirement Title 10

[4.](#_3o7alnk) Evolutionary Requirements 11

[4.1](#_23ckvvd) Functional Requirements 11

[4.1.1](#_ihv636) Requirement Title 11

[4.2](#_32hioqz) Non-Functional Requirements 11

[4.2.1](#_1hmsyys) Requirement Title 11

# Table of Tables

<Generate table here>

# Table of Figures

<Generate table here>

### System Requirements

#### Functional Requirements

*< List all functional requirements in the following example format >*

##### Personal Profile

|  |  |
| --- | --- |
| Title | Creating and Editing a Profile |
| Description | Users should be able to create a personal profile that contains information such as their name, location, as well as a profile picture. They should have the option to set their profile to private or public depending on whether they want to allow others other than friends to see their profile and budgeting plans. |
| Priority | 0 |
| Precondition(s) | The UI of the app must be functional and the app is fully downloaded |
| Basic Flow | * User downloads the app * User opens app * Users are prompted to sign in if they have an existing account or create a new account if it’s their first time using the application. * When creating a new account, the user is asked to enter basic personal information such as name and date of birth to confirm their age. * User is asked to set email and password along with security questions * User is able to personalize their profile as they please * User can choose what information is shown to their close friends e.g. time spent following budget, amount of money going into each category |
| Postconditions(s) | The user now has access to all the functionalities of the app |
| Use Case Diagram | Figure 1 (All use case diagrams will be submitted in the next draft) |

##### Personal Budget

|  |  |
| --- | --- |
| Title | Creating a personal budget |
| Description | Ability to create a personal budget plan and put money into certain savings accounts based on the plan created by the user. |
| Priority | 0 |
| Precondition(s) | The UI of the app must be functional and the user must have a profile |
| Basic Flow | * User clicks a button to enter into a screen for editing budgets * User has a box labeled “category” to type in the name of one of the aspects of the plan e.g. car payments, cell phone bill, credit cards * Under each category, the user can input what percent of the total budget is going towards it * User can choose what color that category will be represented in * A pie chart will be shown to the side to demonstrate the totals of each category in the color chosen * If the user inputs a percent that brings the total over 100%, a warning is given and the number is removed * Once, the user is done and clicks the finish button they are given a notification if their budget plan does not reach 100% total but they are allowed to continue and that remaining money will be left untouched |
| Postconditions(s) | Once the main budget plan is made, auxiliary budget plans are now available to be made either for the use of other users or for personal use at another point in time |
| Use Case Diagram | Figure 2 |

##### Auxiliary Budgets

|  |  |
| --- | --- |
| Title | Creating Auxiliary Budget Plans |
| Description | Ability to create secondary budget plans set for different priorities for personal use or for others to follow |
| Priority | 2 |
| Precondition(s) | User must have a profile and have a main budget plan already created |
| Basic Flow | * Once the main budget is completed, another button will be available to enter into the same process to create a budget * This time, the budget plan will have the option of having a unique name to identify it with * Tags can also be added to the plan, similar to hashtags on traditional social media, that helps users more easily find that plan based on similar search terms * Users will have the option to switch an auxiliary budget plan to their main one at any given moment |
| Postconditions(s) | User is able to have multiple budget plans with different set priorities and be able to invest into whichever they please as well as share those budget plans with others |
| Use Case Diagram | Figure 3 |

##### Followers

|  |  |
| --- | --- |
| Title | Ability to Follow and use Others’ Budget Plans |
| Description | In the given case that successful millionaires/users decide to interact with this app and be some sort of guidance for other users, users should be able to follow those influencers and financial advisor to be able to use as a reference to meet their budgeting plans |
| Priority | 4 |
| Precondition(s) | Must have a profile to follow from |
| Basic Flow | * Users can click on other’s profiles to see their plans as long as the profile is public * If the profile is private, an option to request to follow them is given and a request is sent to that profile owner where they can accept or deny it * If the profile is public, an option to follow is given which when clicked will go through without any further processing * Once a profile is followed, the user will now receive notifications when that profile makes a change to a budget plan or makes a new one * In the settings, the user can choose to receive the notification on their phone or just a message that can only be seen once the app is opened |
| Postconditions(s) | Friending can now be done once a profile is followed |
| Use Case Diagram | Figure 5 |

##### Friends

|  |  |
| --- | --- |
| Title | Ability to Add Others as Friends |
| Description | Users should be able to interact with other users, having the option to add other users as “friends” and be able to see how they are doing with their budget plans and in a way give each other motivation to reach those goals. |
| Priority | 4 |
| Precondition(s) | Two people must follow each other in order to become friends |
| Basic Flow | * Once two users have chose to follow each other, they will become friends |
| Postconditions(s) | Users can now see more detailed information the their friends and choose who in their friends list is on their close friends list |
| Use Case Diagram | Figure 4 |

##### Close Friends

|  |  |
| --- | --- |
| Title | Ability to designate close friends |
| Description | Users can choose who on their friends list is a close friend |
| Priority | 4 |
| Precondition(s) | User must be friends with someone in order to have the choice of adding them to their own close friends list |
| Basic Flow | * When a user looks at their own profile, there will be a tab to see your close friends * Initially it will say “No close friends yet” and prompt them to click on a button that will take them to a list of their friends * From their they can click on all the friends they would like to add to their close friends list and allow to see the previously chosen information |
| Postconditions(s) | A user that has been added to another’s friends list can see more detailed information when they check their profile |
| Use Case Diagram | Figure 5 |

##### Searching

|  |  |
| --- | --- |
| Title | Ability to search for users or budget plans |
| Description | Users will have a search bar they can use to find either a certain type of budget plan or another user’s profile |
| Priority | 1 |
| Precondition(s) | Must have a profile |
| Basic Flow | * One of the main tabs on the app will bring the users to the search bar * On the search bar, users can type a keyword to find a certain type of budget plan e.g. long term investments, saving for an expensive car in a short amount of time, paying off high interest loans or credit cards (tags on budget plans will be used to bring the user to a budget plan most similar to their search) * Users can also search for other users based on their profile name |
| Postconditions(s) | Once a user is found, they can be followed and once a plan is found it can be saved as an auxiliary plan that can be switched over to the main budget plan at any moment |
| Use Case Diagram | Figure 5 |

##### Achievements

|  |  |
| --- | --- |
| Title | Achievement system |
| Description | An achievement system that will reward users for making budget plans, using tags in them, and following their own budget plans for long periods of time |
| Priority | 3 |
| Precondition(s) | A main budget plan must be made and a profile to display the achievements from |
| Basic Flow | * Users will receive a small badge on their profile if they fulfill certain achievements such as   + Following a budget plan for one week, one month, one year, and every subsequent year   + Making 5 different budget plans   + Adding 10, 20, and 50 different tags among all the plans made |
| Postconditions(s) | Other users can now see that badge on their profile if the user who earned it chose to display it |
| Use Case Diagram | Figure 5 |

#### Non-Functional Requirements

*< List all non-functional requirements in the following example format >*

##### Security

|  |  |
| --- | --- |
| Title | Profile Security |
| Description | Users should be able to set a password for their account along with security questions to verify their identity. This is done with the purpose of others not having access to the app as it is planned to be connected with a users bank account/income to better reflect their standing in meeting their goals |
| Priority | 0 |
| Applicable FR(s) | Personal Profile, Personal Budget, Auxiliary Budget |

##### Privacy

|  |  |
| --- | --- |
| Title | Privacy Settings |
| Description | Users should be able to set their profiles to either be public or private and be able to manage who and what information others can see |
| Priority | 0 |
| Applicable FR(s) | Applicable to all functional requirements as the privacy settings affect all the listed functional requirements in it’s own ways |

### System Constraints

#### Tool Constraints

*< List all tool constraints in the following example format >*

##### Requirement Title

|  |  |
| --- | --- |
| Title | <Insert title> |
| Description | <A one or two sentence description> |
| Priority | <Priority from 0 (highest) – 5 (lowest)> |

#### Language Constraints

*< List all language constraints in the following example format >*

##### Requirement Title

|  |  |
| --- | --- |
| Title | <Insert title> |
| Description | <A one or two sentence description> |
| Priority | <Priority from 0 (highest) – 5 (lowest)> |

#### Platform Constraints

*< List all platform constraints in the following example format >*

##### Requirement Title

|  |  |
| --- | --- |
| Title | <Insert title> |
| Description | <A one or two sentence description> |
| Priority | <Priority from 0 (highest) – 5 (lowest)> |

#### Hardware Constraints

*< List all hardware constraints in the following example format >*

##### Requirement Title

|  |  |
| --- | --- |
| Title | <Insert title> |
| Description | <A one or two sentence description> |
| Priority | <Priority from 0 (highest) – 5 (lowest)> |

#### Network Constraints

*< List all network constraints in the following example format >*

##### Requirement Title

|  |  |
| --- | --- |
| Title | <Insert title> |
| Description | <A one or two sentence description> |
| Priority | <Priority from 0 (highest) – 5 (lowest)> |

#### Deployment Constraints

*< List all deployment constraints in the following example format >*

##### Requirement Title

|  |  |
| --- | --- |
| Title | <Insert title> |
| Description | <A one or two sentence description> |
| Priority | <Priority from 0 (highest) – 5 (lowest)> |

#### Transition & Support Constraints

*< List all transition & support constraints in the following example format >*

##### Requirement Title

|  |  |
| --- | --- |
| Title | <Insert title> |
| Description | <A one or two sentence description> |
| Priority | <Priority from 0 (highest) – 5 (lowest)> |

#### Budget & Schedule Constraints

*< List all budget & schedule constraints in the following example format >*

##### Requirement Title

|  |  |
| --- | --- |
| Title | <Insert title> |
| Description | <A one or two sentence description> |
| Priority | <Priority from 0 (highest) – 5 (lowest)> |

#### Miscellaneous Constraints

*< List all miscellaneous constraints in the following example format >*

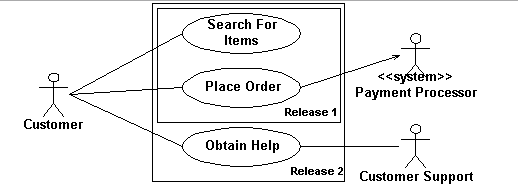
##### Requirement Title

|  |  |
| --- | --- |
| Title | <Insert title> |
| Description | <A one or two sentence description> |
| Priority | <Priority from 0 (highest) – 5 (lowest)> |

### Requirements Modeling

*< List all Use-case diagrams for the functional requirements in the following format>*

##### Requirement Title



### Evolutionary Requirements

#### Functional Requirements

*< List all functional requirements in the following example format >*

##### Requirement Title

|  |  |
| --- | --- |
| Title | <Insert title> |
| Description | <A one or two sentence description> |
| Priority | <Priority from 0 (highest) – 5 (lowest)> |
| Precondition(s) | <What needs to happen before> |
| Postconditions(s) | <What happens as a result> |
| Use Case Diagram | <Link or number, if present> |

#### Non-Functional Requirements

*< List all non-functional requirements in the following example format >*

##### Requirement Title

|  |  |
| --- | --- |
| Title | <Insert title> |
| Description | <A one or two sentence description> |
| Priority | <Priority from 0 (highest) – 5 (lowest)> |
| Applicable FR(s) | <Which functional requirement(s) is this applicable to?> |