Software Requirements Specification

Version 1.0

August20, 2019

Find-Sell-Give Mobile App

# Table of Contents

[Table of Contents i](#_Toc17471835)

[List of Figures iv](#_Toc17471836)

[1. Introduction 1](#_Toc17471837)

[1.1. Purpose 1](#_Toc17471838)

[1.2. Scope of Project 1](#_Toc17471839)

[1.3. Overview of Document 1](#_Toc17471840)

[2. Overall Description 2](#_Toc17471841)

[2.1. System Environment 2](#_Toc17471842)

[2.2. Functional Requirements Specification 2](#_Toc17471843)

[2.2.1. Unauthenticated User Use Case 3](#_Toc17471844)

[2.2.1.1. Use case: Register user 3](#_Toc17471845)

[2.2.1.2. Use case: User sign in 3](#_Toc17471846)

[2.2.2. Authenticated User Use Case 4](#_Toc17471847)

[2.2.2.1. Use case: Display all products 4](#_Toc17471848)

[2.2.2.2. Use case: Display the products to be sold 5](#_Toc17471849)

[2.2.2.3. Use case: Display the products to be given 5](#_Toc17471850)

[2.2.2.4. Use case: Add product to “My Collection” 6](#_Toc17471851)

[2.2.2.5. Use case: Contact User 6](#_Toc17471852)

[2.2.2.6. Use case: Rate user 7](#_Toc17471853)

[2.2.2.7. Use case: Consult wish list 8](#_Toc17471854)

[2.2.2.8. Use case: Comment on wish list 9](#_Toc17471855)

[2.2.2.9. Use case: display user’s profile 9](#_Toc17471856)

[2.2.2.10. Use case: display user’s products 10](#_Toc17471857)

[2.2.2.11. Use case: display user’s wish list 10](#_Toc17471858)

[2.2.2.12. Use case: add product 11](#_Toc17471859)

[2.2.2.13. Use case: add to wish list 12](#_Toc17471860)

[2.2.2.14. Use case: reply to a message 12](#_Toc17471861)

[2.2.2.15. Use case: edit profile 13](#_Toc17471862)

[2.2.2.16. Use case: Display my collection 14](#_Toc17471863)

[2.2.2.17. Use case: log out 14](#_Toc17471864)

[3. Requirements Specification 15](#_Toc17471865)

[3.1. External Interface Requirements 15](#_Toc17471866)

[3.2. Functional Requirements 15](#_Toc17471867)

[3.2.1. Register user 15](#_Toc17471868)

[3.2.2. User sign in 16](#_Toc17471869)

[3.2.3. Display all products 16](#_Toc17471870)

[3.2.4. Display the products to be sold 17](#_Toc17471871)

[3.2.5. Display the products to be given 18](#_Toc17471872)

[3.2.6. Add product to “My Collection” 18](#_Toc17471873)

[3.2.7. Contact User 19](#_Toc17471874)

[3.2.8. Rate user 20](#_Toc17471875)

[3.2.9. Consult wish list 20](#_Toc17471876)

[3.2.10. Comment on wish list 21](#_Toc17471877)

[3.2.11. display user’s profile 22](#_Toc17471878)

[3.2.12. Display user’s products 22](#_Toc17471879)

[3.2.13. Display user’s Wishlist 23](#_Toc17471880)

[3.2.14. Add product 23](#_Toc17471881)

[3.2.15. Add to Wishlist 24](#_Toc17471882)

[3.2.16. Reply to a message 25](#_Toc17471883)

[3.2.17. Edit profile 25](#_Toc17471884)

[3.2.18. Display my collection 26](#_Toc17471885)

[3.2.19. Log out 26](#_Toc17471886)

[3.3 Detailed Non-Functional Requirements 27](#_Toc17471887)

[3.3.1 Logical Structure of the Data 27](#_Toc17471888)

[3.3.2 Security 30](#_Toc17471889)

[Index 31](#_Toc17471890)

# List of Figures

[Figure 1 - System Environment 4](#_Toc77487669)

[Figure 2 - Article Submission Process 6](#_Toc77487670)

[Figure 3 - Editor Use Cases 8](#_Toc77487671)

[Figure 4 - Logical Structure of the Article Manager Data 23](#_Toc77487672)

# Introduction

## Purpose

The purpose of this document is to present a detailed description of the Find-Sell-Give Mobile application. It will explain the purpose and features of the application, its interfaces, what you can accomplish with it, the constraints under which it must operate and how it will react to user’s actions.

## Scope of Project

The Find-Sell-Give software is a mobile application. It is designed to give the users the opportunity to find the products they wish to have. This can be done through purchasing them from other users who put products on sale, getting them for free from users who want to give them away or by putting them in your which list and wait for other users to contact you. The user can also offer some product to sell or to give.

The app is an intermediary between the users to facilitate the transactions and the acquirement of the products. It is an android application, connected to a firebase NoSQL database and authentication system.

## Overview of Document

The next chapter, the Overall Description section, of this document gives an overview of the functionalities of the application. It describes the informal requirements and is used to establish a context for the technical requirements in the next chapter.

The third chapter, Requirements Specification section, of this document describes in technical terms the details of the functionality of the application.

# Overall Description

# System Environment

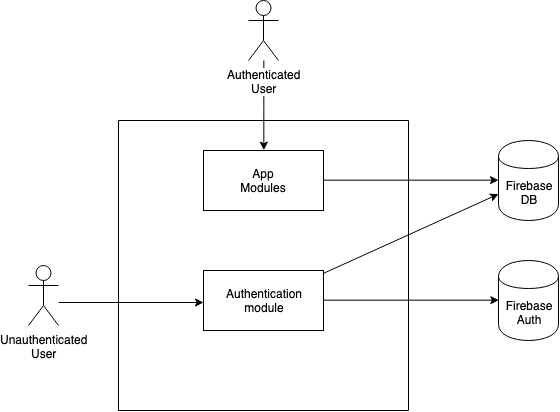


Figure 1 - System Environment

The Find-Sell-Give has two types of users. An unauthenticated user, and an authenticated user. The unauthenticated user can only interact with the authentication module, he can only register or sign into the app if he already has an account. After its authentication he can access all the functionalities of the app.

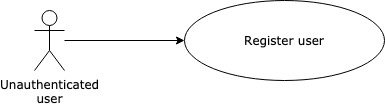
# Functional Requirements Specification

This section outlines the use cases for each unauthenticated user.

# Unauthenticated User Use Case

# Use case: Register user

**Diagram:**



**Brief Description**

The unauthenticated user accesses the mobile app andregisters as a new user.

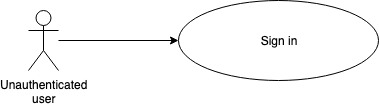
**Initial Step-By-Step Description**

Before this use case can be initiated, the unauthenticated user has already launched the app and clicked on the register button.

1. The userfills the registration form: name, email, password and password confirmation
2. The userclicks on the register button.
3. The app creates the user
4. The user is redirected to the All products page

# Use case: User sign in

**Diagram:**



**Brief Description**

The unauthenticated user accesses the app and enters his credentials to sign in into the app.

**Initial Step-By-Step Description**

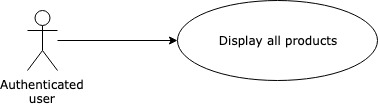
1. The userfills the login form: email, password
2. The userclicks on the sign in button.
3. The user is redirected to the All products page

# Authenticated User Use Case

After a successful registration or sign in the unauthenticated user becomes an authenticated user. This user can access to the different functionalities that the app offers.

# Use case: Display all products

**Diagram:**



**Brief Description**

The user can display all the products that are available on the app.

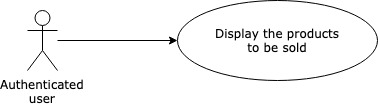
**Initial Step-By-Step Description**

Before this use case can be initiated, the user has already signed in the app.

1. The user is automatically redirected to this page, or he can click on the “ALL” tabof the home screen
2. The app displays all the products available to sell or to give.
3. The User can scroll up and down to see all the products available.

# Use case: Display theproducts to be sold

**Diagram:**

****

**Brief Description**

The User can display only the products to be sold.

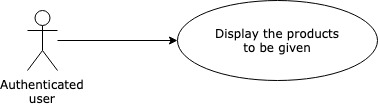
**Initial Step-By-Step Description**

Before this use case can be initiated, the user has already signed in the app.

1. The user clicks on the “On Sale” tabof the home screen
2. The app displays all the products available to sell.
3. The User can scroll up and down to see all the products available for selling.

# Use case: Display theproducts to be given

**Diagram:**

****

**Brief Description**

The User can display only the products to be given.

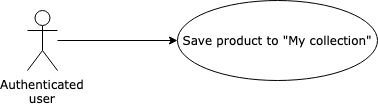
**Initial Step-By-Step Description**

Before this use case can be initiated, the user has already signed in the app.

1. The user clicks on the “Give” tabof the home screen
2. The app displays all the products available to give.
3. The User can scroll up and down to see all the products available for giving.

# Use case: Add product to “My Collection”

**Diagram:**

****

**Brief Description**

The User saves the products that interest him into a collection. Those products can be seen separately under "My collection" section.

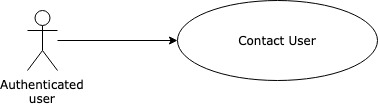
**Initial Step-By-Step Description**

Before this use case can be initiated, the user has already signed in the app, and is in one of the home screen sections (All, On Sale or Give)

1. The user clicks on the “Save” iconof a product.
2. The product is saved to the user’s “My Collection”

# Use case: Contact User

**Diagram:**

****

**Brief Description**

The User can contact another user about a particular item that he is offering for sell or to give.

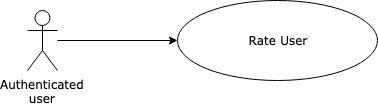
**Initial Step-By-Step Description**

Before this use case can be initiated, the user has already signed in the app, and is in one of the home page sections (All, On Sale or Give)

1. The user clicks on the products he is interested in.
2. The profile of the owner is displayed.
3. The user can choose to contact the user through the app or by email.
4. If the user chooses to contact the owner through email, he has to click on owner’s email. The mobile email app opens.
5. If the user chooses to contact the owner through the app, he has to click on the letter icon.
6. A conversation interface appears. The user can write a message and click on send.

# Use case: Rate user

**Diagram:**

****

**Brief Description**

The User can rate another user on the app.

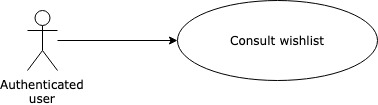
**Initial Step-By-Step Description**

Before this use case can be initiated, the user has already signed in the app, and is in a user’s profile page.

1. The user rates the other users by choosing the number of stars to attribute to him.
2. The system saves the rating and adds it to the user’s ratings.

# Use case: Consult wishlist

**Diagram:**

****

**Brief Description**

The user can consult the wishlist that contains the products wished by the other users.

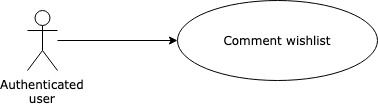
**Initial Step-By-Step Description**

Before this use case can be initiated, the user has already signed in the app, and is in the home page.

1. The user clicks on the wish list icon.
2. The system displays the different categories of the products.
3. The user can choose a category to display.
4. The wished products of the category are displayed then.

# Use case: Comment on wish list

**Diagram:**

****

**Brief Description**

The user comments on the wish list product of another user.

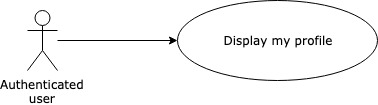
**Initial Step-By-Step Description**

Before this use case can be initiated, the user has already signed in the app, and is in one of the wish list categories pages.

1. The user clicks on a product from the category products.
2. The user writes his comment of the text field and sends it
3. The comment is displayed with the concerned product and can be consulted by other users.

# Use case: display user’s profile

**Diagram:**

****

**Brief Description**

The user displays the information of his own profile.

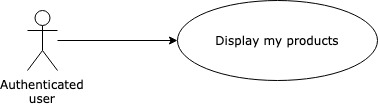
**Initial Step-By-Step Description**

Before this use case can be initiated, the user has already signed in the app.

1. The user clicks on my profile.
2. The user profile displays the name, rating and email of the current user.
3. The inbox tab is displayed with a recap of the messages sent or received from the other users.

# Use case: display user’s products

**Diagram:**

****

**Brief Description**

The user displays the productsthat he had added to the app.

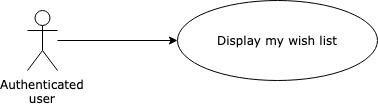
**Initial Step-By-Step Description**

Before this use case can be initiated, the user has already signed in the app, and is in the profile page.

1. The user clicks on my products tab.
2. The app displays the products already added by the user.

# Use case: display user’s wish list

**Diagram:**

****

**Brief Description**

The user displays the wish list products that he had added to the app.

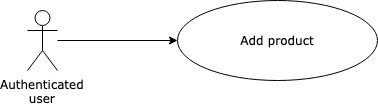
**Initial Step-By-Step Description**

Before this use case can be initiated, the user has already signed in the app, and is in the profile page.

1. The user clicks on my wishlist tab.
2. The app displays the wished products already added by the user.

# Use case: add product

**Diagram:**

****

**Brief Description**

The user can add a product to sell or to give.

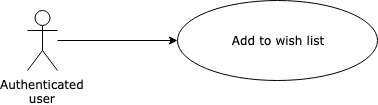
**Initial Step-By-Step Description**

Before this use case can be initiated, the user has already signed in the app, and is in the profile page.

1. The user clicks on the add button, then product.
2. The user fills the product form: description, name, category, location, condition, for selling or giving, price in case of selling and an image.
3. The user submits the form to add the product.

# Use case: add to wish list

**Diagram:**

****

**Brief Description**

The user can add a product to his wish list.

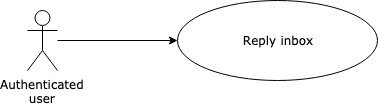
**Initial Step-By-Step Description**

Before this use case can be initiated, the user has already signed in the app, and is in the profile page.

1. The user clicks on the add button, then Wishlist.
2. The user fills the wish list form: post, category.
3. The user submits the form to add the product to the Wishlist.

# Use case: reply to a message

**Diagram:**

****

**Brief Description**

The user can reply to message he received.

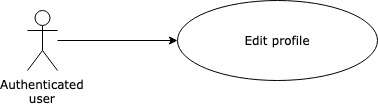
**Initial Step-By-Step Description**

Before this use case can be initiated, the user has already signed in the app, and is in the profile page.

1. The user clicks to a message in the inbox tab.
2. The message is displayed.
3. The user can write his reply and click send.

# Use case: edit profile

**Diagram:**

****

**Brief Description**

The user can edit his profile.

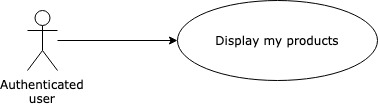
**Initial Step-By-Step Description**

Before this use case can be initiated, the user has already signed in the app, and is in the profile page.

1. The user clicks on the edit icon.
2. The user edits his profile: picture, name.
3. The submits the modifications.

# Use case: Display my collection

**Diagram:**

****

**Brief Description**

The user can display my collection products.

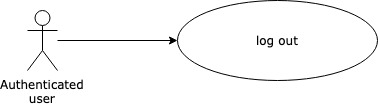
**Initial Step-By-Step Description**

Before this use case can be initiated, the user has already signed in the app.

1. The user clicks on my collection icon.
2. The app displays the products already saved by the user.

# Use case: log out

**Diagram:**

****

**Brief Description**

The user can log out from the app.

**Initial Step-By-Step Description**

Before this use case can be initiated, the user has already signed in the app, and is in the profile page.

1. The user clicks on the log out icon.
2. The user is logged out from the app.
3. The sign in page is displayed.

# Requirements Specification

# External Interface Requirements

The only link to an external system is the link to the Firebase platform, it serves as a NoSQL database to store the different objects manipulated in the app, and also to manage the user authorizations.

# Functional Requirements

# Register user

|  |  |
| --- | --- |
| **Use Case Name** | Register user |
| **XRef** | Section 2.2.1.1, Register user |
| **Trigger** | The Unauthenticated user accesses the registration page |
| **Precondition** | The user has accessed to the app and clicked on the register button |
| **Basic Path** | 1. The app displays a page with the registration form 2. The user fills the form: Name, email, password, password confirmation. 3. The user submits the form. |
| **Alternative Paths** | In step 2, if the user chooses a user mail that already exists.   1. The user submits the form. 2. A message saying that the registration is failed   In step 2, if the users enters a different password and password confirmation, a message saying that the registration failed is displayed. |
| **Postcondition** | The app creates the new user and goes to the home page that displays all the products. |
| **Exception Paths** | The user may abandon the search at any time. |
| **Other** | None |

# User sign in

|  |  |
| --- | --- |
| **Use Case Name** | User sign in |
| **XRef** | Section 2.2.2.1.2, User sign in |
| **Trigger** | The useraccesses the app. |
| **Precondition** | The useraccesses the app. |
| **Basic Path** | This userfills the sign in form and submits |
| **Alternative Paths** | If the userenters a wrong mail/password a message saying that the authentication has failed is displayed. |
| **Postcondition** | The user signs in and the All product page is displayed |
| **Exception Paths** | The attempt may be abandoned at any time. |
| **Other** | None |

# Display all products

|  |  |
| --- | --- |
| **Use Case Name** | Display all products |
| **XRef** | Section 2.2.2.1, Display all products |
| **Trigger** | The user accesses the home page. |
| **Precondition** | The user is logged in. |
| **Basic Path** | 1. The app displays all the products that are on sale or to give 2. The user can scroll up and down to see all the available products. |
| **Alternative Paths** | If the user is an already in the home page but in a different tab, the user has to click on the “All” to be able to display all the products. |
| **Postcondition** | All the products available are retrieved from the database and displayed in this page. |
| **Exception Paths** | The user may abandon the operation at any time. |

# Display theproducts to be sold

|  |  |
| --- | --- |
| **Use Case Name** | Display theproducts to be sold |
| **XRef** | Section 2.2.2.2, Display the products to be sold |
| **Trigger** | The user goes to the home page andclicks on the “On sale”. |
| **Precondition** | The user is logged in. |
| **Basic Path** | 1. The app displays all the products that are on sale. 2. The user can scroll up and down to see all the “on sale”products. |
| **Alternative Paths** | If no product is on sale, a message that says no element is found is displayed. |
| **Postcondition** | All the products on sale are retrieved from the database and displayed in this page. |
| **Exception Paths** | The user may abandon the operation at any time. |

# Display theproducts to be given

|  |  |
| --- | --- |
| **Use Case Name** | Display theproducts to be given |
| **XRef** | Section 2.2.2.3, Display the products to be given |
| **Trigger** | The user goes to the home page andclicks on the “Give”. |
| **Precondition** | The user is logged in. |
| **Basic Path** | 1. The app displays all the products that are marked as to give. 2. The user can scroll up and down to see all the “Give” products. |
| **Alternative Paths** | If no product is to be given, a message that says no element is found is displayed. |
| **Postcondition** | All the products to give are retrieved from the database and displayed in this page. |
| **Exception Paths** | The user may abandon the operation at any time. |

# Add product to “My Collection”

|  |  |
| --- | --- |
| **Use Case Name** | Add product to “My Collection” |
| **XRef** | Section 2.2.2.4, Add product to “My Collection” |
| **Trigger** | The user selects a product to be added to “My collection” |
| **Precondition** | The user has accessed the home page, and displays the products in one of the three tabs available |
| **Basic Path** | 1. The user chooses the product to be added to his collection 2. The user clicks on the save icon related to this product. |
| **Alternative Paths** | None |
| **Postcondition** | The product is saved in the database in the collection of the current user, and a message is displayed. |
| **Exception Paths** | If the product is already in the user’s collection is not duplicated in this list. |

# Contact User

|  |  |
| --- | --- |
| **Use Case Name** | Contact User |
| **XRef** | Section 2.2.2.5, Contact User |
| **Trigger** | The user selects a product. |
| **Precondition** | The user has already accessed the app and is in one of the three sections of the home page. |
| **Basic Path** | 1. The user selects the product he is interested in. 2. The app displays the owner’s profile page. 3. The user clicks on the letter icon. 4. The app opens a new chat page where the user can write his message and send it to the owner. |
| **Alternative Paths** | The user can click on the email of the owner, the mobile mail app opens the give the user the possibility to send an email. |
| **Postcondition** | The message appears in the inbox of both the sender and the owner. |
| **Exception Paths** | The user may abandon the operation at any time. |

# Rate user

|  |  |
| --- | --- |
| **Use Case Name** | Rate user |
| **XRef** | Section 2.2.2.6, Rate user |
| **Trigger** | The user selects a product of the owner’s he wants to rate. |
| **Precondition** | The user has already accessed the app and is in one of the three sections of the home page. |
| **Basic Path** | 1. The user selects the product he is interested in. 2. The app displays the owner’s profile page. 3. The user can choose the number of stars to attribute to the owner. |
| **Alternative Paths** | None. |
| **Postcondition** | The number of stars I added to the owner’s profile, and the result is displayed in his profile. |
| **Exception Paths** | The user may abandon the operation at any time. |

# Consult wish list

|  |  |
| --- | --- |
| **Use Case Name** | Consult wish list |
| **XRef** | Section 2.2.2.7, Consult wish list |
| **Trigger** | The user goes to the Wishlist page |
| **Precondition** | The user has signed into the app |
| **Basic Path** | 1. The app displays all the categories of products, and for each category the number of posts and comments. 2. The user can choose a category, the app displays the list of posts that represent the users wished products. |
| **Alternative Paths** | None. |
| **Postcondition** | The requested information has been displayed. |
| **Exception Paths** | The user may abandon the operation at any time. |

# Comment on wish list

|  |  |
| --- | --- |
| **Use Case Name** | Comment on wish list |
| **XRef** | Section 2.2.2.8, Comment on wish list |
| **Trigger** | The userhad accessed a Wishlist post. |
| **Precondition** | The userhas accessed the Wishlist sectionand selected a category. |
| **Basic Path** | 1. The appdisplays the post with its comments. 2. The usergoes to the comment text field and adds his comment. 3. The user submits his comment. |
| **Alternative Paths** | None. |
| **Postcondition** | The comment is added to the posts, the number of comments on the Wishlist page is updated. |
| **Exception Paths** | The user may abandon the operation at any time. |

# display user’s profile

|  |  |
| --- | --- |
| **Use Case Name** | display user’s profile |
| **XRef** | Section 2.2.2.9, display user’s profile |
| **Trigger** | The usergoes the profile section. |
| **Precondition** | The user has signed into the app. |
| **Basic Path** | 1. The appdisplays the user’s name, rating, email. 2. The tab enabled is the inbox tab. |
| **Alternative Paths** | None. |
| **Postcondition** | The app displays the user’s information and the conversations that he has with the other users. |
| **Exception Paths** | The user may abandon the operation at any time. |

# Display user’s products

|  |  |
| --- | --- |
| **Use Case Name** | Display user’s products |
| **XRef** | Section 2.2.2.10, display user’s products |
| **Trigger** | The user selects the “my products” from his profile |
| **Precondition** | The user has signed into the app and went to the “my profile” section. |
| **Basic Path** | 1. The appdisplays the products that the user has already added, for sale or to give. 2. Only the image and the name are displayed |
| **Alternative Paths** | If no product is present, a message is displayed. |
| **Postcondition** | All the user’s products are displayed in this tab. |
| **Exception Paths** | The user may abandon the operation at any time. |

# Display user’s Wishlist

|  |  |
| --- | --- |
| **Use Case Name** | Display user’s Wishlist |
| **XRef** | Section 2.2.2.11, display user’s Wishlist |
| **Trigger** | The user selects the “my Wishlist” from his profile |
| **Precondition** | The user has signed into the app and went to the “my profile” section. |
| **Basic Path** | 1. The appdisplays the posts that the user has already added to his Wishlist. 2. Only the description is displayed |
| **Alternative Paths** | If no post is present, a message is displayed. |
| **Postcondition** | All the user’s Wishlist posts are displayed in this tab. |
| **Exception Paths** | The user may abandon the operation at any time. |

# Add product

|  |  |
| --- | --- |
| **Use Case Name** | Add product |
| **XRef** | Section 2.2.2.12, add product |
| **Trigger** | The user clicks on the add product button from his profile(“plus” icon + “home” icon) |
| **Precondition** | The user has signed into the app and went to the “my profile” section. |
| **Basic Path** | 1. The appdisplays product form 2. The user fills the product form: description, name, category, location, condition, for selling or giving, price in case of selling and an image 3. The user submits the form |
| **Alternative Paths** | If the form is not filled correctly, a message is diplayed. |
| **Postcondition** | The product is added to the list of products and will be displayed in the All product section and the corresponding section (On Sale or Give) |
| **Exception Paths** | The user may abandon the operation at any time. |

# Add to Wishlist

|  |  |
| --- | --- |
| **Use Case Name** | Add to Wishlist |
| **XRef** | Section 2.2.2.13, Add to Wishlist |
| **Trigger** | The user clicks on the add product button from his profile(“plus” icon + “favorite” icon) |
| **Precondition** | The user has signed into the app and went to the “my profile” section. |
| **Basic Path** | 1. The appdisplays wishlist form 2. The user fills the Wishlist form: post, category, location 3. The user submits the form |
| **Alternative Paths** | If the form is not filled correctly, a message is diplayed. |
| **Postcondition** | The post is added to the wishlist and will be displayed in wishlist page |
| **Exception Paths** | The user may abandon the operation at any time. |

# Reply to a message

|  |  |
| --- | --- |
| **Use Case Name** | Reply to a message |
| **XRef** | Section 2.2.2.14, reply to a message |
| **Trigger** | The user clicks on one of the messages in his inbox |
| **Precondition** | The user has signed into the app and went to the “my profile” section. |
| **Basic Path** | 1. The appdisplays the conversation from the inbox 2. The user writes his reply and submits |
| **Alternative Paths** | If no message in the inbox, this action can’t be done |
| **Postcondition** | The message is sent and will appear in the inbox of the two users |
| **Exception Paths** | The user may abandon the operation at any time. |

# Edit profile

|  |  |
| --- | --- |
| **Use Case Name** | Edit profile |
| **XRef** | Section 2.2.2.15, edit profile |
| **Trigger** | The user clicks on the edit icon |
| **Precondition** | The user has signed into the app and went to the “my profile” section. |
| **Basic Path** | 1. The appdisplays the information of the user. 2. The user can change his profile picture 3. The user submits the changes |
| **Alternative Paths** | None |
| **Postcondition** | The user image is updated |
| **Exception Paths** | The user may abandon the operation at any time. |

# Display my collection

|  |  |
| --- | --- |
| **Use Case Name** | Display my collection |
| **XRef** | Section 2.2.2.16, Display my collection |
| **Trigger** | The user clicks on “my collection” icon |
| **Precondition** | The user has signed into the app |
| **Basic Path** | 1. The appdisplays the products already saved by the user (use case: 2.2.2.4) |
| **Alternative Paths** | If no product is saved a message is displayed |
| **Postcondition** | The app displays the requested data |
| **Exception Paths** | The user may abandon the operation at any time. |

# Log out

|  |  |
| --- | --- |
| **Use Case Name** | Log out |
| **XRef** | Section 2.2.2.16, Log out |
| **Trigger** | The user clicks on “Log out” icon |
| **Precondition** | The user has signed into the app and is in the “my profile” page |
| **Basic Path** | 1. The user is signed off |
| **Alternative Paths** | None |
| **Postcondition** | The app displays the login page |
| **Exception Paths** | The user may abandon the operation at any time. |

## 3.3 Detailed Non-Functional Requirements

### 3.3.1 Logical Structure of the Data

The logical structure of the data to be stored in the internal Article Manager database is given below.

The data descriptions of each of these data entities is as follows (The Database is a NoSQL database, this description is global about each object structure):

**User Data Entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| uid | Text | Technical identifier |  |
| FullName | Text | Full name of the user |  |
| Email | Text | Email of the user |  |
| Password | Text | Password of the user |  |
| ProfileImage | Text | The profile Image |  |
| Rating | Rating | The rating of the user |  |

**Product**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| UID | Text | Technical ID |  |
| ID | Text | ID number of the product |  |
| ImageUrl | Text | Url of the mage associated to the product |  |
| Name | Text | Product Name |  |
| Description | Text | Product description |  |
| Condition | Boolean | New or Used | New = true |
| Category | Integer | Product’s category | 0:Electronics, 1:Housing, 2:Entertainment, 3:Other |
| Purpose | Integer | Sell or give | Sell: 0, Give: 1 |
| City | Text | Location of the product |  |
| Price | Double | Price in case the product is for sale |  |

**Rating**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| RatingOne | Float | Number of one-star ratings |  |
| RatingTwo | Float | Number of two stars ratings |  |
| RatingThree | Float | Number of three stars ratings |  |
| RatingFour | Float | Number of four stars ratings |  |
| RatingFive | Float | Number of five stars ratings |  |

**Wishlist Post**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| UID | Text | Technical ID |  |
| ID | Text | Post’s ID |  |
| Question | Text | Text of the post |  |
| Category | Integer | Category of the wished product | Same as in product |
| CommentsCount | Integer | Number of the comments on this post |  |
| DatePosted | Date | Date of the creation of the post |  |

**Comment (this structure is linked with the wishlist)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| UID | Text | Technical ID |  |
| Test | Text | Comment’s text |  |
| Image Url | Text | Image that can be added to the comment |  |
| DatePosted | Date | Date of the comment |  |

**Rated User (this structure is linked to the user; it represents the users that he rated)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| UID | Text | Technical ID |  |
| Name | Text | Name of the rated user |  |
| Is Rated | Boolean | Rated: true, Unrated: false |  |
| Rating | Rating | Contains the rating of the user |  |

**Chat (Represents the structure of a chat conversation, it will contain the messages)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| ChatName | Text | Represent the chat identifier |  |
| Last Message | Message | Represents the last message of the chat |  |

**Message (this structure is linked to the user; it represents the users that he rated)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| From | Text | UID of the user that sent the message |  |
| To | Text | UID of the user that will receive the message |  |
| Message | Text | The text of the message |  |
| Date | Date | Date of the message |  |