Logo

Description automatically generated

Software Project Management 2021/2022

Flashee

D2.1.1 – Software Requirements Specification

### Authors

* Ângelo Paiva.
* Burak Tinman.
* Jan Frank.
* Álvar San Martin.
* Pedro Henriques.

### Reviewer

* Ângelo Paiva.
* Burak Tinman.
* Jan Frank.
* Álvar San Martin.
* Pedro Henriques.

### Approver

1.0 Tomé Madail

### Table of Contents

[1. Introduction](#_Toc89697297)

[1.1. Purpose](#_Toc89697298)

[1.2. Scope](#_Toc89697299)

[1.3. System overview](#_Toc89697300)

[1.4. References](#_Toc89697301)

[2. Use Cases](#_Toc89697302)

[3. Functional Requirements](#_Toc89697303)

[4. Non-functional requirements](#_Toc89697304)

[5. Mock-ups](#_Toc89697305)

# Introduction

## Purpose

The purpose of this document is to serve as a guide to designers, developers and testers who are responsible for the engineering of the Flashee project. It should give the engineers all the information necessary to design, develop and test the software.

## Scope

This document contains a complete description of the functionality of the Flashee project. It consists of use cases, functional requirements, nonfunctional requirements and mockups, which, taken together, form a complete description of the software.

## System overview

The system is designed as a web application and consists of flashcards which can be used to learn different topics in a fast and short manner. Flashcards can be created or imported by the user itself and then used to learn the new topics by a fast-paced quiz mode. This is achieved by flashcards which consist of two sides: a question side and a translation or answer side to check if the solution of the answer was correct. Furthermore, flashcards can be tagged by the user so they can be accessed and sorted easily.

## References

[Software Development Plan](https://isecpt.sharepoint.com/sites/GPS20212022-Team31/Project%20Files/Deliverables/D1.1.1%20-%20Vision%20and%20Scope.docx): understand the plan of the software development

[Quality Assurance Plan](https://isecpt.sharepoint.com/sites/GPS20212022-Team31/Project%20Files/Deliverables/D1.2.2%20-%20Quality%20Assurance%20Plan.docx): features, metrics (so we can better understand some of the tests)

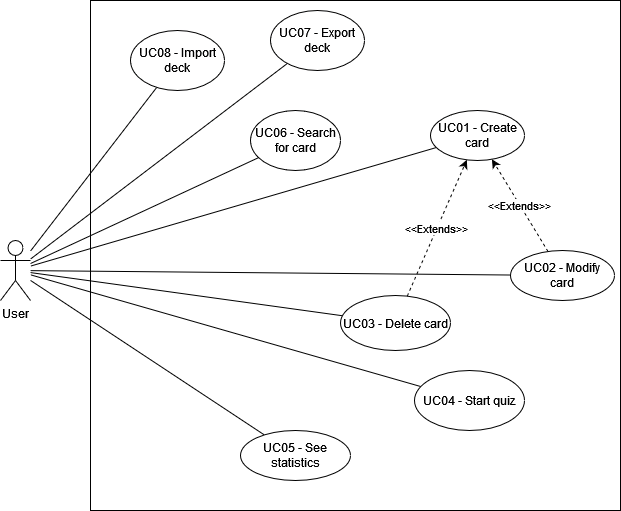
[Quality Assessment Report](https://isecpt.sharepoint.com/sites/GPS20212022-Team31/Project%20Files/Deliverables/D2.3.2%20-%20Quality%20Assessment%20Report.docx):

# Use Cases

The "Use Cases" section contains each of the use cases.

Divide into subsections as needed.

Present **Use Case Diagrams**.



|  |  |
| --- | --- |
| **Name** | **UC01: Create card** |
| Summary | A user creates a card in his deck. |
| Rationale | The most important functionality of the program, the user creates a card from a scaffold with all the fields. |
| Users | Normal user. |
| Preconditions | - |
| Basic course of events | 1. The user clicks in “Create new card” 2. A scaffold of a card is displayed with the fields 3. The user fills the fields and add tags 4. The user confirms the data is correct and the card is created |
| Alternative paths | 3. If the user does not fill a field, the user will not be able to confirm the creation. |
| Postconditions | The card was saved correctly in the local storage as a JSON string. |

|  |  |
| --- | --- |
| **Name** | **UC02: Modify card** |
| Summary | A user modifies the content of a card |
| Rationale | We assume a user can make mistakes while creating a card or wants to make changes after the initial creation, so a way of editing card is added. |
| Users | Normal user. |
| Preconditions | At least a card exists on the deck. |
| Basic course of events | 1. The user clicks in the modification button of the card. 2. The fields will appear as modifiable. 3. The user changes the values. 4. The user confirms the changes. |
| Alternative paths | 1. During the step 3 if any field is empty, the confirm button will not be available. |
| Postconditions | The card was saved correctly in the local storage as a JSON string. |

|  |  |
| --- | --- |
| **Name** | **UC03: Delete card** |
| Summary | A user deletes a card he no longer needs. |
| Rationale | Since we can store an infinite number of cards the user has to be able to choose which cards, he wants to keep and which he no longer needs. |
| Users | Normal user. |
| Preconditions | At least a card exists on the deck. |
| Basic course of events | 1. The user clicks on the delete button of a card. 2. A warning is displayed. 3. The user confirms the deletion. |
| Alternative paths |  |
| Postconditions | The card was removed completely from the storage. |

|  |  |
| --- | --- |
| **Name** | **UC04: Start quiz** |
| Summary | A random quiz is created for the user. |
| Rationale | With all the cards the user will be able to create random quiz to test his knowledge about an issue. |
| Users | Normal user. |
| Preconditions | At least a card exists on the deck. |
| Basic course of events | 1. The user clicks on the “Quiz button” 2. The programs show options about the quiz, tags, number of questions etc. 3. The user chooses the options. 4. During the quiz, the user answers the questions from flashcards. 5. When the quiz finishes the user can see his evaluation. 6. The result is saved. |
| Alternative paths | 1. During the step 3 if any field is empty, the confirm button will not be available. |
| Postconditions | The user saw the evaluation. |
| Name | **UC05: See statistics** |
| Summary | The user can see his personal statistics. |
| Rationale | Since we store the evaluation after each quiz, the user should be able to see his statistics to check his progress. |
| Users | Normal user. |
| Preconditions | No preconditions, but if there are no cards the statistics will appear as empty. |
| Basic course of events | 1. The user clicks on “Check statistics” 2. The program shows the statistics of the user like previous results |
| Alternative paths | 1. During step 2 if any field is empty the program will instead show the user some screen saying “No quizzes done yet, check later” |
| Postconditions | The user saw their personal statistics. |

|  |  |
| --- | --- |
| **Name** | **UC06: Search for card** |
| Summary | The user can search for a specific card. |
| Rationale | For editing or checking up a card, the user must be able to find the specific card he is searching for.  The user will be able to make simple searches with the search bar included, displaying the resulting cards by creation date.  The user will be able to filter by topic, choosing one of the ones listed. |
| Users | Normal user. |
| Preconditions | Nothing, but if no cards will show a message “There are no cards yet”. |
| Basic course of events | 1. The user inputs text in the search bar (max 100 characters) 2. The user clicks on the search/browse button. 3. The user can optionally select a topic in the topic bar to filter by topic 4. The cards that match the text in the search bar will be displayed by date (not shown to the user) |
| Alternative paths | If the specific card is not available, the user will see a message on the screen (“Card not available”) |
| Postconditions | The user obtains the desired card for viewing and / or editing |

|  |  |
| --- | --- |
| **Name** | **UC07: Export deck** |
| Summary | The user exports a group of cards (a deck). |
| Rationale | Since people can create cards, people will be able to share groups of cards so other people can use them without having to create them themselves.  When you export your cards, you export a group of them.  The cards to be exported are the ones selected in the search if none is selected the full deck is exported. |
| Users | Normal user. |
| Preconditions | At least a card exists on the deck. |
| Basic course of events | 1. (Optional) The user browses for a group of cards like in UC06. 2. The user clicks on the “Export” button. 3. The browser shows the default message. 4. The confirmation of the export is shown by the browser. |
| Alternative paths | 1. If no cards are chosen, the “Export” button is greyed out |
| Postconditions | The cards are exported into the specified directory into a single JSON file. |

|  |  |
| --- | --- |
| **Name** | **UC08: Import deck** |
| Summary | The user imports a group of cards (a deck) |
| Rationale | The user does not want to create cards again when using the application on previous sessions. The user can import a previously exported deck so he can continue with his progress.  The program will display the cards that are going to be imported with the total number and the tags imported.  If there is any problem importing the file nothing will be displayed. |
| Users | Normal user. |
| Basic course of events | 1. User clicks on “Import” Button 2. A file manager (provided by the browser) will open so the user can navigate to the specific JSON file. 3. The application shows the card in the screen before importing them. 4. Application confirms imported deck. |
| Alternative paths | 1. If the file is not a JSON file or the JSON file is not in the expected form, an error will be shown instead, and no cards will be imported. |
| Postconditions | The deck is imported to the storage. |

# Functional Requirements

|  |  |
| --- | --- |
| **Name** | **FR01: Non-case sensitive search** |
| Summary | The search feature must not have case sensitivity. |
| Rationale | This will aid users in finding the topic they want to study without worrying if the text is in uppercase, lowercase or a mix of both, thus allowing them to efficiently learn new things.  <https://www.merriam-webster.com/dictionary/search> |
| Requirements | When a user invokes the search function, the software will automatically do a non-case sensitive search to be as easy to use as possible.  The search will match any text that has the same character (numbers, letters, symbols and emojis) as the search term, even if the case is different.  If the search term matches, totally or partially the original text of the card, then it should lead the user to where that card/topic is located, independent of case sensitivity.  The app will display the cards in the home screen, ordered by creation date (not shown). |
| References | UC06: Search, MK07: Browse. |

|  |  |
| --- | --- |
| **Name** | **FR02: Statistics** |
| Summary | The statistics the user sees must be actively updated and detailed. |
| Rationale | Being capable of viewing the statistics will motivate the users to learn even more or even compete with other users. |
| Requirements | These statistics must be accessible:   * Quantity of cards in each deck. * Total number of cards in each topic. * Total number of topics. * Quantity of both successful and unsuccessful tries on each Topic. |
| References | UC05: See Statistics, MK05: Statistics |

|  |  |
| --- | --- |
| **Name** | **FR03: Create card** |
| Summary | The user must be able to create cards. |
| Rationale | This is the basis of our entire application; it covers from the input from the user to the storage in the browser. |
| Requirements | The creation of a card must have:   * Create a title/question. * Ability to assign them to a certain topic and/or deck, at least one to be sortable. * Description (optional). * An answer/translation to appear in the back of the card when clicked or in the quiz, bust be introduced manually by the user.   The title will take ~100 characters and the description 180 max, similar to twitter to keep the cards short and clear. |
| References | UC01: Create card, MK02: card creation |

|  |  |
| --- | --- |
| **Name** | **FR04: Modify Card** |
| Summary | The user must be able to alter existing cards. |
| Rationale | In case the user is unhappy with how a card turned out, or it has a typo, or any problem whatsoever that might leave the user unsatisfied, we want to allow the user to edit the cards to fit them how they wish.  The interface for modifying will be the same as in creation but the fields are already filled with the previous data. |
| Requirements | The Modifying of a card must have:   * Ability to reassign them to a different topic and/or deck. * Modify the Description. * Modify the Question * Alter the answer of a certain card. |
| References | UC02: Modify card, MK02: card creation (from different path) |

|  |  |
| --- | --- |
| **Name** | **FR05: Delete card** |
| Summary | The user must be able to delete cards. |
| Rationale | In case the user finds a card he no longer has any use for, they will be able to remove it. |
| Requirements | Every card will have a button to delete it, showing a confirmation message before deleting the card completely from the storage.  Initially deletion will be only individually, not in bulk. |
| References | UC-3: Delete card, MK01: Home screen. |

|  |  |
| --- | --- |
| **Name** | **FR06: Start quiz** |
| Summary | The user must be able to test himself by doing quizzes. |
| Rationale | Since our app has the intent to motivate the user’s learning, we plan to introduce a way to test the user abilities. |
| Requirements | The app should allow the user to test their knowledge based on a mix of topics the user selects or the whole deck. |
| References | UC04: Start test, MK04: Quiz |

|  |  |
| --- | --- |
| **Name** | **FR07: Export deck** |
| Summary | The user must be able to export his own decks. |
| Rationale | It will allow users to share decks between each other, being capable of learning from one another’s knowledge. |
| Requirements | The exportation of decks of cards will include:   * The ability to export the whole deck. * The ability to export only a group of cards defined by a search, if the user clicks on “Export” after a search only the searched cards will be exported. |
| References | UC07: Export Deck, MK06; Export |

|  |  |
| --- | --- |
| **Name** | **FR08: Import deck** |
| Summary | The user must be able to import other user’s decks. |
| Rationale | It will allow users to import decks created by others. This way they will be able to begin far more efficiently learning new topics, as they will not have to create the cards, decks, or topics themselves. |
| Requirements | The importation of a deck, card or topic must have:   * The ability to import data from other users via JSON files. |
| References | UC08: Import deck, MK03: import deck |

# Non-functional requirements

|  |  |
| --- | --- |
| **Name** | **NFR01: Usability** |
| Summary | Self-explanatory design and error prone application. |
| Rationale | The users want to have an application with an easy-to-understand design and an error prone behaviour so the user can navigate smoothly through the application without any disturbances and unpredicted outcomes. |
| Requirements | The loading time of the application for every screen the user sees should be at a max of 1 second DEFINE BASE SYSTEM/PC. Furthermore, user errors while using the application should be at a max of 3%. Lastly, the number of times the final application fails because of internal errors should be the 0.1%. |
| References | Quality Assessment Report, section 1 |

|  |  |
| --- | --- |
| **Name** | **NFR02: Portability** |
| Summary | Usage across several modern browsers. |
| Rationale | The users want to use the app regardless of their preferred browsers and devices they use. |
| Requirements | The users must be able to use the application by using either the Firefox, Chrome (or Chromium based) or Safari browser. |
| References | Quality Assessment Report, section 1 |

|  |  |
| --- | --- |
| **Name** | **NFR03: Performance** |
| Summary | Fast application. |
| Rationale | The users want to have an application with fast working functions to perform their actions smoothly and without interruptions. |
| Requirements | The complexity of the function used should not exceed O(n²) |
| References | Quality Assessment Report, section 1 |

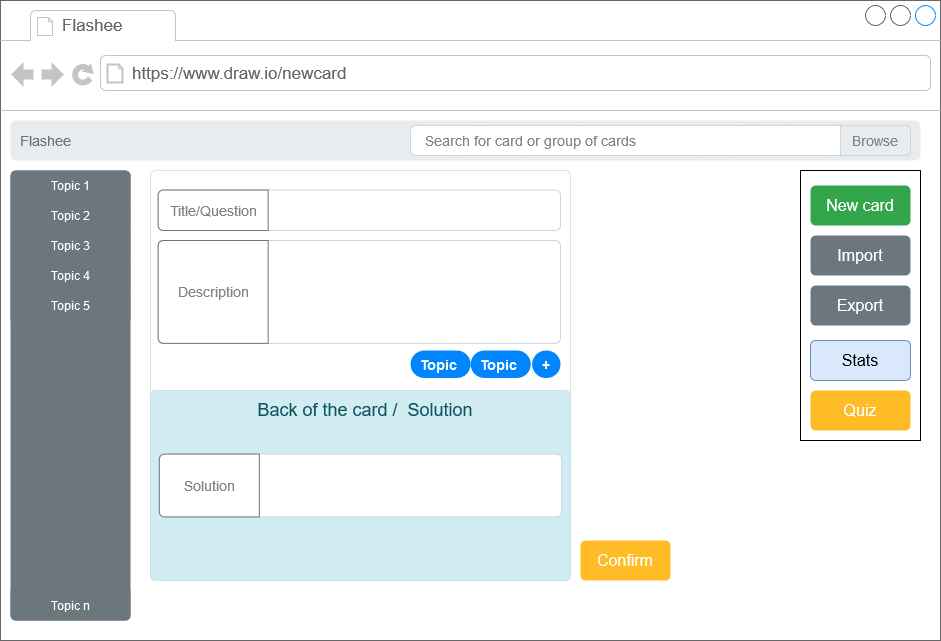
# Mock-ups

### MK01: Home screen

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

**MK02: Card creation**



**MK03: Import deck**

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

**MK04: Quiz**

**Interfaz de usuario gráfica

Descripción generada automáticamente**

**MK05: Statistics**

**Interfaz de usuario gráfica, Gráfico, Aplicación

Descripción generada automáticamente**

**MK06: Export**

**Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente**

**MK07: Browse**

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**