**Software Requirements Specification**

**for**

Memento

**Version 1.0 approved**

**Prepared by Josh Andle, Anthony Attaya-Harris, Evan Hamer, and Mohammed Fazli**

**Team CompEdu**

**10/16/2022**

**Table of Contents**

**Table of Contents ii**

**Revision History ii**

**1. Introduction 1**

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Product Scope 1

1.5 References 1

**2. Overall Description 2**

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 2

2.6 User Documentation 2

2.7 Assumptions and Dependencies 3

**3. External Interface Requirements 3**

3.1 User Interfaces 3

3.2 Hardware Interfaces 3

3.3 Software Interfaces 3

3.4 Communications Interfaces 3

**4. System Features 4**

4.1 System Feature 1 4

4.2 System Feature 2 (and so on) 4

**5. Other Nonfunctional Requirements 4**

5.1 Performance Requirements 4

5.2 Safety Requirements 5

5.3 Security Requirements 5

5.4 Software Quality Attributes 5

5.5 Business Rules 5

**6. Other Requirements 5**

**Appendix A: Glossary 5**

**Appendix B: Analysis Models 5**

**Appendix C: To Be Determined List 6**

**Revision History**

| **Name** | **Date** | **Reason For Changes** | **Version** |
| --- | --- | --- | --- |
| Josh Andle | 10/16 | Filling out remaining sections for Deliverable 2 | 2 |
| Josh Andle | 10/30 | Fixing NFReq IDs, removing template text | 3 |

# **Introduction**

## **Purpose**

This SRS covers the app Memento. Memento’s purpose is to allow users, primarily students to create flashcards from their notes and study those flashcards. The scope of the app covers the creation of cards, organization of cards into separate decks, studying of decks, and the importing and exporting of cards from text files.

## **Document Conventions**

This Document follows the IEEE template for SRS Documents.

No conventions are currently used beyond the name of the app being Memento.

## **Intended Audience and Reading Suggestions**

This document is intended for any reader looking for an overview of the capabilities intended for Memento’s functionality, as well as readers looking for more technical information about detailed requirements, The rest of the SRS reports the scope of the app, a high-level overview of Memento’s functionality, and the requirements implemented to achieve these functions.

## **Product Scope**

The software provides an application which users can interact with to produce and study flashcards. By providing a streamlined service with minimal barrier to entry, the software serves the goal of making studying easier for all types of students. By providing tools for flashcard creation and review without needing an account or subscription, Memento helps achieve this goal.

## **References**

This document makes no external references.

# **Overall Description**

## **Product Perspective**

This product was made to help students, particularly those with lower income, busy schedules, or who are easily overwhelmed by complex software, study more efficiently. Beyond these demographics, anyone who is looking for and interactive studying tool may find Memento beneficial.

## **Product Functions**

**Card Creation and Sharing**

* Convert delimited text files into flashcards
* Export collections of flashcards as text files for backup or sharing
* Import cards from text files created through Memento

**Card Management**

* Create decks and assign cards to them
* Annotate cards with notes or tags
* Remove cards from decks or delete them from the collection

**Study**

* Review flashcards within a selected deck
* Take scored quizzes of a selected deck
* Card review order follows a schedule based on user performance

## **User Classes and Characteristics**

We don’t distinguish between user classes in the design of Memento. This is due in part to the streamlined nature of the app. The experience of Memento is simply the creation and studying of flashcards, and the user is expected to be anyone (whether a professor, student, or other role) who desires to create and study flashcards.

## **Operating Environment**

Memento is intended to function on computer devices across Windows, Linux, and macOS. The app functions through the browser, with local file storage for saving data.

## **Design and Implementation Constraints**

Due to limitations in time, Memento is only expected to be used with English text, and the menu options are only in English.

## **User Documentation**

Memento will provide in-app explanations of key features, and we will provide a Readme file detailing how to use the app.

## **Assumptions and Dependencies**

We assume that Memento is able to read and write within it’s own local directory in order to facilitate importing and exporting text files and accessing the user’s collection. We assume that the user doesn’t manually alter or remove these files from the directory in a way that interferes with the correct functionality of the app.

# **External Interface Requirements**

## **User Interfaces**

A diagram of the UI mockup can be found here: <https://miro.com/app/board/uXjVPT8jhWE=/>

The standard buttons which will appear on each screen along the top of the UI are:

* **Memento:** The name of the app and serves as the home button, taking the user to the initial screen where they can create new cards.
  + It's function as a home button is suggested by being written with larger font than the other buttons while being positioned alongside them.
* **Library**: Takes the user to a list of their sets. From here they can add existing cards to a set.
* **Quiz**: Takes the user to the list of their card sets where they can select one to begin studying.
* **Share**: Lets the user produce a text document storing their cards and associated metadata to an email address or save it locally.

Eventually UIs may be needed for the following software components:

* Adding new cards
  + Selecting sets to add the cards to
* Viewing existing sets
  + Adding a new set
  + Moving cards from the collection to a chosen set
* Quizzing cards
  + Before and after flipping the card over
  + Rating confidence on a card
* Sharing cards
  + By email
  + Locally

## **Hardware Interfaces**

Memento should be independent of the specific hardware on which it’s being run.

## **Software Interfaces**

Memento interfaces with the operating system on which it’s run to the extent that it must be able to read and write text data from a file and be able to locate files given a relative or absolute file path. Additionally, for emailing of notifications and possibly emailing sets of cards, Memento must be able to interface with some emailing system to send those messages.

## **Communications Interfaces**

The only communication function required within the core functions of Memento is email.

# **System Features**

## Importing Notes

4.1.1 Description and Priority

Ensure users can upload or paste in their notes to the app and have them converted to flash cards. This is high priority since it’s required for the app to function.

4.1.2 Stimulus/Response Sequences

When the user opens the app, the main screen shows the input field for notes and a button which the user may press to convert any entered text into flash cards.

4.1.3 Functional Requirements

REQ-1: The flashcard creation system shall be able to parse text documents given some text delimiter(s)

REQ-2: The flashcard creation system shall allow users to use certain flashcard templates

REQ-3:The flashcard creation system shall default to present users with an empty notes field

REQ-4: The flashcard creation system will allow the user to choose from multiple delimiters to differentiate the front and rear of a flashcard

## Creating Decks

4.2.1 Description and Priority

Allow the user to assign created cards to decks so that they can organize information belonging to different subjects. As this improves the studying experience but isn’t necessary for baseline use, it’s medium priority.

4.2.2 Stimulus/Response Sequences

After cards are created the user is given the option to assign them to sets. Choosing to do so takes them to an interface which allows assigning the new cards to sets. This can also be done by going to the library menu and selecting a set.

4.2.3 Functional Requirements

REQ-5: The flashcard creation system will allow users to add new or existing cards to one or more sets of cards, or decks, for organization.

REQ-6: The flashcard creation system will automatically create new sets as required when importing flashcards previously exported by Memento, based on recorded information of what previous decks they belonged to.

## Reviewing Cards

4.3.1 Description and Priority

Allow the user to review cards in a selected set. This is a required feature for the baseline functions of the app, so it should have high priority.

4.3.2 Stimulus/Response Sequences

When the user presses the menu button “Quiz”, theyre shown a selection of decks. By picking a deck, the user begins the cycle of being shown a card, flipping it to see the answer, and progressing through sequential cards in the set. This may include a step where they report their confidence on the card to influence card scheduling.

4.3.3 Functional Requirements

REQ-7: The flashcard creation system shall allow users to rate their comfort level with individual cards

REQ-8: The flashcard creation system shall allow users to make notes or comments on cards or decks

REQ-9: The flashcard creation system shall be able to email the user reminders for studying

REQ-10: The flashcard creation system shall track statistics about user performance for individual cards and decks

REQ-11: The flashcard creation system shall implement scheduling of individual cards based on performance or user confidence level

## Saving Cards

4.4.1 Description and Priority

The user can save their cards and associated metadata, such as scheduling information or notes and associated decks, to the local directory. This allows cards to be reimported at a later date or to serve as a backup. This is a quality-of-life feature, and should have low priority.

4.4.2 Stimulus/Response Sequences

When the user clicks the “Share” button, they will be given the option to have this file emailed to a provided address or saved locally.

4.4.3 Functional Requirements

REQ-12: The data synchronization system shall allow users to access flashcards from computers which have access to the app via browser and can locally store the flashcard data.

REQ-13: The flashcard creation system shall allow users to use all parts of the app without the need of signing up

REQ-14: The flashcard creation system will allow the user to save flashcards to their computer to use later

REQ-15: The flashcard creation system will include metadata such as associated sets and notes when saving or sharing cards.

# **Other Nonfunctional Requirements**

## **Performance Requirements**

NFR-1: The flashcard creation system shall allow users to create decks of at least 100 cards

* 1. Rationale: Decks need to be able to store enough information that users can’t simply memorize all of the cards and to increase the intervals between repetition.

NFR-2: The flashcard creation system shall allow users to create up to 10 decks

* 1. Rationale: Users may need to study material from different subjects, each represented by it's own set of cards.

NFR-3: The flashcard creation system shall take no more than 1 second to load and display the next card

* 1. Rationale: The fast/smooth transition between subsequent cards makes the studying process faster and less frustrating

NFR-4: The app shall crash no more than once per 10 hours of use.

* 1. Rationale: The app crashing may disrupt study sessions and dissuade use of the app. Minimizing it would help avoid these issues.

NFR-5: The flashcard creation system shall be capable of loading the home screen within 4 seconds of start up

* 1. Rationale: Quick startup times serves as a good first impression for the app and helps ensure users aren’t frustrated when starting their study sessions.

NFR-6: The flashcard creation system shall be capable of storing at least 1000 unique flashcards per user

* 1. Rationale: The user should be able to store enough cards to make multiple decks of maximum size.

NFR-7: The flashcard creation system shall be capable of functioning without internet access

* 1. Rationale: Studying cards offline increases the amount of time that the app is available to the user, also providing them a productive way to spend time when unable to access the internet.

NFR-8: The flashcard creation system will take no longer than 1 minute to save flashcard information to the computer once the user has clicked "save"

* 1. Rationale: Serves as a quality of life requirement limiting wait times

NFR-9: The flashcard creation system shall allow users to write up to 100 words words per flashcards

1. Rationale: Larger word allowances allow more flexibility in what information the user chooses to use when studying.

## **Safety Requirements**

NFR-10: The flashcard creation system will be robust to any input text from notes, such that they don’t crash the app.

* 1. Safeguards: Include error handling in the code, consider and account for various common errors with text parsing such as unexpected end-of-file issues.

NFR-11: The flashcard creation system will not write to any directories other than the one created for it.

* 1. Safeguards: Set up any directory pathways within the code to start with a constant string, pointing to the local directory. Ensure that when installed, and possible when run, that the app is in a directory created for its use.

## **Security Requirements**

NFR-12: The flashcard creation system shall keep user data and created cards private by default

* 1. Sharing cards should require direct action from the user, who is authorized by accessing the device. The user could potentially password-protect their data without the need of formal online account.

NFR-13: The flashcard creation system shall function without the need for users to create accounts

* 1. User authentication could be done by local password protection, or simply by having access to the device.

## **Software Quality Attributes**

Some attributes which serve the core intent of Memento, in order of priority, are robustness of data (cards not accidentally getting deleted, decks don’t get duplicates of cards, study loop correctly logs accuracy statistics), simplicity of the UI through streamlining navigation between menus and submenus, portability of data between devices through exporting/importing text files, and interpretability of settings and statistics.

## **Business Rules**

Since Memento is a single-user app, the user has full control over the functionality of the app.

# **Other Requirements**

**Appendix A: Glossary**

**Appendix B: Analysis Models**

Architecture Design: <https://docs.google.com/document/d/1IAiZczxxl17bqMY6RfW2ECTzET5o8sHw/edit?usp=sharing&ouid=103534819591177736091&rtpof=true&sd=true>

Use Cases: <https://docs.google.com/document/d/1ALFtgDP1-P2Ai2NT61xqRmY4r_akCP1h/edit?usp=sharing&ouid=103534819591177736091&rtpof=true&sd=true>

Sequence Diagrams: <https://docs.google.com/document/d/1QHQZ25s5D0hs3AJOSLGYGyBi_LtU3nX2hlyTmg0wGvk/edit?usp=sharing>

**Appendix C: To Be Determined List**