Requirements and Analysis Document for Marc

Thomas Li, Gustav Albertsson,
Alexander Sandberg, Victor Johansson,
Mathias Forsman

2018-10-21
version 5.0

1. Introduction

For students, there will often be situations where they need to learn definitions and vocabulary, and while important, this is a process that can often become tedious and suffer from a lack of cognitive stimulation. *Marc* aims to make studying more rewarding and convenient by providing tools to easily organize the information that they need to study, as well as several different ways of practicing, by way of different kinds of more "game-like" methods, such as quizzes or other challenges. The application also keeps track of the user's performance while studying in order to make it easier to view progression and improvement, encouraging the user to develop more regular review sessions. By showing stats and user progression, it is also easier for the user to feel accomplishment when they achieve goals, whether set up by themselves or as existing in-application challenges.

- The application addresses the difficulty and general lack of stimulation that comes from studying to remember things such as vocabulary, that primarily needs to be memorized.
- The application will provide tools for organizing groups of user-defined information and tools to promote repeated reviewing of said information.
- The main user group will probably be students, who often needs to memorize definitions and vocabulary.
- Since the application will be developed for mobile devices, it is suitable for use on the go. For example, while commuting to and from school, one could use the perhaps otherwise wasted time to do some quick studying. The application should be designed to allow for quickly starting up and closing down review sessions.

1.1 Definitions, acronyms, and abbreviations

- “app” = Shorthand for “application”
- “cards” = Refers to entities containing two sets of information. One of the sets should contain the information that the user can see while reviewing, while the other should contain the information that the user should memorize, with the help of the first set. These sets will often be referenced to as the “front” and the “back” of the card.
- “notes” = Refers to entities containing information used to generate the cards.
- “decks” = Refers to the collection of notes, which are used to play the different game modes.
- “game mode” = Different kinds of exercises in the application for practicing the information contained in the cards.

2. Requirements

2.1 User Stories

User story

Story Identifier: 0

Story Name: Memorization Training Tool

Description

(Epic)

As a student, I want an application where I can practice my different subjects using flashcards and other types of fun exercises such as quizzes, so that I can study more efficiently.

Confirmation

Functional

- The application should allow for doing exercises with both flashcards and any other supported exercise types (such as quizzes).
- The application should allow for adding/removing/editing decks of cards.
- The application should keep stats of the user's progress and exercise results.

Non-functional

Availability:

- The application should be working on a phone using a version of Android of 5.0 or above (Lollipop).
- Cards and stats should persist between application sessions, on the same device.

User story

Story Identifier: 1

Story Name: Usable flashcard

Estimated time: 6 days

Description

As a student, I want a flashcard to be visible in the application, with a predetermined text written on the front, and the answer on the back, so that I can test myself on the information on the card.

Confirmation

Functional

- The card should have some text written on it.
- The text written on the card should fit to the size of the card by being shrunk.
- The card should be clickable, which reveals the correct answer to the question.
- The card should be marked with a Q when the question is displayed.
- The card should be marked with an A when the answer is displayed.

Non-functional

- The card should be well fitted to the current screen.

User story

Story Identifier: 4

Story Name: Show user performance

Estimated time: 6 days

Description

As a student, I want to be able to tell the application if my answer to the card was correct or not on a predefined deck, and for the application to keep track of my performance, so that I am able to see the result.

Confirmation

Functional

- The deck should contain a number of predefined flashcards, with questions and answers on the front and backs.
- The flashcards should each have a “correct” and “incorrect” button below.
- Clicking the “correct” or “incorrect” button should show the next card in the deck.
- The user inputted answer (correct or incorrect) should be saved till the end of the game
- At the end of the game, the results should be displayed.

Non-functional

None.

User story

Story Identifier: 5

Story Name: Building main menu

Estimated time: 3 days

Description

As a student, I want my application to start in a main menu, so that I can navigate to playing my deck.

Confirmation

Functional

- The main menu should have a button to navigate to the decks.
- There should be a button to navigate to Exercise
- There should be a button to navigate to Statistics

Non-functional

- There should be a header which reads “Marc”.
- The buttons should have fitting icons.

User story

Story Identifier: 7

Story Name: Add/remove/edit flashcards

Estimated time: 10 days

Description

As a student, I want to be able to add/remove/edit flashcards in the existing deck, so that I can modify my deck of cards.

Confirmation

Functional

- The text on each card should be editable (by editing the corresponding Note).
- Each card and their corresponding note should be removable.
- There should be a (+) button, which when clicked, shows a prompt to add a note.
- Notes *cannot* be empty on either side.
- Long pressing a card should prompt the user for deletion of the card that has been long pressed.
- Tapping a card should enable editing of the corresponding note:
    - While editing a note, there should be buttons for saving changes.

Non-functional

- There should be a screen where all cards in the deck should be displayed with both the front and back visible.

User story

Story Identifier: 12

Story Name: Add/remove/edit cloze cards

Estimated time: 10 days

Description

As a student, I want to be able to add/remove/edit cloze cards in the existing deck, so that I can modify my deck of cards.

Confirmation

Functional

- The text on each card should be editable.
- Each card should be removable.
- There should be a (+) button, which when clicked, shows the screen for adding Notes. This screen should let the user select Cloze as the Note type.
- The syntax for adding a cloze should be something unusual, which is not easy to write without intent.
- Tapping a card should enable editing of the corresponding Note.
    - While editing a Note, there should be a button for saving changes.
- Long pressing a card should prompt the user for deletion of the corresponding note (and therefore also any other cards associated with the deleted Note).

Non-functional

- There should be a screen where all cards in the deck should be displayed with the current text written on them.
- Removing, editing and adding a card should force the user to do such an action to the Note associated to the card.

User story

Story Identifier: 8

Story Name: Add/remove deck

Estimated time: 5 days

Description

As a student, I want to create/remove decks of flashcards, so that I am able to keep collections of different types/groups of flashcards.

Confirmation

Functional

- Swiping a deck to either direction should prompt the user to confirm deletion of the selected deck.
- A (+) symbol should be visible when you're in the "Decks" screen.
  - After clicking the (+) symbol, you should be presented with a pop-up which allows you to enter the name of the deck.
- Two buttons should exist for the deck creation pop-up, a confirm button and a discard button.
  - Clicking the confirm button should present you with an empty deck, with the title visible on top.
  - Clicking the discard button should present you with the "Decks" screen.

Non-functional

None.

User story

Story Identifier: 13

Story Name: Side menu

Estimated time: 5 days

Description

As a student, I want to be able to navigate the application by using a side menu, so that I can navigate the application more quickly.

Confirmation

Functional

- The side menu should be reached when the toolbars' menu button is clicked, or when the user swipes from the edge.
- The side menu should hold all main menu buttons.
- The side menu should hold a "Logout" button.
- The side menu should be closed by clicking outside the menu, or by swiping the menu back to the left.

Non-functional

- The icons for the menu entries should match the corresponding entries in the home menu (if applicable)

User story

Story Identifier: 11

Story Name: Users system

Estimated time: 5 days

Description

As a student, I want the application to have a system for using multiple (local) users and still have our decks separate for each other, so that I can more easily share a device with someone without having to see the other persons decks.

Confirmation

Functional

- If no user is logged in the app should start at a screen where the user can choose/create a user.
- When the profile has been selected, the main menu should appear.
- The picked user should persist between sessions; the app should automatically log the previous user in at application start, unless the user explicitly logs out from the side menu.
- The user should be able to logout using the side menu.
- The user profiles should be persistent between sessions.
- You should be able to delete a user from the start menu.

Non-functional

- Decks and games shall no longer be associated with “MemorizationTrainingTool”, but rather by different Users. Instead, Users should be associated with ”MemorizationTrainingTool”.

User story

Story Identifier: 9

Story Name: Persistent storage of decks

Estimated Time: 2 days

Description

As a student, I want my decks/cards to persist between sessions, so that I can easily practice my decks without needing to rewrite them everytime.

Confirmation

Functional

- The Decks are saved between restarts of the app
- The Notes within a deck should be saved between restarts of the app

Non-functional

None.

User story

Story Identifier: 10

Story Name: Saving statistics

Estimated time: 3 days

Description

As a student, I want my stats to be saved, and saved between sessions, so that I know how im performing on my different decks.

Confirmation

Functional

- Each user should have their own stats page, where they can view various statistics.
- The following stats should be tracked:
    - Total times played
    - Times played, per deck, per game mode
    - High score per deck, per game mode
    - Average score per deck, per game mode
- The stats should be displayable in a stats view
- The stats after a game is played should be saved to the stats object.
- When a deck is removed the stats for that deck should be removed.

Non-functional

None.

User story

Story Identifier: 14

Story Name: Quiz Mode

Estimated Time: 6 days

Description

As a student, I want a multiple choice-type game mode so that I can review my cards in different ways.

Confirmation

Functional

- There should be multiple choice questions, with 4 alternative answers
- The questions should have random answers taken from other answers in the deck
- When an answer is selected the correct answer is shown by being highlighted.
- When an answer is selected the user needs to manually proceed to the next question.
- Like the flashcard game mode, you will see your result in the end.
- The game mode should only be playable on Decks with at least 4 Cards.

Non-functional

- The layout should be similar to the flashcard mode layout.

User story

Story Identifier: 15

Story Name: Achievements

Estimated Time: 6 days

Description

As a student, I want the app to have achievements to give me milestones in my progress, so that I get motivated.

Confirmation

Functional

- When a game is played the achievements should be updated
- There should be different achievements for:
  - Get all answers correct on a deck
  - Played 10 games in total (milestones)
  - Played 20 games in total (milestones)
  - Played 50 games in total (milestones)
  - Played 100 games in total (milestones)
  - Created your first deck
  - Played your first deck
  - Unlock all achievements

Non-functional

- The achievements should be displayed as icons without text.
- When you have received an achievement you should not be able to lose it.
- When an achievement is clicked, the requirement should be visible.

User story

Story Identifier: 2

Story Name: Animated flipping of flashcards

Description

As a student, I want a good looking animation for when I flip a flashcard, so that the application better resembles real life use of flashcards.

Confirmation

Functional

- When the text of the flashcard changes by tapping the card, the change should be animated by rotating the card and the text.

Non-functional

None.

2.2 User interface

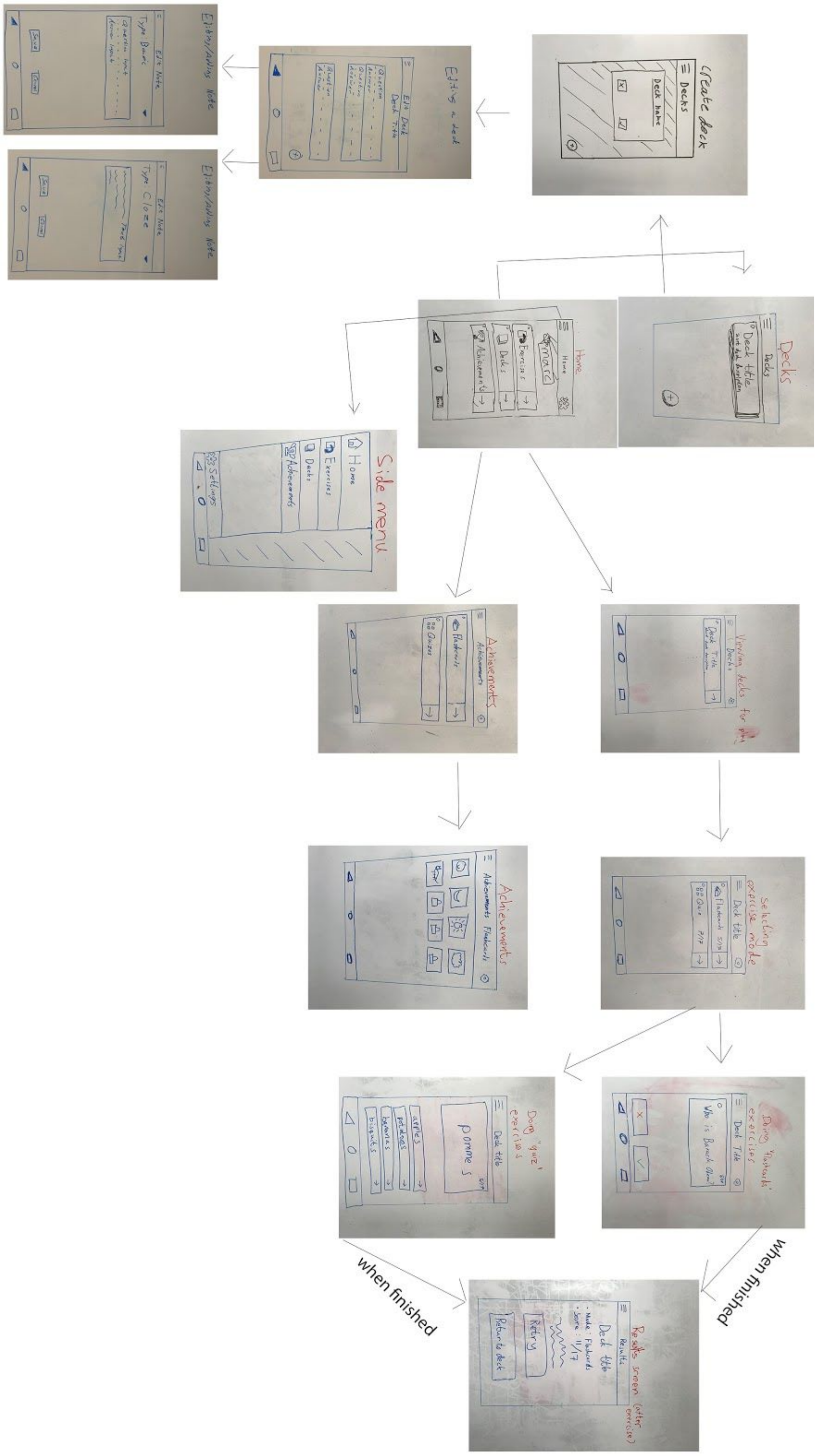
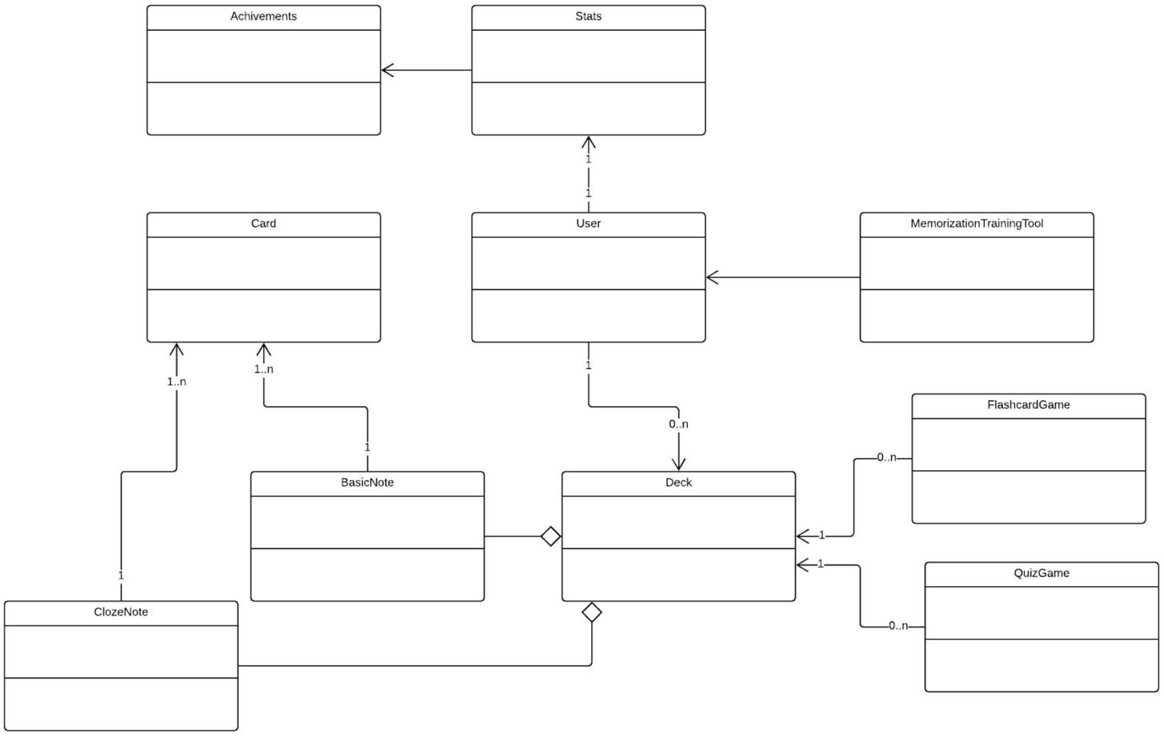


figure 1: a sketch of the user interface

3. Domain model



3.1 Class responsibilities

The Deck class has the responsibility of holding cards that can be used in a exercise.

The Card class holds the information for a card, which includes text for the the front and the back side of the flashcard.

The BasicNote class holds all the information for a flashcard, IE just one card with a front and a back.

The ClozeNote class holds all the information needed to generate cards with cloze deletions.

The User class holds the information about a user, their name, which decks they have, how they have performed on these decks.

The Stats class holds the statistics for a users, how they performed on their different decks in different game modes.
The Achievement class holds the achievements that the user are able to receive and are responsible for keeping track of which one are completed and which one are not.

The QuizGame class contains rules needed for the Quiz game mode, using this class one should be able to play a game of Quiz on a deck. The idea then that the player wont be able to flip the card to see right answer but instead get four alternatives where one is right answer and the other 3 is randomized answers from another questions.

The FlashcardGame class contains rules needed for the Flashcard game mode. If you use this class, these rules will be applied to the game.

The MemorizationTrainingTool class has the responsibility of keeping track of all the users and have a sense of which user is active at the moment.