DRABT Acceptance Testing  
The Oryx Group

The following criteria are used to judge if all the product requirements have been met:

[+ Registering a New User 1](#_Toc154356970)

[+ Loging In 2](#_Toc922471941)

[+ Creating a Card 2](#_Toc1820671669)

[+ Editing a Card 3](#_Toc543956925)

[+ Creating a Flashcard 3](#_Toc311969578)

[+ Editing a Flashcard 3](#_Toc1723550465)

[+ Creating a Deck 4](#_Toc678157921)

[+ Editing a Deck 4](#_Toc1028003333)

[+ Review the Flashcards in a Deck 4](#_Toc737683878)

## + Registering a New User

Users need to be able to create accounts.

### Interaction Algorithm:

* The user navigates to the login page on the website, or is redirected from the app, and selects “Register Account”.
* The system displays a popup asking for **email/username** and **password**.
* The user enters and submits their credentials.
* The system validates the input and:
* If the username and email are unique, a new account is registered, and the user is redirected to their new dashboard.
* If the username or email are not unique, or the entries are invalid, the user is prompted to try again.

## + Loging In

Users need to be able to log into their accounts.

### Interaction Algorithm:

* The user navigates to the login page on the website or accesses the app.
* The product displays input fields for **email/username** and **password**.
* The user enters their credentials.
* The user clicks the **“Login”** button.
* The system validates the input against the database and:
* If the credentials are **correct**, the system grants access and redirects the user to their dashboard.
* If the credentials are **incorrect**, the system displays an error message and prompts the user to try again.

## + Creating a Card

Users should be able to create cards that can store information for studying. This interaction is only available through the web platform.

### Interaction Algorithm:

* The user clicks on a button labeled **"Create Card"**, and an empty card appears.
* The system displays input fields for the text and/or image.
* The user can enter information into the text field.
* The user can upload an image for the card.
* Optionally, the user can reformat the text, including size, font, and colors.
* The user clicks the **"Save"** button.
* The system validates the input.
  + If valid, the system saves the card and displays a success message.
  + If invalid, the system prompts the user to complete the required fields.
* The new card is added to the user’s collection and becomes available for creating flashcards.

## + Editing a Card

Users should be able to modify cards in their collection. This interaction is only available through the web platform.

### Interaction Algorithm:

* The user selects a card from their collection and clicks “Edit”.
* The system reopens the card editor with the fields already filled.
* This interaction then proceeds as if creating a card.

## + Creating a Flashcard

Users should be able to create a flashcards consisting of a front and back. This interaction is only available through the web platform.

### Interaction Algorithm:

* The user clicks on a button labeled **"Create Flashcard"**, and an empty card appears.
* The system displays selection fields for the flashcard front and back.
* The user selects cards from their collect for the front and back of the of the flashcard.
* The user clicks the **"Save"** button.
* The new flashcard is added to the user’s collection and becomes available for creating study decks.

## + Editing a Flashcard

Users should be able to modify flashcards in their collection. This interaction is only available through the web platform.

### Interaction Algorithm:

* The user selects a flashcard from their collection and clicks “Edit”.
* The system reopens the flasahcard editor with the card selection fields already filled.
* This interaction then proceeds as if creating a flashcard.

## + Creating a Deck

Users should be able to create a deck of flashcards. This interaction is only available through the web platform.

### Interaction Algorithm:

* The user clicks the “**Create Deck**” button.
* A popup appears with a text entry for a deck name.
* The user submits a deck name, and the system validates it
  + If valid, a new deck is added to their saved collection, and the user is redirected to the deck editor.
  + If blank or invalid, the user is prompted to try again.
* The deck editor displays all the cards in the deck, with buttons to
  + Add flashcards from the user’s collection to the deck.
  + Edit the deck name.
* All changes to the deck are automatically saved.

## + Editing a Deck

Users should be able to modify decks in their collection. This interaction is only available through the web platform.

### Interaction Algorithm:

* The user selects a deck from their collection and clicks “Edit”.
* The system reopens the deck editor with the flashcard selection fields already filled.
* This interaction then proceeds as if creating a deck.

## + Review the Flashcards in a Deck

Users should be able to view their decks using a polished, animated interface.

### Interaction Algorithm:

* The user naviagtes to their dashboard on the website or app.
* The user selects a **deck** from their collection.
* The system displays the first flashcard in the deck, and presents several navigation options, including:
  + **“Show Answer”** or tapping the card, reveal the back side of the flashcard using a smooth animation.
  + **“Next”** and **“Previous”** (or swiping on the app), move between flashcards with a smooth, fluid transitions.
  + **“Exit”,** which returns the user to their dashboard.