

# Release Plan

## Cognoso

- **High level goals:**
  - User authentication + login page
  - Study set functionality
    - Create and practice with flashcards
      - Includes create, delete, and flip for flashcards
    - Save and share sets with other users
  - Additional Features
    - Learning minigames
    - Tag and Favorite Study Sets
    - Rating System for Study Sets
- **User stories defining the scope of the release: (MVP User Stories)**
  - As a user, I want to be able to make an account and save my study sets
    - Tasks
      - Make a account creation page for the user
      - Store user details securely
      - Make a login page for the user
  - As a user, I want to create a study set that can help me prepare for my courses efficiently
    - Tasks
      - Implement key functionality for flashcards
        - Create
        - Flip
        - Delete
      - Let users save sets in database
      - Let users share flashcard sets
- **Complete list of user stories**
  - Sprint 1
    - As a user, I want to be able to make an account and save my study sets (10 points)
      - Tasks
        - Make a account creation page for the user
        - Store user details securely
        - Make a login page for the user
    - As a user, I want to create a study set that can help me prepare for my courses efficiently (20 points)
      - Tasks
        - Implement key functionality for flashcards
          - Create
          - Flip

- Delete
    - Let users save sets in database
  - Sprint 2
    - As a student, I want to have fun while learning (5 points)
      - Tasks
        - Create minigame
          - Drag and drop question+answer game
      - Spike: front end implementation of minigame
    - As a student, I want to make a study set quickly and easily (10 points)
      - Tasks
        - Allow users to upload PDFs to create flashcards
        - Allow users to easily create flashcards on their own (refine UI on frontend)
      - Spike: PDF parsing
    - As a user, I want to be able to manage an account and save my study sets (10 Points)
      - Tasks
        - Make a account creation page for the user
        - Store user details securely
        - Make a login page for the user
        - Make a delete account button with confirmation
        - Make change password option
    - As a student, I want to make a study set quickly and easily (10 points)
      - Tasks
        - Allow users to upload PDFs to create flashcards
        - Allow users to easily create flashcards on their own (refine UI on frontend)
      - Spike: PDF parsing
      - Spike: Run LLM locally
  - Sprint 3
    - As a student, I want to have fun while learning (15 points)
      - Tasks
        - Create minigame design on frontend
          - Drag and drop of question and answer “cards”
          - Button to submit questions
        - Implement backend functionality of minigame (taking info from flashcards)
          - Get info on flashcards from selected deck
        - Button to play minigame from a selected flashcard set
      - Spikes:
        - UX on frontend (graphic design)
        - Calling info from backend
    - As a student, I want to make a study set quickly and easily (10 points)

- Tasks
    - Allow users to upload PDFs to create flashcards
    - Allow users to easily create flashcards on their own (refine UI on frontend)
  - Spike: PDF parsing
  - Spike: Run LLM locally
- As a user, I want students to be able to find and use study sets easily (3 points)
  - Tasks
    - Implement search bar to look for sets by name
    - Backend search implementation
- As a user I want to be able to delete flashcard sets I don't want to use (1 points)
  - Task
    - Delete flashcard set from backend
    - Implement button to delete set on frontend
      - Send confirmation to backend for deletion
- As a user I want to be able to delete flashcards that I no longer want to keep (2 points)
  - Task
    - Delete single flashcards with a button
- As a user I want to be able to know what deck I am working on while in flashcard viewer mode (3 points)
  - Task
    - Implement backend endpoint to give flashcard set name
    - Allow this to be displayed as title on the frontend in flashcard viewer
- Sprint 4
  - As a student, I want to make a study set quickly and easily (40 points)
    - Tasks
      - Allow users to upload PDFs to create flashcards
      - Allow users to easily create flashcards on their own (refine UI on frontend)
    - Spike: PDF parsing
    - Spike: Run LLM locally
  - As a user I want to see all my flashcard decks on the home page (2 points)
    - Tasks
      - Show decks on home page upon login
      - Favorites and other flashcard decks shown separately
  - As a User I want to be able to see other user-made flashcard sets (3 points)
    - Tasks
      - Show flashcard sets made by other users on home page
      - Show flashcard sets made by other users on search\_results page
      - Create a new page to view other user decks?
  - As a user I want to save my favorite flashcard sets (2 points)
    - Tasks:

- Favorite button added to flashcard viewer (star button on top right)
  - Send the deck id to the backend to save in a separate category
  - Unfavorite decks
- As a User I want to be able to rate certain flashcard sets that were made by other users (2 points)
  - Tasks
    - Implement 5 star system to rate certain flashcard sets
    - Send info to backend
- As a user I want to be able to see what other users have rated my flashcard set(s) (3 points)
  - Tasks
    - Show what other users have rated my flashcard set in the deck manager set.
  - Show as a popup on the bottom right of deck icon when hovering over it with mouse
    - Make rating stars that are disabled and cannot be edited (MUI rating)
- As a user I want to be able to set icons for my decks (4 points)
  - Tasks
    - Edit button on each deck in deck manager
    - Popup that allows users to choose an icon for selected deck
    - Set icon and send info to backend
- As a user I want to be able to ask a chatbot questions (1 point)
  - Tasks
    - New page for ai chat
    - Send prompt to backend and get response
- **Product backlog (all high level goals and user stories that did not make it into release at this point. May change in priority!!)**