CMPT 276 – GROUP PROJECT PROPOSAL

PROJECT TITLE

Cirkle – Collaborative Study for Better Results

Group Members

1. Julianna Morena
2. Mudasser Mashal
3. Yogya Agrawal
4. Nafeesa Leena

Link to Github Repository

https://github.com/CMPT-276-SPRING-2025/final-project-07-hills

Project Overview

Description

Circle is a web application designed to enhance collaborative learning. It does this by providing an integrated platform with tools for collaboration, such as shared notes, file uploads, flashcards, and a group pomodoro timer with leaderboards for your group. Many people either struggle to effectively collaborate or use 2-3 different platforms to meet all their needs. Cirkle aims to address this problem, an all-encompassing platform with only the essential features to make learning your next course/skill a breeze.

Idea and Importance

The idea for Cirkle came from personal anecdotes of students being disorganized and feeling overwhelmed when studying in groups. Apps like Notion exist for taking notes together and including links to files. On the other hand, apps like Google Drive exist for uploading files and sharing them with a group. Finally, we have flashcard generators and pomodoro websites that claim to improve productivity and help you revise better. Most students who are serious about their preparation would benefit from all these resources. However, there is no such service that puts all these essential services under one umbrella and packages them in a user-friendly way. Cirkle aims to do that. We believe that with seamless integration to all of these essential features, users would benefit greatly from collaborative studying. With cirkle, users do not have to worry about logging into multiple different services and sharing resources across multiple emails. It can all be done with a single website.

With the addition of a leaderboards feature under our flashcards and pomodoro timer, we aim to generate healthy competition that will motivate users to do better than their peers, which has a positive net result. It also gamifies the process, adding an extra incentive to push users that extra bit further.

Potential Users and Their Needs

1. **University Students:** University students often engage in group studying to improve their understanding of course material. However, to do so, they require tools to work on notes together and share their resources with each other. They also need tools to improve their learning, such as flashcard generators and AI that can explain the concepts to them. They would also benefit from productivity hacks such as pomodoro timers that keep them focused for a fixed period.

2. **Lifelong Learners:** Lifelong learners often pursue knowledge outside of formal education, including learning new skills, languages, or subjects. They may study independently or in peer groups and need effective tools to stay organized and motivated. They need an area where they can store all of their notes and resources. They also need to have the ability to join groups with peers who have the same goals and motivation as them. As they are learning a new skill out of their own will, they would also like to have some sort of resource that keeps track of their progress, preferably automatically. They would also benefit from constantly revising the material and using tools like pomodoro timers to boost their productivity.

Personas

1. **University Student – Josh Gilford**
    - Age: 21
    - Major: Computer Science
    - Needs: A platform to take notes together in class and have the ability to share the resources he finds with the rest of his peers. Josh would also like to study together with his peers, but they always get distracted when they do group study sessions. Maybe some sort of flashcard game with the ability to compete with his friends would prevent that?
    - Goals: Get good grades and understand the course material

2. **Language Learner – Diana Chen**
    - Age: 29
    - Occupation: Marketing Professional
    - Needs: Spending time learning new languages with friends. She would want to share the resources she finds online with everyone. She would also like a platform that would keep her accountable about how much time she is dedicating to learning the language. Maybe a pomodoro timer with leaderboards showing who has spent the most time keep her accountable?
    - Goals: Achieve fluency in a new language whilst having fun learning it together with her peers

Chosen APIs

1. **Google Drive API:** Facilitates secure file storage and sharing within groups whilst providing access control.
2. **Notion API:** Facilitates collaborative note-taking with group members whilst including many features that ensure proper note-taking and structured pages.
3. **Cohere API:** Summarize notes and file uploads and present the summary in user-friendly format. Also, turn the notes and file uploads into questions and answers that can be used for flashcards.
4. **AnkiConnect API:** Generate flashcards with the provided questions and ensure proper deck management and spaced repetition abilities.

Features

1. **Google Drive API**
   a. **Name:** Upload Files
   Description: The user will be able to upload files when they click “Upload File” on the group dashboard. The app sends the file to Google Drive using the Drive API and the API returns a file ID and access link.
   User Story: As a user, I want to be able to upload files to my group so that everyone can access them. I would also like to generate flashcards using the information in my uploaded files.

   b. **Name:** Set Access Permissions
   Description: The API allows the application to configure permissions for the group for each file that has been uploaded (eg. view-only, view and edit)
   User Story: As the owner of a file, I might want it to be view-only or have other people be able to edit it.

   c. **Name:** View Uploaded Files
   Description: The app can display file previews directly using Google Drive’s file preview API.
   User Story: As a user, I might want to quickly preview a file to make sure it is the correct one before downloading it or using it for other cases such as flashcard generation.

2. **Notion API**
    a. **Name:** Create Notes
    Description: A user can start a new note in the group. The app then sends a request to Notion to create a page within the group.
    User Story: As a user, I want to be able to collaborate on notes with all the people in my group. This involves starting a new note and allowing everyone in the group to be able to edit it.

    b. **Name:** Organize Notes
    Description: The user can organize notes using Notion's structured format (eg. nested sections, bullet points, headings, etc)
    User Story: As a user, I want to be able to manipulate my notes in a way that would make it better than using pen and paper. This would involve creating tables, using headings and sub-headings, and using other techniques that would make my notes easier to review.

    c. **Name:** Link Files to Notes
    Description: The collaborative notes can include links to files uploaded through Google Drive, enabling seamless access to related study materials.
    User Story: As a user, during the process of writing a note I might notice the fact that a file that has been uploaded explains the same concept that I am writing down the notes for. At this point, I can just refer to the file by including its link, and makes it very easy for the user to review the material later.

3. **Cohere NLP API**
    a. **Name:** Question Answer Generation
    Description: Automatically generate question-answer pairs by extracting key points from the user notes or uploaded files.
    User Story: As a user, I want an AI that can generate questions and answers from my notes, so I can efficiently create flashcards without having to worry about creating them manually.

    b. **Name:** Paraphrase Assistance
    Description: Provide paraphrased versions of note content to help reframe complex ideas into simpler terms for better understanding.
    User Story: As a user, I want my notes to be paraphrased into simpler language, so I can better understand and memorize complex concepts.

    c. **Name:** Personalized Study Recommendations

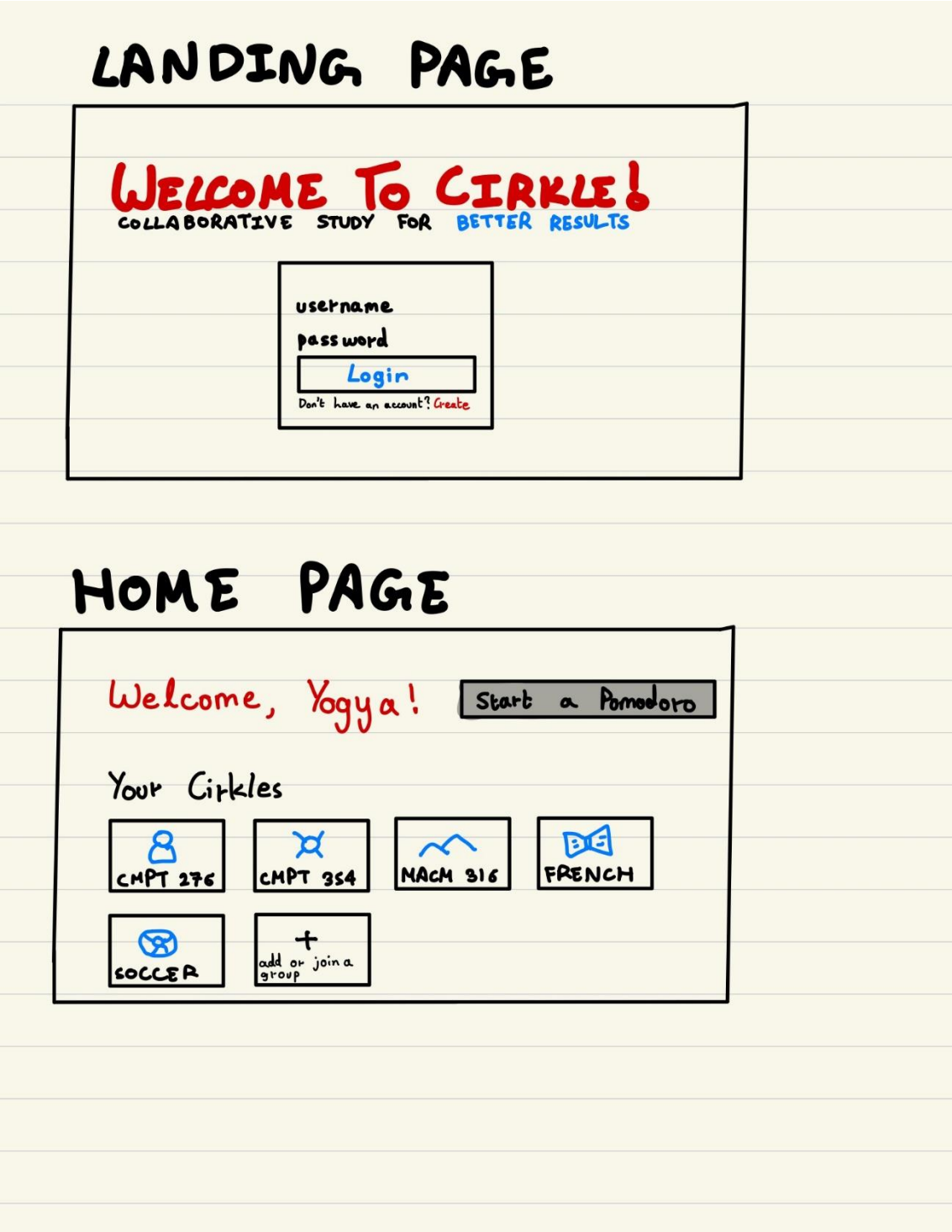
Description: Provide personalized study recommendations based on the user's notes and generated flashcards.
User Story: As a user, I want the system to recommend additional topics to study, so I can ensure I fully understand the material covered in my notes.

4. **AnkiConnect API**
   a. **Name:** Flashcard Generation
   Description: Add new flashcards and create a new deck with the question-answers that have been provided by another API
   User Story: As a user, I would like the application to be able to generate flashcards without having to worry about manually doing them.

   b. **Name:** Deck Management
   Description: Allow users to create different decks for different groups or topics and switch between them seamlessly.
   User Story: As a user, I would like to have different decks for different topics (eg. Frontend, Backend, UI/UX) so that I can prepare for each topic separately and organize my studies better.

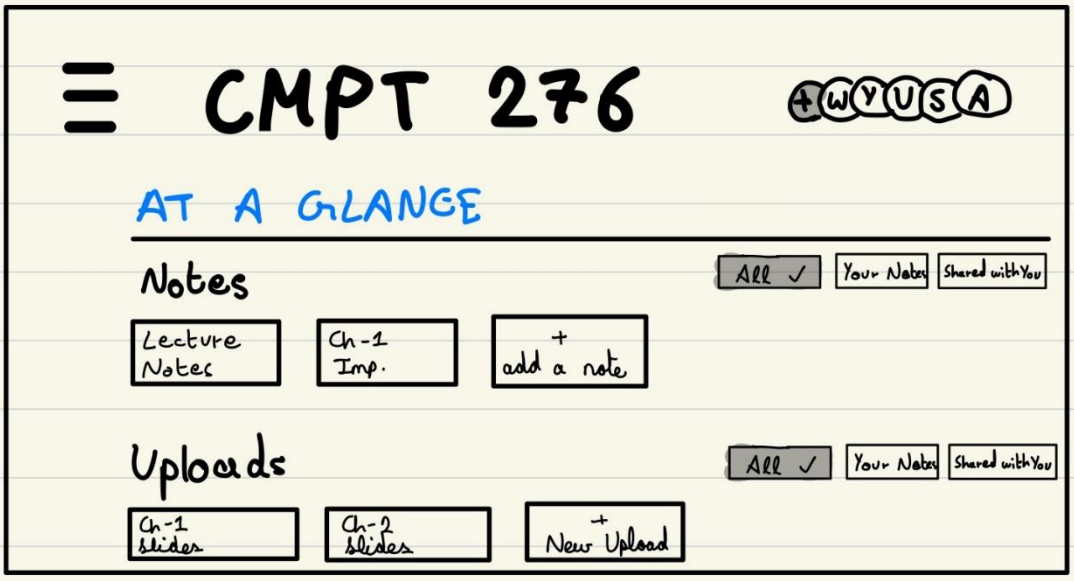
   c. **Name:** Review Flashcards
   Description: Enable users to review flashcards with spaced repetition. The API prompts the user to review flashcards at optimal intervals for retention based on how the user performed.
   User Story: As a user, I would like to practice the concepts that I struggle with more. However, I also struggle to motivate myself to do them on my own as I do not perform well in them.

Low Fidelity Storyboard of the Application

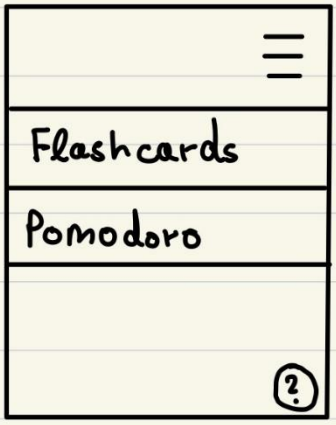


ADDING A GROUP

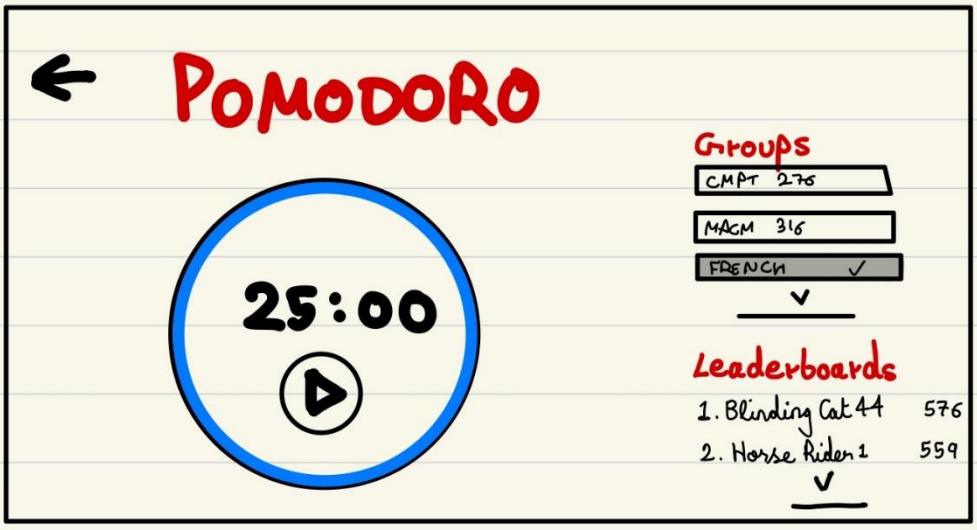
"YOUR GROUP" PAGE



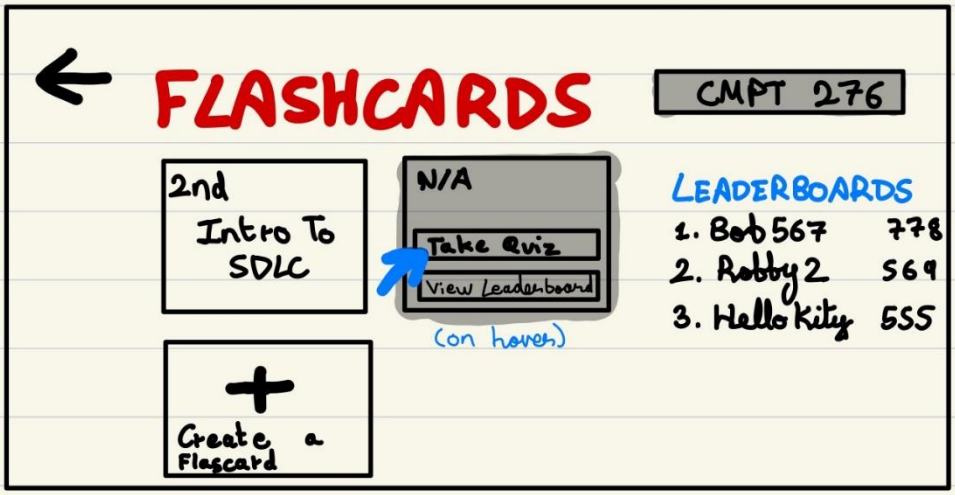
NAVIGATION BAR



POMODORO FEATURE



FLASHCARDS FEATURE



Front-end Technology Stack

Stack: React with Tailwind CSS

Explanation: For this project, we chose React because it helps build dynamic, reusable components, which is perfect for integrating APIs for features like Collaborative Notes and File Uploads. The components are also extremely useful in building the pomodoro timer and leaderboards for the flashcards, as well as managing the overall design of the application. React also makes it easier to manage content that changes quickly and works well with state management. State management will be especially useful to show the group and collaborative note features using two accounts in the same device as the project does not have a proper back end. For styling, we are using Tailwind CSS since it provides ready-to-use design classes, making the layout simple and quick to build. This stack allows us to create a clean, responsive, and efficient user interface without needing a back end.