Proposal for Orbital 23

Team Name:

Noted!

Proposed Level of Achievement:

Artemis

Motivation

As students from interdisciplinary residential colleges, we were intrigued by the connections between various subjects and how they come together to create a final product. However, we noticed that many majors and faculties within NUS face segregation, which prompted us to create a platform that fosters a more interdisciplinary approach beyond residential colleges. By allowing students to access resources from different faculties, we hope to encourage independent, interdisciplinary learning, ultimately creating a more fulfilling and diverse learning experience for NUS students.

Furthermore, as students passionate about our majors and broadening our horizons, we are always looking for new opportunities to learn things beyond the scope of our modules. This has led us to realise that there is a shortage of consolidated resource materials available to assist us in expanding our knowledge beyond what we have learnt in class.

Hence, by taking into consideration these points, we wanted to create a one stop shop for all NUS students, to collectively contribute to this encyclopaedia and even network with students who have the same interest.

Aim

[Please describe what you aim to achieve.]

Our goal is to develop an app with a user-friendly interface that is intuitive and easy to navigate. Our app is primarily designed to serve as an encyclopaedia that offers NUS course materials and even student-written articles with secondary features for networking. By leveraging on `reputation` scores, we aspire to cultivate a supportive learning environment that enables NUS students to make meaningful contributions to NUS' very own rendition of an encyclopaedia.

User Stories

[Please describe what the users would be able to do with your system.]

1. As a current NUS student, I am able to access publicly accessible course materials for any NUS modules.
2. As a current NUS student, I am able to publish my own informative articles or notes on my course related topics, with due credit given and in return, I would receive a better reputation score.

3. As a current NUS student who is not the author of the note, I am able to offer suggestions regarding a particular section of the notes, in return, I would receive a better reputation score.
4. As a current NUS student, I am able to publish articles that I wrote on any topics that I am interested in, in return, I would receive an improved reputation score.
5. As the author of the note, I am able to verify and accept the suggestion of fellow NUS students.
6. As a user, I would be able to easily find fellow schoolmates who have the same interests and chat with them.

Features and Timeline

The application has several main features that are designed to enhance the user experience.

An overview of the features:

1. Accessibility to publicly accessible course materials cross-faculty, meaning that course instructors and authors of course materials has approved its distribution
2. Ability to make suggestions with an additional layer of verification on the credibility and reliability of the suggestion
3. Based on their activity, each user is given a reputation score
    a. Activities that increases reputation score:
        i. Publishing insightful or helpful work that has been complemented by other users
        ii. Suggesting useful comments to posts and articles
    b. Activities that decreases reputation score:
        i. Publishing articles with no due credit that have been flagged by users.
        ii. Publishing unreliable and factually wrong articles that have been flagged by users.
4. Ability to publish their own articles on any topics that they find interesting
5. Tracking users' activities to suggest people with similar interests, making it easier to connect with like-minded individuals.

Timeline:

Implementation I [21 May - 20 June]

- Enabling users to upload course materials and their own articles and notes on the app. (Firebase Cloud Storage)
    - Firebase Cloud Storage is a scalable and secure cloud storage solution provided by Google, which will ensure that user data is stored safely and efficiently
- Using an SQL database, each user will be assigned their own reputation score.
- Implementation of a feedback system that automates the process of adjusting their score based on their contributions on the app. (Cloud Functions for Firebase)
    - These functions can be triggered based on user actions, such as uploading a course material or an article, and can update the user's reputation score accordingly. Hence, managing and adjusting user reputation scores in real-time.
- Implement a network function
    - Graph Data Structure
        - To represent the relationships between users in the app, which can help users to discover new connections and expand their network

        - Can also be used to suggest relevant content and connections to users based on their interests and engagement
        - How can we implement this?
            1. Represent the user interests as a graph
                a. Once you have stored the user interests in a database, you can represent them as a graph, where each user is a node and the edges represent connections between users.
            2. Use Jaccard similarity to compute the similarity between the interests of each pair of users and use these similarities as weights for the edges
            3. Apply the modularity clusterer in Flutter (modularity-based algorithm) to identify groups of users with similar interests. This will partition the graph into communities of nodes that are highly connected to each other, but less connected to nodes outside the community
            4. We can then suggest links between users in the same community, i.e. users chat with each other
    - Private Chats for networking (Firebase Realtime Database)
- UI/UX of the app (FlutterBoost)
    - Profile (NUS Email, Name, Reputation Score, Interests, Chats)
    - Article feed
        - Allows users to browse and discover articles and materials uploaded by other users on the app.
        - Can be personalised based on user interests and engagement, and can also include curated content based on the topics that they frequently engage in
    - Search Page
        - Allows users to search for articles, materials, and other users on the app.
        - Can be customised to include filters and categories that make it easy for users to find what they are looking for
    - Editor's Page (to publish your own materials)

Implementation II [21 June - 20 July]
- Implement the authentication feature to ensure only NUS students can register and access the app (Firebase Authentication)
- Security Checks to ensure that data provided are protected. (Firebase Security Rules to control access to your Firebase resources)

Testing & Review [21 July - 20 August]
- System Integration Testing
    - Firebase Test Lab to automatically test across multiple devices and configurations
    - Ensure that all features are working together as expected
- User Acceptance Testing
    - Get friends to test out our app and provide feedback on the features and ui/ux

Tech Stack

[Please list down the technologies that you are planning to use.]

Flutter and Dart: FrontEnd Mobile App Development Framework
Firebase: BackEnd
MySQL: Database
Firebase Authentication: Authentication

Qualifications

[Please elaborate on your technical skills and project experience of your team. This is required only for teams targeting Apollo 11 and Artemis.]

Lee Yi Lin

- Prior experience doing iOS development on XCode (Swift), creating over 10 different applications, ranging from storyline games to xylophone apps.
- Familiar with application building using StreamLit
- Experience in Machine Learning
    - Computer Vision (Transfer learning on ResNet 50 to optimise the model to categorise flowers, YoloV7 for object detection)
    - NLP (LangChain, HuggingFace open source repos to incorporate google search results into natural language, and OpenAI's api to build a chatbot to automate data analysis)
- HacknRoll 2023, where I used Telegram's API to fetch users' inputs and conduct sentiment analysis (NLTK and Textblob) and SQLite3 to create a central database to keep track of users' scores.

Kirthiga

- Prior experience doing iOS development on XCode, created various applications such as a Calculator app.
- Prior experience in creating programs using Python
    - Used the Luhn Algorithm to create an application that can identify the type of credit card being used and determine if it is valid
- Prior experience in creating Telegram bots and using Telegram API