Proposed Level of Achievement: **Gemini**

Target Audience: **Members of NUS (freshmen especially)**

**Project Scope:**

An interactive android application which serves as a vehicle to allow students to book, join, and gain from various study sessions occurring across the campus. Moreover, apart from being used as a platform to organise study sessions between various groups of students across NUS. It gives NUS students, especially undergraduates and freshmen, the ability to collaborate and consolidate their strengths and weaknesses within certain areas of their academic learning.

**Core Features Developed:**

* User Authentication (Log in with an email and password)
* Create user profile
* Logout of current user profile
* A tab to show a list of joinable sessions, each session can be joined by the
* Tapping on a session on the list will prompt a question asking if the user wants to join the session
* A tab to show a list of sessions the user has joined
* A tab to show a list of sessions the user has hosted
* Session can be created by the user

**Problems Encountered**

1. Finding the location of bugs
   1. The main problem we have faced during prototyping is our lack of knowledge about the Java language. Although we can understand the logic and code for our various features, they are often plagued with bugs. Debugging the application is a hassle as we do not know the exact location of our code which causes the bug.
2. Designing the layout
   1. We also faced problems in designing the layout of the application. The layout appears differently on different computers and it is difficult to pinpoint where the problem arises. This would cause a lot of difficulty especially if we are not able to meet often due to personal schedules.
3. Mastery over Version Control
   1. Lastly, we are still unsure of how to utilise git properly and to the best of its ability and thus have faced a few problems in version control.
4. Technical Difficulties using Android Studio
   1. Implementing a Navigation Drawer View was extremely painful as the MainActivity already had section fragments, and implementing the drawer view required us to manoeuvre around the fragment code, which lead to further bugs and errors
   2. Extremely slow emulator speed performance was caused due to the excessively high quality of drawable elements used in the application
   3. When switching from one tab to another in MainActivity, the individual lists duplicate themselves for no apparent reason, even though all the lists are separate with respect to each other in the Java code.

**Development Plan (timeline since Orbital Ignition):**

* Week 1: finalise on ideas and meet up with advisor to consolidate our idea
* Week 2: prepare for ideation by finalising major features of the product
* Week 3: finalise the list of major software and respective technicalities needed
* Week 4: finalise initial frontend sketches and establish user authentication
* Week 5: refine initial design ideas and work extensively on backend database development
* Week 6: establish list feature for study sessions (both frontend and backend)
* Week 7: implement additional features (e.g. modules taken, create and host study sessions)
* Week 8: continue backend development, while concurrently developing frontend UI
* Week 9: finalise initial prototype of our product
* Week 10: refine both frontend and backend, with emphasis on frontend development

**Task list for designing the system:**

|  |  |
| --- | --- |
| Task | Duration (Hours) |
| Lift-off Day 1 | 9 |
| Lift-off Day 2 | 9 |
| Initial ideation and testing of various software to design the product (Firebase and CRUD, Android Studio, IOS with Swift 4, Piskel, Photoshop, Illustrator, Excel) | 20 |
| Designing main frontend framework as well as structuring backend mainframe | 32 |
| Gathering data for databases to implement listing feature, as well as create database for modules in conjunction with prerequisites | 16 |
| Building and developing application, with full user-interface and backend support | 14 |
| Testing and debugging (backend, as well as after integration with frontend UI) | 40 |
| Total number of hours (tentative) | 140 |

**Necessary technologies needed:**

1. Firebase
2. Android Studio
3. Photoshop/Illustrator (for front end UI design)
4. Figma (designing User Interface panels)
5. Spark Adobe (designing FrontEnd of the application)

*Please note, these are the required technologies that will be essential for the design at this stage. If further technologies are required in the future, this list may be expanded appropriately as and when required. At this stage, we have finalised our idea, ideated which software and platforms we will be using for product development, as well as sketched a rough timeline to achieve major requirements of the solution.*