

## Project Description:

Stubby is a smart, school-specific study matching application that connects students enrolled in the same college or university based on similar subjects, study preferences, and availability. Unlike traditional social or messaging platforms, Stubby is designed with academic goals, moderation, and structured learning in mind. Users are matched either with a study partner or a group and can schedule sessions in approved campus locations. All activity is verified via school credentials and feedback mechanisms to ensure safety and productivity.

## Requirements Summary:

<b>MINIMUM REQUIREMENTS</b>	Processor Cores	Single Core
	OS	Android 6.0 (Marshmallow)
	RAM	2 GB
<b>RECOMMENDED REQUIREMENTS</b>	Processor Cores	Quad Core
	OS	Android 10
	RAM	4 GB
<b>OTHER REQUIREMENTS</b>	School login	GPS and Storage

Table 1. System Requirements

Stubby runs smoothly on both low-end and modern Android devices to ensure accessibility for all students.

## Prototype Description:

The prototype was developed by coding the GUI directly using React, Tailwind CSS, with the help of HTML, and Visual Studio Code. This approach allowed for a more realistic simulation of the user experience and better alignment with the final product's intended behavior. Major interface components include:

- Profile creation with subject, schedule, and study preference inputs

- Match suggestions for either partner or group sessions
- Venue map and session booking UI
- Feedback and reporting system
- Moderator tools and badge rewards (conceptual)

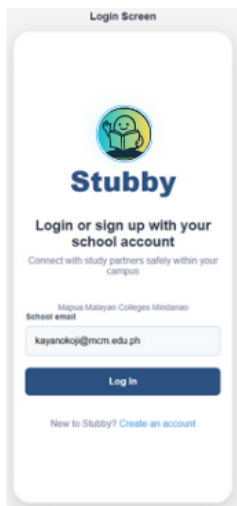
### Stubby HTML Link:

[https://cdn.fsbx.com/v/t59.2708-21/506793846\\_1435899484212666\\_6634242342077511554\\_n.html/stubby\\_app\\_ui.html?\\_nc\\_cat=109&\\_nc\\_cb=47395efc-74c935b2&ccb=1-7&\\_nc\\_sid=2b0e22&\\_nc\\_ohc=gzAwngYHIAcQ7kNvwFz-L5l&\\_nc\\_oc=AdmcghmSh5qolGwcfuNLgqqO0g8KO\\_o0O66Riax4nguuPgteOiUYCIQM4FahwVpNAPQ&\\_nc\\_zt=7&\\_nc\\_ht=cdn.fsbx.com&\\_nc\\_gid=ckCgORIO1PjxAaQ6N2ByYQ&oh=03\\_Q7cD2gFuVe2j40zPtdyDudlAmc4BSg5xZCByimn\\_6Fw9D7-t0w&oe=684DF740&dl=1](https://cdn.fsbx.com/v/t59.2708-21/506793846_1435899484212666_6634242342077511554_n.html/stubby_app_ui.html?_nc_cat=109&_nc_cb=47395efc-74c935b2&ccb=1-7&_nc_sid=2b0e22&_nc_ohc=gzAwngYHIAcQ7kNvwFz-L5l&_nc_oc=AdmcghmSh5qolGwcfuNLgqqO0g8KO_o0O66Riax4nguuPgteOiUYCIQM4FahwVpNAPQ&_nc_zt=7&_nc_ht=cdn.fsbx.com&_nc_gid=ckCgORIO1PjxAaQ6N2ByYQ&oh=03_Q7cD2gFuVe2j40zPtdyDudlAmc4BSg5xZCByimn_6Fw9D7-t0w&oe=684DF740&dl=1)

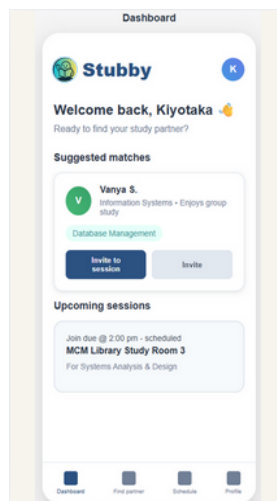
### User Scenario:

Dana, a 1st-year education major who transferred from another school, has difficulty joining study groups because of unmatched course codes. She uses Stubby to enter her subject preferences, and the app's subject-aware algorithm finds an equivalent match for her at her new university. Dana then schedules a session at the library with the matched students, attends the session, and leaves positive feedback after.

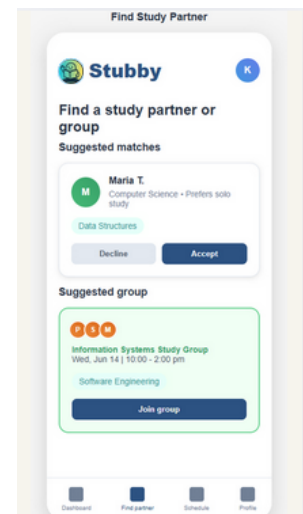
### Stubby Mock-up/Prototype:



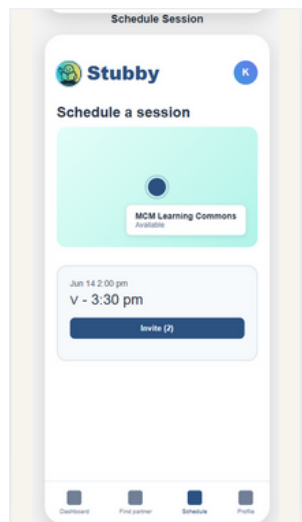
**Login Screen  
Partner**



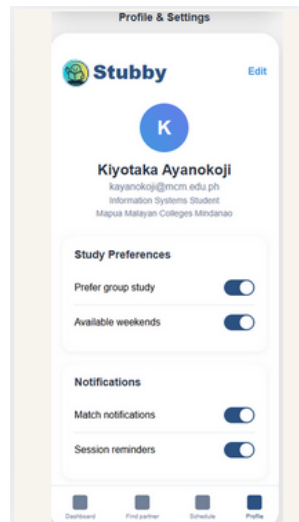
**Dashboard**



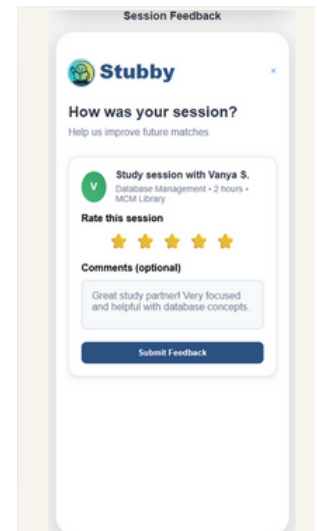
**Find Study**



**Schedule Session**



**Profile & Settings**



**Session Feedback**

**Rationale:**

We opted to code the GUI manually to gain full control over the layout, logic, and responsiveness of the interface. This method allowed us to implement dynamic features and interactions that a static prototype tool might not support, while also preparing the team for future integration with real backend systems. The design combines elements from previous design alternatives:

- Dashboard-central for efficiency
- Venue-first for campus integration
- Gamified progress for engagement

**Changes to the Requirements:**

- Added: Subject equivalency logic and gamification elements
- Removed: Real-time chatroom due to moderation difficulty
- Refined: Study preference toggle, rating system, and admin moderation flow

**Initial Evaluation Plan:**

The evaluation will be conducted remotely via Discord or Microsoft Teams. Participants will screen-share and perform specific tasks with real-time observation by team members.

With that said, the Evaluation plan is split into three separate parts: Usability Specifications, Heuristics Evaluation, and Participant Survey and Feedback.

### **Usability Specifications**

The creation of this prototype will aim to achieve the following measures when it appeals to the use:

**Learnability:** New users can set up a profile and match in under 5 minutes

**Efficiency:** Session booking takes 3 minutes or less

**Safety:** All users are school-verified and sessions happen on campus

**Flexibility:** Can choose between group or partner setups

**Feedback:** Every session ends with a rating prompt

## Population

Targeting 10–15 college students across different programs. They will complete the following benchmark tasks:

- Create a verified profile
- Match with a buddy or group
- Schedule a session using venue map
- Submit session feedback

Developer / UI Designer Member	Task(s)
Shelley Magdalan	Will be recording time users interact with a task section, taking notes of the user's experience, and relay the task that the participant will do.
Cedric Josh Leal	Will be recording time users interact with a task section, taking notes of the user's experience, and relay the task that the participant will do.
Angeline Kaquilala	Will observe user navigation flow, document usability issues or confusion, and help summarize participant feedback post-session.

Table 2. Team Member Tasks

Profile Setup	Within 3 minutes or Below	Highly Acceptable	Successful
---------------	---------------------------	-------------------	------------

	Above 3 minutes	Not Acceptable	Unsuccessful
Matching	Within 2 minutes or Below	Highly Acceptable	Successful
	Above 2 minutes	Not Acceptable	Unsuccessful
Booking a Session	Within 5 minutes or Below	Highly Acceptable	Successful
	Above 5 minutes	Not Acceptable	Unsuccessful
Feedback Submission	Within 2 minutes or Below	Highly Acceptable	Successful
	Above 2 minutes	Not Acceptable	Unsuccessful

### Heuristic Evaluation

Evaluation of Stubby will use the 10 Usability Heuristic method of Evaluation.

#### Visibility of System Status

The prototype keeps participants informed about what is happening by providing clear, timely feedback on actions, such as session confirmation and form submissions.

#### Match Between System and Real World

Stubby uses natural language and student-friendly terminology rather than technical jargon. Labels and interface elements follow familiar patterns, making navigation logical and intuitive.

#### User Control and Freedom

Users are given clear options to cancel, go back, or log out at multiple stages. The app avoids trapping users in any process and ensures that navigation is reversible.

#### Consistency and Standards

The interface uses consistent colors, icons, and layout standards throughout all screens. Terms like "match," "session," and "feedback" are used uniformly.

#### Error Prevention

Forms are validated with warnings and input constraints to reduce the chance of user errors. Only valid options are selectable to prevent common mistakes.

**Recognition Rather Than Recall**

Icons, subject tags, and user data are always visible on-screen, reducing the cognitive load. Users do not have to memorize steps or data across pages.

**Flexibility and Efficiency of Use**

Stubby caters to both novice and experienced users with simple onboarding and quick-access buttons for repeat tasks such as booking or viewing sessions.

**Aesthetic and Minimalist Design**

The interface avoids unnecessary text or clutter, using visual hierarchy to focus the user’s attention on the most important actions and information.

**Help Users Recognize, Diagnose, and Recover from Errors**

Messages are written in plain language and offer clear advice when inputs are missing or incorrect. Users are guided to correct issues without technical terms.

**Help and Documentation**

Helpful tips are embedded in the form fields and screen instructions. While minimal documentation is needed, assistance is available in the UI where relevant.

**Participant Survey and Feedback**

After conducting the online test,

DATA GATHERING METHOD	DESCRIPTION
Survey (Quantitative)	After the Online Testing, the team will be handing out a survey to the participants to gather data for the user’s experience with the prototype which the team will be interpreting in a 5-point Likert scale
Feedback (Qualitative)	The survey that the team provided will support a Feedback section to help users/participants speak out concerns or issues with the prototype that needs to be addressed.

Table 2. Data Gathering Methods

Question	Method of Answer
Section 1	
Participant Number	Short Answer
On a scale of 1 to 5, how would you rate	5-Point Scale

your experience with Stubby?	
On a scale of 1 to 5, how was the UI design of the prototype?	
How easily were you able to follow the tasks provided?	
<b>Section 2: Features of the Prototype</b>	
Navigation and Screen Transitions	5-Point Scale
Viewing Match Suggestions	
Scheduling a Study Session	
Selecting a Campus Venue	
Submitting Feedback After Sessions	
Using Dashboard to View Sessions	
Accessing and Editing Profile Information	
<b>Section 3: Feedback Section</b>	
Your Feedback	Short Answer

**Interpretation Scale:**

Score	Rating	Outcome
5	Highly Acceptable	Excellent
4	Acceptable	Meets expectations
3	Neutral	Needs improvement
2	Fairly Acceptable	Needs redesign
1	Not Acceptable	Failed usability



### **Implementation Challenges and Justification:**

While coding the Stubby GUI gave us full flexibility, it also introduced certain challenges. For instance, handling dynamic filtering for matches in real time was limited to placeholder data. Additionally, we could not implement the full backend functionality like session confirmation or user authentication due to time constraints. However, by focusing on high-priority screens and flows, we ensured that the prototype could support a meaningful usability test.

This prototype was built to highlight Stubby's core value: smart, safe, and structured peer learning. Feedback from peers during early review sessions indicated that users appreciated the dashboard clarity and the group versus partner toggle. This confirmed that the direction we took coding the interface manually for realism was effective in communicating our design intent.

### **Team MALEKA Members:**

- Shelley Magdalan
- Cedric Leal
- Angeline Kaquilala