# Finals Project:

Partners: Eric and Naveen

[**Finals Project: 1**](#_rq5qul1svez)

[Charter Docs: 2](#_titfbz1hj46x)

[Project Statement: 2](#_sg3ofid8rhid)

[Concise Project Overview: 2](#_is1jnttys7xo)

[Scope Statement: 2](#_w76ktm9s01u6)

[Stakeholders: 3](#_5l3c3f5g0xc)

[Timeline: 3](#_9jpzso4eogei)

[Budget Estimate(s): 3](#_7zqj8lp0uchb)

[Risks and Contingency Plans: 3](#_9fd4u1axwo0)

[Success Criteria: 3](#_f430eivrjmze)

## Charter Docs:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | **Team Member** | **Strengths** | **Weaknesses** | | **Eric** | planning and organization  coding non-OOP  UML organization | OOP coding  debugging  Making complex algorithms | | **Naveen** | Project Management OOP coding (to an extent) Debugging | Multi-class processes  Algorithmic Structuring | |
| Group Communication: Discord, Imessage, Google suite, etc |
| Team Norms:  Musts:   * Everyone in the group must agree to norms * Consequence: I shall slap u (jk but the teacher will be informed)   Norms:   * Do not be lazy * Always communicate, especially for errors * Respect everyone's ideas and always try to work together |
| We, Eric and Naveen, as group members of the Software Development Project ScholasticTinder agree to follow the above team norms along with class norms to stay committed to completing the tasks on time and as required |

## Project Statement:

Idea brainstorm:

* Group Project Matcher
  + Application will have a login page where users can enter their student roster and add students to create a list, then sort them into groups in however many groups or indvidiuals per group they like

## Concise Project Overview:

Efficiently and effectively matching students and projects to streamline the collaborative working process. This process will take place over the course of around three weeks, with a fiscal budget of $0. We will be using the IntelliJ IDE and VS Code alongside desktop computers provided by our organization and personal computers provided at our own expense. The major stakeholders involve high school students who are more academically inclined, so our product must be tailored to meet their needs. Our main constraints are developer inexperience and time, as with a short deadline of June 2nd, we will need to curtail some of our ambitions to effectively complete our work.

## Scope Statement:

The scope of this product includes a homepage page, a database to hold user information, and a matchmaker system in which users are matched to projects and other individuals randomly. The matchmaking algorithm will be relatively simplistic accounting for both our time and inexperience constraints, and we will keep our user base relatively minimal to avoid bandwidth problems.

## Stakeholders:

Client: Students at Skyline High School

Team Members: Eric Ma, Naveen Challa

Roles: Developers

Communication Needs: iMessage, Discord (as applicable)

Satisfaction Requirements: A functional matchmaking application

## Timeline:

(using May 25th as general deadline to incorporate aspects of idea management and discussion until then)

May 25th: Charter Document

May 25th: Idea solidified

May 25th: Flow chart finished

May 29th : backend code finished

May 31th: frontend code finished

June 2nd: all code finished and consolidated

June 4th: testing finished

## Budget Estimate(s):

* Computers in class
* Github
* Requisite materials
* Time (project due June 2nd)
* Interview with students and teachers to find what works best for the entire project

## Risks and Contingency Plans:

There are several risks and areas for which we need to have contingencies in the event of something unfortunate happening. For one, the tight timeline of this project makes it a possibility that we won’t be able to complete all of our goals to their entirety. Another risk is an assumed one as part of the collaborative working process - with data being shared constantly, data loss is a constant risk that must be accounted for at all times. A way we can mitigate this is by utilizing and learning the various mechanisms within Github, specifically the version history and backup components, to ensure our project is up-to-date and secure at all times.

## Success Criteria:

* Successfully be able to add members from the application
* Successfully group members through a complex sorting algorithm
* Be able to sort and add members continuously
* Implement button mechanisms to transport user from page to page; Be able to navigate through a javaFX website like application
* Organize in a simplistic user interface

Testing Results:

Round 1 Testing: June 4th:

* Navigation: negative
* Adding Members: positive
* Sorting Members: negative
* Button Appearance: positive
* Simplistic Design: negative
* Readable Code: positive
* Continuous Use of Application: negative

Round 2 Testing: June 4th

* Navigation: positive
* Adding Members: positive
* Sorting Members: positive
* Button Appearance: positive
* Simplistic Design: positive
* Readable Code: positive
* Continuous Use of Application: positive