Development Plan
Flow

Team 9, min-cut
Ethan Patterson
Hussain Muhammed
Jeffrey Doan
Kevin Zhu
Chengze Zhao

Table 1: Revision History

Date	**Developer(s)**	**Change**
Date1	Name(s)	Description of changes
Date2	Name(s)	Description of changes
...	...	...

[Put your introductory blurb here. Often the blurb is a brief roadmap of what is contained in the report. —SS]

[Additional information on the development plan can be found in the lecture slides. —SS]

1 Confidential Information?

The project does not use any confidential information.

[State whether your project has confidential information from industry, or not. If there is confidential information, point to the agreement you have in place. —SS]

[For most teams this section will just state that there is no confidential information to protect. —SS]

2 IP to Protect

There is no IP to protect in this project.

[State whether there is IP to protect. If there is, point to the agreement. All students who are working on a project that requires an IP agreement are also required to sign the "Intellectual Property Guide Acknowledgement." —SS]

3 Copyright License

The group has agreed that the project should use a MIT License.

[What copyright license is your team adopting. Point to the license in your repo. —SS]

4 Team Meeting Plan

[How often will you meet? where? —SS]

The team will meet 1-3 times per week with one in person meeting reoccurring at the same time every week. The meeting will be at a previously agreed upon location that will be discussed privately. Alternatively the meeting may be done virtually. We will meet with industry advisers and other experts on a scheduled basis. These meetings will be announced beforehand.

[If the meeting is a physical location (not virtual), out of an abundance of caution for safety reasons you shouldn't put the location online —SS]

[How often will you meet with your industry advisor? when? where? —SS]

[Will meetings be virtual? At least some meetings should likely be in-person. —SS]

[How will the meetings be structured? There should be a chair for all meetings. There should be an agenda for all meetings. —SS]

5 Team Communication Plan

[Issues on GitHub should be part of your communication plan. —SS]

The team will use a private discord server along with GitHub for communication. Team members are expected to check both of these daily for updates to meeting times or other milestones.

6 Team Member Roles

We will need a dedicated Team Lead, Keynoter, Team Liaison, Reviewer, Meeting chair. The team lead is in charge of scheduling meetings and other events that may be needed to complete the project. They Keynoter is in charge of taking notes on any meetings for later use. The Team Liaison is in charge of communicating with the Professor and other external helpers for the project. The reviewer is in charge of reviewing work to ensure it is of high standard. The meeting chair is in charge of managing meeting topics and making sure that the meeting stays productive and relevant.

[You should identify the types of roles you anticipate, like notetaker, leader, meeting chair, reviewer. Assigning specific people to those roles is not necessary at this stage. In a student team the role of the individuals will likely change throughout the year. —SS]

7 Workflow Plan

The project will be using GitHub for version control and issue tracking. It will be setup with a main branch that is always stable and a dev branch shall be used for integration of new features. Each feature or fix will first be developed in a separate branch off of dev. Once the feature is complete and tested, a pull request shall be created to merge the feature branch into dev. The pull request would be reviewed by at least one other team member before being merged. For smaller changes, the dev branch may just be used directly. The dev branch will be periodically merged into main after thorough testing to ensure stability.

The GitHub Issues will be used to track the various tasks and features that need to be completed, as well as any meetings that are held. Templates are to be used for each issue to ensure relevant information is included. Issues will also be labeled based on their type, such as bug, feature, meeting, etc, where applicable. This will help in organizing and prioritizing the work that needs to be done. Assigning of issues will be done where necessary to track who is responsible for what.

GitHub Actions will be used for CI/CD mainly for linting and running tests, as well as updating the pdfs from the tex files. Whenever a pull request is created, the CI/CD pipeline will automatically run the actions which would include linter and any tests that have been written. This will help catch any problems early, making sure any code merged is the correct format. Testing will most likely be setup later on as the project progresses and more code is written.

8 Project Decomposition and Scheduling

We will use GitHub projects in the Kanban board format to organize our issue items. To begin, the columns will be ”To Do”, ”In Progress”, and ”Done”. As we work the the project, more columns may be added to better suit our workflow. Each issue will be added to the ”To Do” column, and when a team member starts working on it, it will be moved to ”In Progress”. Once the issue is completed and reviewed, it will be moved to ”Done”. This will help us keep track of what needs to be done, what is currently being worked on, and what has been completed. The project board can be found at https://github.com/users/ritual-17/projects/1/views/2.

9 Proof of Concept Demonstration Plan

What is the main risk, or risks, for the success of your project? What will you demonstrate during your proof of concept demonstration to convince yourself that you will be able to overcome this risk?

The main risks for the success of the project have to do with factors adding to the time it takes to take a note down, making it longer than taking notes by other means. As the project involves making and rendering geometry in a GUI, there is a risk that the performance of the application may be too slow to be practical. Another risk is that users may also find the syntax of the application too cumbersome or unintuitive to use effectively. To mitigate these risks, the proof of concept demonstration will focus on showing that notes can be taken down quickly and easily using the application. It will involve having someone technically proficient testing writing different notes using our application and comparing the time it takes to write the same notes using handwriting. A successful demonstration will show that notes can be taken down in faster or comparable time to handwriting, while still being legible and well formatted.

10 Expected Technology

The programming language we expect to use is Electron with TypeScript. These languages are chosen due to their ability to create cross-platform desktop applications with graphical user interfaces, which is essential for our note-taking application. Some libraries we may use include React for building the user interface components, and others for rendering shapes. GitHub and GitHub Projects will be used for version control and project management, respectively. Testing and linting are important aspects of our development process. Jest and ESLint are potential choices for unit testing and linting, respectively, but these decisions will be finalized as the project progresses. As previously mentioned, GitHub Actions will be used for CI/CD to automate testing and linting processes, as well as updating documentation.

11 Coding Standard

The Standard JavaScript Style Guide will be used as the coding standard for this project. This guide provides a simplistic and easy to follow set of rules for writing clean and consistent code. It covers various aspects of coding style, including naming, variables, functions and more. The standard style can also be integrated with ESLint, which may be used for linting the code to ensure it adheres to the coding standard.

Appendix — Reflection

[Not required for CAS 741 —SS]

The purpose of reflection questions is to give you a chance to assess your own learning and that of your group as a whole, and to find ways to improve in the future. Reflection is an important part of the learning process. Reflection is also an essential component of a successful software development process.

Reflections are most interesting and useful when they're honest, even if the stories they tell are imperfect. You will be marked based on your depth of thought and analysis, and not based on the content of the reflections themselves. Thus, for full marks we encourage you to answer openly and honestly and to avoid simply writing "what you think the evaluator wants to hear."

Please answer the following questions. Some questions can be answered on the team level, but where appropriate, each team member should write their own response:

1. Why is it important to create a development plan prior to starting the project?
2. In your opinion, what are the advantages and disadvantages of using CI/CD?
3. What disagreements did your group have in this deliverable, if any, and how did you resolve them?

Appendix — Team Charter

[borrows from University of Portland Team Charter —SS]

External Goals

[What are your team's external goals for this project? These are not the goals related to the functionality or quality fo the project. These are the goals on what the team wishes to achieve with the project. Potential goals are to win a prize at the Capstone EXPO, or to have something to talk about in interviews, or to get an A+, etc. —SS]

Attendance

Expectations

[What are your team's expectations regarding meeting attendance (being on time, leaving early, missing meetings, etc.)? —SS] Show up to all meetings unless they have an acceptable excuse or an emergency.

Acceptable Excuse

[What constitutes an acceptable excuse for missing a meeting or a deadline? What types of excuses will not be considered acceptable? —SS] For in person meetings the person must be unable to reach the meeting position and have the excuses vetted by 3/4 of the other members. The excuses should be communicated as soon as possible.

In Case of Emergency

[What process will team members follow if they have an emergency and cannot attend a team meeting or complete their individual work promised for a team deliverable? —SS]

If a team member has a emergency that impairs their ability to do work they should communicate the other team mates as soon as possible to work out an alternative plan.

Accountability and Teamwork

Quality

[What are your team's expectations regarding the quality of team members' preparation for team meetings and the quality of the deliverables that members bring to the team? —SS] Team mates should have proper tools for meetings to discuss and work on the project. Code will be reviewed by other team-mates and code of poor quality will be rewritten until it is of proper quality.

Attitude

[What are your team's expectations regarding team members' ideas, interactions with the team, cooperation, attitudes, and anything else regarding team member contributions? Do you want to introduce a code of conduct? Do you want a conflict resolution plan? Can adopt existing codes of conduct. —SS]

Stay on Track

[What methods will be used to keep the team on track? How will your team ensure that members contribute as expected to the team and that the team performs as expected? How will your team reward members who do well and manage members whose performance is below expectations? What are the consequences for someone not contributing their fair share? —SS]

[You may wish to use the project management metrics collected for the TA and instructor for this. —SS]

[You can set target metrics for attendance, commits, etc. What are the consequences if someone doesn't hit their targets? Do they need to bring the coffee to the next team meeting? Does the team need to make an appointment with their TA, or the instructor? Are there incentives for reaching targets early? —SS]

Team Building

[How will you build team cohesion (fun time, group rituals, etc.)? —SS]

Decision Making

[How will you make decisions in your group? Consensus? Vote? How will you handle disagreements? —SS]