Development Plan
Software Engineering

Team 24, Cowgnition
Sylvia Kamel
Rosa Chen
Karim Elbasiouni
Claire Nielsen
Safwan Khan

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?

[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

[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

[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]

[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

Our team's means of communication will primarily comprise of a Microsoft Teams group chat, an iMessage group chat, and GitHub issues. Our Microsoft Teams group chat is suited for discussions about the project. Some examples of discussions are: someone asking for help on how to resolve an error and someone confirming with team members before taking action on an important task. Our iMessage group chat is useful for situations such as: confirming an in-person

meeting's location and someone sending a text to indicate that they will be arriving a couple of minutes late to the in-person meeting. Our iMessage group chat has also allowed us to exchange phone numbers with each other. Having each others' phone numbers is useful for one to one communication situations. Some examples of these situations are: calling someone to ask if their assigned task is done and if someone has an assigned task that depends on another team member's work, it might text or call the team member to get clarity and understand the team member's work. Our team will not be using email to communicate with each other. Email will only be used to communicate with our project supervisor, other project stakeholders, the course instructor, and our assigned TA. That being said, we still have each others' emails. GitHub issues will primarily be used for putting meeting agendas for meetings, summarizing a meeting's key points, and indicating the people that attended the meeting. Moreover, since lectures are also considered as meetings, GitHub issues will be created for lectures. For a lecture, the issue would summarize the lecture's key points and indicate the people that attended the lecture.

6 Team Member Roles

[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

- How will you be using git, including branches, pull request, etc.?
- How will you be managing issues, including template issues, issue classification, etc.?
- Use of CI/CD

8 Project Decomposition and Scheduling

- How will you be using GitHub projects?
- Include a link to your GitHub project

[How will the project be scheduled? This is the big picture schedule, not details. You will need to reproduce information that is in the course outline for deadlines. —SS]

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?

10 Expected Technology

[What programming language or languages do you expect to use? What external libraries? What frameworks? What technologies. Are there major components of the implementation that you expect you will implement, despite the existence of libraries that provide the required functionality. For projects with machine learning, will you use pre-trained models, or be training your own model? —SS]

[The implementation decisions can, and likely will, change over the course of the project. The initial documentation should be written in an abstract way; it should be agnostic of the implementation choices, unless the implementation choices are project constraints. However, recording our initial thoughts on implementation helps understand the challenge level and feasibility of a project. It may also help with early identification of areas where project members will need to augment their training. —SS]

Topics to discuss include the following:

- Specific programming language
- Specific libraries
- Pre-trained models
- Specific linter tool (if appropriate)
- Specific unit testing framework
- Investigation of code coverage measuring tools
- Specific plans for Continuous Integration (CI), or an explanation that CI is not being done
- Specific performance measuring tools (like Valgrind), if appropriate
- Tools you will likely be using?

[git, GitHub and GitHub projects should be part of your technology. —SS]

11 Coding Standard

[What coding standard will you adopt? —SS]

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?

   Safwan

   The development plan is a great way to establish our team environment. It helps team members gain a detailed overview of how the team is going to communicate with each other and operate and the expected technologies. Investing time towards this will help us better tackle the problem and build a great solution.

2. In your opinion, what are the advantages and disadvantages of using CI/CD?

   Safwan

   CI/CD enables consistent integration testing and figuring out if our code works on multiple operating systems. Consistent integration testing helps us as a team better understand if our individual contributions still ensure that subsystems and thus the system as a whole are functioning as expected. If our system has bugs, the testing on different operating systems can help us find out if the bug is due to a problem in the logic or if it is an operating system specific issue.

   In terms of disadvantages, the main one is that upfront time needs to be spent on both setting up CI/CD for our project and getting comfortable with using it.

3. What disagreements did your group have in this deliverable, if any, and how did you resolve them?

   Safwan

   In this deliverable, one disagreement that we had was setting up a concrete day and time for weekly meetings. At first, as a team, none of us could meet weekly at a given day and time. To resolve this, we discussed

this problem as a group over a call. Discussing it over a call helped us better understand each others' circumstances. Over time, we became more flexible over availabilities. In the end, we ended up finding a day and time for our weekly meetings.

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]

Acceptable Excuse

[What constitutes an acceptable excuse for missing a meeting or a deadline? What types of excuses will not be considered acceptable? —SS]

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]

Accountability and Teamwork

Quality

For the quality of preparation for team meetings, our team's expectation is that team members are aware of what is to be discussed and done in the meeting and work on what is necessary to effectively contribute in the meeting. For example, if a meeting involves discussion about a milestone document, a reasonable expectation is that team members go through the document so that they have the necessary context to effectively contribute in the meeting. In the meeting, if too much time is spent on gathering context, it makes it difficult to achieve the meeting's goals. Another example is a meeting that involves providing individual updates. In this type of meeting, a reasonable expectation is that team members should be able to clearly give updates about their assigned tasks and explain if they are stuck on something. For the quality of team members' deliverables, team members should fullfill the purpose and requirements of the deliverables and have clear reasoning behind their approaches towards their deliverables.

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]