Development Plan
ProgName

Team #, Team Name
Student 1 name
Student 2 name
Student 3 name
Student 4 name

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

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

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?

   Stanley: It is important to create a development plan because a development plan reduces chaos and helps the team organize themselves better. It allows for smoother integration of parts that have been planned, and the team also has a direction and set of expectations thta they can rely on.

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

   Stanley: An advantage of CI/CD would be that bugs are caught faster because of the automated testing. Because of this, code quality is generally better. Pipelines are also a lot more reliable than manual testing. A disadvantage of CI/CD would be cost. Setting it up takes time, and it running tests every commit could be costly.

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

   Stanley: During this deliverable, there were not any major disagreements, as we were able to allocate work smoothly since there was not too much work to do yet.

Appendix — Team Charter

[borrows from University of Portland Team Charter —SS]

External Goals

Our goal for this course is to achieve a 12 on the McMaster GPA scale. We also hope to create a project that would be impressive on our resumes and be something that could be talked about during our interviews for new grad jobs.

Attendance

Expectations

Our team plans to meet once a week for roughly two hours every Thursday at 4:30 during our tutorial time. We also scheduled a weekly meeting every Saturday for an hour so that we could discuss all the work that we've been working on individually and to finalize certain changes. In terms of punctuality, we've given a grace period of 10 minutes for being late in terms of being late, since there may be technical difficulties when joining or being pardoned for a bathroom break.

Acceptable Excuse

Acceptable excuses would include any medical, personal, or family related issues that arise. It would be like a Self-Report MSAF. That or, if a team member lets our group know in advance, they can opt out of a meeting for any reasonable issues.

In Case of Emergency

If a team member cannot meet up, they would be informed by one of the other team members about what had been discussed in the meeting and informed about what work they had been assigned. They would still need to complete the assigned work by the deadline assigned. If they cannot do that because of emergency reasons, they will be assigned more work in latter stages of the project to make up for the missed effort.

Accountability and Teamwork

Quality

Team members are expected to come prepared with all materials and expectations set from the previous meetings. The quality of the work should at least satisfy all the requirements set in the rubric, as well as any expectations set by the team in previous meetings.

Attitude

Team members are expected to treat each other with respect and handle conflict maturely. All members will be respected and have a voice, and all opinions will be considered. They should be able to work co-operatively when assigned tasks together, as well as be punctual and respect each other's time.

Stay on Track

Team members will be assigned tasks so that they can contribute around an equal amount. To ensure quality work, we will all go over the individual work that we contributed together as a group so that any poorly done will not be overlooked. There will not be a reward and punishment system for the quality of work. Members are expected to perform, and if they cannot, they can rely on other group members for assistance. As long as all of the work assigned has been completed, then there will be no issues. If a group member is not co-operating, it may be addressed to the group first, and if need be, could be bought up to administrative figures such as the TA's.

Team Building

Most of the members worked together at Citi as interns during the summer, so they have experience working together. As per team building, we have already hiked together and plan on meeting up more during the school year to play sports such as badminton.

Decision Making

How our group handles decision making is through discussion during the meetings and voting. When we were choosing our project idea, we decided to use a decision matrix to see the most desired project idea.