

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

For the remainder of this project, we will work collaboratively as a team following assumed roles to develop the Cow-puter Vision project. These defined roles may have different assignees at different stages of the project, as team members may wish to rotate between the roles. The table below contains a list of all the roles and respective descriptions that are believed to be essential to ensure successful collaboration and achievement of the project goals:

6.1 Non-technical Roles

Table 2: Non-technical roles and descriptions

Role	Description
Team Liaison	Acts as the main point of contact between the team and supervisors and stakeholders.
Notetaker	Records meeting minutes, lecture notes, and sets agenda.
Meeting Chair	Facilitates team discussions, ensuring that the team remains focused on the meeting agenda without going off on tangents.
Backlog Manager	Creates, tracks, and assigns project-related GitHub issues based on deliverable sections and sets their priority based on deadlines and business value.

Important Note: Backlog Manager is not solely responsible for creating GitHub issues, and is only responsible for creating issues based on the deliverable sections. Example: Backlog Manager is responsible for creating issues such as "Complete section 1.1" and "Review section 1.1". Issues involving very specific tasks, that are not clearly outlined or sectioned on deliverable templates, such as fixing specific bugs, fall under the responsibility of the team member(s) who encounters the issue or task to begin with.

6.2 Technical Roles

Table 3: Technical roles and descriptions

Role	Description
Computer Vision & ML Engineer	Responsible for leading design and development of the ML model, including model training and evaluation of the computer vision models using the pre-processed cow data, as well as ensuring that the model achieves satisfactory performance.
Data Pipeline Engineer	Responsible for leading the collection, organization and preprocessing of the raw image/video data, ensuring it is properly labeled and cleaned, and preparing it into standardized datasets ready for model training and evaluation.
Backend & Integration Engineer	Responsible for leading the development of APIs or services that connect the trained model to the frontend, and manages a database to store any necessary data, ensuring smooth communication between components and managing data flow through the system
Frontend & UI/UX Engineer	Responsible for leading the design and development of interface for farmers to interact with, and ensure that the necessary results are clearly displayed to them.
Deployment & Testing Lead	Responsible for ensuring that the model deploys to the target environment, testing and validating the accuracy and reliability of the system, running end-to-end tests, and fixing integration issues

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?
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 of 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

[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]

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

Meeting Deadlines: The team will employ the use of the official capstone calendar to keep track of dates and deadlines. Each team member is responsible for ensuring that they are aware of upcoming deadlines and potential postponement of these deadlines. Regardless, the team will collectively make an effort to communicate any reminders regarding dates, deadlines, and any announcements made by the professor through the team's private Microsoft Team's channel.

Tracking Progress: GitHub issues will be created to split milestone tasks to ensure that the volume of work is evenly distributed. Brief progress check-ins will be done at the beginning of every weekly meeting, through which every team member is required to share their current progress on their assigned tasks, including any concerns or difficulties that they may have encountered. The team reserves the right to adjust division of responsibilities if a task appears to be more difficult to complete than expected, or to accommodate for another team member's conflicting schedules, such as in the case when a team member may have external commitments or circumstances that other team members do not share. Examples include conflicting midterm schedules, important medical appointments, etc.

Rewards for High Performance: Team members who perform beyond their expectations will be celebrated in team meetings and given the recognition they rightfully deserve, as well as assigned leadership roles in future milestones that better reflect their commitment to the team. Following the conclusion of the project, the team may pitch in and buy the high performing individual any candy of their choice. Moreover, the team will make a collective request to the professor that the performing individual is given a bonus grade, though this may or may not be accepted by the professor.

Dealing with Unmet Expectations: If no extenuating circumstances are present, and a team member is not meeting their workload commitments for longer than two weeks or up to the closest milestone, then the following procedure will be followed to address unmet expectations:

1. Unmet expectations will be addressed briefly as the first item on the agenda of the closest upcoming meeting. If the issue persists, or the team member does not attend the meeting proceed to step 2.
2. The team will have an in-person meeting hosted before, during (if lecture slot is empty), or after lecture to discuss with the team member in question. If the issue persists, or the team member does not attend the meeting proceed to step 3.
3. The team liaison will contact the TA to seek advice on handling the situation. If after applying the TA's advice the issue still persists, proceed to step 4.

4. The team liaison will contact the professor to address the situation.

Consequences for Under-performing: The consequences for not meeting the expected workload commitment include:

- Negative peer evaluation
- Loss of grades
- Academic dishonesty punishments

Incentives for Meeting Deadlines Early: Team members who meet deadlines early will be given the option to have a lighter workload for the next deliverable or milestone. This does not apply for the winter semester, as the team expects to have a tougher workload around that time period, due to final implementation deadlines.

Target Metrics: Each team member is responsible to achieve the following metrics to show their contribution:

- Attend 100% of all meetings (except if there is an acceptable excuse)
- Confirm review of all documents prior to submission, ensuring adherence to rubric
- Contribute to roughly 20% of the total volume of work, tracked through weekly number of closed issues.

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]