Development Plan
SFWRENG 4G06

Team 28, Cowvolution Minds
Aryan Patel
Harshpreet Chinjer
Krish Patel
Martin Ivanov
Shazim Rahman

Table 1: Revision History

Date	Developer(s)	Change
September 22 2024	Martin Ivanov	Added team member roles and POC demonstration plan
September 22 2024	Harshpreet Chinjer	Added meeting plan, Expected technology and Coding standard
September 24 2024	Martin Ivanov	Added Accountability and Teamwork goals to the Appendix
...	...	...

[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

1 Frequency of Meetings

- **Team Meetings**: The team will meet virtually once a week at discussed time, with in-person meetings if required
- **Industry Supervisor Meetings**: Meetings with the industry supervisor will occur virtually once a week, with additional meetings or cancellations based on the supervisor's availability and project progress.
- **Additional Meetings**: The team or supervisor may call extra meetings as needed, especially during critical project milestones.

2 Meeting Location

- **Virtual Meetings**: Conducted via platforms Discord or Microsoft Teams.
- **In-Person Meetings**: Arranged at a mutually agreed location like Mills Memorial Library, H.G. Thode Library and ITB Hallway Space

3 Meeting Structure

- **Chair Rotation**: Each team member will take turns chairing the meetings. This includes leading discussions, ensuring the agenda is followed, and managing time.
- **Agenda**: The chair of the upcoming meeting is responsible for preparing the agenda and circulating it to all members at least **24 hours prior** to the meeting. The agenda will include:
  - Project updates
  - Task allocation
  - Deadlines review
  - Challenges and risks
  - Next steps
  - Supervisor queries (for meetings with the industry advisor)

4 Meeting Protocol

- **Attendance**: All members are expected to attend unless unable, in which case they should inform the members ahead of time.
- **Minutes**: One team member will be assigned to take minutes, including key decisions, action items, and next steps. Minutes will be shared within **24 hours** after the meeting on GitHub.

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]

Liaison

Responsible for communicating with the industry supervisor, ensuring that the project is on track, and addressing any concerns or questions from the supervisor.

Chair

Responsible for leading meetings, ensuring the agenda is followed, and managing time. Team members will take turns being the chair for each meeting.

Meeting Minutes

Responsible for taking minutes during meetings, including key decisions, action items, and next steps. Meeting minutes will be taken by a single member at a time, and the member taking minutes will rotate every week.

Machine Learning Lead

Responsible for leading the machine learning component of the project, including researching machine learning models, and ensuring that the machine learning component meets the requirements.

Developer

Responsible for developing the project, including coding, testing, and documentation. Developers must ensure that the code produced for the project is of high quality and is well-documented.

Tester

Responsible for testing the project, including unit testing, integration testing, and system testing. This role will be responsible for ensuring that the project meets the requirements and is free of defects.

Path to Changing Roles

Team members can change roles based on their interest, availability, and skill set. The team will discuss and decide on role changes as needed. Furthermore, any team member may be asked to help out anywhere else in the system if the need arises. For this reason, it is important that all team members stay up to date with context for what is happening with different parts of the team. Team members may fulfill more than one role at once.

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

9   Proof of Concept Demonstration Plan

A few potential challenges that may arise during the course of this project include the availability and quality of historical dairy farm data. While we will be provided with some datasets related to animal health, breeding success, and productivity, these may suffer from inconsistencies, missing values, or lack of relevant features. Our strategy to overcome this involves performing thorough data preprocessing steps to ensure the dataset is suitable for building a reliable model. Additionally, the complexity of predicting outcomes like breeding success or the likelihood of an animal leaving the herd poses a challenge in terms of model accuracy. To address this, we plan to conduct some experiments using various machine learning algorithms to identify the most accurate and robust model for these predictions.

For our POC demonstration in November, we will build a prototype model using a limited dataset that predicts breeding success rates. The POC will include a data pipeline that handles preprocessing steps, a prediction model that outputs breeding success probabilities, and means for displaying the predictions via a dashboard or command line output. This will allow us to demonstrate the viability of the system and our ability to overcome challenges related to data quality and prediction accuracy. In the final product, the model will be scaled to handle more complex data and generate additional insights such as health risks or productivity forecasts, but if the POC can show accurate predictions for a small subset of outcomes, it serves as proof that the system will work for more complex scenarios.

10   Expected Technology

Initial Implementation Plan

- **Programming language**: The project will primarily use **Python** due to its strong support for machine learning and data science tasks. For web development tasks, the project will use **React**, a **JavaScript** library.

- **Libraries**: Expected key libraries include **Pandas** and **Numpy** for data manipulation, **Scikit-learn**, **TensorFlow**, or **PyTorch** for building the machine learning model, and **Matplotlib** or **Seaborn** for data visualization.

- **AI Model**: The project will create a **custom AI model** specifically tailored to dairy farming data, focusing on predicting outcomes such as breeding success rates and herd attrition.

- **Linter**: **Flake8** or **Black** will be utilized to maintain code quality and ensure adherence to Python's PEP 8 standards.

- **Unit testing framework**: **PyTest** will be used to implement unit tests, focusing on validating data pipelines, model training, and prediction accuracy.

- **Continuous Integration (CI)**: **GitHub Actions** will be used to automate tests and ensure code quality in a continuous integration pipeline.

- **Version Control and Project Management**: **Git** will be used for version control and **GitHub** for repository management and collaboration throughout the project.

11   Coding Standard

The project will follow the **PEP 8** coding standard for Python, which is the official style guide for Python code. PEP 8 outlines guidelines and best practices for writing clean, readable, and maintainable Python code. Key recommendations include:

- Using 4 spaces per indentation level.

- Limiting line length to 79 characters.

- Proper use of blank lines to separate functions and classes.

- Consistent use of lower_case_with_underscores for variable and function names.

- Avoiding extraneous whitespace.

- Using meaningful comments to explain the purpose of code blocks.

- Keeping code simple and readable, and adhering to Python's philosophy of clarity.

For a full reference to PEP 8 guidelines, visit the official document: PEP 8 Style Guide.

Appendix — Reflection

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]

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

Our team expects all members to come to meetings well-prepared, having reviewed any relevant materials and completed their assigned tasks. Deliverables should be of high quality, adhering to the agreed-upon standards and guidelines. If a member is unable to meet these expectations, they should communicate this to the team in advance and seek assistance if needed. The code submitted by team members will be reviewed by other team members to ensure quality and consistency via comments on GitHub Pull Requests.

Attitude

Our team expects all members to maintain a positive and respectful attitude towards each other. This includes being open to different ideas, providing feedback, and listening during discussions.
Code of Conduct:

- Treat all team members with respect and courtesy.

- Be receptive to feedback and willing to make improvements.

- Maintain professionalism in all interactions, both within the team and with external stakeholders.

Stay on Track

To ensure that the team stays on track, we will implement the following methods:

- **Regular Check-ins**: We will use our Discord channel to discuss progress, address any issues, and adjust plans as necessary.

- **Task Tracking**: Use of GitHub Projects to track tasks and deadlines. Each task will be assigned to a team member, and progress will be monitored.

- **Performance Metrics**: Metrics such as the number of commits, issues resolved, and pull requests reviewed will be tracked to ensure active participation.

- **Peer Reviews**: Code and deliverables will be peer-reviewed to maintain quality and provide constructive feedback.

- **Consequences**: Members who do not contribute their fair share will be given a warning. Continued lack of contribution will result in a meeting with the TA or instructor to discuss further actions.

Team Building

To build team cohesion, we plan to communicate primarily via Discord for regular updates, discussions, and casual interactions. Additionally, we will occasionally work together on campus, which will provide opportunities for face-to-face collaboration. These in-person sessions will help strengthen our teamwork and ensure that we stay aligned with our project goals.

Decision Making

We plan to use consensus for decision-making. However, if there is a major disagreement, we will resolve it by voting. Each team member will have an equal vote, and the majority decision will be implemented. In case of a tie, the team will discuss further to reach a consensus or consult the industry supervisor for guidance.