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 23 2024	Martin Ivanov	Added description for the Copyright License section
September 24 2024	Martin Ivanov	Added Accountability and Teamwork goals to the Appendix
September 24 2024	Harshpreet Chinjer	Added External goals and Attendance expectations
September 24 2024	Shazim Rahman	Added IP to protect, team communication plan, and workflow plan
...	...	...

1 Confidential Information?

2 IP to Protect

This project falls entirely under the intellectual property of CATTLEytics. The specific IP agreement has not yet been shared with the team. Once the agreement is provided, this document will be updated to reflect its terms and conditions accordingly.

3 Copyright License

Having discussed with the industry supervisor, the project will be licensed under a proprietary license to protect the intellectual property and ensure that the project remains confidential. The proprietary license will allow the industry partner to use the software for their internal purposes which will allow them to have a competitive edge over the competing firms in the cattle management software industry. The proprietary license will also prevent the software from being open-sourced or made publicly available. Currently, the team is awaiting the license agreement from the industry supervisor, which will be included in the project repository once received. This document will be updated to reflect that the license was received.

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

GitHub Issues

GitHub Issues will serve as the primary platform for task management and project tracking. Each task, whether it pertains to a bug fix, feature implementation, or a request for modification, will be logged as an issue. The team will utilize labels, milestones, and assignees to ensure tasks are categorized and prioritized effectively. This transparent system allows for real-time collaboration, ensuring that all members are informed of current developments, changes, and future features, promoting traceability and accountability across the project lifecycle.

Online Communication

For internal team communication, Discord will be the primary medium, fostering collaboration and quick exchanges. Channels will be structured to organize discussions by topics such as development progress, code reviews, and troubleshooting. Discord's voice and text capabilities will facilitate real-time problem-solving, and informal check-ins. This method ensures agility in communication, allowing the team to address and resolve issues promptly as they arise during the development process.

Formal communications with the project supervisor or any external stakeholders will be conducted via email. This ensures a professional tone and provides a record of all significant decisions and updates shared externally. Emails will serve as the official means of sharing reports, progress updates, and seeking feedback or approvals, thus maintaining a clear, documented line of communication with parties outside the core development team.

Weekly Meetings

The team will hold weekly meetings to discuss the current progress of the project and review any outstanding tasks. These meetings will focus on assessing the status of ongoing work, resolving any blockers, and assigning new tasks based on priorities. If necessary, older tasks will be migrated or reassigned to reflect changes in project scope or focus. These structured meetings will ensure that the team remains aligned on project goals and timelines.

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

Our project will follow a structured Git workflow using two primary persistent branches: *main* and *develop*. The *main* branch will serve as the target for deployment, ensuring that only validated, production-ready code is merged. The *develop* branch will act as the integration branch where all new changes are implemented and tested before they are merged into *main*.

Branching Strategy

For any new changes, developers will create a new branch from the *develop* branch. This new branch must be appropriately named to reflect the feature or bug being addressed. All development work and commits will be performed

within this branch. When the feature or fix is complete, the developer will push their branch to the remote repository and open a Pull Request (PR) to merge it back into *develop*.

Pull Request Guidelines

The PR must be linked to its corresponding GitHub Issue, which will have been labeled accordingly (e.g., bug, enhancement, documentation). Each PR will require at least one team member's approval before it is eligible to be merged. Additionally, the following checks must pass:

- All unit and integration tests are successful.

- Linter checks for code quality are successful.

- If changes involve *.tex files, the corresponding *.pdf files must also be updated.

Once these conditions are met and the PR is approved, the developer can proceed with merging the branch into *develop*. After a successful merge, the branch will be automatically deleted, and the linked issue must be closed.

Merging into Main and CI/CD Integration

Periodically, the develop branch will be merged into *main*, but only after the code passes comprehensive validation, including integration and system-level tests. GitHub Actions will handle the Continuous Integration/Continuous Deployment (CI/CD) process, ensuring that any validated changes are automatically deployed to the production environment. No changes will be merged directly into *main* outside this process, thus preventing untested code or bugs from reaching production.

GitHub Issue Labels

To organize and maintain all open and closed issues, each issue must implement one of the provided templates and must be labelled using at least one of the following:

1. **bug:** Used for tracking issues related to errors or malfunctions in the code.

2. **documentation:** For issues related to writing or updating project documentation.

3. **enhancement:** Indicates requests for new features or improvements to existing ones.

4. **duplicate:** Marks issues that have already been reported and tracked elsewhere.

5. **invalid:** Denotes issues that are not applicable or are irrelevant.

6. **wontfix:** Used for issues that the team has decided not to address.

7. **help wanted:** Signifies a need for additional assistance or contributions.

8. **question:** For inquiries or clarification requests related to the project.

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

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

External Goals

- **Capstone EXPO Success**: The team aims to present the project at the Capstone EXPO and potentially win a prize by showcasing the innovative use of machine learning in the agricultural domain.

- **Portfolio and Interview Value**: Each member of the team intends to use this project as a strong portfolio piece, demonstrating their technical skills in machine learning, data science, and web development. The project will also provide valuable content to discuss in job interviews, highlighting problem-solving capabilities and teamwork.

- **High Academic Achievement**: The team is working toward achieving an A+ grade on this project, ensuring that all components meet the highest academic standards.

- **Networking and Industry Exposure**: Through regular meetings with the industry supervisor, the team hopes to build professional connections that could lead to future opportunities or collaborations within the industry.

- **Learning and Skill Enhancement**: Each member of the team is committed to learning new technologies and improving their skill set, particularly in machine learning, AI, and frontend development, as part of the project's process.

Attendance

Expectations

- Team members are expected to attend all scheduled meetings, whether virtual or in-person, and be on time.

- Leaving early or missing a meeting without prior notice should be avoided unless absolutely necessary.

- Consistent attendance and active participation are critical to the success of the project, and all members are responsible for keeping up with the team's progress.

Acceptable/Unacceptable Excuse

- **Acceptable Excuses**: Valid reasons include illness, family emergencies, or unavoidable academic or professional obligations (e.g., exams, presentations). In such cases, the member must notify the team as soon as possible.

- **Unacceptable Excuses**: Personal convenience, poor time management, or non-urgent conflicts will not be considered acceptable reasons for missing a meeting or failing to meet a deadline.

In Case of Emergency

- The team member must immediately inform the rest of the team through Discord or Microsoft Teams.

- The team will determine how to redistribute the missing member's responsibilities for that week or project milestone to ensure no delays in the overall timeline.

- The team will aim to support the member in catching up as soon as possible once the emergency is resolved.

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.