Software Requirements Specification for SFWRENG 4G06: subtitle describing software

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

October 7, 2024

Contents

Revision History

Date	Version	Notes
October 6, 2024	1.0	Initial SRS document

1 Purpose of the Project

1.1 User Business

Insert your content here.

1.2 Goals of the Project

Insert your content here.

2 Stakeholders

2.1 Client

Insert your content here.

2.2 Customer

Insert your content here.

2.3 Other Stakeholders

Insert your content here.

2.4 Hands-On Users of the Project

Insert your content here.

2.5 Personas

Insert your content here.

2.6 Priorities Assigned to Users

Insert your content here.

2.7 User Participation

Insert your content here.

2.8 Maintenance Users and Service Technicians

Insert your content here.

3 Mandated Constraints

3.1 Solution Constraints

Insert your content here.

3.2 Implementation Environment of the Current System

Insert your content here.

3.3 Partner or Collaborative Applications

Insert your content here.

3.4 Off-the-Shelf Software

Insert your content here.

3.5 Anticipated Workplace Environment

Insert your content here.

3.6 Schedule Constraints

Insert your content here.

3.7 Budget Constraints

Insert your content here.

3.8 Enterprise Constraints

Insert your content here.

4 Naming Conventions and Terminology

4.1 Glossary of All Terms, Including Acronyms, Used by Stakeholders involved in the Project

Insert your content here.

5 Relevant Facts And Assumptions

5.1 Relevant Facts

Insert your content here.

5.2 Business Rules

Insert your content here.

5.3 Assumptions

Insert your content here.

6 The Scope of the Work

6.1 The Current Situation

The current state of dairy farming presents challenges in efficiently predicting the health, productivity, and breeding outcomes of cattle. Farmers typically rely on historical records, but the analysis is done manually, often leading to reactive management. The existing systems do not utilize advanced technologies such as machine learning for predictive analytics. As a result, there is limited proactive management regarding milk production, breeding success, and herd longevity, which directly impacts farm profitability and sustainability.

The current solution environment lacks integration of large datasets from multiple sources, such as individual cow health records, breeding history, and environmental conditions, into a single system that can offer actionable predictions.

6.2 The Context of the Work

This project aims to develop a machine learning model that will leverage historical herd data to predict important traits such as milk yield, breeding success rates, and the likelihood of a cow leaving the herd. This model will be integrated into a farm management system, providing farmers with actionable insights. The goal is to move from reactive to proactive herd management.

The model will use data such as the health, breed, and genetic history of both the mother and father to predict traits in calves. The software will be developed as part of a partnership with CATTLEytics Inc., ensuring seamless integration into their existing platform used by dairy farmers.

6.3 Work Partitioning

The project will be divided into several key components:

1. **Data Collection and Preprocessing:**
    - Collection of historical data from existing systems, including cow health records, breeding data, and productivity metrics.
    - Cleaning and standardizing the data for input into the machine learning model.
2. **Model Development:**
    - Designing and implementing the machine learning model for trait prediction (e.g., milk production, herd retention).
    - Training and validating the model on historical datasets.
    - Iterative testing and refinement.
3. **Integration:**

- Integrating the prediction model into the CATTLEytics Inc. farm management system.
- Ensuring that outputs are presented in a user-friendly format for farmers to make decisions.

4. **Testing and Validation:**
    - Testing the software in real-world farm environments to validate predictions and refine the user interface.
5. **Documentation and Training:**
    - Providing clear documentation for users and training for farmers to effectively use the system.

6.4 Specifying a Business Use Case (BUC)

Title: Predicting Cow Traits for Optimized Herd Management
Primary Actor: Dairy farmer using the CATTLEytics Inc. system.
Precondition: The farmer has access to a herd management system integrated with the prediction model. Historical data on breeding, milk production, and herd turnover are available.
Trigger: The farmer initiates the model to predict the outcomes of a planned breeding or evaluates the likelihood of an existing cow leaving the herd.
Main Success Scenario:

1. The farmer selects cows for breeding and inputs the necessary data (e.g., parent traits).
2. The system processes the input and returns predictions for milk yield and herd retention likelihood.
3. Based on the model's predictions, the farmer makes informed decisions on breeding strategies or management actions to prevent herd loss.

Postconditions:

- The farmer has actionable insights to improve herd productivity and manage herd turnover proactively.

7 Business Data Model and Data Dictionary

7.1 Business Data Model

This section will be completed once the relevant data model details are available.

7.2 Data Dictionary

This section will be completed once the relevant data model details are available.

8 The Scope of the Product

8.1 Product Boundary

The product's primary function is to predict cow traits such as milk production, breeding success, and herd retention based on parental data. The machine learning model will integrate into a platform like CATTLEytics Inc., and its boundaries will include:

- **Included**: The product will use historical data to generate predictive insights on milk production, breeding success rates, and herd retention likelihood. Farmers will be able to input relevant data to receive predictions.
- **Not Included**: Real-time data collection and analysis, live health monitoring, or any complex integrations with external devices (e.g., IoT sensors).

This product will focus solely on predictive analytics based on historical data and will not handle aspects like external raw data collection or advanced herd management automation beyond providing insights.

8.2 Product Use Case Table

This table outlines the core use cases currently identified for the product.

ID	Title	Actor	Description
PUC1	Predict Breeding Success	Dairy Farmer	Farmers input breeding data to get predictions on the likelihood of successful breeding.
PUC2	Forecast Milk Production	Dairy Farmer	Farmers input cow data to get a prediction of future milk production.
PUC3	Predict Herd Retention Likelihood	Dairy Farmer	Farmers receive predictions on whether cows are likely to stay in or leave the herd.

8.3 Individual Product Use Cases (PUC's)

8.3.1 PUC1: Predict Breeding Success

- **Primary Actor**: Dairy Farmer
- **Preconditions**: Farmer has historical data about cow and parental traits available for input.
- **Trigger**: The farmer initiates a request to predict breeding success.
- **Main Success Scenario**:

1. The farmer enters relevant breeding data.
2. The system processes the input using historical records.
3. A prediction is generated on the likelihood of breeding success.

- **Postcondition**: The farmer gets an actionable prediction to decide whether to proceed with the breeding.

8.3.2 PUC2: Forecast Milk Production

- **Primary Actor**: Dairy Farmer
- **Preconditions**: Historical data for milk production and parental traits is available for input.
- **Trigger**: The farmer requests a prediction for future milk production.
- **Main Success Scenario**:
  1. The farmer inputs the cow's data.
  2. The system processes the input data.
  3. A prediction on future milk production is generated.
- **Postcondition**: The farmer receives a prediction that helps in planning milk yield expectations.

8.3.3 PUC3: Predict Herd Retention Likelihood

- **Primary Actor**: Dairy Farmer
- **Preconditions**: Health and productivity data is available for the cows in question.
- **Trigger**: The farmer requests predictions on herd retention likelihood.
- **Main Success Scenario**:
  1. The farmer selects a cow or group of cows for analysis.
  2. The system processes the available data.
  3. A prediction is generated on whether the cows are likely to stay in or leave the herd.

- **Postcondition**: The farmer receives predictions to assist in managing herd turnover.

8.3.4 PUC1: Predict Breeding Success

- **Primary Actor**: Dairy Farmer
- **Preconditions**: Farmer has historical data about cow and parental traits available for input.
- **Trigger**: The farmer initiates a request to predict breeding success.
- **Main Success Scenario**:
  1. The farmer enters relevant breeding data.
  2. The system processes the input using historical records.
  3. A prediction is generated on the likelihood of breeding success.
- **Postcondition**: The farmer gets an actionable prediction to decide whether to proceed with the breeding.

8.3.5 PUC2: Forecast Milk Production

- **Primary Actor**: Dairy Farmer
- **Preconditions**: Historical data for milk production and parental traits is available for input.
- **Trigger**: The farmer requests a prediction for future milk production.
- **Main Success Scenario**:
  1. The farmer inputs the cow's data.
  2. The system processes the input data.
  3. A prediction on future milk production is generated.
- **Postcondition**: The farmer receives a prediction that helps in planning milk yield expectations.

8.3.6 PUC3: Predict Herd Retention Likelihood

- **Primary Actor**: Dairy Farmer
- **Preconditions**: Health and productivity data is available for the cows in question.
- **Trigger**: The farmer requests predictions on herd retention likelihood.
- **Main Success Scenario**:
  1. The farmer selects a cow or group of cows for analysis.
  2. The system processes the available data.
  3. A prediction is generated on whether the cows are likely to stay in or leave the herd.
- **Postcondition**: The farmer receives predictions to assist in managing herd turnover.

9 Functional Requirements

9.1 Functional Requirements

FR1: Predict Breeding Success

- **Description**: The system shall predict the likelihood of a successful breeding event between two cows based on input data regarding parental traits and historical breeding records.
- **Rationale**: This feature will help farmers make more informed breeding decisions, improving breeding efficiency and reducing failures.
- **Fit Criterion**: The system will output a probability of breeding success based on parental data, and this probability must be verified by comparing predicted outcomes with actual breeding success over time.

FR2: Forecast Milk Production

- **Description**: The system shall forecast the milk production of a cow based on historical milk yield and parental traits.

- **Rationale**: Accurate predictions of future milk yield will enable farmers to better plan for production and make decisions on herd management.
- **Fit Criterion**: The forecasted milk production must be within 10% (to be determined) accuracy when compared to actual milk yield over a specified period.

FR3: Predict Herd Retention Likelihood

- **Description**: The system shall predict the likelihood of cows leaving the herd based on health records, productivity, and other historical data.
- **Rationale**: This feature will enable farmers to proactively manage their herd, reducing unexpected departures and improving herd stability.
- **Fit Criterion**: The system will provide a prediction score (e.g., high, medium, low) for herd retention, which can be evaluated by tracking actual herd retention over a six-month period.

FR4: Data Input for Predictions

- **Description**: The system shall allow the farmer to input relevant data, such as breeding records, milk production history, and health records, into the prediction model.
- **Rationale**: To generate accurate predictions, the system requires access to a range of historical data that can be inputted by the user.
- **Fit Criterion**: The input form must successfully accept and validate required fields for at least 95% of user inputs, with clear error handling for missing or incorrect data.

FR5: Report Generation

- **Description**: The system shall generate a report summarizing predictions for breeding success, milk production, and herd retention for selected cows.

- **Rationale**: Farmers need a consolidated report that provides actionable insights based on the predictions generated by the system.
- **Fit Criterion**: The system will generate reports that can be exported to a PDF format and include all requested predictions in a structured layout.

FR6: User Access Control

- **Description**: The system shall provide secure login and role-based access control, ensuring that only authorized users can access or modify the prediction data.
- **Rationale**: Farm management data is sensitive and should only be accessible by authorized personnel.
- **Fit Criterion**: The system must enforce unique login credentials for each user and restrict access based on roles (e.g., farmer, supervisor), with at least 99% reliability in access control enforcement.

FR7: Integration with Farm Management System

- **Description**: The system shall be designed to integrate with existing farm management platforms, such as CATTLEytics Inc, allowing seamless data exchange.
- **Rationale**: Integration with existing platforms will enable the system to leverage historical data and provide predictions without requiring manual data entry.
- **Fit Criterion**: The system should successfully exchange data with the farm management platform 90% of the time during testing, without errors in data transmission.

9.2 Formal Specification

Specification 1: Breeding Success Prediction

- **Description**: The system must be able to predict the likelihood of breeding success between two cows based on historical data, such as parental traits and previous breeding records.

- **Formal Specification**:
  Let X represent a breeding event.
  Let Y represent the set of all possible breeding events.
  Let P represent the predicted probability of success.

  $$\forall X \in Y : \text{Prediction}(X) \rightarrow P \in [0, 1]$$

  The system shall compute the probability P for each breeding event X.

Specification 2: Milk Production Forecast

- **Description**: The system shall forecast future milk production for a given cow based on historical data of both the cow and its parents.

- **Formal Specification**:
  Let C represent a cow in the herd.
  Let Y represent historical milk production data.

  $$\forall C : \text{Forecast}(C, Y) \rightarrow \text{PredictedMilkProduction}(C)$$

  The system shall provide a forecast of future milk production for each cow C based on input data Y.

Specification 3: Herd Retention Likelihood

- **Description**: The system must predict the likelihood of a cow staying within or leaving the herd, based on its health, productivity, and historical data.

- **Formal Specification**:
  Let H represent a cow's health record.
  Let P represent the predicted probability of retention.

  $$\forall H : \text{RetentionPrediction}(H) \rightarrow P \in [0, 1]$$

  The system shall compute the retention probability P for each cow based on its health records and other historical data.

Specification 4: Data Input Validation

- **Description**: The system must validate the input data for cows and breeding events to ensure it is accurate and complete before generating predictions.

- **Formal Specification**: Let D represent the input data for a cow or breeding event.

  $\forall D : \text{InputValid}(D) = \begin{cases} \text{True} & \text{if data passes validation checks} \\ \text{False} & \text{otherwise} \end{cases}$

  The system must ensure that all data D is valid before processing it for predictions.

Specification 5: Report Generation (TBD)

- **Description**: The system must generate a report summarizing predictions for breeding success, milk production, and herd retention likelihood.

- **Formal Specification**:
  Let R represent the report generated.

  $$\forall P, C : \text{GenerateReport}(P, C) \to R$$

  The system shall generate a report R based on the predictions P and input data C.

10 Look and Feel Requirements

10.1 Appearance Requirements

LFR1: Dashboard Display of Predictions

- **Description**: The system's dashboard shall present the predicted cow traits (e.g., milk production, breeding success) in a structured and organized manner, clearly showing individual predictions for each cow.

- **Rationale**: Farmers need to quickly and easily interpret the predictions without searching through large amounts of data. An organized display ensures that all predictions can be understood at a glance.

- **Fit Criterion**: The system shall display predictions for multiple cows in a table format, with clear labels for each trait, such as milk production and herd retention likelihood.

LFR2: Text Contrast for Readability

- **Description**: All text displayed on the system interface shall use a high-contrast color scheme to ensure readability.
- **Rationale**: Farmers and users may access the system in various lighting conditions. High contrast, such as black text on a white background, will ensure clarity.
- **Fit Criterion**: The system shall use a high-contrast color scheme for all text, ensuring that it meets standard readability guidelines under different lighting conditions.

LFR1: Dashboard Display of Predictions

- **Description**: The system's dashboard shall present the predicted cow traits (e.g., milk production, breeding success) in a structured and organized manner, clearly showing individual predictions for each cow.
- **Rationale**: Farmers need to quickly and easily interpret the predictions without searching through large amounts of data. An organized display ensures that all predictions can be understood at a glance.
- **Fit Criterion**: The system shall display predictions for multiple cows in a table format, with clear labels for each trait, such as milk production and herd retention likelihood.

LFR2: Text Contrast for Readability

- **Description**: All text displayed on the system interface shall use a high-contrast color scheme to ensure readability.
- **Rationale**: Farmers and users may access the system in various lighting conditions. High contrast, such as black text on a white background, will ensure clarity.
- **Fit Criterion**: The system shall use a high-contrast color scheme for all text, ensuring that it meets standard readability guidelines under different lighting conditions.

10.2 Style Requirements

LFS1: Consistent Formatting for Input Fields

- **Description**: All data input fields, such as for entering cow or parental traits, should follow a consistent format with clear labels and input validation.
- **Rationale**: A consistent layout for input fields will minimize errors and ensure ease of use when farmers input or update data.
- **Fit Criterion**: Input fields shall maintain a uniform format, with clear labels and consistent spacing throughout the interface.

LFS2: Minimalist Design for the Dashboard

- **Description**: The dashboard interface shall maintain a clean and minimalist design, avoiding unnecessary clutter or decorative elements.
- **Rationale**: A simplified interface will allow farmers to focus on the essential data (predictions) without distractions, ensuring ease of use.
- **Fit Criterion**: Over 80% of users in a usability test shall report that the dashboard is free from unnecessary elements and easy to navigate.

LFS1: Consistent Formatting for Input Fields

- **Description**: All data input fields, such as for entering cow or parental traits, should follow a consistent format with clear labels and input validation.
- **Rationale**: A consistent layout for input fields will minimize errors and ensure ease of use when farmers input or update data.
- **Fit Criterion**: Input fields shall maintain a uniform format, with clear labels and consistent spacing throughout the interface.

LFS2: Minimalist Design for the Dashboard

- **Description**: The dashboard interface shall maintain a clean and minimalist design, avoiding unnecessary clutter or decorative elements.
- **Rationale**: A simplified interface will allow farmers to focus on the essential data (predictions) without distractions, ensuring ease of use.
- **Fit Criterion**: Over 80% of users in a usability test shall report that the dashboard is free from unnecessary elements and easy to navigate.

11 Usability and Humanity Requirements

11.1 Ease of Use Requirements

Insert your content here.

11.2 Personalization and Internationalization Requirements

Insert your content here.

11.3 Learning Requirements

Insert your content here.

11.4 Understandability and Politeness Requirements

Insert your content here.

11.5 Accessibility Requirements

Insert your content here.

12 Performance Requirements

12.1 Speed and Latency Requirements

Insert your content here.

12.2 Safety-Critical Requirements

Insert your content here.

12.3 Precision or Accuracy Requirements

Insert your content here.

12.4 Robustness or Fault-Tolerance Requirements

Insert your content here.

12.5 Capacity Requirements

Insert your content here.

12.6 Scalability or Extensibility Requirements

Insert your content here.

12.7 Longevity Requirements

Insert your content here.

13 Operational and Environmental Requirements

13.1 Expected Physical Environment

Insert your content here.

13.2 Wider Environment Requirements

Insert your content here.

13.3 Requirements for Interfacing with Adjacent Systems

Insert your content here.

13.4 Productization Requirements

Insert your content here.

13.5 Release Requirements

Insert your content here.

14 Maintainability and Support Requirements

14.1 Maintenance Requirements

Insert your content here.

14.2 Supportability Requirements

Insert your content here.

14.3 Adaptability Requirements

Insert your content here.

15 Security Requirements

15.1 Access Requirements

Insert your content here.

15.2 Integrity Requirements

Insert your content here.

15.3 Privacy Requirements

Insert your content here.

15.4 Audit Requirements

Insert your content here.

15.5 Immunity Requirements

Insert your content here.

16 Cultural Requirements

16.1 Cultural Requirements

- The primary language for the product will be English, tailored specifically to Canadian dairy farmers.
- All data and measurements will follow Canadian standards, including the use of liters for milk production, kilograms for weight, and hectares for land area Other relevant units such as Celsius for temperature and metric tons for larger quantities may also be used

17 Compliance Requirements

17.1 Legal Requirements

- The project must comply with the Code of Practice for the Care and Handling of Dairy Cattle, which is a government-regulated standard in Canada. This code outlines mandatory guidelines for the ethical treatment, health, and welfare of dairy cattle. Any management recommendations or actions suggested by the machine learning model will align with these regulations to ensure ethical practices in dairy farming.
- The project must comply with PIPEDA (Personal Information Protection and Electronic Documents Act) for any personal information related to dairy farmers or other individuals involved. This includes the handling of contact details, financial information, and other personally identifiable data.

17.2 Standards Compliance Requirements

- There are no specific standards for collecting dairy farming data in this project. All relevant aspects of data collection and handling are already covered under Legal Requirements, specifically in compliance with PIPEDA for managing sensitive information about dairy farmers, and the Code of Practice for the Care and Handling of Dairy Cattle for ensuring the welfare of the animals.

- For coding standards, the project will adhere to PEP8 to ensure consistent and readable Python code. More information on PEP8 can be found here.

18 Open Issues

- **Data Availability and Quality:** The accuracy of predictions will heavily depend on the quality and completeness of the data obtained from CATTLEytics and Lactanet. Inconsistent or missing data might affect the performance of the model.

- **Model Accuracy:** The machine learning model may need to be fine-tuned multiple times to achieve high accuracy in predicting cow traits. This requires testing with diverse datasets to ensure the model generalizes well.

- **User Interface Usability:** The graphical representation of the family tree and predicted traits needs to be intuitive and user-friendly for farmers with varying levels of technical skill. Determining the best design and ensuring it meets users' needs could take time.

- **Integration with CATTLEytics:** Seamlessly integrating the tool into the existing CATTLEytics system without causing disruptions or requiring major system changes could be technically challenging.

- **Regulatory Compliance:** Ensuring that the predictions and recommendations made by the model comply with Canadian regulations for dairy farming (Code of Practice for the Care and Handling of Dairy Cattle) will require thorough review and potential adjustments during development.

- **Model Interpretability:** Farmers may need clear explanations for how predictions are made to trust and use the tool effectively. Ensuring the model's predictions are explainable is an open issue.

- **Performance Considerations:** The tool needs to be efficient and scalable, handling large amounts of data without significant lag or performance issues, especially as it gets adopted by multiple farms.

19 Off-the-Shelf Solutions

19.1 Ready-Made Products

- There are no fully ready-made products that address the predictive capabilities being developed in this project. While tools like Lactanet provide dairy farm data, they do not offer predictive models based on genetic and health data. Lactanet data will be used primarily for training the custom machine learning model.

19.2 Reusable Components

- Machine learning libraries, such as PyTorch or TensorFlow, will be utilized to develop the custom AI model for cow trait prediction. Additionally, front-end libraries such as D3.js or React Tree Visualization libraries could be considered for visualizing the family-tree diagrams.

19.3 Products That Can Be Copied

- There are no existing products to be copied for this project. However, open-source family-tree visualization tools might serve as inspiration for the graphical aspects of the project.

20 New Problems

20.1 Effects on the Current Environment

- Introducing this system could change how farmers currently select sires or evaluate herd performance. Some may resist adopting new technology due to unfamiliarity.

20.2 Effects on the Installed Systems

- The project will be integrated into the existing Cattleytics software, which is already used to manage dairy farms. The machine learning tool will act as an additional module within Cattleytics, allowing farmers to visualize the family tree of cows and predict future traits based on genetic data. Seamless integration with the current system will be prioritized to ensure smooth adoption and ease of use.

20.3 Potential User Problems

- Users may face difficulties interpreting complex AI model outputs, so ensuring the tool's recommendations are easy to understand is key.

20.4 Limitations in the Anticipated Implementation Environment That May Inhibit the New Product

- The tool will need to function effectively on standard farm computing systems, which may have limited processing power or internet connectivity.

20.5 Follow-Up Problems

- Continuous updates may be needed to improve the model based on feedback from farmers. Future updates may also need to address changes in farming practices

21 Tasks

21.1 Project Planning

The project will be scheduled to follow the deliverables deadlines as outlined in the SFWRENG 4G06 course outline.

21.2 Planning of the Development Phases

The development of the project will be conducted in four primary phases:

Table 1: Project Documentation Tasks

Phase	Task	Due Date
Phase 1	Hazard Analysis	October 23, 2024
	Verification and Validation Plan	November 1, 2024
	Proof of Concept Demonstration	November 11–22, 2024
	Design Documentat	January 15, 2025
	Revision 0 Demonstration	February 3–14, 2025
Phase 2	Verification and Validation Report	March 7, 2025
	Final Demonstration	March 24–30, 2025
	Final Documentation	April 2, 2025

1. Data Acquisition and Initial Integration: This initial phase focuses on acquiring, cleaning, and integrating historical dairy farm data into the system. Activities include:

- Collecting and preprocessing datasets from CATTLEytics and Lactanet to ensure data quality and consistency.
- Developing scripts to handle data normalization, unit conversion (e.g., to liters and kilograms), and encryption for compliance with PIPEDA.
- Implementing an initial data pipeline to allow seamless integration between the existing CATTLEytics system and the new module.

This phase will conclude with a Proof of Concept (POC) demonstration, where the pipeline and initial data handling capabilities will be validated.

2. Development of Predictive Model and Core Functionality: The second phase is dedicated to developing and integrating the core machine learning model responsible for predicting key outcomes such as breeding success rates. This phase involves:

- Experimenting with various algorithms (e.g., logistic regression, decision trees, neural networks) using Python libraries such as Scikit-learn, TensorFlow, or PyTorch to determine the most suitable approach for our data.

- Developing the model's training, validation, and testing processes, ensuring the system is optimized for accuracy and performance.

- Implementing backend APIs for data retrieval and model output generation to support the frontend development in the next phase.

This phase will culminate with an evaluation of the model's initial accuracy and robustness, and any necessary refinements will be documented for the next sprint.

3. System Integration and Frontend Development: In the third phase, the focus shifts to building the user-facing components and integrating the machine learning model into the existing CATTLEytics system. Activities include:

- Developing a React-based frontend interface that allows farmers to input data, visualize predictions, and interpret results in an intuitive manner.

- Implementing interactive visualizations using libraries like D3.js to display herd health trends, genetic family trees, and predictive insights.

- Integrating the frontend with the backend APIs, ensuring seamless data flow and responsiveness across the system.

- Initial system testing to verify that the user interface works well with the machine learning outputs and data pipeline.

This phase will end with a functional demo of the frontend and backend integration, allowing for early user feedback and adjustments.

4. Testing, Deployment, and Refinement: The final phase focuses on refining the system, conducting extensive testing, and preparing for deployment. Key activities include:

- Performing unit, integration, and system-level testing to validate the accuracy, performance, and reliability of the predictive model and frontend interface.

- Setting up a Continuous Integration/Continuous Deployment (CI/CD) pipeline using GitHub Actions to automate testing and deployment to a staging environment.

- Conducting user testing sessions with industry stakeholders to gather feedback, refine features, and enhance the usability of the system.
- Finalizing user documentation, training materials, and preparing the system for deployment to the production environment.

This phase will conclude with a complete system demonstration and the final handoff to stakeholders, ensuring the solution meets all requirements and is ready for the EXPO presentation.

All phases will run concurrently with documentation development to meet capstone deliverables and ensure that non-functional requirements, such as performance and security, are integrated throughout the development process. Each phase will have some overlap to allow flexibility and responsiveness to evolving requirements.

22 Migration to the New Product

22.1 Requirements for Migration to the New Product

The new predictive module that is being developed for this project will need to be integrated into the existing CATTLEytics software. Key requirements include:

- Ensuring compatibility of the new module with existing CATTLEytics data formats and APIs.
- Ensuring that the frontend interface seamlessly integrates with the existing user interface of CATTLEytics.
- Developing a rollback plan to ensure that, if issues occur during migration, the system can revert to the previous stable state.
- Testing the migration process on a staging environment to validate functionality before deployment.

22.2 Data That Has to be Modified or Translated for the New System

This section will be completed once the relevant data model details are available.

23 Costs

The goal for this project is to minimize costs by leveraging open-source tools and resources wherever possible. The primary development tools, including Python, TensorFlow, PyTorch, React, and GitHub for version control, are open-source and free to use. Continuous Integration will be managed through GitHub Actions, which offers a free tier suitable for our needs.

Any additional datasets needed from third-party providers (e.g., Lactanet) may incur small fees, but these will be minimized through the use of open or freely available datasets when possible. The team at CATTLEytics will provide access to a dataset that they have scraped from Lactanet.

At this time, no other significant direct monetary costs are anticipated. The project plan remains flexible to adapt if unforeseen expenses arise during development, but the team will prioritize cost-effective and open-source solutions whenever feasible.

24 User Documentation and Training

24.1 User Documentation Requirements

The user documentation for the system will be concise and user-friendly, aimed at ensuring farmers and stakeholders can easily understand and utilize the tool. The documentation will include:

- Step-by-step instructions on accessing and utilizing the platform.
- Instructions on interpreting reports generated by the system.
- A troubleshooting section to address common issues.

24.2 Training Requirements

No formal training will be required as the tool is designed to be intuitive and straightforward. The provided documentation will be sufficient for users to operate the system without additional assistance.

25 Waiting Room

This section will be updated to highlight requirements that need to be put on hold due to time constraints or other factors.

26 Ideas for Solution

Insert your content here.

Appendix — Reflection

The information in this section will be used to evaluate the team members on the graduate attribute of Lifelong Learning. Please answer the following questions:

1. What knowledge and skills will the team collectively need to acquire to successfully complete this capstone project? Examples of possible knowledge to acquire include domain specific knowledge from the domain of your application, or software engineering knowledge, mechatronics knowledge or computer science knowledge. Skills may be related to technology, or writing, or presentation, or team management, etc. You should look to identify at least one item for each team member.

2. For each of the knowledge areas and skills identified in the previous question, what are at least two approaches to acquiring the knowledge or mastering the skill? Of the identified approaches, which will each team member pursue, and why did they make this choice?