Problem Statement and Goals
ProgName

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

Table 1: Revision History

Date	Developer(s)	Change
Sept. 16, 2025	Changhao Wu	Initial Draft
Sept. 20, 2025	Changhao Wu	Updated Problem Statement and Goals document
Sept. 20, 2025	Changhao Wu	Completed Reflection section
...	...	...

1 Problem Statement

[You should check your problem statement with the problem statement checklist. —SS]

[You can change the section headings, as long as you include the required information. —SS]

1.1 Problem

On modern dairy farms, many routine tasks require farmers to quickly locate and separate specific cows from the herd. Examples include providing medical treatment to a sick cow, monitoring the recovery of an injured animal, collecting milk samples for quality testing, administering vaccines, or tracking cows with unusual feeding or reproductive behaviors.

Currently, locating specific animals is highly time-consuming. Farmers often have to walk through an entire pen, visually check ear tag numbers one by one, and manually identify the right cow. This process is not only slow and labor-intensive, but also stressful for both workers and animals.

The problem is compounded by the fact that cows are large, mobile herd animals. In a crowded barn or pasture, they are constantly moving, blocking each other from view, or clustering together, making manual identification even more difficult. Under these conditions, locating a single cow can take far longer than planned and disrupt other farm operations.

1.2 Inputs and Outputs

- **Inputs**
  - Visual data streams from on-farm environments.
  - Supplementary annotated datasets for training and evaluation.
- **Outputs**

- Information that distinguishes individual cows.
- Consistent identification of cows across time.
- Visual outputs annotated with each cow's predicted identity

[Characterize the problem in terms of "high level" inputs and outputs. Use abstraction so that you can avoid details. —SS]

1.3 Stakeholders

Direct Stakeholders

1. **Cattleytics Inc. (Industry Partner)**: The primary stakeholder and end user of the system. They benefit directly from improved efficiency in locating, identifying, and monitoring individual cows for treatment, feeding management, and general herd health.

Indirect Stakeholders

1. **Veterinarians**: By enabling faster and more reliable identification of target animals, veterinarians can administer treatments and health checks more efficiently, improving overall herd health management.
2. **Other Dairy Farmers and Ranchers**: Since the system is designed with relatively low deployment cost, it can be adopted by other farms to reduce labor demands, increase operational efficiency, and ultimately improve profitabilit.
3. **Animal Welfare and Protection Organizations**: Better tracking and monitoring ensure that cows receive timely care, reducing stress and discomfort. These outcomes align with the goals of organizations advocating for animal well-being and ethical treatment in agriculture.

1.4 Environment

Hardware Environment

- Deployment will take place in dairy farm environments such as barns, feeding areas, or open pens, where lighting, shadows, and occlusion from other animals may affect visibility.
- Input devices will consist of fixed surveillance cameras installed in barns or portable cameras (e.g., smartphones).
- Development and training will primarily be conducted on a GPU-enabled workstation (e.g., a desktop equipped with an NVIDIA RTX 4060 or higher-class GPU).
- For later testing, the system may also be evaluated on lightweight hardware (e.g., NVIDIA Jetson) to explore potential edge deployment, but the primary focus is on farm-level camera input rather than embedded control.

Software Environment

- Operating System: Ubuntu Linux (WSL2 on Windows will also be used for development).
- Programming Language: Python.
- Core Libraries and Frameworks: PyTorch, Ultralytics YOLOv8, OpenCV, scikit-learn, and ByteTrack.
- Development Tools: Git (for version control), GitHub (for remote collaboration and repository hosting), GitHub Projects (for task management), GitHub Actions (for CI/CD), and VS Code as the primary IDE.
- Testing and Evaluation Tools: pytest for unit testing, flake8/black for linting and formatting, and TensorBoard/Matplotlib for model evaluation and visualization.

[Hardware and Software Environment —SS]

2 Goals

The primary goal of this project is to design and implement a computer vision system that can:

1. Accurately locate cows within visual inputs (images or video), ensuring that the presence of animals can be detected without manual inspection.
2. For video data, maintain continuous tracking of individual cows across frames, ensuring that the same animal is consistently recognized over time.
3. Provide a practical application scenario: given the information of a target cow (e.g., name or identifier), the system highlights this cow within the input data, eliminating the need for manual searching.

3 Stretch Goals

- Enable deployment on portable or embedded platforms, making the system accessible directly on-farm without requiring high-end computing infrastructure.
- Broaden applicability by extending recognition capabilities beyond cows to other livestock or farm management contexts.
- Deliver a farmer-oriented interface that visualizes animal activity and identity records, and allows data export to support decision-making and reporting.

4 Extras

4.1 Performance Report:

A comprehensive performance report will be created to document the system's accuracy, speed, and reliability. This will include quantitative evaluation of detection, tracking, and identification models under different conditions (e.g., lighting, occlusion, herd density). The report will provide insights into system limitations and suggest areas for future improvement.

4.2 User Manual:

A user-friendly manual will be prepared to guide non-technical users, such as farmers and dairy workers, through system setup and operation. The manual will include step-by-step installation instructions, examples of typical use cases, troubleshooting tips, and recommendations for best practices when deploying the system on a farm.

[For CAS 741: State whether the project is a research project. This designation, with the approval (or request) of the instructor, can be modified over the course of the term. —SS]

[For SE Capstone: List your extras. Potential extras include usability testing, code walkthroughs, user documentation, formal proof, GenderMag personas, Design Thinking, etc. (The full list is on the course outline and in Lecture 02.) Normally the number of extras will be two. Approval of the extras will be part of the discussion with the instructor for approving the project. The extras, with the approval (or request) of the instructor, can be modified over the course of the term. —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. What went well while writing this deliverable?

   One positive outcome was that our team was able to refine the level of detail in our problem statement and goals. Initially, we included too many technical specifics, which risked restricting our future implementation choices. Under the guidance of the TA, we revised these sections to be more abstract yet still measurable, ensuring they provide direction without prematurely narrowing our design space. In addition, we carried out early tests on potential external components that the project might depend on. This gave us a baseline understanding of the tools and libraries available and allowed us to set realistic goals that stretch our abilities without being unachievable.

2. What pain points did you experience during this deliverable, and how did you resolve them?

   A major challenge was the relatively low and inconsistent engagement from some team members, likely due to heavy workloads in other courses this term. This limited the amount of sustained progress we could make week to week. As individuals, we recognize that we need better time planning to avoid leaving tasks until just before deadlines. Such last-minute work does not leave sufficient time to polish the deliverables to a high standard. Another difficulty was our limited experience with disciplined use of version control. While we managed to use Git and GitHub for collaboration, our workflow lacked consistency, which could make the project harder to sustain in the long run. To resolve this, we intend to adopt clearer commit conventions, branch organization, and more frequent pushes so that our repository remains well-structured. In addition, we plan to protect the `main` branch by requiring that all changes go through a pull request, and that every pull request must be reviewed and approved before merging. This process will help us maintain code quality and accountability throughout the project.

3. How did you and your team adjust the scope of your goals to ensure they are suitable for a Capstone project (not overly ambitious but also of appropriate complexity for a senior design project)?

   When defining our goals, we initially considered ambitious objectives such as pursuing both the current project and an additional module for lameness detection in cows. However, after further investigation, we realized that this direction would be far too complex given the resources and circumstances currently available to us. Because lameness detection has high commercial value, there are very few publicly available projects or datasets we could reference. Building such a system would likely require us to collect data from scratch, for example by visiting farms, recording videos, and learning how to reliably label lameness under veterinary guidance. Given the scale of this work and the heavy course workload that most of our team members are already carrying this term, it became clear that attempting both projects in parallel would not be feasible.

After reflection, we scaled our scope down to focus on three core capabilities: detecting cows, maintaining continuous tracking in video, and identifying individuals. Stretch goals were then defined for deployment on embedded devices and broader applicability to other livestock. Importantly, the scope we chose remains within a reasonable range of our current technical capacity, challenging us to go slightly beyond our comfort zone without being unrealistic. This balance ensures the project is both feasible and sufficiently challenging to be worthy of a Capstone project.