Our team brings together experience from diverse project domains. In this inception phase, we are consolidating prior work and evaluating where AI can be strategically integrated into the development stages of our ongoing modules. As a group, we will conduct a comprehensive analysis of AI use cases-requirements analysis, implementation, and testing, using both functional and non-functional measures (e.g. productivity, quality, reproducibility). We will apply the same evaluation framework across our ongoing Master's projects to compare results across domains and build a more holistic perspective.

## Brief projects overview:

Animal Ecology

This project aims to reliably track wildlife and understand behavior so ecologists can act in time to help animal species. We're building an end-to-end pipeline that connects field sensors to a simple, dependable data flow: detect events in the field, capture follow-up evidence when it matters, sync everything to a cloud for review, and then send updated models and guidance back to the field. The goal is better, faster observations of rare behaviors with less disturbance and manual effort, so insights move from the field to decisions more smoothly.

Adaptive Optics OCT (AO-OCT)

We're strengthening a retinal imaging pipeline from intake to review so results are cleaner, faster to obtain, and easier to trust. The emphasis is on smoothing the preprocess -> train -> infer -> quality-check flow, reducing handoffs, and improving visibility into failures on HPC. We also keep careful guardrails—such as subject-wise splits and privacy checks and record each run with its settings and outputs so findings can be traced and reproduced.

Digital Agriculture

The goal here is a dependable way to bring together varied farm data—photos, video, logs, sensor streams—and move it reliably to the right compute resources without breaking downstream work. The system accepts inputs from field devices, adds just-enough metadata, stores them safely, registers entries for easy lookup, and routes jobs to the appropriate workflows. Outputs are published with clear lineage so results stay consistent and repeatable as new data types come online.

## Research Objectives:

1. **Requirements analysis & scoping:**
    - **What:** Summarize stakeholder inputs, extract requirements, detect conflicts/duplicates, draft acceptance criteria/user stories.
    - **Measure:** Requirement coverage; conflicts/duplicates identified; time-to-first draft; rework due to missed requirements.

2. **Design support (architecture & data flow)**
    - **What:** Draft architecture options, sequence/data-flow diagrams, and interface contracts from requirements.
    - **Measure:** To be decided or could be merged into either extension of Requirement analysis.

3. **Implementation assistance (coding)**
    - **What:** Generate boilerplate, suggest functions/tests, provide inline explanations, and small refactors.
    - **Measure:** Task completion time; delivery velocity; first-pass test pass rate; defects per change.

4. **Test generation & maintenance**
    - **What:** Propose unit/integration tests from specs/code; update tests when APIs change; identify gaps.
    - **Measure:** Coverage; time to create/update tests; flaky test rate; pre-merge failures caught.

5. **Documentation & knowledge capture**
    - **What:** Generate/refresh READMEs, API docs, runbooks, and change logs from code and commits.
    - **Measure:** Doc freshness vs. last commit; onboarding time; broken link/missing section rate.

6. **Compliance, risk, and quality gates**
    - **What:** Check licenses, provenance, policy violations, and required approvals before release.
    - **Measure:** Violations caught pre-release; audit duration; % releases with complete provenance.

7. **Project management assistance**
    - **What:** Turn meeting notes into tasks, align tasks with milestones, and flag schedule risks early.
    - **Measure:** Planning time saved; on-time delivery rate; reduction in slipped/unassigned tasks.

## Timeline

**Phase 1 : Frame, Gather, Scan**

- Finalize objectives, scope, and success criteria.
- Draft the evaluation rubric.
- Scan tools by use case and shortlist 2–3 per area.

**Phase 2 : Compare & Set Guardrails**

- Run small, prompt-level trials on representative scenarios.
- Build a comparison matrix and observations log.
- Define human-in-the-loop vs. automation guardrails.

**Phase 3 : Synthesize & Deliver (Weeks 5–6)**

- Compile the tools report with side-by-side comparisons and control-vs-automation analysis.
- Peer review, refine, and prepare the final documentation and presentation.

## Inventory of Risks:

- Limited familiarity with the AI-tool landscape, making holistic comparison difficult.
- Hallucinations or incorrect guidance from AI assistants.
- Outcomes can become person-dependent if prompts, context, and decisions aren't captured, making it hard for others to continue or replicate the work.
- Tool instability and rapid version churn/deprecations, our observations can become stale quickly.
- Multiple tools and its subscriptions complicate access and make fair benchmarking difficult and costly.
- Over-dependence on AI tools at the expense of human judgment.
- Privacy concerns when sending configuration data to third-party services.
- Overfitting to easy metrics can undermine real quality and user value.