Assignment Template: Project Assignment 1

Team Name: Drone Based Development
Team Members: Logan Sutton, Gil Rezin
Course Code: Cpts 421
Date: 9/24/25

1. Functional Requirements (FRs)

(List at least 8–10. Use FR-01, FR-02 … numbering)

- **FR-01:** The system shall allow users to upload drone imagery
- **FR-02:** The system shall stitch drone imagery into a field mapping
- **FR-03:** The system shall generate a pesticide prescription based on the field mapping
- **FR-04**: The system shall calculate an optimal path for pesticide application on the given field
- **FR-05**: The system shall upload the pesticide prescription to the physical robot sprayer
- **FR-06**: The system shall upload the optimal pesticide application path to the physical robot
- **FR-07**: The system shall notify the robot when to start its pesticide application service
- **FR-08:** The system shall allow users to view field mappings
- **FR-09**: The system shall allow the user to manually override sprayer settings
- **FR-10**: The system must store field mappings and optimal paths for later use
- **FR-11**: The system shall support multi-file uploading and processing
- **FR-12**: The system shall support a button for uploading prescriptions to robotic sprayers
- **FR-13 [GenAI]**: The system shall learn to find the optimal pesticide prescription via machine learning simulations.
- **FR-14 [GenAI]:** The system shall log all pesticide applications, including time, dosage, and area coverage, for compliance reporting.
- **FR-15 [GenAI]:** The system shall provide a visual preview of the spraying path overlayed on the field map before execution.

2. Non-Functional Requirements (NFRs)

(List at least 5–7. Use NFR-01, NFR-02 … numbering)

- **NFR-01:** The system must run on a singular machine with an I7 CPU (> or equivalent) and a Nvidia 3060 GPU (> or equivalent)
- **NFR-02:** The image stitching computations shall take no longer than 5 minutes for a 20 acre field

- **NFR-03:** The system must load previous field mappings in under 1 minute
- **NFR-04**: The system must calculate optimal path in under 2 minutes
- **NFR-05:** The system must be independent and run offline (no support from cloud computing or external services)
- **NFR-06 [GenAI]:** The system shall be designed with a modular architecture so that components (mapping, prescription generation, robot interface) can be independently updated.
- **NFR-07 [GenAI]:** The user interface must remain responsive (no freezing longer than 3 seconds) during background computations.

  …

3. User Stories with Acceptance Scenarios

(Write as many as reasonably possible at this stage. Each story must be peer-reviewed in the team. Add ***at least two scenarios per story*** *— one positive, one negative/edge case.)*

US-01: Upload Drone Imagery

As an end-user farmer, I want an Upload Drone Imagery button so that I can easily upload my drone's images to generate a field map.

Acceptance Scenarios (Gherkin):

Scenario 1: Positive Flow

Given I am on the main page of the application,

When I press on the Upload button,

Then I should be prompted to upload my drone imagery files.

Scenario 2: Negative/Edge Case

Given that I am now selecting my files from the File Explorer,

When I add an excessively large file count/size,

Then the application should prompt me to upload the files in smaller batch sizes.

Scenario 3: Invalid Data

Given that I am selecting files from File Explorer,

And given that drone data may be corrupted or the wrong file may be uploaded,

When I add an invalid file,

The application should prompt that the given file is invalid.

US-02: Transfer Prescription to Robot

As an end-user farmer, I want to have the system upload the generated prescription to my spraying robot, so I do not have to manually upload the prescription for each spray.

Acceptance Scenarios (Gherkin):

Scenario 1: Positive Flow

Given that I have a prescription generated and ready to upload,

When I click Upload Prescription,

Then I expect the prescription to be uploaded to the robot sprayer.

Scenario 2: Negative/Edge Case

Given that I have a prescription generated and ready to upload, but the robot sprayer is not connected,

When I click Upload Prescription,

Then the system should display that the robot needs to be connected to upload a prescription.

US-03: Individual Nozzle Control

As a sprayer robot, I want to be able to control each sprayer nozzle individually so I can minimize pesticide resource expenditure.

Acceptance Scenarios (Gherkin):

Scenario 1: Positive Flow

Given that the sprayer robot is moving along the prescribed path,

When the prescription specifies different rates for nozzles,

The robot should adjust spraying rates across the boom accordingly

Scenario 2: Negative/Edge Case

Given that the prayer robot is moving along its prescribed path,

If prescribed path causes overhang outside of the field boundary,

The nozzle control for the “out of bounds” parts of the boom should shut off.

US-04: Evaluate Spraying Effectiveness

As a team, we want to be able to measure the effectiveness of the robotic sprayer in order to continually improve our ML model.

Acceptance Scenarios (Gherkin):

Scenario 1: Positive Flow

Given that the robot sprayer has completed a spraying cycle,

When the team collects the spraying results,

Then the results should be compared against the coverage requirements.

Scenario 2: Negative/Edge Case

Given that the robot has completed a spraying cycle with a sensor failure,

When the team attempts to validate the effectiveness,

The system should flag the missing data

US-05 [GenAI]: Visual Path Preview
As a farmer, I want to preview the spraying path on top of my field map so that I can verify the prescription is correct before sending it to the robot.

Acceptance Scenarios (Gherkin):

Scenario 1: Positive Flow

Given that a prescription has been generated,

When I click “Preview Path,”

Then the spraying path should be displayed overlayed on the field map.

Scenario 2: Negative/Edge Case

Given that the spraying path overlaps areas outside the mapped boundary,

When I preview the path,

Then the system should highlight the error and prevent uploading until corrected.

(Repeat for all user stories US-03 … US-XX)

---

4. Brainstorming and GenAI Reflection

Step A: Team Brainstorming

(Document initial FRs, NFRs, stories, and scenarios created without AI. Record 3–4 key decision points from your discussion.)

- Decision Point 1: Use WebODM instead of Pix4d as WebODM is open source
- Decision Point 2: Necessity for a completely offline system (no cloud processing)
- Decision Point 3: Automated backend workflow from when the user uploads their drone imagery data.

Step B: GenAI-Assisted Brainstorming

(Summarize what GenAI suggested — new FRs, NFRs, user stories, or scenarios.)

- GenAI Suggestion 1: Using a simulated environment, train a machine learning model to navigate the environment in an optimal manner.

- GenAI Suggestion 2: Increase program transparency through usage of visualizations.
- GenAI Suggestion 3: Allow for ease of expansion through modular code and logged data.

Step C: Refined Requirements & Stories

*(Update Step A with insights from GenAI. Clearly mark new/modified items — e.g., FR-06 **[Added after GenAI]**.)*

Done

Step D: Reflection (200–300 words)

- How did you feel about using GenAI in this exercise (e.g., empowering, surprising, confusing, over-reliant)?
- In what ways did GenAI change or improve your brainstorming compared to your team's initial work? (Consider clarity, creativity, and coverage.)
- Did GenAI help you uncover new functional, non-functional, or business requirements that you had not considered before? Provide examples.

Utilizing GenAI in this exercise was beneficial, as it provided additional insight and sparked new requirement ideas for our project. While some of the ideas generated were not quite what we were looking for, we were surprised by the number of unique suggestions GenAI provided us. GenAI also allowed us to carefully review our previous requirements by allowing us to refine and clarify our original ideas. Additionally, we felt like GenAI did a decent job of ensuring we had full coverage of our scenarios. For example, GenAI suggested that we add a user story for a visual path preview, which we thought was an excellent suggestion and a good requirement to have for our project. GenAI also allowed us to explore additional functional requirements, such as

1. usage of machine learning via simulations within our optimal spraying calculations
2. ease of codebase expansion and debugging during hardware integration
3. end-user experience via route visualizations

Additionally, Gen AI provided new non-functional requirements, such as

1. designing the software architecture in a modular fashion to allow the mapping, prescription generation, and robot interface components to have independence and update independently
2. limit the amount of input delay experienced on the UI to 3 seconds or less, even while performing computationally expensive tasks

Overall, we felt like GenAI contributed positive and helpful results to our project requirements.

Final Submission Checklist

- Functional Requirements (8–10+)
- Non-Functional Requirements (5–7+)
- User Stories (all possible at this stage, with acceptance scenarios)

- Gherkin Scenarios (≥2 per story, reviewed)
- Brainstorming & GenAI Section (Steps A–D complete)
- Reflection (200–300 words)
- PDF format, file named correctly

File name format:
TeamName_Assignment1_Requirements.pdf