FarmBot Thermal Imaging Integration Testing

Testing Overview

This document outlines the testing strategies for the FarmBot project, including unit tests, integration tests, and acceptance tests. Each type of test is designed to ensure that our system meets the functional and performance requirements of the project.

1. Unit Testing

Objective

To verify that each independent module functions as expected and is error-free.

Methodology

- **Software**: Test each functionality (such as data processing, user interface responses, etc.) separately.
- **Hardware Interface Code**: Test segments of code that control hardware operations to ensure they execute commands correctly.

Tools

- Use RSpec, a popular testing tool for Ruby applications.

Plan

- Define test cases for each module.
- Implement automated test scripts.
- Record test results and any issues found.

2. Integration Testing

Objective

To ensure that all modules work together as intended, particularly focusing on the integration of software and hardware components.

Methodology

- **Camera to Raspberry Pi**: Verify that camera data is correctly transmitted to and processed by the Raspberry Pi.
- **Raspberry Pi to Server**: Test that data can be securely transmitted through internal network penetration to the server and is correctly displayed on the web app.

Tools

- Manual testing will be used due to the complexity and the need to monitor hardware and software interactions in real-time.

Plan

- Design test cases covering all critical integration points.
- Conduct tests and document the results.
- Analyze potential issues and adjust integration strategies as necessary.

3. Acceptance Testing

Objective

To verify that the entire system meets the specific requirements of the client and user stories.

Methodology

- Design acceptance tests based on client requirements and user stories.
- Verify that users can remotely view real-time thermal imaging and interact with the interface as intended.

Tools

- Manual testing.
- Use user stories and acceptance criteria to guide test planning.

Plan

- Develop acceptance criteria for each user story.
- Perform tests, recording videos or taking screenshots as evidence of functionality.
- Compare results to expected outcomes to confirm requirement fulfillment.