

Part I : Overall Test Plan

We will be approaching our test plan by building upon each part. We will be testing our sensors individually at first to make sure that we understand the communication channels and then we will test all the sensors together to make sure they can all work collectively and there are no issues with timing. The next part of our testing will be testing the application. This will involve integrating Grafana into our project to properly read the collected data, the data can be visualized within the application, and notifications can be sent to the app user when critical conditions are met. Once we know that our sensor system is working properly and can communicate with the app, we can then conduct our final test at the vineyard and make sure that the fully integrated system is working properly.

Part II : Test Case Descriptions

| | |
|----------------|--|
| THST1.1 | Temperature Humidity Sensor Test |
| THST1.2 | This test will ensure that the sensor is accurately communicating data to the microcontroller. |
| THST1.3 | This test will have the temperature and humidity sensor properly wired to our microcontroller and programmed to collect data points of the environment at a desired rate |
| THST1.4 | Inputs: Environmental Temperature, Environmental Humidity |
| THST1.5 | Outputs: Analog signal(0-5v) sent from the sensor to the microcontroller |
| THST1.6 | Normal |
| THST1.7 | Whitebox |
| THST1.8 | Functional |
| THST1.9 | Unit |
| THST1.10 | Results: Accurate data received on the micro controller |
| LST1.1 | Light Sensor Test |
| LST1.2 | This test will ensure that the sensor is accurately communicating data to the microcontroller |
| LST1.3 | This test will have the light sensor properly wired to our microcontroller and programmed to collect data points of the environment at a desired rate. |
| LST1.4 | Inputs: Sunlight exposure |
| LST1.5 | Outputs: Analog (0-5v) sent from the sensor to the microcontroller |
| LST1.6 | Normal |
| LST1.7 | Whitebox |
| LST1.8 | Functional |
| LST1.9 | Unit |
| LST1.10 | Results: Accurate data received on the micro controller |
| RFST1.1 | Rain Fall Sensor Test |
| RFST1.2 | This test will ensure that the sensor is accurately communicating data to the microcontroller |
| RFST1.3 | This test will have the rain sensor properly wired to our microcontroller and programmed to collect data points of the environment at a desired rate. |

| | |
|-----------------|--|
| RFST1.4 | Inputs: Rainfall |
| RFST1.5 | Outputs: Analog (0-5v) sent from the sensor to the microcontroller |
| RFST1.6 | Normal |
| RFST1.7 | Whitebox |
| RFST1.8 | Functional |
| RFST1.9 | Unit |
| RFST1.10 | Results: Accurate data received on the micro controller |
| SMST1.1 | Soil Moisture Sensor Test |
| SMST1.2 | This test will ensure that the sensor is accurately communicating data to the microcontroller |
| SMST1.3 | This test will have the soil moisture sensor properly wired to our microcontroller and programmed to collect data points of the environment at a desired rate. |
| SMST1.4 | Inputs: Moisture of the soil |
| SMST1.5 | Outputs: Analog (0-5v) sent from the sensor to the microcontroller |
| SMST1.6 | Normal |
| SMST1.7 | Whitebox |
| SMST1.8 | Functional |
| SMST1.9 | Unit |
| SMST1.10 | Results: Accurate data received on the micro controller |
| CST1.1 | Collective Sensor Test |
| CST1.2 | This test is to ensure that all the sensors can work at the same time and collect data without impacting the other sensors |
| CST1.3 | This test will have all the sensors wired to our microcontroller and be able to manage the data points of multiple different types of sensors and different number of sensors at different locations to get different readings |
| CST1.4 | Inputs: Temperature, humidity, light, rainfall, soil moisture, at different locations |
| CST1.5 | Outputs: Analog (0-5v) sent from the sensor to the microcontroller |
| CST1.6 | Normal |
| CST1.7 | Whitebox |
| CST1.8 | Functional |
| CST1.9 | Integration |
| CST1.10 | Results: Accurate data points from different locations |
| GDCT 1.1 | Grafana Data Completion Test |
| GDCT 1.2 | This test will ensure that all data being collected by all sensors is being received by Grafana. |
| GDCT 1.3 | Once all sensor tests are completed, each datapoint collected by each sensor will be compared to the datapoints being collected in Grafana. |
| GDCT 1.4 | Inputs: All sensor datapoints, expected Grafana datapoints. |
| GDCT 1.5 | Outputs: Number of Grafana datapoints missing compared to sensor datapoints. |
| GDCT 1.6 | Normal |
| GDCT 1.7 | Blackbox |
| GDCT 1.8 | Functional |
| GDCT 1.9 | Integration |

| | |
|-------------------|---|
| GDCT 1.10 | Results: Successful interface between sensors, microcontroller, and Grafana |
| DVAUTS 1.1 | Data Visualization Application Unit Test Suite |
| DVAUTS 1.2 | This test will ensure that all parts of the data visualization application are working successfully for the end user. |
| DVAUTS 1.3 | Comprehensive unit tests will be implemented for each module in the application. |
| DVAUTS 1.4 | Inputs: Mocked data structures necessary for module execution. |
| DVAUTS 1.5 | Outputs: Indication of success or failure for each module in the application. |
| DVAUTS 1.6 | Normal |
| DVAUTS 1.7 | Whitebox |
| DVAUTS 1.8 | Functional |
| DVAUTS 1.9 | Unit |
| DVAUTS 1.10 | Results: Complete application functionality for the end user. |
| AST1.1 | Alarm System Test |
| AST1.2 | This test will ensure that when hazardous weather is present to send a notification to the app user to make the proper preparations |
| AST1.3 | We will create a range for the sensors to read and if the data read by the sensor falls within that range, a notification will be sent out |
| AST1.4 | Inputs: Sensor data, and simulated hazardous conditions |
| AST1.5 | Outputs: A notification being sent to the user when a hazardous condition is present |
| AST1.6 | Boundary |
| AST1.7 | Whitebox |
| AST1.8 | Performance |
| AST1.9 | Integration |
| AST1.10 | Results: A system to check for hazardous conditions and notifications sent when needed |
| SMATS 1.1 | Supporting Microservice API Test Suite |
| SMATS 1.2 | This test will ensure that the APIs of all supporting microservices are all functioning correctly. |
| SMATS 1.3 | Comprehensive API tests will be implemented for each microservice supporting the application. |
| SMATS 1.4 | Inputs: Mock API inputs and expected API responses |
| SMATS 1.5 | Outputs: Indication of success or failure for each call to each API. |
| SMATS 1.6 | Normal |
| SMATS 1.7 | Blackbox |
| SMATS 1.8 | Functional |
| SMATS 1.9 | Unit |
| SMATS 1.10 | Results: Complete microservice functionality in support of the application. |
| NFT1.1 | Normal Field Test |
| NFT1.2 | This test will ensure that the sensors are correctly placed within the vineyard and wired correction and able to send data to the app |
| NFT1.3 | The sensors will be placed in desired locations in the vineyard and make sure that they are wired properly, and data can be collected in the wider area |

| | |
|---------|--|
| NFT1.4 | Inputs: Sensors across the vineyard, microcontroller, canopy, application |
| NFT1.5 | Outputs: Fully integrated system |
| NFT1.6 | Normal |
| NFT1.7 | Whitebox |
| NFT1.8 | Functional |
| NFT1.9 | Integration |
| NFT1.10 | Results: Fully functioning sensor monitoring system |

Part III : Test Case Matrix

| | Normal/Abnormal | Blackbox / Whitebox | Functional/ Performance | Unit/Integration |
|---------|-----------------|------------------------|----------------------------|------------------|
| THST1 | Normal | Whitebox | Functional | Unit |
| LST1 | Normal | Whitebox | Functional | Unit |
| RFST1 | Normal | Whitebox | Functional | Unit |
| SMST1 | Normal | Whitebox | Functional | Unit |
| CST1 | Normal | Whitebox | Functional | Integration |
| GDCT1 | Normal | Blackbox | Functional | Integration |
| DVAUTS1 | Normal | Whitebox | Functional | Unit |
| AST1 | Boundary | Whitebox | Performance | Integration |
| SMATS1 | Normal | Blackbox | Functional | Unit |
| NFT1 | Normal | Whitebox | Functional | Integration |