

Software Requirements Specification (SRS)

Revision Notice

Project Name	Automated Testing System
Version No.	1
Date Of Release	04/26/2022
Author	Christian Apostoli
Notice	This SRS was made as part of the Requirements Engineering and Analysis course at Florida Gulf Coast University

1. Introduction

1.1 Purpose

The software will reduce labor costs and time, for employees will spend less time on testing activities, and such testing activities will only require one employee to supervise the system. The software will reduce the time that it takes to test a unit which consists of creating, running, and documenting tests that the software will accomplish. This minimization of testing time will bring products to the market faster which will improve customer satisfaction. The software will minimize human error during manual validation by focusing all testing activities through the system. The system will automate documentation of test cases by recording test states (Pass/Fail) and test results which will minimize human error in running and documenting tests.

1.2 Scope

Product Name

Test User Interface, Test manager, Test logger/recorder

Overview

- Test User Interface
 - Get commands from the user.
 - Get a range of accepted values from user.
 - Display test states (Pass/Fail) and property readings.
- Test Manager
 - Runs tests or iterations of tests on the test unit.
 - Tests will be on an individual property or all properties.
 - Collect readings from physical instruments.
 - Validate that readings are within or not within the range of accepted values.
- Test Logger/Recorder
 - Records test cases, test states, and physical readings.

Goals

The software will achieve business goals of minimizing resources used in manual testing and getting to the product market faster by hosting all testing activities through the system. The software will minimize human error by allowing the human user to have the smallest part possible in testing. The proposed system aims to reduce the expertise and time needed to test a product such that users with minimal technical experience can interact with the system. The minimization of resources parallels with the company's desire for growth by cutting costs and increasing profit.

- The time it takes to validate a test unit will decrease by at least 20% by three months after the system has been developed.
- The time that it takes for a product to reach the market will decrease by 10% by three months after the system has been developed.
- The cost of resources in automated testing 1 unit will be 30% smaller than the cost of manual testing for 1 unit by Jan 1, 2023.

Out of Scope

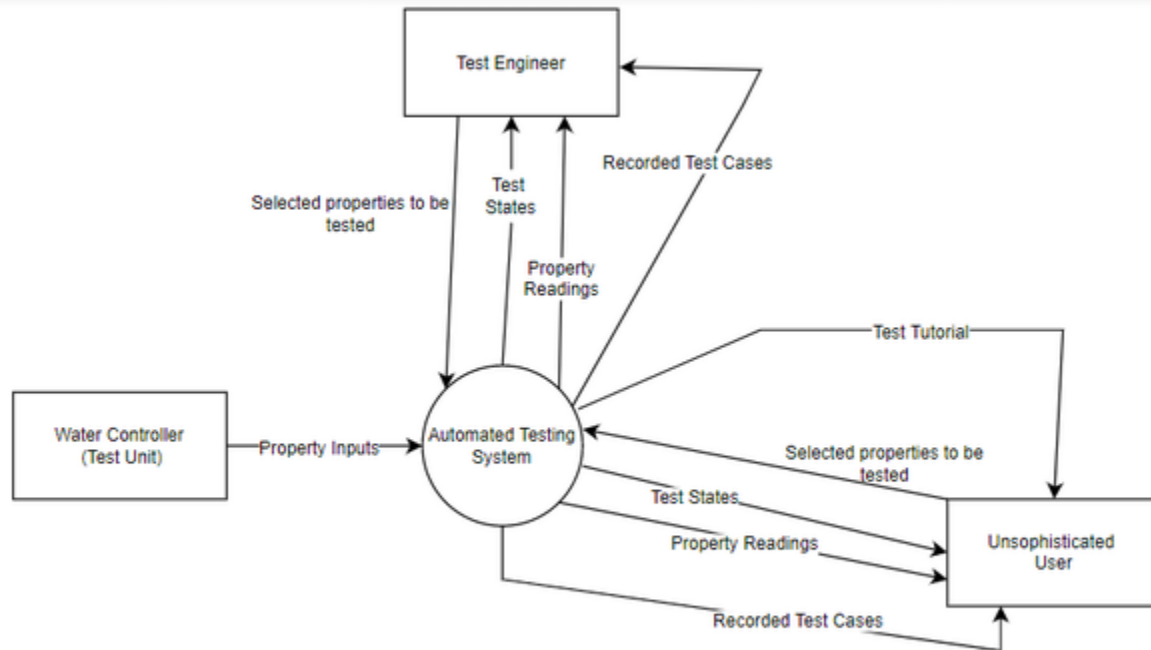
- The solution will not contain a database component.
- The solution may send the records of tests through a third-party system in a later iteration.
- The solution will not distinguish between what type of user is operating the system through a login feature.
- The solution will not be used for other types of test units beyond controllers.

- The solution may test for other properties, but instruments that collect data on these properties will have to be integrated into the system for a later iteration.
- The solution will not test multiple units at the same time.

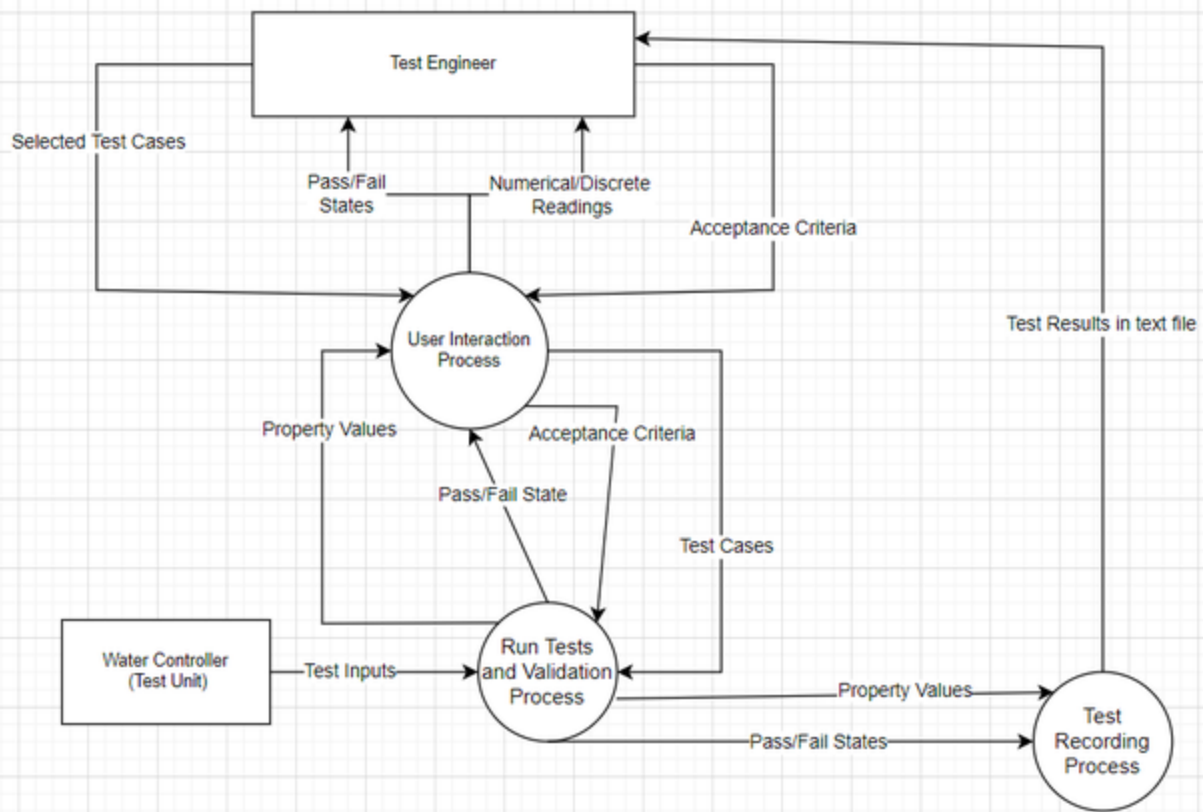
1.3 Product Overview

1.3.1 Product Perspective

Context Diagram/Data Flow Diagram Level 0



Data Flow Diagram Level 1



Key	Summary	T	Created	Updated	Due	Assignee	Reporter	P	Status	Resolution
PULSAF DR-55	Testing documentation provided by the system must be approved by development team		05/Apr /22 2:24 PM	08/Apr /22 4:33 PM		Unassigned	Christian Apostoli		TO DO	Unresolved
PULSAF DR-54	There must be a unique identifier for each test case		05/Apr /22 2:12 PM	11/Apr /22 6:39 PM		Unassigned	Christian Apostoli		TO DO	Unresolved
PULSAF DR-53	The system must show a property name, property value, and pass/fail state for each test case		05/Apr /22 2:12 PM	11/Apr /22 6:39 PM		Unassigned	Christian Apostoli		TO DO	Unresolved
PULSAF DR-52	All GUI programming must be done with the Qt framework		05/Apr /22 2:10 PM	05/Apr /22 2:10 PM		Unassigned	Christian Apostoli		TO DO	Unresolved
PULSAF DR-51	The system must be compatible with the Windows 10 operating system		05/Apr /22 2:07 PM	05/Apr /22 2:07 PM		Unassigned	Christian Apostoli		TO DO	Unresolved
PULSAF DR-40	The system must show that the result of the test case is a Pass state or a Fail state		14/Mar /22 5:55 PM	11/Apr /22 6:39 PM		Unassigned	Christian Apostoli		TO DO	Unresolved
PULSAF DR-38	The system must wait to return to its initial state until the user has verified that the current test is complete		12/Mar /22 12:19 PM	11/Apr /22 6:39 PM		Unassigned	Christian Apostoli		TO DO	Unresolved
PULSAF DR-37	All embedded software must be written in C++		12/Mar /22 12:09 PM	12/Mar /22 12:09 PM		Unassigned	Christian Apostoli		TO DO	Unresolved
PULSAF DR-36	The system may only test properties that Pulsafeeder has instruments to collect readings for.		12/Mar /22 12:06 PM	11/Apr /22 6:39 PM		Unassigned	Christian Apostoli		TO DO	Unresolved
PULSAF DR-35	The system must incorporate all properties that are currently tested		12/Mar /22 12:	11/Apr /22 6:39		Unassigned	Christian Apostoli		TO DO	Unresolved

10 issues

1.3.2 Product Functions

- Display properties that can be tested on the test unit through the user interface
 - Receives property selected by the user
 - May receive all properties to test
- Allows the user to select a range of accepted values for property
 - Displays change in value as the user makes the change
 - Allows the user to save the range of values for that test iteration
- Displays if each individual test passed (was within accepted range) or failed (was not within accepted range)
 - System will logically validate if the inputs from instruments are within acceptance criteria or not
- Reads physical input from instrument probe
- Display continuous values as a numerical reading and discrete values as an (on/off) state
- Runs multiple iterations of a test at once if desired by the user
- Provide documentation that guides users to accurately test a controller

1.3.3 User Characteristics

Education Level: The education level is assumed to not have a significant impact on the user's experience with the proposed system

Experience: The experience with the system and the application domain will vary greatly from the sophisticated user and the unsophisticated user.

Disabilities: It is assumed by the stakeholders that the users will not have major physical, mental, or emotional disabilities. However, there are limitations to the skills of the sophisticated user and naive user.

- Testing personnel (sophisticated user)
 - The sophisticated user has the ability to know the expected result of the test
 - They will be familiar with the controller and the manual testing environment
 - These users will be familiar with what is the range of accepted values
 - These users will be familiar with what are the appropriate properties to test
 - These users will be familiar with the regulations that the product must adhere to
 - An Example of these users may be a dedicated test engineer
- Unsophisticated user (naive user)
 - It is assumed that the naive user has no background knowledge of the controller or testing process
 - They are not familiar with the controller
 - They may not know what properties to test or the range of accepted values for each property
 - This user may not accurately predict the correct result due to the lack of experience or domain knowledge
 - They may not have domain knowledge
 - It may be recommended to these users that they do not attempt to operate the proposed system without the accompaniment of an experienced test engineer
 - They may not be able to anticipate the misuse of a product by a customer
 - An example of this type of user may be employees that are new hires with limited experience or interns.

1.3.4 Limitations

Key	Summary	T	Created	Updated	Due	Assignee	Reporter	P	Status	Resolution
PULSAF DR-24	Embedded programs must be written in C++	☑	04/Mar/22 9:31 PM	04/Mar/22 9:31 PM		Unassigned	Christian Apostoli	==	TO DO	Unresolved
PULSAF DR-23	The system can not add new properties unless the team has access to instruments for those properties.	☑	04/Mar/22 8:06 PM	08/Apr/22 4:33 PM		Unassigned	Christian Apostoli	==	TO DO	Unresolved
PULSAF DR-22	The system must read inputs from controllers that are developed by Pulsafeeder	☑	28/Feb/22 3:57 PM	08/Apr/22 4:33 PM		Unassigned	Christian Apostoli	==	TO DO	Unresolved
PULSAF DR-21	The system's functionality must be open to all users	☑	28/Feb/22 3:56 PM	08/Apr/22 4:33 PM		Unassigned	Christian Apostoli	==	TO DO	Unresolved
PULSAF DR-20	The system cannot go down for maintenance during business hours	☑	28/Feb/22 3:54	08/Apr/22 4:33		Unassigned	Christian Apostoli	==	TO DO	Unresolved

5 issues

1.4 Definitions

Core values, trust, team, excellence Look on the website

Technology - key to delivering responsible products to the market that we serve

Innovation - finding a new approach for an old problem

Diversity - offering a full range of products to address a wide variety of customer requirements

Excellence -to extend far beyond the competition

Controller – a device that manages water level to regulate water flow






























Property - characteristic of the controller that is importance to pulsafeeder, and/or can be chained to some numerical value

2. References

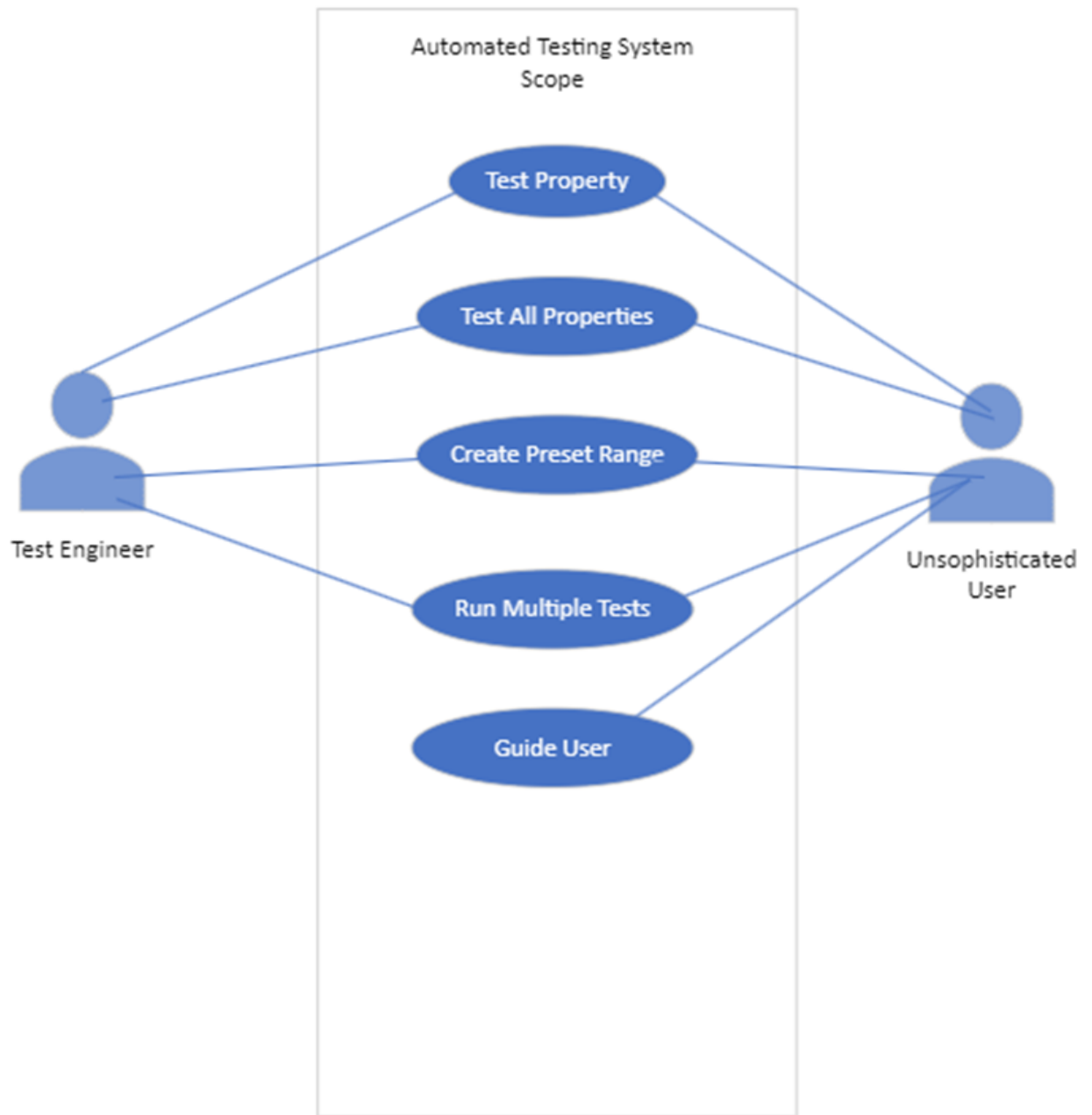
1. <https://www.pulsatron.com/overview>
2. Moises Martinez, Junior Software Developer at Pulsafeeder
3. Fabrice Pryor, Manager of New Product Development & Controllers at Pulsafeeder
4. Michael Smith, Senior Product Manager at Pulsafeeder
5. <https://www.idexcorp.com/about-idex/the-idex-story/>
6. <https://www.epa.gov/ground-water-and-drinking-water/national-primary-drinking-water-regulations#one>
7. <https://www.nsf.org/standards-development/standards-portfolio/water-wastewater-standards>
8. Sample_Test_Case_Pulsafeeder.xlsx (Not attached for confidentiality reasons)

3. Requirements

Key	Summary	Description	T	Linke d Issues	P	Labels
PULSAF DR-56	When a user selects the option to test all properties, the system shall gather range of accepted values and number of test iterations for each all test properties.	When the system tests all properties, the system shall iterate through each property and replicate the testing process for a single property. This includes the input of a range of accepted values and the number of iterations to test.	☰		==	functiona l, product
PULSAF DR-50	When the user is finished seeing documentation, the system shall bring the user back to the initial state	The user may access documentation to learn how to use the system to successfully test a controller. Once they are finished with documentation, they can come back to the initial state to begin testing.	☰		^	functiona l, product
PULSAF DR-49	The system shall allow the user to select how many iterations to perform a single test (ranging from 1 to 99)		☰	PULSAF DR-14	v	functiona l, product
PULSAF DR-48	When the user selects the option to test all properties, the system shall collect input values from all property instruments	Once the user selects the option to test all properties, the system must collect readings on all properties to run the tests for each.	☰		v	functiona l, product
PULSAF DR-47	Logged test cases, within the text file created by the system, shall create a unique identifier for each test case	Test cases shall have a unique identifier so they can be identified properly. This helps in keeping accurate records and visiting previous test cases.	☰	PULSAF DR-11	^	Interface, non- functiona l, product
PULSAF DR-46	Logged test cases, within the text file created by the system, shall show a property name, property input value, and a pass/fail state	Test cases should show a property name, the reading value, and the pass/fail state as these have been deemed as the minimum amount of information to analyze a test case.	☰	PULSAF DR-11	==	Interface, functiona l, product
PULSAF DR-45	When tests are verified as complete, the system shall allow the test unit to be disconnected from the system		☰	PULSAF DR-13 , PULSAF DR-29	==	functiona l, product

PULSAF DR-44	The system shall provide the same set of features for all users such that any user can use the system for testing			PULSAF DR-18		non-functional, product, usability
PULSAF DR-43	When a test has finished, the system shall display non-discrete input values as numerical values			PULSAF DR-17		non-functional, product
PULSAF DR-42	The system shall offer documentation on how to use the system before tests are run			PULSAF DR-15		non-functional, product, usability
PULSAF DR-41	When the user has selected the property/properties to test, the system shall allow the user to enter a range of accepted values			PULSAF DR-9		functional, product
PULSAF DR-34	When the system has collected inputs from the test unit, the system shall verify if the input is within the user's range of accepted values	The system shall run a test/s on the property selected by the user and determine test results. The system shall determine if the inputs are within the user's range of accepted values or not within the range of accepted values.				functional, product
PULSAF DR-33	When the user has selected a property, the system shall collect input from the instrument corresponding to that property.			PULSAF DR-13		Interface, functional, product
PULSAF DR-32	The system shall offer the user the option to run an individual test a set number of iterations.			PULSAF DR-14		non-functional, product, usability
PULSAF DR-31	When the user has verified that a test is complete, the system shall go back to its initial state	Once the test/s is finished, the user will be presented the option to complete the test notifying the system that the user is satisfied. At this point, the system shall return to its initial state such that the system can continue its functionality.				functional, product
PULSAF DR-30	When a test has finished, the system shall display discrete readings as ON or OFF states depending on a properties input value.			PULSAF DR-16		non-functional, product
PULSAF DR-29	The system shall allow a test unit to be connected to the system			PULSAF DR-45 , PULSAF DR-13		functional, product
PULSAF DR-28	When a test has finished, the system shall present the user the option to complete the current test.	The system shall present the user option to complete the test notifying the system that the test is complete. Once a test is complete, the system may go to its initial state.				functional, product
PULSAF DR-27	When a test has finished, the system shall display the test results as a Pass or Fail state.			PULSAF DR-10		functional, product
PULSAF DR-26	The system shall display the option to test all properties that are possible within the system's available instruments	Use Cases_4_2_Christian_Apostoli.docx  This requirement is related to use case PFUSECASE-2				functional, product
PULSAF DR-25	The system shall display all test unit properties that can be tested within the instruments that the system has available			PULSAF DR-8		Interface, functional, product

21 issues



Use Cases

ID:	PFUSECASE-1
Title:	Test Property
Description:	User connects the test unit to run tests on an individual property of the test unit.
Preconditions:	The test unit has been manufactured and approved for testing. The system has an instrument to read physical inputs for that property.
Postconditions:	Tests have been run on the individual property for the test unit. The results of the test and their numerical/discrete values have been recorded. The result of the test is shown as pass or fail.

Basic Flow	<ol style="list-style-type: none"> 1. User connects the test unit to the system 2. User selects a property to test 3. System reads in physical inputs from a probe for the selected property 4. User selects range of accepted values for property 5. User approves range of accepted values 6. System designates if value from reading is within range of accepted values 7. System displays if test is a pass or fail. System displays the value from the reading.
Alternate Flow:	<p>A1: System fails at reading physical input</p> <p>A2: System fails at determining if the physical input is within range of accepted values</p> <p>A3: Property does not exist within system's available properties</p> <p>A4: System fails at displaying test state and result</p>
Actors:	Test engineer, unsophisticated user.
Goals:	User is able to test an individual property if a test needs to be redone or if test was never done for specific property.

ID:	PFUSECASE-2
Title:	Test All Properties
Description:	User connects the test unit to run tests on all properties of the test unit within the possible properties that the system can test.
Preconditions:	The test unit has been manufactured and approved for testing.
Postconditions:	Tests have been run on all properties for the test unit. The results of each test and their numerical/discrete values have been recorded. The results of each test are shown to be a pass or fail.
Basic Flow	<ol style="list-style-type: none"> 1. User connects the test unit to the system 2. User selects to test all properties 3. System reads in physical inputs from a probe for all properties 4. User selects range of accepted values for each property 5. User approves range of accepted values for each property 6. System designates if value from reading is within range of accepted values 7. System displays if test is a pass or fail for each property. System displays the value from the reading for each property.

Alternate Flow:	<p>A1: System fails at reading physical input for any of the properties</p> <p>A2: System fails at determining if the physical input is within range of accepted values for any of the properties</p> <p>A3: System fails at displaying test state/results for any of the properties tested</p>
Actors:	Test engineer, unsophisticated user.
Goals:	User is able to test and record values for all properties of the test unit. User does not have to run tests for all individual properties.

ID:	PFUSECASE-3
Title:	Create Preset Range
Description:	User creates a set range of accepted values for an individual property. The user shall select the preset in later tests
Preconditions:	No preconditions currently known for this test case
Postconditions:	The preset range is able to be selected. The preset range populates the range of accepted values that were selected by the user.
Basic Flow	<ol style="list-style-type: none"> 1. User selects to create a preset 2. User selects a property that corresponds to the preset 3. User assigns a name to the preset 4. User selects the range for the preset
Alternate Flow:	<p>A1. System fails at assigning the range</p> <p>A2. System fails at giving the preset a name</p>
Actors:	Test engineer, unsophisticated user.
Goals:	User will be able to quickly select the preset for a test on a property.

ID:	PFUSECASE-4
Title:	Run Multiple Tests
Description:	User runs an integral number (1 or higher) of tests on an individual property or all properties.
Preconditions:	The test unit has been manufactured and approved for testing.
Postconditions:	Tests have been run on the individual property for the test unit. The results of the test and their numerical/discrete values have been recorded.

Basic Flow	<ol style="list-style-type: none"> 1. User connects the test unit to the system 2. User selects the amount of iterations for tests to be ran 3. User selects a property to test 4. System reads in physical inputs from a probe for the selected property 5. User selects range of accepted values for property 6. User approves range of accepted values 7. System designates if value from reading is within range of accepted values 8. System displays if test is a pass or fail. System displays the value from the reading.
Alternate Flow:	<p>A1: User selects a number that is larger than the max amount of tests allowed</p> <p>A2: User selects a negative number of tests</p> <p>A3: System fails at reading physical input for any of the test iterations</p> <p>A4: System fails at determining if the physical input is within range of accepted values for any of the test iterations</p> <p>A5: System fails at displaying test state and result for any of the test iterations</p>
Actors:	Test engineer, unsophisticated user.
Goals:	Users will have the ability to run tests multiple times at once. This can increase the accuracy of tests.

ID:	PFUSECASE-5
Title:	Guide User
Description:	System guides the user through testing a controller with detailed steps.
Preconditions:	The test unit has been manufactured and approved for testing.
Postconditions:	Tests have been run on the individual property for the test unit. The results of the test and their numerical/discrete values have been recorded.
Basic Flow	<ol style="list-style-type: none"> 1. User connects the test unit to the system 2. User selects to run guided tests 3. System will show steps to test an individual property 4. System proceeds with step 4 of Test Property use case

Alternate Flow:	<p>A1: User selects a number that is larger than the max amount of tests allowed</p> <p>A2: User selects a negative number of tests</p> <p>A3: System fails at reading physical input for the property</p> <p>A4: System fails at determining if the physical input is within range of accepted values for any of the test iteration</p> <p>A5: System fails at displaying test state and result for any of the test iterations</p>
Actors:	Unsophisticated user.
Goals:	Users can test a controller accurately without necessary testing experience or domain knowledge

4. Verification

Key	Summary	Verification Approach
PULSAFD R-56	When a user selects the option to test all properties, the system shall gather range of accepted values and number of test iterations for each all test properties.	Demonstration
PULSAFD R-50	When the user is finished seeing documentation, the system shall bring the user back to the initial state	Demonstration
PULSAFD R-49	The system shall allow the user to select how many iterations to perform a single test (ranging from 1 to 99)	Test
PULSAFD R-48	When the user selects the option to test all properties, the system shall collect input values from all property instruments	Test
PULSAFD R-47	Logged test cases, within the text file created by the system, shall create a unique identifier for each test case	Inspection
PULSAFD R-46	Logged test cases, within the text file created by the system, shall show a property name, property input value, and a pass /fail state	Inspection
PULSAFD R-45	When tests are verified as complete, the system shall allow the test unit to be disconnected from the system	Test
PULSAFD R-44	The system shall provide the same set of features for all users such that any user can use the system for testing	Analysis
PULSAFD R-43	When a test has finished, the system shall display non-discrete input values as numerical values	Test
PULSAFD R-42	The system shall offer documentation on how to use the system before tests are run	Inspection
PULSAFD R-41	When the user has selected the property/properties to test, the system shall allow the user to enter a range of accepted values	Inspection
PULSAFD R-34	When the system has collected inputs from the test unit, the system shall verify if the input is within the user's range of accepted values	Test
PULSAFD R-33	When the user has selected a property, the system shall collect input from the instrument corresponding to that property.	Test
PULSAFD R-32	The system shall offer the user the option to run an individual test a set number of iterations.	Inspection
PULSAFD R-31	When the user has verified that a test is complete, the system shall go back to its initial state	Demonstration

PULSAFD R-30	When a test has finished, the system shall display discrete readings as ON or OFF states depending on a properties input value.	Test
PULSAFD R-29	The system shall allow a test unit to be connected to the system	Test
PULSAFD R-28	When a test has finished, the system shall present the user the option to complete the current test.	Demonstration
PULSAFD R-27	When a test has finished, the system shall display the test results as a Pass or Fail state.	Demonstration
PULSAFD R-26	The system shall display the option to test all properties that are possible within the system's available instruments	Analysis
PULSAFD R-25	The system shall display all test unit properties that can be tested within the instruments that the system has available	Analysis

[21 issues](#)

5. Appendices

5.1 Assumptions and dependencies

- The automated testing system is to test controllers solely made by Pulsafeeder.
- The system will test properties that are tested in manual testing.
- The system will use instruments for reading physical inputs that Pulsafeeder already has access to.
- The system will not be accessed by entities outside of Pulsafeeder.
- Market competitors would benefit from a system that automates the manual testing of controllers.
- The unsophisticated user has little to no experience with testing a controller.
- Both types of users have the ability to make errors in the testing process.

5.2 Acronyms and abbreviations

IDEX (NYSE: IEX) - Innovation, Diversity, and Excellence. The core values of the parent corporation.

GUI - Graphical User Interface. A form of user interface that allows users to interact with graphical icons.

API - Application Programming Interface. A software component that allows two applications to communicate with each other.

UI - User Interface. The space where the user and the device interact. This can include all text, buttons, images and other elements of the system.

EPA - United States Environmental Protection Agency

NPDWR - National Primary Drinking Water Regulations

MCLG - Maximum Contaminant Level Goal. The level of a contaminant in drinking water below which there is no expected health risk.

MCL - Maximum Contaminant Level. The highest level of a containment that is allowed in drinking water.

MRDLG - Maximum Residual Disinfectant Level Goal. The level of a drinking water disinfectant below which there is no expected health risk.

TT - Treatment Technique. A required process intended to reduce the level of a contaminant

MRDL - Maximum Residual Disinfectant Level. The highest level of a disinfectant allowed in drinking water.

NSF - National Science Foundation

ANSI - American National Standards Institute

LED - Light Emitting Diode