System Verification and Validation Plan
Formulate

Team 25, MECHTRON 4TB6
Ahmed Nazir, nazira1
Stephen Oh, ohs9
Muhanad Sada, sadam
Tioluwalayomi Babayeju, babayejt

April 5, 2023

1 Revision History

Date	Developer	Notes
10/30/22	Stephen	Added General Information and Plan
04/03/23	Ahmed	Final Revision
04/04/23	Muhanad	Grammer fixes, added reflection
04/05/23	All	Revision 1

Contents

2 General Information

2.1 Summary

Formulate consists of four subsystems, one hardware subsystem and three software subsystems, which interact to provide the user with a testing device designed to eliminate automatable processes in common testing procedures.

A physical data collection device is the hardware subsystem used as the first point of contact with the measured quantity through a sensor. The sensor obtains physical quantities for the device to buffer, before sending the data to a desktop application for user verification.

The user has the ability to view the data collected by the physical device after a completed test using a desktop application software subsystem. The desktop application enables the user to either accept the test results to then store a collection of data from a test to a database, or reject the test and prevent the test from being stored to a database.

Accepted test data sent from the desktop application aggregates and saves verified test results to a database software subsystem. Users can query the database to obtain common statistics in test data and generate new or obscure relationships by leveraging database language capabilities.

A final dashboard software subsystem then queries the database to visualize key performance indicators on the test data collected and stored in the database.

2.2 Objectives

The objective of the system Verification and Validation (VnV) plan for Formulate is to ensure the intended project qualities are present.

Ease in user understanding is a quality Formulate will achieve to support the system's usability. Specifically, ease in user understanding of each subsystem's function and how subsystems interface will be key qualities of the overall project.

The system will also demonstrate the quality of adequate portability and physical robustness to support system maintainability, portability, and operationality.

2.3 Relevant Documentation

This document references a variety of requirements generated during the Software Requirements Specification (SRS) process and the Hazard Analysis (HA) process for the Formulate system.

3 Plan

3.1 Roadmap

The intention of testing for Formulate is to generate confidence that the project meets qualities relating to usability, maintainability, portability, operationality, and safety set out as requirements in SRS and HA documentation. Through sets of unit and system tests that will prove if the system has met the above requirements, Formulate will understand if the project has achieved the desired qualities.

Specifically, requirements that are functional, non-functional, and safety-security related from the SRS and HA documents will be referenced in the Plan, System Test Description, and Unit Test Description sections of this document.

3.2 Verification and Validation Team

Name	**Role**	**Explanation**
Stephen	Desktop Application Tester	VnV for software application design and integration with hardware and database
Ahmed	Hardware Device Tester	VnV for embedded program design, chassis design, and integration with desktop application
Muhanad	Database Application Tester	VnV for database design and integration with desktop application and dashboard
Tioluwalayomi	Dashboard Application Tester	VnV for dashboard application design and integration with database
Timofey	Project and Course Teaching Assistant	Detailed low level feedback on planned VnV tests
Dr. Smith	Course Instructor	General high level feedback on planned VnV tests

3.3 SRS Verification Plan

SRS verification will be composed of two approaches to verify that functional and non-functional requirements are met. The first approach is engaging in read through's of the SRS document each month. Individual member progress will be evaluated against the relevenat sections(s) of the SRS document to ensure system development is on track to meet the requirements. The second approach is evaluating issues created by classmates on GitHub and incorporating their concerns and suggestions as seen fit.

Stephen and Tioluwalayomi will lead the group wide discussion for SRS verification activities on the first Tuesday of each month.

3.4 Design Verification Plan

Design verification will be composed of two approaches. The first planned approach is completing read through's of each individual's high level design documentation for their respective sub-system. The design documentation covered during these read through's will likely entail mechanical, electrical, and or software schematics or diagrams outlining the architecture of the subsystem. Members will voice concerns during the read through of design decisions made by the individual responsible for the sub-system architecture. The second planned approach is evaluating issues created by classmates on GitHub and incorporating their concerns and suggestions as seen fit.

Ahmed will lead the group wide discussion for design verification activities on the second Tuesday of each month.

3.5 Implementation Verification Plan

Implementation verification will be composed of techniques in both static and dynamic analysis.

Content walkthrough is the primary type of static implementation verification technique the group plans on using. Three similar types of content walkthrough's are planned for use depending on the sub-system under analysis. Software implementation's will receive a code walkthrough, mechanical implementation's will receive a Computer Aided Design (CAD) spin, and electrical implementation's will receive a schematic walkthrough. During the content walkthrough, unit and system tests relevant to the implementation will be considered to critique the quality of the implementation. A meeting will be organized for each content walkthrough once the implementation has reached a notable milestone worthwile for group analysis.

Live execution of the implementation using a proof of concept style demonstration to the group is the primary type of dynamic implementation technique the group plans on using. During the live demonstration of the implementation, system and unit tests relevant to the implementation will be considered to critique the quality of the implementation. Using the initial state and inputs of the test outlined in the system and unit test sections, the quality of implementation is passed or failed depending on if the actual output of the implementation matches the expeted output.

Ahmed and Muhanad will lead the group wide discussion for design verification activities on the second Tuesday of each month.

3.6 Automated Testing and Verification Tools

Tools will not be used to automate testing, profile, or code coverage for software flashed on embedded hardware because of the available tools incur high additional overhead to testing effort and times. The group plans on completing extensive unit testing for embedded software to compensate for the absence of automated testing and coverage tools.

The desktop application will likely use Visual Studio's memory usage tool to profile the program during execution.

Code coverage will be completed manually for the desktop, database, and the dashboard programs. This decision will be feasible as the expected size of software for the applications listed above is relatively small. As a result, it is reasonable for to manually check the amount of code coverage achieved through tests.

3.7 Software Validation Plan

There are no current plans to use external data for validation.

4 System Test Description

4.1 Tests for Functional Requirements

4.1.1 Hardware Device

Sensor Validation

1. **ST-SV 1**

   Type: Manual

   Initial State: Device is on, measuring and sending values to the application, and connection to database has been verified

   Input: The device is placed in temperature controlled box at 25°C

   Output: The temperature sensor on the device should consistently read 25°C within a 5% error for ten successive readings

   How test will be performed: The device will be placed in a styrofoam insulated box and a hair dryer will be used to heat the box to 25°C. Using a digital thermometer we will compare the results of our device to the digital thermometer

2. **ST-SV 2**

   Type: Manual

   Initial State: Device is on, measuring and sending values to the application, and connection to database has been verified

   Input: The device will be placed in a humidity controlled box which will be 40% humidity

   Output: The humidity sensor on the device should consistently read 40% humidity within a 5% error for 10 successive readings

   How test will be performed: The device will be placed in a styrofoam insulated box and a hair dryer will be used to heat the box to 50°C. Using a digital humidity monitor we will compare the results of our device to the digital sensor

3. **ST-SV 3**

   Type: Manual

   Initial State: Device is on, measuring and sending values to the application, and connection to database has been verified Input: The device will be placed on a constantly vibrating platform at 1 Hertz for 10 seconds.

   Output: The accelerometer sensor should consistently read a vibration of 1 Hz within a 5% error for 10 seconds

   How test will be performed: The device will be placed on a constantly vibrating platform which is set to a known vibration. We will start a test on our device and compare the output of our accelerometer with the known vibration

Device Telemetry

1. **ST-DT 1**

   Type: Manual

   Initial State: Device is on and is connected to a PC via cable or via Wi-Fi

   Input: The start button is pressed

   Output: All readings from all sensors should be sent to the PC for 30 consecutive readings within 1 second of pressing the start button

   How test will be performed: While the device is connected to the PC the start button will be pressed to check if telemetry data is being sent to the PC. The emphasis is on the ability of data transmission and not correctness

2. **ST-DT 2**

   Type: Manual

   Initial State: Device is on and is connected to a PC via cable or via Wi-Fi

   Input: The start button is pressed 10 times consecutively

   Output: The device should continue to send data to the PC regardless of how many times the start button is pressed

   How test will be performed: While the device is connected to the PC the start button will be pressed and then after a few seconds pressed again 10 more times to check if telemetry data transmission to the PC cannot be broken by pressing the start button

3. **ST-DT 3**

   Type: Manual

   Initial State: Device is on and is connected to a PC via cable or via Wi-Fi

   Input: The stop button is pressed

   Output: All sensor readings should stop being sent to the PC within 1 second of pressing the stop button

   How test will be performed: While the device is connected to the PC the stop button will be pressed to check if telemetry data is stopped

4. **ST-DT 4**

   Type: Manual

   Initial State: Device is on and is connected to a PC via cable or via Wi-Fi

   Input: The stop button is pressed 10 times

   Output: The device should not send data to the PC regardless of how many times the stop button is pressed

   How test will be performed: While the device is connected to the PC the stop button will be pressed 10 times in an interval of 2 seconds to check if telemetry data is has stopped

Device Hardware

1. **ST-DH 1**

   Type: Manual

   Initial State: Device is on and is connected to a PC, no sensors are attached

   Input: A sensor is connected using the snap on connector and start is pressed

   Output: The device should start sending data to the PC within 1 second of pressing the start button and should be firmly attached to the device at a vibration of 1 Hertz.

   How test will be performed: While the device is connected to the PC a sensor is connected to the modular port on the device and the start button is pressed, the sensor will be shaken at 1 Hertz and it should still continue to send data

2. **ST-DH 2** Type: Manual

   Initial State: Device is on but the battery has less than 5 minutes of charge

   Input: User swaps the battery

   Output The battery should provide a full 2 hours of charge in testing time

   How test will be performed: We will continue to use the device until the battery is completely drained, once its low we will start to charge it and check if the battery increases

3. **ST-DH 3**

   Type: Manual

   Initial State: Device is fully mounted on a test platform

   Input: User will apply 49N of force on every side of the device

   Output: The fastened device should not show signs of deformation or changes in position

   Rationale: This will test the rigidity and durability of our mounting mechanism and ensure that our device can withstand the forces it will face while mounted on a Formula E vehicle.

   How test will be performed: Once the device is mounted we will place 5kg on top of different sides of the device to check if the mount can withstand it

4. **ST-DH 4**

   Type: Manual

   Initial State: Device is not mounted on anything

   Input: User will begin to mount the device on the Formula SAE car

   Output: The device should be fully installed under 5 minutes

   How test will be performed: While someone is mounting the device another person is timing them

4.1.2 Desktop Application

1. **ST-DA 1**

   Type: Manual

   Initial State: Device is on and is connected to a PC, the desktop application is open and connected to the device

   Input: User selects the retrieve test case

   Output: A table with all the raw test data should populate the screen

   Rationale: This tests if the desktop application can read the last test performed from the local storage incase connectivity was lost

   How test will be performed: After a test is completed on the device, it will be connected to the PC and the user should be able to view the most recent test after clicking retrieve

2. **ST-DA 3**

   Type: Manual

   Initial State: Device is on and is connected to a PC, the desktop application is open and test data is open

   Input: User hits the submit to database button

   Output: The test data in the table should be sent to the database

   Rationale: This will test if the desktop application can connect with the database and send data

   How test will be performed: After a test is conducted and the data will be viewed on the desktop app, the user will preview the data and click submit to database

3. **ST-DA 4**

   Type: Manual

   Initial State: Device is on and is connected to a PC, the desktop application is open but no tests are done

   Input: User hits the submit to database button

   Output: The application should alert the user that no data was recorded and that nothing was sent to the database

   How test will be performed: No tests will be performed and an empty table will be shown. Submit to database will be clicked

4. **ST-DA 5**

   Type: Manual

   Initial State: Device is on and is connected to a PC, the desktop application is open

   Input: User clicks the start test button on the application

   Output: The application should start to receive live data from the device and display it on the table

   Rationale: This test verifies that our desktop application can read live data as the sensors are outputting them

   How test will be performed: The hardware device will be plugged into the PC and once we hit the start button on the application all the sensors will display their output

4.1.3 Data Dashboard

1. **ST-DD 1**

   Type: Manual

   Initial State: The dashboard contains test data and is connected to the database and has all the visualized data of the tests.

   Input: User logs into the analytics platform and selects a test

   Output: The test data is displayed showing the results over time for that particular test

   Rationale: This tests verifies that our data dashboard will be able to be viewed by the appropriate user.

   How test will be performed: We will populate the database with different tests and the user once logged in will select a specific test to view

2. **ST-DD 2**

   Type: Manual

   Initial State: The database contains test data and is connected to the analytics platform

   Input: Unauthorized user attempts to login

   Output: The website alerts the user they are not authorized to view the dashboard

   Rationale: This tests verifies that our data dashboard will be only viewed by the appropriate users.

   How test will be performed: One team member will not have privileges to view the dashboard and will attempt to login

4.1.4 Database

1. **ST-D 1**

   Type: Manual

   Initial State: Database is active

   Input: User submits data multiple times within 5 seconds

   Output: Database stops accepting new data and alerts user

   Rationale: This test will check if it is possible to spam the database with data, it keeps the database safe from unintended purposes

   How test will be performed: After a test is completed on the device, it will be previewed using the desktop app and the submit to database button will be pressed repeatedly

4.2 Tests for Nonfunctional Requirements

System tests for nonfunctional requirements are broken down into performance and usability subsets. Performance tests are used to measure speed, modularity, responsiveness, and stability of the device, application and database. Usability tests are more user-related in the sense that they are validated through users operating the device and taking measurements. These tests allow the Formulate system to be evaluated through user's perspective.

4.2.1 Performance

1. **ST-P 1** Type: Dynamic, Manual

   Initial State: Device is on and mounted, has connected to the application and is waiting to start measuring.

   Input/Condition: Recording of measurements begin

   Output/Result: Device is operational and stays physically intact in conditions at 20% greater than threshold values and in all types of weather.

   How test will be performed: The device will be tested outdoors under various weather conditions including rain, wind, etc. The device will also be tested in temperature and vibration conditions that are above threshold values. This will be performed by placing the device in a hot environment and vigorously shaking it.

2. **ST-P 2**

   Type: Dynamic, Manual

   Initial State: Device is on and mounted, has connected to the application and is waiting to start measuring.

   Input/Condition: Recording of measurements begin

   Output/Result: Data latency should be less than 10 seconds to simulate viewing live data.

   How test will be performed: The amount of time for data to be viewable on the application will be inspected to be less than 10 seconds. The application UI will also be inspected to ensure that data is smooth and not lagging while measurements are being performed.

3. **ST-P 3**

   Type: Dynamic, Manual

   Initial State: Device measuring and sending values to the application, and connection to the database has been verified

   Input/Condition: One or two of either the device, application, or database are disconnected or turned off

   Output/Result: The other two components are still functional even though communication between them is broken.

   How test will be performed: While device, application, and database are fully functional and communicating successfully, different combinations of either one or two components will be turned off. The other component(s) will be inspected to ensure that they are operational and indicating that the other component(s) are disconnected.

4.2.2 Usability

1. **ST-U 1-A**

   Type: Dynamic, Manual

   Initial State: Device is turned off and nothing is connected, only the application is loaded on to the computer and the correct Arduino code to the test setup is flashed onto the micro-controller

   Input/Condition: Users will be asked to setup device and start taking measurements, and the rate setup process using a survey

   Output/Result: Time for setup and data to appear on the application should be less than 5 minutes and an average score of 3 should be achieved on each user's survey.

   How test will be performed: A test group will be educated on the setup and connection of the device, then they will attempt to do that process. Each person will be timed and compared to the 5 minute threshold. In addition, they will be given a survey to rate the difficulty of the setup process on a scale from 1 to 5 on the following categories: overall setup, sensor mount, device mount, start up procedure, and measuring procedure (refer to SQ1).

2. **ST-U 1-B**

   Type: Dynamic, Manual

   Initial State: Device is turned off and nothing is connected, only the application is loaded on to the computer and the Arduino code must be adjusted to the test setup is flashed onto the micro-controller

   Input/Condition: Users will be asked to setup device and start taking measurements, and the rate setup process using a survey

   Output/Result: Time for setup and data to appear on the application should be less than 5 minutes and an average score of 3 should be achieved on each user's survey.

   How test will be performed: A test group will be educated on the setup and connection of the device, then they will attempt to do that process. Each person will be timed and compared to the 5 minute threshold. In addition, they will be given a survey to rate the difficulty of the setup process on a scale from 1 to 5 on the following categories: overall setup, sensor mount, device mount, start up procedure, and measuring procedure (refer to SQ1).

3. **ST-U 2-A**

   Type: Dynamic, Manual

   Initial State: Device is given to McMaster's Formula Electric team to use with a wired connection to the laptop

   Input/Condition: Using a survey, Formula Electric members will compare their current testing process to the Formulate process

   Output/Result: All users need to select Formulate in at least 3 of the 4 categories

   How test will be performed: Formula E members will select which process is preferred in the following categories: speed, data collection, ease of use, and portability (refer to SQ2).

4. **ST-U 2-B**

   Type: Dynamic, Manual

   Initial State: Device is given to McMaster's Formula Electric team to use with a wireless connection to the laptop

   Input/Condition: Using a survey, Formula Electric members will compare their current testing process to the Formulate process

   Output/Result: All users need to select Formulate in at least 3 of the 4 categories

   How test will be performed: Formula E members will select which process is preferred in the following categories: speed, data collection, ease of use, and portability (refer to SQ2).

5. **ST-U 3-A**

   Type: Dynamic, Manual

   Initial State: Device is set up and waiting to start measuring via a wired connection, application is loaded onto the computer

   Input/Condition: Users are asked to log into the application and use the UI to observe data. The wired connection is broken by removing the physical connection. Users are asked through a survey how easy the recovery method for test data collected onto the local memory and submitting the data to the database was

   Output/Result: An average score of 3 should be achieved on each user's survey

   How test will be performed: A test group will be asked to use the application UI and rate it on the following categories on a scale from 1 to 5: login process, responsiveness, accessibility, and sending results to the database (refer to SQ3).

6. **ST-U 3-B**

   Type: Dynamic, Manual

   Initial State: Device is set up and waiting to start measuring, application is loaded onto the computer, the user is logged in

   Input/Condition: Users are asked to use the UI to observe data and send results to the database, and rate the UI using a survey when the connection between the UI and the database was broken

   Output/Result: An average score of 3 should be achieved on each user's survey

   How test will be performed: A test group will be asked to use the application UI and rate it on the following categories on a scale from 1 to 5: login process, responsiveness, accessibility, and sending results to the database (refer to SQ3).

4.2.3 Security

1. **ST-S 1**

   Type: Dynamic, Manual

   Initial State: The device will have measurements stored in the database after a test is completed.

   Input/Condition: User will input multiple usernames and passwords to try open the database and see the data that has been stored.

   Output/Result: Depending on the username and password inputed the user will either be able to see the data or be prompted with an error message.

   How test will be performed: After the user has conducted a test and wants to view their data they will be asked for a username and password to verify that they are the intended user of the device. Depending on the username and password they will either be allowed to view the data or be blocked off.

2. **ST-S 2**

   Type: Dynamic, Manual

   Initial State: The device will have measurements stored in the database after a test is completed.

   Input/Condition: Multiple fake data points will be attempted to be added by an unauthorized user.

   Output/Result: The data points taken from the device will not be affected and will remain the same regardless of attempts to alter it.

   How test will be performed: A user will attempt to change and compromise the data points that are being stored in the database. A security feature will stop this from happening unless the user is authorized to do so.

4.3 Traceability Between Test Cases and Requirements

Requirement	Test
FR 1	ST-SV 1, ST-SV 2, ST-SV 3
FR 2	ST-DT 1, ST-DT 2, ST-DT 3, ST-DT 4
FR 3	ST-DT 1, ST-DT 2
FR 4	ST-DT 3, ST-DT 4
FR 5	ST-SV 1, ST-SV 2, ST-SV 3
FR 6	ST-DA 1
FR 7	ST-DA 3, ST-DA 4
FR 8	ST-DD 1
FR 9	ST-DH 3
FR 10	ST-DH 4
FR 11	ST-DH 2
FR 12	ST-DT 1, ST-DT 2, ST-DT 3, ST-DT 4
FR 13	ST-DH 4
FR 14	ST-DA 5
FR 15	ST-DH 1, ST-DA 5
FR 16	ST-DD 2
FR 17	ST-DD 1
FR 18	ST-DA 3
NFR 1	ST-U 1-A, ST-U 1-B
NFR 2	ST-U 2-A, ST-U 2-B, ST-U 3-A, ST-U 3-B
NFR 3	ST-P 2
NFR 4	ST-P 1, ST-P 3
NFR 5	ST-U 2-A, ST-U 2-B
NFR 6	ST-P 1
NFR 7	ST-P 3
NFR 8	ST-U 2-A, ST-U 2-B
NFR 9	ST-S 1, ST-S 2

5 References

Oh, S., Nazir, A., Sada, M., amp; Babayeju, T. (n.d.). (rep.). Software Requirements Specification MECHTRON 4TB6: Formulate.

Oh, S., Nazir, A., Sada, M., amp; Babayeju, T. (n.d.). (rep.). Hazard Analysis MECHTRON 4TB6.

6 Appendix

6.1 Usability Survey Questions

SQ1 How difficult was the setup process? Rate the difficulty of the following components of the setup process on a scale from 1 to 5: overall setup, sensor mount, device mount, start up procedure, and measuring procedure.

SQ2 Which testing process do you prefer based on the following categories: speed, data collection, ease of use, and portability.

SQ3 How did you enjoy your experience using Formulate's application UI? Rate the UI on the following categories from 1 to 5: login process, responsiveness, accessibility, and sending results to the database.

6.2 Reflection

6.2.1 Knowledge Required

1. In order to ensure the success of our project, it is essential that our team enhances our understanding of how sensors operate, as we work to complete our verification and validation plan. It is imperative that we accurately capture and interpret data from the sensors, which entails mitigating any potential interference, optimizing calibration, and implementing the sensors effectively. Only by mastering these critical aspects of sensor technology can we ensure that our device functions properly and delivers optimal performance.

2. In order to successfully complete the verification and validation of the Formulate project, team members need to acquire knowledge in Structured Query Language (SQL). SQL is the standard language for creating, reading, and writing manipulating values in databases. It is important to know at least the basics of SQL to be able to enter the Formulate database and ensure that the correct test values have been submitted from the device and desktop application. In addition, other test cases require the manipulation of data to verify results which means team members need to understand which values to access and how to alter them using query language.

3. In order to complete electrical hardware verification, our team needs to acquire knowledge of how to check for electrical continuity between connected points. Electrical continuity checks are required to verify that the built electrical hardware matches that of the electrical schematic design.

4. In order to full complete our project, our team will have to acquire knowledge on the best methods to visualize data with test data coming from a database. We also required to know data analysis and have a good understanding of data, so we can create quality, structured and clear data that is visualized. In addition we are requiredto know our

target user and understand what type of data they would want to see after completing a test. Only after acquiring this knowledge we will successfully be able to finish our project.

6.2.2 Approach

1. Our team aims to elevate our sensor expertise through a twofold approach. Firstly, we plan to conduct extensive research on the workings of digital and analog sensors, utilizing a range of resources such as scholarly papers and informative YouTube videos. This will enable us to establish a firm theoretical foundation of sensor fundamentals, allowing us to comprehend and apply sensor concepts more effectively. Secondly, we will embark on a practical journey by conducting experiments on our current sensors to explore the impact of various environmental factors on their performance. By subjecting these sensors to different conditions, we can identify the optimal configuration to obtain the most accurate readings. This hands-on approach will supplement our theoretical knowledge and provide valuable insights into the real-world application of sensor technology. From our team Ahmed will be pursing this because of his previous coop background utilizing different sensors and his general interest in the area.

2. The two approaches that were considered for acquiring the required SQL knowledge included a teaching session from one of the Formulate team members who have extensive knowledge in query language and going through a SQL online course hosted on an educational programming platform such as Codeacademy. The team collectively chose the second approach since the online lessons were found to be much more suited for an educational setting as they are interactive, organized, and paced to the liking of the team member so no one would feel overwhelmed.

3. The first approach our team can make to gain the required knowledge on how to complete continuity checks is to learn from the MakerSpace coordinator on how electrical continuity checks can be completed. Since the MakerSpace coordinator's job is to help students complete their projects in McMaster's MakerSpace and is well versed in a variety of testing methods, learning from the coordinator is a good first approach to understanding how continuity checks can be completed to verify our circuit. Secondly, our group can refer to online articles that show the knowledge and the technique required to perform continuity checks on electric circuits.

4. There were two approaches our team planned to use in order to create the optimal data visuals for our Formulate project. The first approach was to understand and learn how to create visualized that a user would want to see. We did this by researching the topic, looking at examples of other people's data visualizations and went through online tutorials on how collect data from a database and create visuals from it. The second approach we used when creating our data visuals was communicating with our target user to determine exactly what they would like to see when testing, this approach was really effective cause it allowed us to create a clear and effective data visuals.