Verification and Validation Report: MECHTRON 4TB6

Team 25, Formulate
Ahmed Nazir, nazira1
Stephen Oh, ohs9
Muhanad Sada, sadam
Tioluwalayomi Babayeju, babayejt

March 8, 2023

1 Revision History

Date	Version	Notes
Date 1	1.0	Notes
Date 2	1.1	Notes

2 Symbols, Abbreviations and Acronyms

symbol	**description**
T	Test

Contents

List of Tables

List of Figures

3 Functional Requirements Evaluation

Test Number	Type	Input	Output	Result
ST-SV 1	Manual	Room at a constant 25 °C ambient	Constant 25.4 °C reading from temperature sensor	Pass
ST-SV 2	Manual	Room at constant 40% humidity °C	Constant 43% reading from humidity sensor	Pass
ST-SV 3	Manual	4 seperate phonecalls with accelerometer measuring haptic feedback	Maximum error between acceleration profiles of phone calls within 0.2 meters per second squared	Pass

Table 1: Sensor Validation

Test Number	Type	Input	Output	Result
ST-DT 1	Manual	The device is connected via Wi-Fi and the start button is pressed on the GUI	The device beings to send and display the sensor data in the GUI	Pass
ST-DT 2	Manual	The device is connected via Wi-Fi and the start button is pressed multiple times in the GUI	The start button greys out not allowing the user to press it multiple times and the sensors still send data to the GUI	Pass
ST-DT 3	Manual	The device is connected via Wi-Fi and the stop button is pressed on the GUI	The device stops sending data to the GUI	Pass
ST-DT 4	Manual	The device is connected via Wi-Fi and the stop button is pressed multiple times in the GUI	The stop button greys out not allowing the user to press it multiple times and the device does not send data	Pass

Table 2: Device Telemetry

Test Number	Type	Input	Output	Result
ST-DH 1	Manual	Temperature sensor male end JST connector unconnected to the device's female JST connector	Device measured the room's ambient temperature at 23.4 °C	Pass
ST-DH 2	Manual	Device collecting sensor data at low battery charge	Device unoperational upon battery depletion and halted sensor data collection	Fail
ST-DH 3	Manual	5 kg dumbbell placed on each corner of the device chassis	No plastic failure in device chassis	Pass
ST-DH 4	Manual	A member on the formula electric team was given a cross screwdriver, M4 screw, the device, and DIN rail	The formula electric team member mounted the device in 40 seconds	Pass

Table 3: Device Hardware

ST-DH3 failed the functional test case. The team estimated the time required to design and integrate a rechargeable battery subsystem within the project's timeline was 1.5 weeks and decided the effort was not worthwhile relative to other project objectives. As a result, the failure of this test case lead the expected operation at low battery to be adjusted. Users are now expected to replace the battery on a regular schedule to prevent a failure in test integrity due to charge depletion.

4 Nonfunctional Requirements Evaluation

4.1 Usability

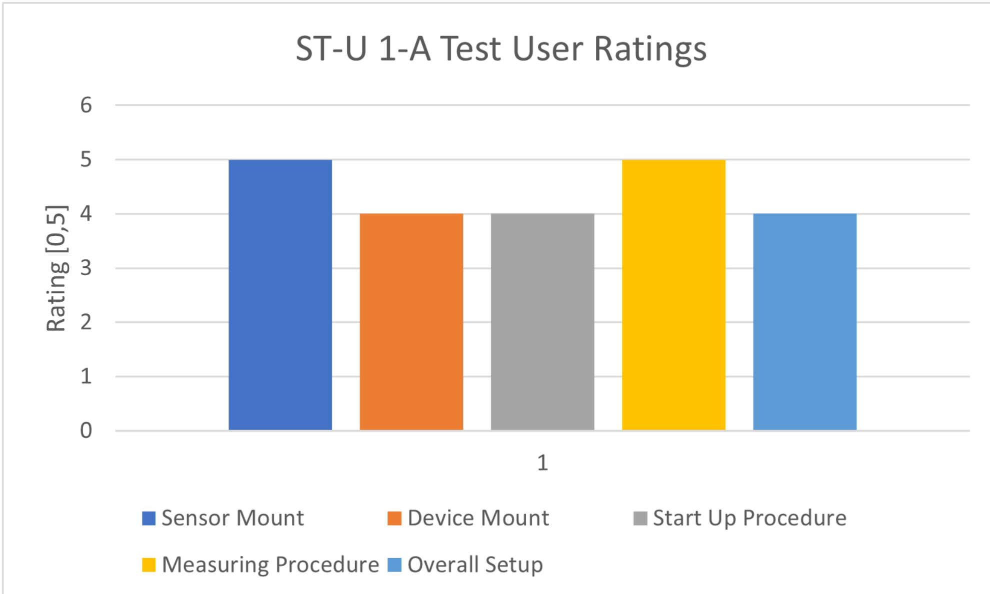

Figure 1: ST-U 1-A Test User Ratings

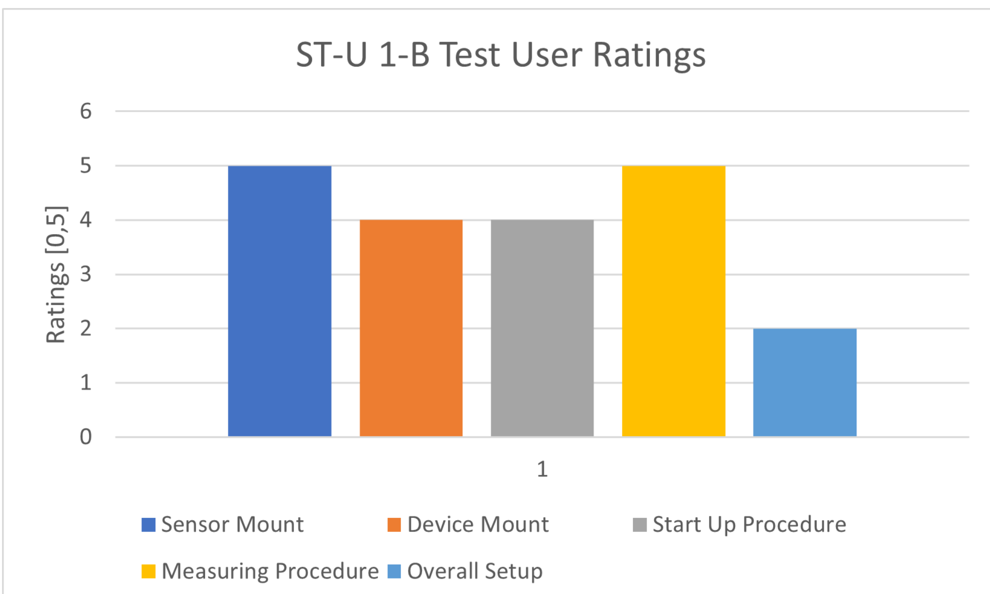

Figure 2: ST-U 1-B Test User Ratings

ST-U 1-B failed the usability test case. The difference between ST-U 1-B and ST-U 1-A was the user requirement to adjust and write the Arduino code for a sensor not previously used with the device. Notably, the user experienced difficulty and was intimidated with adjusting existing code to integrate a new sensor. This effect was shown in both the increased time to complete the overall process and the decreased rating in the overall test experience category. As a result, the project will have an approach whereby the user should only interact with the graphical user interface to reduce the user's feelings of complexity and intimidation when integrating a new sensor. The goal is to abstract the adjustments in the backend when implementing a new sensor by having the user interact only with the GUI and following guided steps in plain English to fill in the required information to integrate a new sensor.

4.2 Performance

4.3 Security

5 Unit Testing

connect wireless(self) - Tested by clicking connect button on wireless page

Test No.	Input	Expected Output	Actual Output	Result
U1	Connected to Formulate Wi-Fi	The application should connect to the device and display that it is connected	The application shows that the device is connected	Pass
U2	Not Connected to Formulate Wi-Fi	The application should display a pop up error indicating to the user that they need to connect to the Formulate Wi-Fi	A popup is displayed to the user to indicate they need to connect to Wi-Fi	Pass
U3	-Connected to Formulate Wi-Fi -Device is connected via Serial (Wired)	The application should disconnect from serial and follow U1	The application disconnects from serial and follows U1	Pass

Table 5: Unit Test

disconnect wireless(self) - Tested by clicking disconnect button on wireless page

Test No.	Input	Expected Output	Actual Output	Result
U4	-Connected to Formulate Wi-Fi -The disconnect button is clicked	The application should disconnect from the board and display that it is disconnected	The application disconnects from board and shows its disconnected	Pass
U5	Not Connected to Formulate Wi-Fi	The application should not display the disconnect button on the connectivity widget	The application does not display the disconnect button	Pass

Table 6: Unit Test

connect wired(self) - Tested by clicking connect button on wired page

Test No.	Input	Expected Output	Actual Output	Result
U6	Connected to PC via USB and the correct COM port is selected in the Wire connectivity widget	The application should connect to the device and display that it is connected	The application shows that the device is connected	Pass
U7	Not Connected to PC via wire	The application should not display any COM port in the widget	The wired drop down is empty and shows no COM port	Pass
U8	-Connected to PC via USB and the correct COM port is selected in the Wire connectivity widget -Device is connected via WiFi	The application should disconnect from WiFi and follow U6	The application disconnects from WiFi and follows U6	Pass

Table 7: Unit Test

disconnect wired(self) - Tested by clicking disconnect button on wired page

Test No.	Input	Expected Output	Actual Output	Result
U9	-Connected via wired -The disconnect button is clicked	The application should disconnect from the board and display that it is disconnected	The application disconnects from board and shows its disconnected	Pass

Table 8: Unit Test

ping(self) - Tested by clicking connect button

Test No.	Input	Expected Output	Actual Output	Result
U10	Connected via WiFi	The application should read which sensors are flashed on the board and display them	The application gets which sensors are flashed and displays them	Pass
U11	Connected via wired	The application should read which sensors are flashed on the board and display them	The application gets which sensors are flashed and displays them	Pass

Table 9: Unit Test

startTest(self) - Tested by clicking startTest button

Test No.	Input	Expected Output	Actual Output	Result
U12	Connected via WiFi	The application should read data from the bytestring sent from the ESP8266 and display it in the table as the data is coming	The application reads the data and displays it correctly	Pass
U13	Connected via wired	The application should read data from the bytestring sent from the Arduino UNO and display it in the table as the data is coming	The application reads the data and displays it correctly	Pass

Table 10: Unit Test

stopTest(self) - Tested by clicking stopTest button

Test No.	Input	Expected Output	Actual Output	Result
U14	Connected via WiFi	The application should stop reading data sent from the ESP8266	The application stops reading the data	Pass
U15	Connected via wired	The application should stop reading data sent from the Arduino UNO	The application stops reading the data	Pass

Table 11: Unit Test

declineData(self) - Tested by clicking stopTest button

Test No.	Input	Expected Output	Actual Output	Result
U16	There is data populated in the table	The application should clear all the data in the table and display an empty table	The application clears the data and displays an empty table	Pass
U17	There is no data populated in the table	The application should clear all the data in the table and display an empty table	The application clears the data and displays an empty table	Pass

Table 12: Unit Test

6 Changes Due to Testing

6.1 Functional Requirements

FR 11 was changed such that the battery under expected operational use is non-rechargeable. Batteries will continue to be used but will change from rechargeable to non-rechargeable batteries with a scheduled replacement timeline close to charge depletion. This change followed the test ST-DH 2 results.

6.2 Nonfunctional Requirements

Additional considerations to the GUI must be made in response to NFR1, NFR2, and the test result ST-U 1-B. In particular, the team plans to improve the GUI to improve the user experience by minimizing and ultimately eliminating interaction with Arduino code when integrating new sensors and making the user work with the GUI to integrate new sensors.

7 Trace to Requirements

8 Trace to Modules

References

Appendix — Reflection

The information in this section will be used to evaluate the team members on the graduate attribute of Reflection. Please answer the following question:

1. In what ways was the Verification and Validation (VnV) Plan different from the activities that were actually conducted for VnV? If there were differences, what changes required the modification in the plan? Why did these changes occur? Would you be able to anticipate these changes in future projects? If there weren't any differences, how was your team able to clearly predict a feasible amount of effort and the right tasks needed to build the evidence that demonstrates the required quality? (It is expected that most teams will have had to deviate from their original VnV Plan.)

   The Verification and Validation Plan for usability differed from the actual activities conducted in the VnV. For example, the usability test ST-U 1 did not originally account sensor code integration by the user during the test. But during VnV, the team realized that we needed to know both the total process time taken when the user already had the required code and when the user had to integrate the code to adequately determe the project's ability to meet the FR's and NFR's defined in the SRS. In the future, prototyping earlier in the design process can help identify important test cases when generating the VnV Plan.

   For the most part, our VnV plan was closely aligned with the actual VnV activities we conducted. During the VnV planning session, our team worked diligently to clearly outline the specific objectives and requirements for our final product. We conducted extensive research, and actively engaged with Formula E team members to gain a deeper understanding of the type of device they were looking for. This helped us to develop a comprehensive VnV plan that was well-tailored to the specific needs and expectations of the project. While there were some slight changes to the plan as we progressed, the core elements remained largely unchanged. This was largely due to the thoroughness of our planning process and our ability to anticipate potential challenges or modifications that might arise over the course of the project. Ultimately, this allowed us to execute our VnV activities with precision and confidence, and to demonstrate the high level of quality that was required for the final product.

Test Number	Type	Input	Output	Result
ST-U 1-A	Manual	User connects a thermistor to device, begins collecting sensor test data and gathers data for 1 minute, completes collecting sensor test data, adds remarks to the test, and submits test data to database	User completed the overall process in 3 minutes 43 seconds and rated the sensor mount and measuring procedure a 5. All other categories were given a 4	Pass
ST-U 1-B	Manual	User adjusts Arduino code for a fluid flow rate sensor, connects the sensor to the device, begins collecting sensor test data and gathers data for 1 minute, completes collecting sensor test data, adds remarks to the test, and submits test data to database	User completed the process in 49 minutes 15 seconds and rated the overall experience a 2. The sensor mount and measuring procedure categories were given a 5. All other cattegories were given a 4	Fail
ST-U 2-A	Manual	Device collecting sensor data with wired connection to a laptop running the python application	Python application GUI displayed at minimal latency. Less than 1 second latency for changes in sensor measurements to display on GUI	Pass
ST-U 2-B	Manual	Device collecting sensor data with wireless connection to a laptop running the python application	Python application GUI displayed at minimal latency. Less than 1 second latency for changes in sensor measurements to display on GUI	Pass
ST-U 3-A	Manual	Device connection to application is broken. User submits previous test data held on the micro-SD module to the database	Test data submits and test contents can be viewed on dashboard	Pass
ST-U 3-B	Manual	Application connection to database is broken. User can connect a temperature sensor, start a test, and stop a test	Device recognizes sensor, starts test, and stops test	Pass

Table 4: Usability