# Verification and Validation Report: 4TB6 - Mechatronics Capstone

Team #5, Locked & Loaded
Abi Nevo, nevoa
Elsa Bassi, bassie
Steffi Ralph, ralphs1
Abdul Iqbal, iqbala18
Stephen De Jong, dejons1
Anthony Shenouda, shenoa2

March 8, 2023

# 1 Revision History

| Date | Developer(s) | Notes |
|---|---|---|
| 04-03-23 | Abi, Anthony, Elsa, Steffi | Testing |
| 05-03-23 | Abi, Elsa, Steffi | Writeup |

## 2 Symbols, Abbreviations and Acronyms

Refer to Section 1 of the SRS documentation for a full reference section on units, symbols and abbreviations/acronyms.

# Contents



# List of Tables

# List of Figures

This document will follow the Verification and Validation Plan that was outlined to prove that the SmartLock device is a successful product. This document will provide the results/data that correspond with the given tests. At the completion of this document, we will have information regarding future changes for improvements and functionality as well as what requirements are already met.

# 3 Functional Requirements Evaluation

This subset of tests were used to validate the functional requirements of our product. Completing these tests proves various aspects of our product's needs. These aspects include primary SmartLock functionality namely App, electrical circuit and mechanical locking features. Note that each test references, and is directly mapped to, at least one requirement, showing that these test cases robustly cover the defined requirements in the SRS.

### 3.0.1 Area of Testing: User Input Related

1. DisengageLock:

   FR1: LockDisengage input must disengage the lock on the bike.

   Expected Results: The lock will successfully disengage upon receiving the DisengageLock Signal.

   Actual Result: Pass – The lock successfully disengaged upon receiving the DisengageLock Signal.

2. LockLocation:

   FR2: Location (coordinates) of user's phone must be able to be saved in the smartphone application as UserPosition.

   Expected Results: The App places a geotag within ten metres of the user's current location when prompted.

   Actual Result: Pass – The App successfully places a geotag within ten metres of the user's current location when the associated screen button is pressed.

3. EffectiveLock

FR3: Effective Bike Lock: The lock is sturdy and cannot be manually opened by the average human once engaged.

Expected Results: Perform the following user tests and achieve an average score at or above 90%. Optional scores of 1-4 for the following cases: a fail if the lock disengages and breaks, a fail if the lock disengages, a pass if the lock stays engaged but breaks, and a pass if the lock system can stay engaged without breaking.

Actual Results:

Three test subjects' scores from 1-4 out of 50 trials:

| Score | Subject 1 | Subject 2 | Subject 3 |
|---|---|---|---|
| 1 | 0 | 0 | 0 |
| 2 | 0 | 0 | 1 |
| 3 | 1 | 2 | 3 |
| 4 | 49 | 48 | 46 |

Figure 1: EffectiveLockChart

Therefore, result of pass due to average engagement without breaking of 95% of the time when the test force was applied.

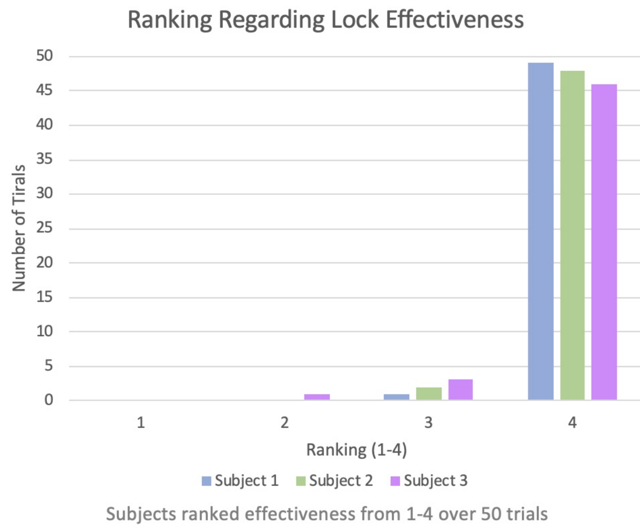


Figure 2: EffectiveLockGraph

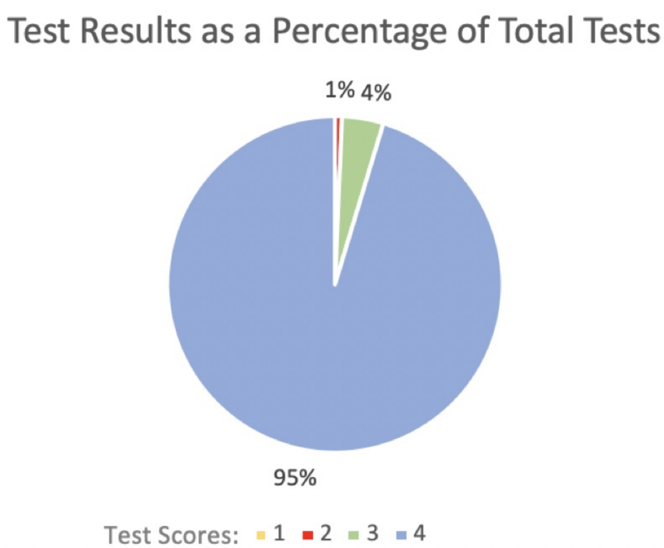


Figure 3: EffectiveLock Percentage of Tests Resulting in a result of 4

4. EffectiveLockSimulation

   FR3: Effective Bike Lock: The lock is sturdy and cannot be manually opened by the average human once engaged.

   Expected Results: The CAD simulation meets the 200-400 N threshold.

   Actual Results: Pass – The CAD simulation is able to meet the 200-400 N threshold.

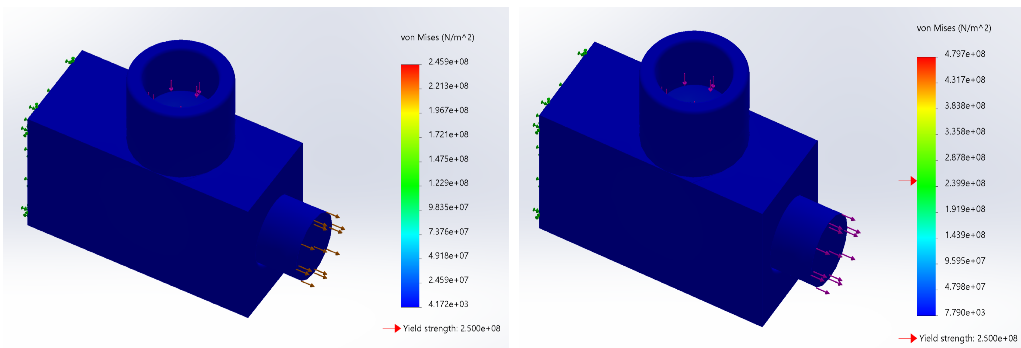


Figure 4: Stress Results with 200N & 400N Applied Force

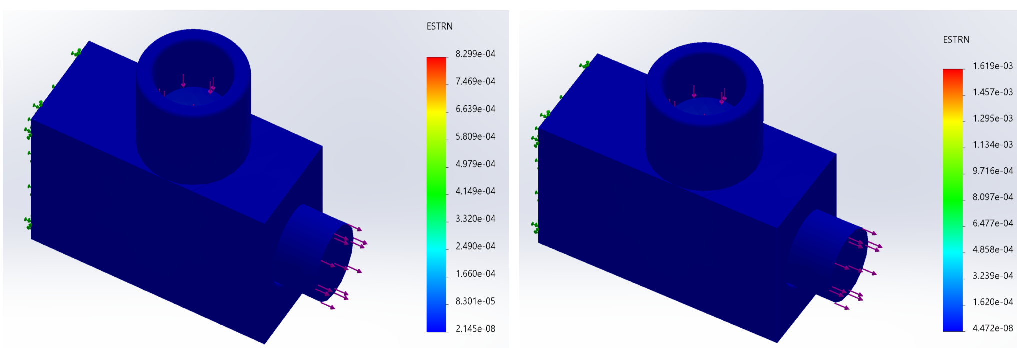


Figure 5: Stress Results with 200N & 400N Applied Force

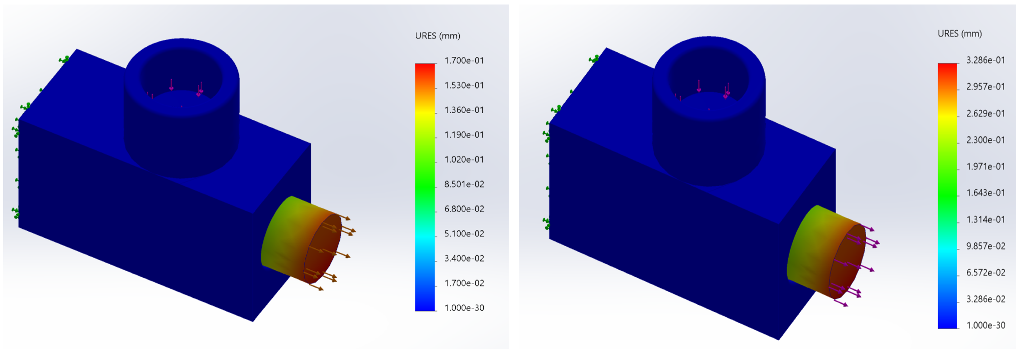


Figure 6: Stress Results with 200N & 400N Applied Force

5. Security

   FR4: Lock must only be engaged/disengaged by the intended user(s).

   Expected Results: A Bluetooth developer app is downloaded on an unintended user's phone. The unintended user will be able to connect to the Arduino but not be able to disengage the lock.

   Actual Results: Pass – The unintended user was able to connect to the device but not able to unlock the device.

#### 3.0.2 Area of Testing: Bike Input Related

1. LockMount

   FR5: The lock can be mounted to the bike's frame.

   Expected Results: A pass/fail per bike tested if the lock is/isn't able to be mounted onto the bike as intended. A score from 0-the number of bikes tested will be given.

   Actual Results: Three test subjects' results for the number of bikes where successful lock mounting occurred:

   1 – 3/3

   2 – 3/3

   3 – 3/3

   Therefore, of nine bikes tested, all passed.

### 3.0.3 Area of Testing: Output Related

1. BatteryLevel

   FR6: Battery level must be shown on the phone app.

   Expected Results: The user must be able to view the battery level of the lock.

   Actual Results: Pass – The user is able to view the battery level of the lock on the App. It is represented as four levels.

2. LocationOnApp

   FR7: Location (coordinates) of the bike must be shown on the app as BikePosition.

   Expected Results: The user must be able to view the saved coordinates.

   Actual Results: Pass – The user is able to access and view the saved coordinates of the lock on the App.

3. PowerOutput

   FR8: Battery must output enough power to engage the lock.

   Expected Results: Circuit connected. Battery is successfully able to meet the threshold voltage of the electromagnet to engage the locking mechanism.

   Actual Results: Pass – The battery successfully supplied enough voltage to engage the locking mechanism in 5/5 tests.

   Note: During one test where the Arduino was powered on for more than an hour, the battery was supplying the correct voltage, however, the Arduino became defective, (likely due to a current spike), and the locking mechanism was not engaged successfully. Moreover, it was determined that the failed test was due to the Arduino frying, so it was not counted as a failed PowerOutput test. Our learnings from this result are mentioned below in Changes Due to Testing.

# 4 Nonfunctional Requirements Evaluation

This subset of tests will be used to validate the nonfunctional requirements of our product. Completing these tests will prove various aspects of our product's functionality. These aspects include smartphone app features, the physical design attributes, accuracy, and the usability of the product. Note that each test references, and is directly mapped to, at least one requirement, showing that these test cases robustly cover the defined requirements in the SRS.

#### 4.0.1 Area of Testing: Smart Phone

1. LimitedInstructions

   NFR1: Can reasonably be used without requiring an instruction manual.

   Expected Results: The lock is successfully engaged and disengaged, and the bike is securely locked by all test users.

   Actual Results: Three test users were able to successfully engage and disengage the lock without instruction. The bikes were then tested to be securely locked by attempting to unlock them manually, (disengage the locking mechanism), with an average human's arm strength.

2. AppStorage

   NFR2: App storage under 50 megabytes. A small mobile app should not take up significant space on the user's phone.

   Expected Results: App storage is less than 50 MB.

   Actual Results: Pass – App storage does not exceed 50 MB.

#### 4.0.2 Area of Testing: Physical Lock

1. VisualAppeal

   NFR3: The design must be visually appealing.

   Expected Results: Survey users on their opinions of the visual appeal of the device; see User Testing, question 6. A majority of surveyed users should deem the visual appeal satisfactory.

   Actual Results: The majority of surveyed users deemed the visual appeal of the SmartLock satisfactory or above expectations.

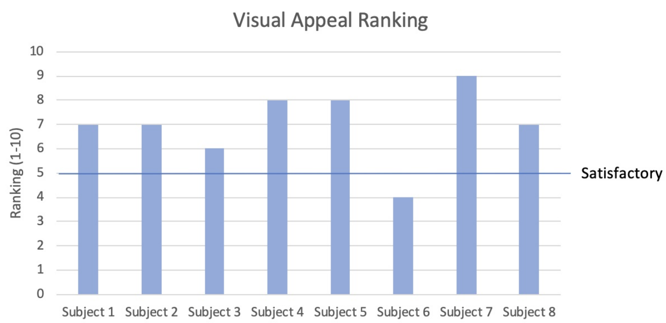


Figure 7: Visual Appeal Ranking

2. NormalBikeFunction

   NFR4: The lock must not impede normal bike functions.

   Expected Results: The SmartLock does not impede normal bike functionality and operation.

   Actual Results: Pass – The SmartLock does not impede normal bike functionality and operation.

3. Safety

   NFR5: The design must not inflict harm to the user in any way, such as clamping down on a finger, or moving at a force or speed that could cause injury.

   Expected Results: The design must not inflict harm to the user in any way, such as clamping down on a finger, or moving at a force or speed that could cause injury.

   Actual Results: Pass – User is not harmed or pinched when SmartLock is in use.

#### 4.0.3 Area of Testing: Accuracy

1. BatteryAccuracy

   NFR6: Battery level must be calculated accurately within 25% since it uses four battery levels.

Expected Results: The number of lock engages possible with 1% battery matches our specification for the number of lock engages possible with 100% battery (multiply the measured # of lock engages possible with 1% battery by 100), within 25% accuracy.

Actual Results: The number of lock engages possible with 1% and 100% battery level was manually tested and compared to the calculated number. It was within the required 25% accuracy.

#### 4.0.4 Area of Testing: Usability

1. QuickLock

   NFR7: The SmartLock must be quicker to use than a typical keyed or combination bike lock.

   Expected Results: The SmartLock must be quicker to use than both a sample keyed and combination lock. One test user will attempt three times each to lock and unlock typical keyed and combination as well as the SmartLock. All tests must be faster for the SmartLock.

   Actual Results:

| | Keyed lock Time: | Keyed Unlock Time: | Combo Lock Time: | Combo Unlock Time: | SmartLock Lock Time: | SmartLock Unlock Time: |
|---|---|---|---|---|---|---|
| Test 1 | 28.75 | 24.52 | 15.6 | 18.21 | 12.31 | 11.36 |
| Test 2 | 26.65 | 20.43 | 18.45 | 18.63 | 11.34 | 11.41 |
| Test 3 | 30 | 21.75 | 16.3 | 19.2 | 11.61 | 12.26 |

Figure 8: QuickLockChart

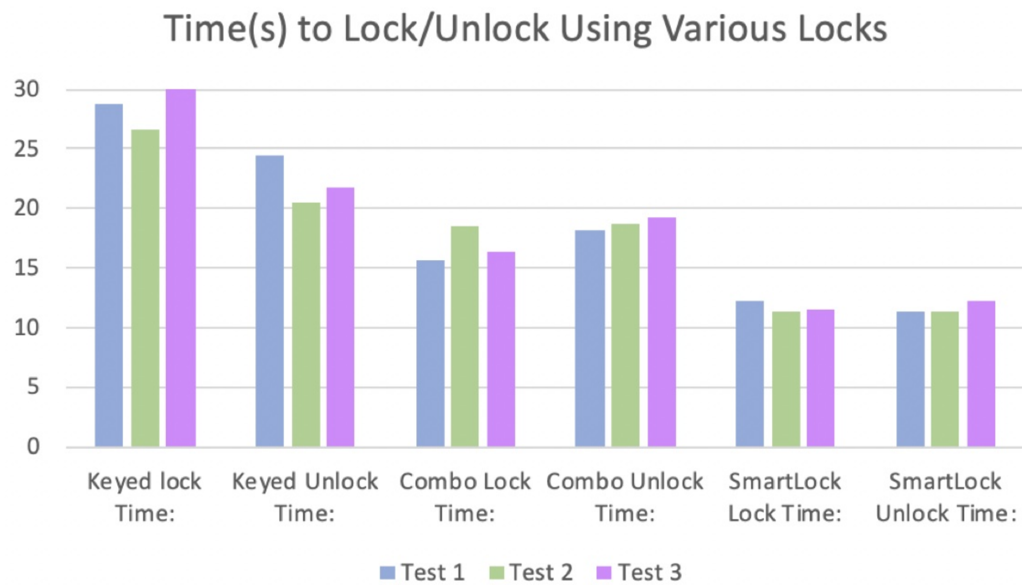


Figure 9: QuickLockGraph

2. UseForce

   NFR8: Opening and closing lock must require similar force to a typical keyed/combo.

   Expected Results: The amount of force required to open and close the lock frame is comparable to that of a typical keyed or combination lock. Test users score the force required out of 5 where 5 is the maximum amount of force they can physically provide and 1 is the minimum amount of force they can physically provide. The SmartLock should not vary more than 1 score unit from the most different score in each test.

   Actual Results: Three tests subjects' results for the amount of force required to open and close the lock frame for the SmartLock, a typical keyed lock and a combination lock respectively out of 5:

   1- 2, 1, 2

   2- 3, 3, 3

   3- 1, 1, 2

Therefore, the SmartLock did not vary more than one score unit from the most different score in each test.

3. BatteryLife

   NFR9: Battery must last for greater than 1 month and/or 60 rides before needing to be replaced or charged.

   Expected Results: The battery lasts for longer than 1 month and/or 60 rides before needing to be replaced or charged.

   Actual Results: Pass – The battery lasted longer than 1 month before needing to be replaced or charged.

4. ComponentAccessibility

   NFR10: Batteries and other internal components must be accessible to replace and/or chargeable.

   Expected Results: The battery can be replaced, and internal components can be accessed, as intended.

   Actual Results: Pass – Three test users were able to access and replace the battery and all other internal components.

5. NoSpecialTools

   NFR11: The lock must be easily mounted on the bike frame. It does not require special tools, (i.e., those not found in a typical toolbox, such as power tools), to be installed and does not take more than twenty minutes to install.

   Expected Results: The lock is easily mounted on the bike frame. It does not require special tools, (i.e., those not found in a typical toolbox, such as power tools), to be installed and does not take more than twenty minutes to install.

   Actual Results: Three test subjects' results for three individual tests for the time to mount the lock on their bike frame without the use of special tools.

   1 – 6, 5, 4 min

   2 – 7, 7, 4 min

   3 – 5, 3, 3 min

Therefore, all tests were successful and completed within twenty minutes.

6. BikeVersatility

   NFR12: The lock can be used for many different models of mountain, city, and road bikes.

   Expected Results: The SmartLock can be mounted on three different categories of bikes (road, hybrid and mountain) successfully by each test user.

   Actual Results: Three test subjects' results for the number of bikes where successful lock mounting occurred:

   1 – 3/3

   2 – 3/3

   3 – 3/3

   Therefore, of nine bikes from three different types of bikes (road, hybrid and mountain) tested, all passed.

7. AppOS

   NFR17: The App should run on iOS and Android.

   Expected Results: App operates on iOS and Android.

   Actual Results: Pass – App operates successfully on iOS and Android as expected.

# 5 Unit Testing

## 5.1 Physical Locking Mechanism

1. When small pin in "locked" position, large pin cannot be pulled out.

   Result: PASS

## 5.2 Microcontroller

1. When Arduino receives signal from Bluetooth development app "nRF Connect", on-board LED turns on.

   PASS

2. When Arduino receives signal from Bluetooth development app ”nRF Connect”, high voltage signal is sent to desired pin.

   PASS – validated using multimeter.

### 5.3 Circuit

1. When high power signal is transmitted to gate of transistor from power source (NOT Arduino), LED connected to drain of transistor turns on.

   PASS

2. When high power signal is transmitted to gate of transistor from power source (NOT Arduino), solenoid connected to drain of transistor is enabled.

   PASS

3. When high power signal is transmitted to gate of transistor from Arduino, solenoid connected to drain of transistor is enabled.

   PASS

### 5.4 Mobile App

1. Able to connect to Arduino from mobile app.

   PASS

2. Arduino recieves signals transmitted to it from the mobile app.

   PASS – validated using the on-board Arduino LEDs.

## 6 User Testing

Eight users were asked to try the SmartLock, and were asked several questions about their experience. These questions can be found in the VnV Plan, and are also reiterated here, with the common answers we received.

1. How long does it take you to open the lock? Is a 15-second window too short? Too long?

15 seconds was found to be just slightly too long of an unlock window. It was found that 10 seconds was enough time to pull the pin out of the body of the lock. This change will be implemented.

2. Is the app intuitive to use? Would you change any features or any part of the app design?

   Many users commented on the confusing nature of listing all active Bluetooth devices in the area, rather than only the SmartLock. In order to limit this confusion, the list of Bluetooth devices will be filtered so that only the SmartLock appears.

3. How did you like the whole product use experience? Is the locking/unlocking process smooth, convenient, and efficient?

   Especially due to the current cold weather, users enjoyed not needing to fiddle with a key or combination lock, and just being able to press a button to unlock their bike.

4. Was any aspect of using the product confusing?

   Again, users commented on the listing of all Bluetooth devices as being confusing. Users also mentioned that it was not clear that the lock would only be disengaged for a limited period of time after pressing "Unlock". This concept must be clarified either within the app, or in a seperate instruction manual.

5. Would you be comfortable using this product to lock your bike? What would make you feel more secure?

   Due to all of the internal components currently being not secured properly, and the solenoid being exposed to the outside, most users said they were not comfortable using the SmartLock. In order to make the lock more secure, the housing could be redesigned to hold all of the components internally.

6. Rate this device's visual appeal on a scale of 1-10, with 10 being the most visually appealing, and 1 being the least visually appealing.

   Most users were satisfied with the visual appeal of the App and the SmartLock's physical features. The results of this survey can be found in the Visual Appeal Ranking graph.

# 7 Changes Due to Testing

Changes due to Requirement Testing are summarized below:

1. Add a diode across the electromagnet in order to protect the transistor from the inductive voltage spike which occurs as the solenoid de-energizes.

2. Add a moderately-sized gate resistor between the gate of the MOSFET and the Arduino to protect the Arduino from current spikes.

Changes due to User Testing are summarized below:

1. Change the lock disengaged window (how long the lock is disengaged after pressing "Unlock") to 10 seconds from 15 seconds.

2. Filter the Bluetooth devices shown so that only SmartLock appears. Alternatively, make the mobile app only able to connect to the SmartLock.

3. Redesign the lock housing so that all components fit internally properly and securely.

## 8 Trace to Requirements

Table 1: Requirements Traceability Table

| Test Case | Functional Requirement(s) | Non-Functional Requirement(s) |
|---|---|---|
| DisengageLock | FR1 | |
| LockLocation | FR2 | |
| EffectiveLock, EffectiveLockSimulation | FR3 | |
| Security | FR4 | |
| LockMount | FR5 | |
| BatteryLevel | FR6 | |
| LocationOnApp | FR7 | |
| PowerOutput | FR8 | |
| LimitedInstructions | | NFR1 |
| AppStorage | | NFR2 |
| VisualAppeal | | NFR3 |
| NormalBikeFunction | | NFR4 |
| Safety | | NFR5 |
| BatteryAccuracy | | NFR6 |
| QuickLock | | NFR7 |
| UseForce | | NFR8 |
| BatteryLike | | NFR9 |
| ComponentAccessibility | | NFR10 |
| NoSpecialTools | | NFR11 |
| BikeVersatility | | NFR12 |
| AppOS | | NFR17 |

Note: NFR 13-16 and 18 were deemed out of scope on the SRS, and thus are not present in this Verification and Validation Plan.

# 9 Trace to Modules

Table 2: Modules Traceability Table

| Test Case | Module(s) |
|---|---|
| DisengageLock | M12 |
| LockLocation | M1, M4, M8, M9 |
| EffectiveLock, EffectiveLockSimulation | M1, M2, M8, M9 |
| Security | M9, M10, M11, M12 |
| LockMount | M2, M5 |
| BatteryLevel | M12 |
| LocationOnApp | M1, M3, M6, M8, M9 |
| PowerOutput | M1, M3, M5, M7, M8 |
| LimitedInstructions | N/A |
| AppStorage | M1, M2, M3, M4, M5, M6, M8 |
| VisualAppeal | M2, M3 |
| NormalBikeFunction | M3 |
| Safety | M3 |
| BatteryAccuracy | M6, M9 |
| QuickLock | M1, M2, M12 |
| UseForce | M10, M11, M12 |
| BatteryLife | M6, M9 |
| ComponentAccessibility | N/A |
| NoSpecialTools | N/A |
| BikeVersatility | M2, M5 |
| AppOS | M1 |

Note: NFR 13-16 and 18 were deemed out of scope on the SRS, and thus are not present in this Verification and Validation Plan.

# 10 Appendix

## 10.1 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 VnVPlan was different than the actual VnV because several of the tests were initially hoped to be broader, however the group decided to utilize our time more effectively by making the tests more concise but still useful. In other words–we had to limit our scope. For example, instead of conducting surveys over 50 or 100 people, we limited our user testing to less than 10. We overestimated how much time we would have to complete testing, and were not thinking about completing the tests realistically.

   Some of our tests were also not extremely founded or based in reality. For example, we said in the VnV Plan we would test the durability of the product by dropping/kicking/hitting it, but in reality, we obviously don't want to accidentally break our project in doing so. Instead, we used a CAD simulation to show how strong the SmartLock is, and didn't have to physically try to break the product.

   In future projects, when creating the VnV plan, it would be helpful to realistically think about what your team can accomplish in your time frame, and with the current resources you have access to. If you say, "apply a force of XX N", take the time to think through how you will apply this force, how you will know if the correct amount of force is being applied, and also why this is the specific amount of force necessary to test. Further, think about the consequences if the test fails–if you try to test the limits of your product and it breaks, what happens then? Is there an alternative simulation/test that can be completed instead?