Software Requirements Specification for 4TB6 - Mechatronics Capstone: Smart Bike Lock

Abi Nevo
Elsa Bassi
Steffi Ralph
Abdul Iqbal
Stephen De Jong
Anthony Shenouda

October 5, 2022

Contents

Revision History

Date	Version	Name	Notes
02-10-22	1.0	Elsa	Drafted first draft
03-10-22	1.1	Stephen	Drafted 1 & 5, and formatting updates
03-10-22	1.1	Abi	Drafted Likely and Unlikely Changes
04-10-22	1.2	Steffi	Drafted Appendix
04-10-22	1.3	Abi	Drafted Dev Plan and added Reflection
04-10-22	1.3	Steffi	Added Reflections and Section 3 Removed unnecessary subsections from section4

1 Reference Material

This section records information for easy reference.

1.1 Table of Units

Throughout this document SI (Système International d'Unités) is employed as the unit system. In addition to the basic units, several derived units are used as described below. For each unit, the symbol is given followed by a description of the unit and the SI name.

symbol	**unit**	**SI**
m	length	metre
kg	mass	kilogram
s	time	second
J	energy	joule
P or W	power	watt ($\mathrm{W} = \mathrm{J\,s^{-1}}$)
A or I	current	ampere
Ω or R	resistance	ohm
V	voltage	volt
N	force	newton
NM	torque	newton meter

1.2 Table of Symbols

The table that follows summarizes the symbols used in this document along with their units. The choice of symbols was made to be consistent with the heat transfer literature and with existing documentation for solar water heating systems. The symbols are listed in alphabetical order.

symbol	**unit**	**description**
B_L	hour	battery lifein hours
B_C	mAH	battery capacity in amp hours
L_C	A/actuation	current drawn per motor actuation
A_{num}	actuation/charge	number of actuations per charge

1.3 Abbreviations and Acronyms

symbol	description
SRS	Software Requirements Specification

2 Introduction

The purpose of the SmartLock project is to design and build a product that will provide bicycle users with a safer, easier, and more accessible way to secure their bike through their smartphone. Additionally, it will provide users with a GPS feature to locate the lock in case of bike theft or misplacement. It will consist of a physical lock that mounts to a bike and a smartphone application that will function as the user interface through which the lock can be engaged and disengaged wirelessly, as well as located. The project will provide an engineering solution using wireless communication, mechanical design, and smartphone application development. More broadly, it seeks to encourage members of society to pursue biking, in both a transportation and recreational capacity, improving the health of society's citizens and its environment.

2.1 Purpose of Document

The Requirements Documentation seeks to provide a complete overview of the requirements of the Smart Lock system so as to define the project. The assumptions, inputs and outputs will also be defined in order to outline the problem. Several models will be developed and lastly, likely and unlikely changes will be described. This will communicate a unified and documented plan for the project that can be used after the design stage to verify its functionality and provide an important benchmark.

2.2 Scope of Requirements

The SmartLock project will build off existing designs to tie together principles of wireless communication, GPS, a mechanical locking mechanism, electrical actuation, and smartphone application development. The system will provide all functions necessary for effective and secure locking and location.

The SmartLock project will be designed to perform the following functions for the user.

1. Wireless communication from the smartphone to the lock and vice versa.
2. Display of lock and battery status information on the app.
3. Display of location on the app.
4. Store and use a replaceable battery.

5. House a mechanical lock frame.
6. Perform electrical engagement/disengagement of a locking mechanism.
7. Be waterproof.

Not included in the project scope will be Bluetooth wireless connective capabilities, a fully autonomous locking mechanism (one that will both open/close and engage/disengage the lock wirelessly) and a rechargeable battery. We will also be ignoring extreme weather and temperature conditions and their impact on the material properties.

2.3 Characteristics of Intended Reader

The reader will have basic knowledge of how bikes are secured in public settings, namely using mechanical locks and keys or combination locks. They will understand why bike locking is necessary and what it means for a bike to be vulnerable to theft, i.e. that all main parts of the bike must be secured including the frame and wheels. They will also have basic knowledge of wireless communications that can be accessed through smartphone applications. Lastly, the reader will have a high-school level understanding of kinematic and electronics-related physics and math such that they can grasp the mechanical and electrical functionality of the system.

2.4 Organization of Document

The document will follow the base template provided by the Capstone 4TB6 course GitLab. It will be organized with an increasing order of specificity, beginning with the General System Description which will include an overview of the Smart Lock project. The Specific System Description will make up the bulk of the document, describing the project in greater detail with its assumptions, inputs and outputs, goals and models. Finally, since the context of the project is previously stated, the Functional and Non-Functional Requirements will be outlined, as well as any foreseen Likely and Unlikely Changes to the project. Lastly, the Traceability Matrix is included to define the relationships between various relevant formulas, sections and information in the document and how they are used to build the project.

3 General System Description

This section provides general information about the system. It identifies the interfaces between the system and its environment describes the user characteristics and lists the system constraints.

3.1 System Context

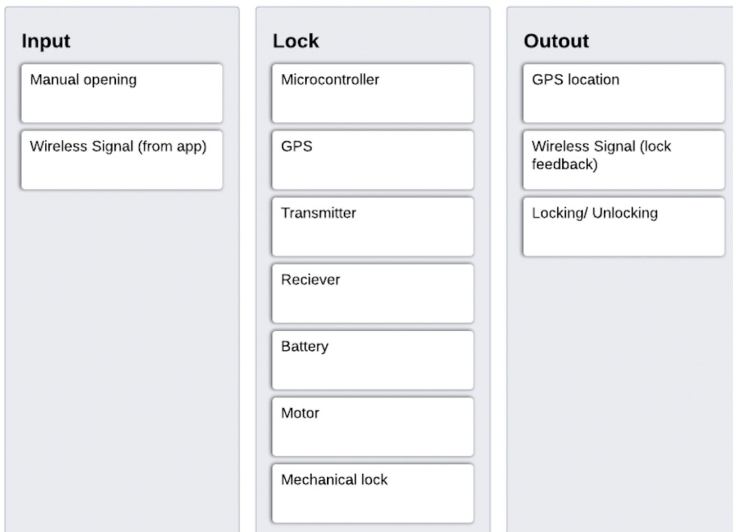

Figure 1: The System Context: Shows the inputs, the entities that manipulate them, and the outputs.

- User Responsibilities:
  - Opening and Closing the physical mechanism
  - Keeping their smartphone with them and charged to be able to use the application
  - Disengaging the lock with their smartphone
  - Leaving their bike in a location where it is possible to connect to an external frame
- 4TB6 - Mechatronics Capstone Responsibilities:
  - Develop a mechanism(s) that closes around the wheels and connects to an external frame
  - Developing a functional lock so that users can feel secure in leaving their bike
  - Developing an app that is user-friendly that disengages the lock and stores the location of the locked bike

3.2 User Characteristics

The user of the SmartLock is anyone who owns a bike and a smartphone of any age, gender or biking ability. The locking mechanism of the SmartLock functions similarly to that of any other locking device, so that no special skills are required to perform the engagement/disengagement of the lock. The physical mechanism is designed to be simple so that anyone who has the dexterity and physical strength to ride a bike will have the capability to open and close it with no strain. The application needs to be on a smartphone that supports Bluetooth v4.0 or higher (ie. any iPhone 4S or newer or any Android with OS 4.3 or newer), the user will be expected to understand how basic applications work on these devices. However, the application itself will be designed to be user-friendly and simple so that the functionality is clear and users do not have to spend lots of time learning the app, and so that the use time of the app when locking your bike is short.

3.3 System Constraints

The system has the following constraints.

- Locate the bike within a tolerance of 50 meters.
  - Justification: Complex software like Google Maps ensures accuracy of around 20 meters. Considering users only require the general location and our software will not be as complex as Google's, the team added this constraint.
- Total mass of lock under 3 lbs.
  - Justification: Average weight of a bike lock is just under 3 pounds. To ensure the lock was not significantly heavier than average the team added this constraint.
- App storage under 50 megabytes.
  - Justification: A small mobile app should not take significant space on the user's phone. Similar GPS applications are around 50 megabytes.
- The budget for testing, designing, and equipment is $750.
  - Justification: SmartLock has a limited budget to work within to encourage efficient use of materials and ensure low costs.

3.4 Normal Use Cases

In general, for a normal use case, the SmartLock system must be able to engage/disengage the SmartLock remotely through the SmartLock mobile interface after which the user must be able to open/close the SmartLock. The SmartLock must also be able to send "bike parked" location coordinates as well as current SmartLock statuses back to the mobile application.

3.4.1 Lock is Engaged/Disengaged

The lock is to be engaged/disengaged through the SmartLock mobile application. Given that the user is within range of the system, the SmartLock is designed to receive the corresponding signal request from the user's smartphone and perform the corresponding action of engaging/disengaging the lock. The system then sends the current engaged/disengaged status to the mobile application for the user to view.

3.4.2 Lock is Open/Closed

Once the request to engage/disengage the lock is sent, the system is not designed to perform the swing action of the lock. The system is designed to accept the user's manual input of swinging the lock to the desired open/closed position. The system then sends the current open/closed status to the mobile application for the user to view.

3.4.3 SmartLock location is requested

The user's smartphone's current coordinates can be requested through the mobile application by the user. When the user is near the lock, then they can tag their coordinates, for later use if they forget where their bike is locked.

4 Specific System Description

4.1 Problem Description

There are many problems associated with bike locks today. People often forget or lose their keys, lock or combination. Additionally, current locking systems are often not comprehensive – they may not lock all parts of the bike that can be stolen, namely the seat, front and back wheels, and frame.

Furthermore, bike locks can be bulky, heavy and dangerous to carry around. It can also be tedious to find and lock one's bike to an external frame. The combination of these nuisances can lead to individuals leaving their bikes without properly securing them. The city of Toronto reports an average 3625 stolen bikes annually, and the Canadian Cycling Magazine estimates that only 15-20% of stolen bikes are reported, which indicates a rather expansive problem that can be solved [1,2].

4.1.1 Terminology and Definitions

The terms listed below will be used throughout the scope of this document and should be referenced when referring to specific components.

- Smart Lock: The scope of this project and document referring to both the mechanical lock and the mobile application communicating with it.

- Lock: The physical component of the Smart Lock; includes the hardware, battery, motor, and locking mechanism.
- Team: Refers to the members working on Smart Lock
- App: The mobile application used to communicate with the lock
- Open/Close: The process of physically moving the lock frame into a state where it can be latched onto something else.
- Open/Close Status: The current state of the lock frame.
- Engage/Disengage: The automated process to actuate the lock to secure the frame.
- Wireless Signalling: The means of communication between the lock and the mobile application.
- Battery: The power source used to power the motor for engaging and disengaging.
- Battery Status: A measure of the quantity of charge in the battery.
- Git: The platform used to store the documentation and software and track changes made to the files.
- XCode and Flutter: The IDE used to develop the mobile application.
- Cross-Platform Development: The means of developing the mobile application on both iOS and Android devices.
- Transmitter: The device used to send signals to and from the app to give status information and to send the engagement and disengagement signals.
- GPS: The satellite-based navigation system used to locate the bike.
- Receiver: The device used to receive the signals from the app for engagement and disengagement.
- Microcontroller: The programmable device used to control the motor.

4.1.2 Physical System Description

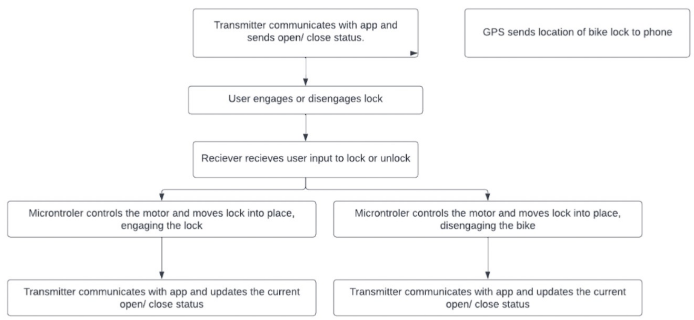

Figure 2: The Physical System

4.1.3 Goal Statements

Given the [inputs —TPLT], the goal statements are:

GS1: Effective Bike Lock: The lock is sturdy and cannot be manual opened by the average human once engaged.

GS2: Long Lasting Battery Life: The power source is used efficiently to last a sufficient period of time without requiring replacement.

GS3: Bicycle Versatility: The lock can be used on mountain, city, kids, and road bikes.

GS4: Easily Mountable on Bike Frame: Does not require special tools to be installed.

GS5: Effective GPS Tracking: The lock can accurately transmit location to app regardless of location.

GS6: Weatherproof: The lock is not easily damaged by wind, rain, snow, heat, and cold weather.

4.1.4 Assumptions

- The bike will be left in locations with strong GPS signal. Dependencies: Locations such as garages and underground parking.
- The user's phone will always be able to connect to the lock (wireless connection works as intended). Dependencies: All users have modern phones that can operate the application. The phones will also have functional location settings.

- Users will have no limiting physical disabilities. Dependencies: Average person is strong enough to manually lock and unlock the bike.
- The lock will be weatherproof. Dependencies: Rain, snow or extreme temperatures will not damage the lock.
- Battery life is only reduced when the motor is being used. Dependencies: Battery life is independent of weather and time.
- Efficiency is always constant. Dependencies: The battery's power does not decrease over time.

4.1.5 Theoretical Models

4.1.5.1 RefName:

Torque of a Motor

4.1.5.2 Label:

$\mathbf{T = F \cdot d = I \cdot k_T}$

4.1.5.3 Equation:

4.1.5.4 Description:

The above equation gives the torque T (N m) required to actuate the opening and closing mechanism of the lock through the motor. Torque is equal to the force F (N) to open or close the lock, multiplied by the distance d (m) from the motor's axis of rotation. To extract current, torque is also equal to the load (motor) current I (A) multiplied by the motor's torque Constant k_T ($\mathrm{N\,m\,A^{-1}}$).

The net force will be calculated once the design is finalized. The distance and torque constant will be dependent on the motor. These equations will be used to calculate the current required from the battery. They must be computed twice; once for engaging, and again for disengaging.

4.1.5.5 Notes:

None.

4.1.5.6 Source:

https://www.motioncontroltips.com/faq-whats-the-relationship-between-current-and-dc-motor-output-to

4.1.5.7 Ref. By:

GD??

4.1.5.8 Preconditions for Torque of a Motor:

None

4.1.5.9 Derivation for Torque of a Motor:

Not Applicable

4.1.5.10 RefName:

Ohm's Law

4.1.5.11 Label:

$V = I \cdot R$

4.1.5.12 Equation:

4.1.5.13 Description:

The above equation gives the voltage V (V), which is equal to the load (motor) current I (A) multiplied by the resistance R (Ω) of the motor. This can be used to find the rating of battery needed to power the motor. This calculation must also be done twice as outlined above.

4.1.5.14 Notes:

None.

4.1.5.15 Source:

4.1.5.16 Ref. By:

GD??

4.1.5.17 Preconditions for Ohm's Law:

None

4.1.5.18 Derivation for Ohm's Law:

Not Applicable

4.1.5.19 RefName:

Battery Life

4.1.5.20 Label:

$\mathbf{BL} = \frac{\mathbf{BC}}{\mathbf{I}}$

4.1.5.21 Equation:

4.1.5.22 Description:

The above equation gives the battery life BL (h), which will be calculated by dividing the battery capacity BC (A h) selected and the load current I (A).

4.1.5.23 Notes:

None.

4.1.5.24 Source:

4.1.5.25 Ref. By:

GD??

4.1.5.26 Preconditions for Battery Life:

None

4.1.5.27 Derivation for Battery Life:

Not Applicable

5 Requirements

This section provides the functional requirements, the business tasks that the software is expected to complete, and the nonfunctional requirements, the qualities that the software is expected to exhibit.

5.1 Functional Requirements

5.1.1 User Input Related

FR1: LockButton input must engage the lock on the bike.

Rationale: When you press the lock button to engage the lock, the bike lock must engage.

FR2: UnlockButton input must disengage the lock on the bike.

Rationale: When you press the unlock button to dissengage the lock, the bike lock must be disengaged.

FR3: When lock is engaged, change lock status to engaged.

Rationale: Lock must give correct status as engaged when engaged.

FR4: When lock is disengaged, change lock status to disengaged.

Rationale: Lock must give correct status as disengaged when disengaged.

FR5: Moving locking bar closed must move OpenClosedstatus to Closed.

Rationale: Closing the lock must result in status of lock to change to closed.

FR6: Moving locking bar open must move OpenClosedstatus to open.

FR7: Location (coordinates) of user's phone must be able to be saved in the smartphone application as UserPosition. Doing so enables memory of where the user's phone was at a specific time, such as when locking the bike, which can then later be used to determine where the bike is parked/locked.

Rationale: Doing so enables memory of where the user's phone was at a specific time, such as when locking the bike, which can then later be used to determine where the bike is parked/locked.

FR8: Effective Bike Lock: The lock is sturdy and cannot be manually opened by the average human once engaged.

Rationale: If the lock could be opened by the average human when engaged, then the lock would not prevent theft, and be rendered useless.

Verification: Apply a force between 200-400N, which is the average human's arm strength, to pry open the lock over 50 trials.

FR9: Lock must only be engaged/disengaged by the intended user(s) (I.e., not everyone with the app can engage/disengage the lock).

Rationale: Opening the lock must result in status of lock to change to open.

FR10: Location (coordinates) of user's phone must be able to be saved in the smartphone application as UserPosition.

Rationale: The iphone app needs to save the location of phone when locking for user to locate bike later.

FR11: Effective Bike Lock: The lock is sturdy and cannot be manually opened by the average human once engaged.

Rationale: If the lock is not resistance to external forces it will not be an effective lock.

FR0: Lock must only be engaged/disengaged by the intended user(s).

Rationale: Not everyone with the app can engage/disengage your lock on your bike.

5.1.2 Bike Input Related

FR13: The lock can be mounted to the bike's frame.

Rationale: If the lock can not be mounted to the users bike it will not be effective as a lock.

5.1.3 Output Related

FR14: Battery percentage must be shown on the phone app.

Rationale: User must be able to view the battery percentage of the bike lock

FR15: Location (coordinates) of bike must be shown on the app as BikePosition.

Rationale: User must be able to view the location of the bike so that teya re able to find bike.

FR16: Battery must output enough power to engage/disengage the lock.

Rationale: The locking system will not be functional if the battery is not able to disengage the lock.

5.2 Nonfunctional Requirements

5.2.1 Smart Phone

NFR1: Can be used by people of any language.

Rationale: Our lock and app must be accessible to people who can not read english.

Verification: App will be verified by test users who do not know english.

NFR2: Can reasonably be used without requiring an instruction manual.

Rationale: Our lock and app must be easy to use with a minimal learning curve for it to be useful.

Verification: App will be verified by test users, and learning period for app will be tracked

NFR3: App storage under 50 megabytes. A small mobile app should not take significant space on the user's phone.

Rationale: We want our app to fit on anyones phone, and use up minimal storage.

5.2.2 Physical Design

NFR4: The design must be visual appealing.

Rationale: Our lock must be visual appealing for it to be a desirable product.

Verification: Design willbe verified with test users who will rank visual how appealing the lock is.

NFR5: The design must not impede normal bike functions.

Rationale: Our lock will not be used if it negatively impacts the normal function of the bike.

Verification: Lock will be verified by mounting it on test users bikes, then having them use the bike.

NFR6: The lock must be waterproofed to withstand normal rainfall.

Rationale: Our lock must be able to be left in the rain without it breaking.

Verification: Lock will be verified by leaving bike with lock out in the rain.

NFR7: The lock must be waterproofed to withstand normal splashing while riding.

Rationale: Our lock must be able to be splashed without it breaking

Verification: Lock will be verified by splashing it to simulate a splash from riding through a puddle.

5.2.3 Accuracy

NFR8: Accuracy of bike lock status must be above 95%.

Rationale: The status must report the correct status in order to function optimally. The figure of 95% was taken as it is the classic confidence interval used.

Verification: The status will be manually verified over 100 trials.

NFR9: Where location services are available, the accuracy of location coordinates must be within 10M.

Rationale: In order for the locating feature to be useful, it must be accurate. The figure of 10M was obtained since a bike within a distance of 10M will be visible.

Verification: The accuracy of location coordinates will be manually verified over 50 trials.

NFR10: Battery percentage must be calculated accurately within 10%.

Rationale: In order to avoid the case where the user uses the lock and it dies while it must be locked, battery percentage must be accurate. The average user would replace their battery when it only has 10% charge left.

Verification: Observe battery charge shown right before battery dies.

5.2.4 Usability

NFR11: iPhone app locking must be quicker to use than a typical keyed/combo bike lock.

Rationale: If using the SmartLock takes longer than a typical lock, it is no longer more convenient than a typical lock, which is a key selling feature.

Verification: Time use of both the SmartLock and compare to timed use of key and combo lock by average user.

NFR12: Opening and closing lock must require similar force to a typical keyed/combo.

Rationale: If opening and closing the lock takes more strength than a typical lock, it is no longer convenient for the user.

Verification: Ask regular use of typical bike locks to use SmartLock and compare force.

NFR13: Battery must last for greater than 1 month and/or 60 rides before needing to be replaced or charged.

Rationale: It would be inconvenient to ask for the user to replace the batteries more frequently than this.

NFR14: Batteries must be accessible to replace or chargeable.

Rationale: If the batteries are not accessible to replace or charge, then when the battery dies initially, the user can no longer use the SmartLock.

NFR15: The lock must be easily mounted on the bike frame. It does not require special tools (those not found in a typical toolbox, such as power tools) to be installed and does not take more than 20 minutes.

Rationale: Required for convenience for user.

NFR16: The lock can be used for many different models of mountain, city, and road bikes.

Rationale: The SmartLock must fit a wide variety of bikes so that user's can use the SmartLock with their bike, no matter what that might be.

Verification: Test the SmartLock on as many different bikes as accessible. Ensure that this range includes all typical bike frame dimensions and shapes. For those not accessible, compare dimensions of typical bike frame/dimensions found online with dimensions/shape of SmartLock to ensure the lock can be used.

5.2.5 Maintainability

NFR17: Changes to the items below should take less than X_FRACTION of total development time.

Rationale: This rationale applies for all of the requirements in this subsection. In order for the lock to be used for a long period of time, when it breaks or needs improvements, maintenance must be able to be made.

NFR18: GUI_FRACTION, Making changes to the GUI.

NFR19: CONTROLLER_FRACTION, Motor/Battery Controller Software.

NFR20: ELCTRICAL_FRACTION, Electrical Circuit.

NFR21: MECHANICAL_FRACTION, Mechanical physical design.

5.2.6 Portability

NFR22: The app should run on iOS.

Rationale: The team's skillset is in development of iOS apps, so the requirement reflects the app only running on iOS. However, should the designer have a different skillset, an alternative requirement might be: the app should run on Android, or the app should run on iOS and Android. This requirement is constrained to only iOS due to this project's development time constraint of eight months.

NFR23: The app should be easily maintained through iOS updates.

Rationale: The app should be able to be used even when a user updates their phone, and hence should be maintained through iOS updates.

Verification: maintenance to app in preparation for iOS updates must be less than the initial development time. Maintenance will be done for every iOS update to ensure the app runs on the updated iOS.

6 Likely Changes

LC1: Depending on the battery life, showing battery percentage on the phone app, as required in FR11, might not be necessary. If the battery life is very long, then perhaps a warning that battery is low is sufficient.

LC2: Having never created an app like this before, the amount of storage needed for the smartphone application is unknown. Requirement NFR3 provides ample space, if any change, this requirement might be revised so that it be required that the app take up even less storage.

LC3: Even though aesthetics is a selling point of any product, it is not a priority for this product. As such, the requirement NFR4 is subject to change.

LC4: The degree of accuracy mentioned in NFR8, NFR9, and NFR10

LC5: Given that we will likely not be maintaining the system, requirements regarding maintability will be difficult to test and verify. While products should be designed with maintability in mind, it is not a priority for this project. Thus, requirements NFR17, NFR18, NFR19, NFR20, NFR21 are subject to change.

LC6: Requiring an iOS application as stated in NFR22 and NFR23 was chosen given the team's skillset; however, the product's nature does not require the application to be iOS based. Given different preferences and/or skillset, this application could be substituted or extended to an Android application. Note that NFR23 is dependent on NFR22

7 Unlikely Changes

ULC1: These requirements are core to the functionality of the product. The product would not accomplish it's purpose without meeting these requirements. Thus, these requirements are unlikely to change: FR1, FR2, FR3, FR4, FR5, FR6, FR7, FR8, FR9, FR13

ULC2: These requirements are key selling featueres of the system, and are therefore unlikely to change: FR10, FR11

ULC3: These requirements are necessary to ensure accessibility for all users: NFR1, NFR2

ULC4: Dependent on requirement FR10, that the lock can be mounted on the bike's frame, NFR5 that the design must not impede normal bike function is necessary for the bike to function at all, and, if the bike does not function, the SmartLock product is useless. As such, this requirement is unlikely to change.

ULC5: To maintain usability in imperfect weather conditions, requirements NFR6 and NFR7, and are therefore unlikely to change.

ULC6: The following requirements must be satisfied to maintain an edge over typical, manually engaged/disengaged bike locks, and are therefore unlikely to change: NFR11, NFR12, NFR14, NFR13, NFR15, NFR16

8 Traceability Matrices and Graphs

The purpose of the traceability matrices is to provide easy references on what has to be additionally modified if a certain component is changed. Every time a component is changed, the items in the column of that component that are marked with an "X" may have to be modified as well. Table 1 shows the dependencies of theoretical models, general definitions, data definitions, and instance models with each other. Table 2 shows the dependencies of instance models, requirements, and data constraints on each other. Table 3 shows the dependencies of theoretical models, general definitions, data definitions, instance models, and likely changes on the assumptions.

[You will have to modify these tables for your problem. —TPLT]

[The traceability matrix is not generally symmetric. If GD1 uses A1, that means that GD1's derivation or presentation requires invocation of A1. A1 does not use GD1. A1 is "used by" GD1. —TPLT]

[The traceability matrix is challenging to maintain manually. Please do your best. In the future tools (like Drasil) will make this much easier. —TPLT]

The purpose of the traceability graphs is also to provide easy references on what has to be additionally modified if a certain component is changed. The arrows in the graphs represent dependencies. The component at the tail of an arrow is depended on by the component at the head of that arrow. Therefore, if a component is changed, the components that it points

	T??	T??	T??	GD??	GD??	DD??	DD??	DD??	DD??	IM??	IM??	IM??	IM
T??													
T??			X										
T??													
GD??													
GD??	X												
DD??				X									
DD??				X									
DD??													
DD??								X					
IM??					X	X	X				X		
IM??					X		X		X	X			X
IM??		X											
IM??		X	X				X	X	X		X		

Table 1: Traceability Matrix Showing the Connections Between Items of Different Sections

	IM??	IM??	IM??	IM??	??	R??	R??
IM??		X				X	X
IM??	X			X		X	X
IM??						X	X
IM??		X				X	X
R??							
R??						X	
R??					X		
R??	X	X				X	X
R??	X						
R??		X					
R??			X				
R??				X			
R??			X	X			
R??		X					
R??		X					

Table 2: Traceability Matrix Showing the Connections Between Requirements and Instance Models

	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??
T??	X																		
T??																			
T??																			
GD??		X																	
GD??			X	X	X	X													
DD??							X	X	X										
DD??			X	X						X									
DD??																			
DD??																			
IM??											X	X		X	X	X			X
IM??												X	X			X	X	X	
IM??														X					X
IM??													X					X	
LC??				X															
LC??								X											
LC??									X										
LC??											X								
LC??												X							
LC??															X				

Table 3: Traceability Matrix Showing the Connections Between Assumptions and Other Items

to should also be changed. Figure **??** shows the dependencies of theoretical models, general definitions, data definitions, instance models, likely changes, and assumptions on each other. Figure **??** shows the dependencies of instance models, requirements, and data constraints on each other.

9 Development Plan: Prioritizing and Phasing of Requirements

9.1 Phase 1

Phase 1 consists of implementing requirements that, together, make up the minimum viable product: a lock that can be engaged remotely. These requirements are the highest priority, and should be completed first, over the course of November and December. These requirements include: FR1, FR2, FR3, FR4, FR5, FR6, FR9, NFR22

9.2 Phase 2

Phase 2 consists of requirements informing the design of the minimum viable product of the physical bike lock. These are the second highest priority, but will be completed in parallel with Phase 1 requirements. These requirements include: FR8, FR10, FR13, NFR5, NFR12, NFR15,NFR16

9.3 Phase 3

Phase 3 consists of requirements regarding the locating feature. These are lower priority, but a key selling feature. These requirements will be completed after Phase 1 and Phase 2, in January. These requirements include: FR7, FR12

9.4 Phase 4

Phase 4 consists of requirements that ensure accuracy, usability, and accessibility of features. This phase will be completed after Phase 3, likely in February. These requirements include: FR11, NFR1, NFR2, NFR3, NFR6, NFR7, NFR8, NFR9, NFR10, NFR11, NFR12, NFR13, NFR14

9.5 Phase 5

Phase 5 consists of requirements regarding the aesthetics of the product, and are lowest priority. These requirements include: NFR4

9.6 Phase 6

Phase 6 consists of requirements that ensure maintainability, which is likely outside of the scope of this project. These requirements include: NFR17, NFR18, NFR19, NFR20, NFR21

10 Values of Auxiliary Constants

[Show the values of the symbolic parameters introduced in the report. —TPLT]

[The definition of the requirements will likely call for SYMBOLIC_CONSTANTS. Their values are defined in this section for easy maintenance. —TPLT]

[The value of FRACTION, for the Maintainability NFR would be given here. —TPLT]

11 Appendix A

11.1 Table of Monitored Variables

Variable Name	Description	Type	Units
m_SignalEngaged	Monitors whether or not the locking mechanism is engaged	Digital	Boolean
m_SignalDisengaged	Monitors whether or not the locking mechanism is disengaged	Digital	Boolean
m_SignalOpen	Monitors whether or not the physical mechanism is open	Digital	Boolean
m_SignalClosed	Monitors whether or not the physical mechanism is closed	Digital	Boolean
m_Location	Monitors the location of the bike when it is locked	Analog	Coordinates
m_BatteryPower	Monitors the current battery percentage	Analog	Percentage

11.2 Table of Controlled Variables

Variable Name	Description	Type	Units
c_LockEngaged	Engages the lock	Digital	Boolean
c_LocklDisengaged	Disengages the lock	Digital	Boolean
c_LockOpen	Indicates to the user that the latch is open	Digital	Boolean
c_LockClosed	Indicates to the user that the latch is closed	Digital	Boolean
c_BikePosition	Marks the location of the bike when it is locked	Analog	Coordinates
c_BatteryPercentStatus	Indicates what the percentage of the battery is	Analog	Percentage

11.3 Constants – NA

11.4 Stimulus and Responses

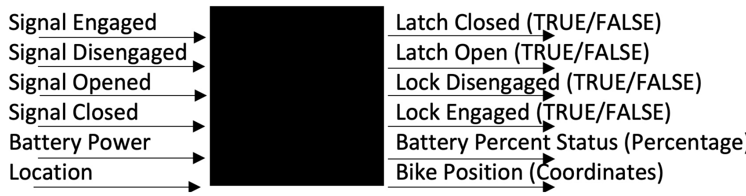

Figure 3: Stimulus and Responses

11.5 Notation

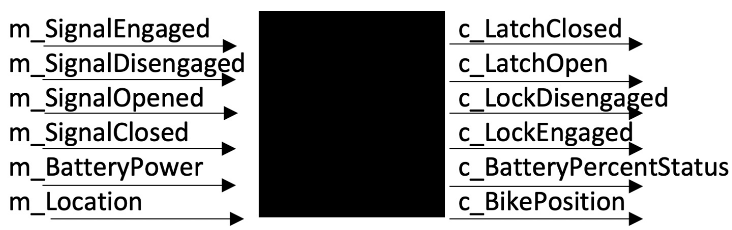

Figure 4: Notation

11.6 Responses

	c_LatchClosed	c_LatchOpen	c_LockDisengaged	c_LockEngaged	c_BatteryPercentStatus	c_BikePosition
No stimulus	FALSE	TRUE	TRUE	FALSE	No Percent	No Coordinates
m_SignalClosed	TRUE	FALSE	FALSE	FALSE	No Percent	No Coordinates
m_SignalClosed. m_SignalEngaged	TRUE	FALSE	FALSE	TRUE	No Percent	No Coordinates
m_SignalClosed. m_SignalEngaged. m_SignalDisengaged	TRUE	FALSE	TRUE	FALSE	No Percent	No Coordinates
m_SignalClosed. m_SignalEngaged. m_SignalDisengaged. m_SignalOpen	FALSE	TRUE	TRUE	FALSE	No Percent	No Coordinates
m_BatteryPower	-	-	-	-	Battery Percent Shown	No Coordinates
m_Location	-	-	-	-	No Percent	Coordinates Shown

Figure 5: Responses

11.7 Effect of Each Stimulus

Current Stimulus \ Current State	Latch open – Lock Engaged	Latch open – Lock Disengaged	Latch closed – Lock Engaged	Latch closed – Lock Disengaged
m_SignalEngaged	-	Bike ready to ride	-	Bike Secure
m_SignalDisengaged	Device ready to be closed	-	Device ready to be opened	-
m_SignalOpened	-	-	-	Bike removed from fixed frame
m_SignalClosed	-	Bike attached to fixed frame	-	-
m_Location	BikePosition	BikePosition	BikePosition	BikePosition
m_BatteryPower	BatteryPercentStatus	BatteryPercentStatus	BatteryPercentStatus	BatteryPercentStatus

Figure 6: Effect of Each Stimulus

12 Appendix B

12.1 Reflection

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

1. What knowledge and skills will the team collectively need to acquire to successfully complete this capstone project? Examples of possible knowledge to acquire include domain specific knowledge from the domain of your application, or software engineering knowledge, mechatronics knowledge or computer science knowledge. Skills may be related to technology, or writing, or presentation, or team management, etc. You should look to identify at least one item for each team member.

   Steffi:
   My lead role on this project is in regard to Documentation and Latex. While I have experience with some project documentation, the extent to which it is necessary for this project is not something I have done before. Looking at this project in such detail will help to more deeply understand every component of the project and help to better see the connection between every aspect. Additionally, technical communication is a skill I have never used before so learning to use Github and Latex will be critical to the delivery of the project and developing skills that will be useful in future technical projects.

   Abi:
   My lead role for this project is surrounding wireless communication, specifically Bluetooth. I will need to learn how to implement Bluetooth communication between the SmartLock and the user's phone so that the lock can be disengaged remotely. I have never worked with this type of technology before, so I will need to have a very steep learning curve.

   Elsa:
   I am the Embedded Systems Lead of the project, and as such I will be adding to my existing skills in order to design, build and provide support to other members of my team related to this topic. This project will require significant knowledge in the realm of electronics, wiring, microprocessor programming and general hardware systems. I am also a Support Lead of Documentation, which includes technical writing and typesetting in Latex.

   Anthony:
   The mobile application will be developed using XCode Ver.14.0, utilizing the Swift programming language for iOS Development. For linting, SwiftLint will be implemented as it is commonly used in the industry for iOS development utilizing XCode. Basic background in programming is a required skill needed to be able to effectively learn

the IDE and programming language.

Abdul:
Being on the app development sub-team, I will be required to acquire a working knowledge of the Swift programming language to develop the mobile application for the Smart Lock System. In addition to that, I would need to research Bluetooth technology and its integration within our system.

2. For each of the knowledge areas and skills identified in the previous question, what are at least two approaches to acquiring the knowledge or mastering the skill? Of the identified approaches, which will each team member pursue, and why did they make this choice?

   Steffi:
   Documentation - Two approaches that I can use to improve my skillset for documentation are reading old documentation that other groups have produced, or examples of documentation online, and writing documentation and paying careful attention to the focus of the section. The emphasis will be placed on writing documentation because with this skillset I believe the best approach is to try and then learn from what was difficult and feedback on what was produced. However, old documentation will still be referenced as well.
   Latex - Two approaches that can be used to improve my latex skill set are referencing external explanations (videos & Tutorials), and trial and error with external reference materials. I will be focusing on trial and error using some external materials. I believe that seeing what happens for yourself when you make a change is the best way to understand what is going on. The external resources will be used to give an idea of what it is possible to try to accomplish in latex, for example, certain formatting that I may not have considered existed.

   Abi:
   There are many different approaches to learning technical skills, such as Bluetooth. I plan to approach learning this skill by researching on the Internet, asking our supervisor for potential resources or personal knowledge, seeking help from peers who have more knowledge on the subject. After researching which hardware is needed to implement wireless communication, I will learn by doing, slowly building up my skills. Rather than choosing just one approach, I plan on using all of these approaches in parallel.

   Elsa:
   I will acquire this expertise by learning through resources such as previous Embedded Systems and Electronics courses I have taken, online research and journals that I have access to through McMaster. Additionally, I find YouTube to be a useful experiential resource for learning how to apply acquired knowledge in personal projects. I will

improve my technical writing and typesetting skills by practicing using Latex and challenging myself to continue researching and learning new modes of practice and editing.

Anthony:
Apple has created their own modules to teach individuals how to use their IDE and programming language. Other resources include free YouTube tutorials and private companies offering courses. For the scope of this project and considering the complexity of the application, Apple's free course and YouTube tutorials would be sufficient. Additionally, there are several public forums where individuals could search similar problems others have experienced before and find various solutions contributed by the community. Team member Anthony Shenouda has experience with the IDE as they have developed multiple mobile applications before. They will be responsible for the overall development and ensuring the application is completed successfully and timely.

Abdul:
Swift - To learn Swift effectively, I plan to take a few starter courses as well as practise some basic programs using Swift. Doing so would familiarize myself with the Syntax and libraries. Most of the general programming basics I have already developed through my use of other languages.

Bluetooth - My approach to this would be to look at a few existing projects and adopt a similar approach within our system. Bluetooth is widely used, and there are a variety of open-source libraries of which we can take advantage.