

System Design for 4TB6 - Mechatronics Capstone

Team #5, Locked & Loaded
Abi Nevo, nevoa
Elsa Bassi, bassie
Steffi Ralph, ralphs1
Abdul Iqbal, iqbalal18
Stephen De Jong, dejons1
Anthony Shenouda, shenoa2

January 18, 2023

1 Revision History

Date	Version	Developer
January 16, 2023	1.0	Steffi

2 Reference Material

This section records information for easy reference.

2.1 Abbreviations and Acronyms

symbol	description
SRS	Software Requirements Specification
FR	Functional Requirements
NFR	Nonfunctional Requirements
LC	Likely Changes
ULC	Unlikely Changes
SC	System Constraints
A	Assumptions
MV	Monitored Variables

Contents

1 Revision History	i
2 Reference Material	ii
2.1 Abbreviations and Acronyms	ii
3 Introduction	1
4 Purpose	1
5 Scope	1
6 Project Overview	1
6.1 Normal Behaviour	1
6.2 Undesired Event Handling	2
6.3 Component Diagram	2
6.4 Connection Between Requirements and Design	2
7 System Variables	3
7.1 Monitored Variables	3
7.2 Controlled Variables	3
7.3 Constants Variables - NA	3
8 User Interfaces	4
9 Design of Hardware	5
10 Design of Electrical Components	6
11 Design of Communication Protocols	6
12 Timeline	7
13 Appendix	8
13.1 Interface - Included in Section 8	8
13.2 Mechanical Hardware	8
13.3 Electrical Components	9
13.4 Communication Protocols - Included in Section 11	10
13.5 Reflection	10

List of Tables

List of Figures

1	System Context	1
2	Application User Interface	4
3	Hardware User Interface	5
4	Locking Mechanism Components Disassembled and Assembled	8
5	Circuit without the Arduino	9
6	Rechargeable Battery	9
7	Circuit Diagram	10

3 Introduction

For more information on the project breakdown, planning or delivery refer to the following documentation: [Problem Statement and Goals](#), [Development Plan](#), [SRS](#), [HA](#), [MG](#), [MIS](#).

4 Purpose

The purpose of this documentation is to break down the components that will come together to create the final product, as well as how they interact and why they are used. This document will show everything that the SmartLock is made from and when it will be assembled in order to provide transparency throughout the entire design project.

5 Scope

This lock and application are designed for the use of the average bike rider. It is simple and easy to use and locks the primary components of the bike so that the user can be confident in its security.

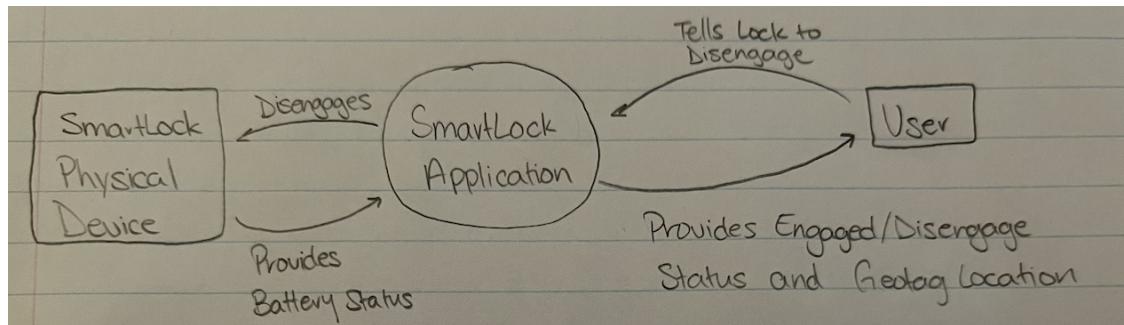


Figure 1: System Context

6 Project Overview

6.1 Normal Behaviour

The normal behaviour that is expected with the SmartLock is multistep:

1. After the first purchase of the SmartLock the user will have to attach the lock to the bike, and download the application from the AppStore or Google Play and pair the application with the physical device (this only happens once)
2. The user will put the cable around the wheel and external frame and then push the pin into the locking hole

3. The user will be able to geotag the location of the locked bike on the application
4. The user will be able to disengage the lock on the application so that they can remove the pin from the hole and release the bike
5. The application will warn the user when there is low battery so that the lock can be recharged

6.2 Undesired Event Handling

In the case of an undesired or unexpected event, the lock will stay in the locked position and the application will block the unlocking process. While this does inhibit the user's ability to be able to use their bike, it does prevent the bike from being stolen which is more important in an uncertain situation. The user will be able to reset the lock to eliminate the stimulus that caused the event and be able to use their app as normal after.

6.3 Component Diagram

6.4 Connection Between Requirements and Design

[The intention of this section is to document decisions that are made “between” the requirements and the design. To satisfy some requirements, design decisions need to be made. Rather than make these decisions implicit, they are explicitly recorded here. For instance, if a program has security requirements, a specific design decision may be made to satisfy those requirements with a password. —SS]

7 System Variables

7.1 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_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

7.2 Controlled Variables

Variable Name	Description	Type	Units
c_LockEngaged	Engages the lock	Digital	Boolean
c_LockDisengaged	Disengages the lock	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

7.3 Constants Variables - NA

8 User Interfaces

There are two user interfaces related to our product. The first is through an application (SmartLock) and the second is the lock itself where the user will be required to manually open/close the chain to secure the bike.

The application is where the user will be able to disengage their lock and locate where it was left with the Geotagging feature.

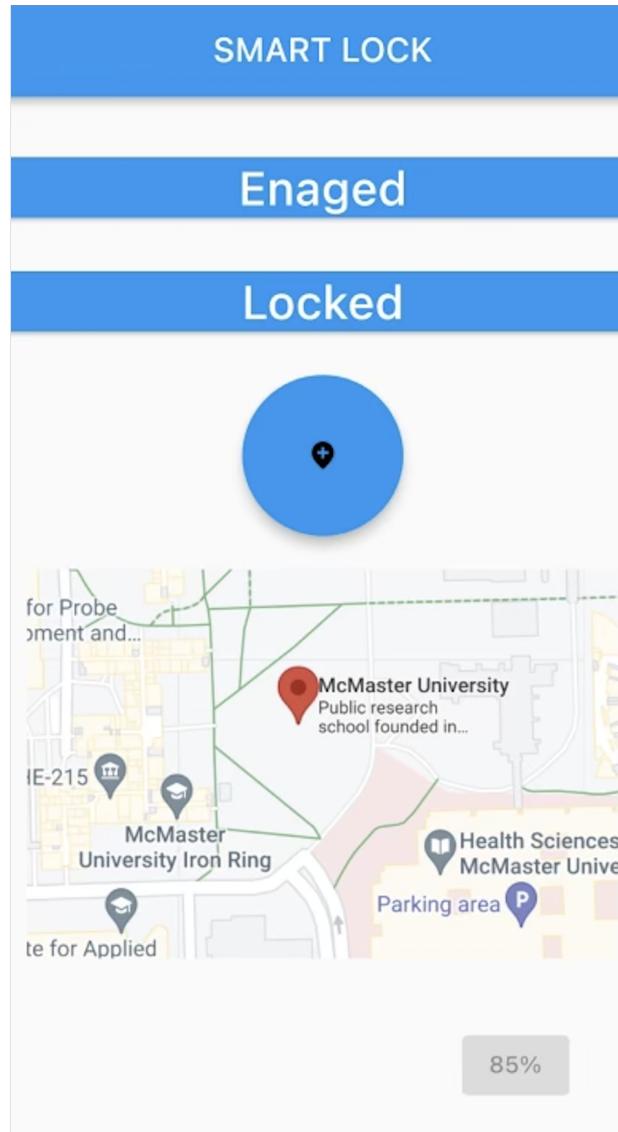


Figure 2: Application User Interface

The hardware will be mounted to the bike, which will require user interaction upon the purchase of the SmartLock. Additionally, the user must push the locking pin into the hole to engage the lock, and disengage the lock with their phone to be able to remove the pin.

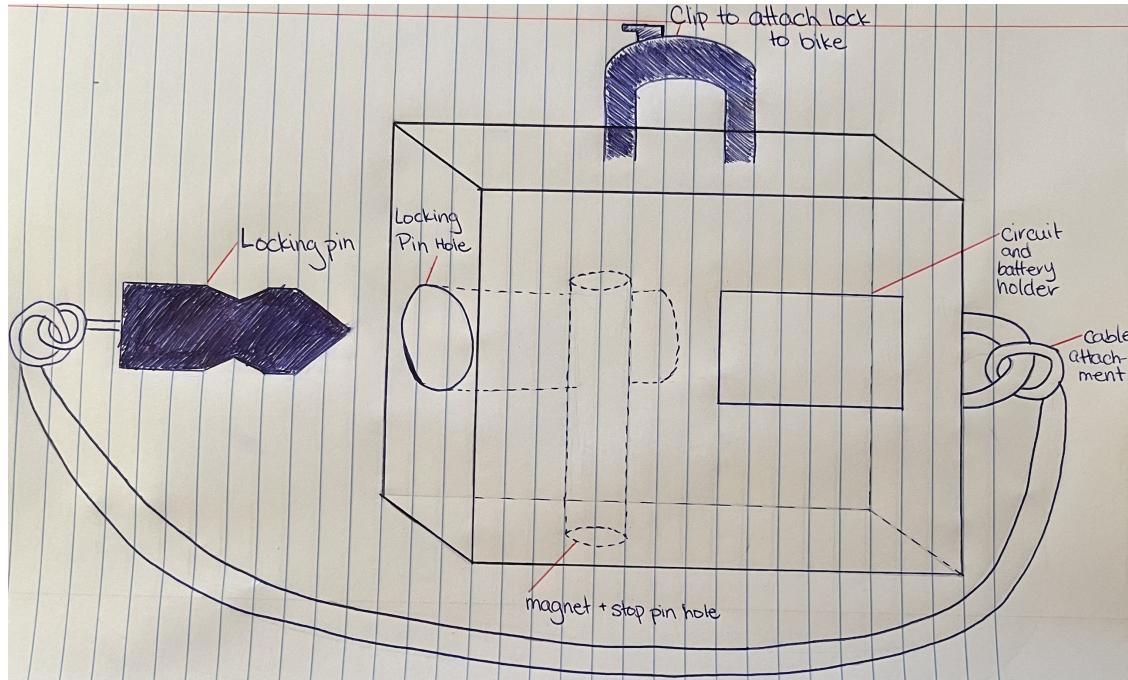


Figure 3: Hardware User Interface

9 Design of Hardware

The following is a list of the hardware components that will be acquired to build the SmartLock:

- Arduino
- Locking Cable
- Bike Attachment

The following is a list of the hardware components that will be built for the SmartLock:

- Circuit (discussed below)
- Lock Housing (CAD)
- Locking Mechanism (CAD)

10 Design of Electrical Components

The following is a list of the electrical components that will be used in combination with the aforementioned Arduino to make up the circuitry used to control the SmartLock:

- Battery
 - Needs to be acquired
- Transistor
 - Previously owned
- Wiring
 - Previously owned
- Magnet
 - Needs to be acquired
- Charging board (maybe)
 - Needs to be acquired
- Resistors (maybe)
 - Previously owned

11 Design of Communication Protocols

In order for the SmartLock to work there are signals that have to be used to communicate between the application and the physical lock. The features of our lock include:

- Locking the bike from the application
 - A Bluetooth microcontroller is utilized so that the application is able to communicate with the physical device
 - The microcontroller is wired directly to an electromagnet that controls the disengagement of the lock
- Geotagging the location of the bike
 - Integration with Google Maps will be used to save the coordinates of the locked bike

12 Timeline

Date	Description	Group Member Assigned
January 16	Housing Design	Abi, Elsa & Steffi
January 18	Design Documentation	MG - Elsa MIS - Abi & Anthony SystDes - Steffi
January 22	CAD of Housing Design	Steffi
January 22	Circuit	Stephen
January 22	App	Anthony
January 22	Arduino Coded	Abi
January 25	Arduino and Circuit Testing	Abi & Stephen
January 29	Housing 3D Printed	Abdul
February 1	Assemble Housing and Circuit	Steffi & Stephen
February 1	All Documentation has been updated to reflect the current project including Git issues & battery calculations	Elsa
February 4	Rev 0 Testing	Everyone
February 6	Rev 0 Demonstration	App & Arduino - Anthony & Abi Circuit - Stephen Housing - Steffi & Elsa Documentation Elsa & Steffi
March 8	V & V Report Rev 0	Everyone (Reqs divided by area of expertise on Rev 0 Demonstration)
March 20	Final Demonstration	Everyone (Divided by areas of work)
April 5	Final Documentation	Everyone (Divided by areas of work)
TBD	EXPO	Everyone (Divided by areas of work)

13 Appendix

13.1 Interface - Included in Section 8

13.2 Mechanical Hardware

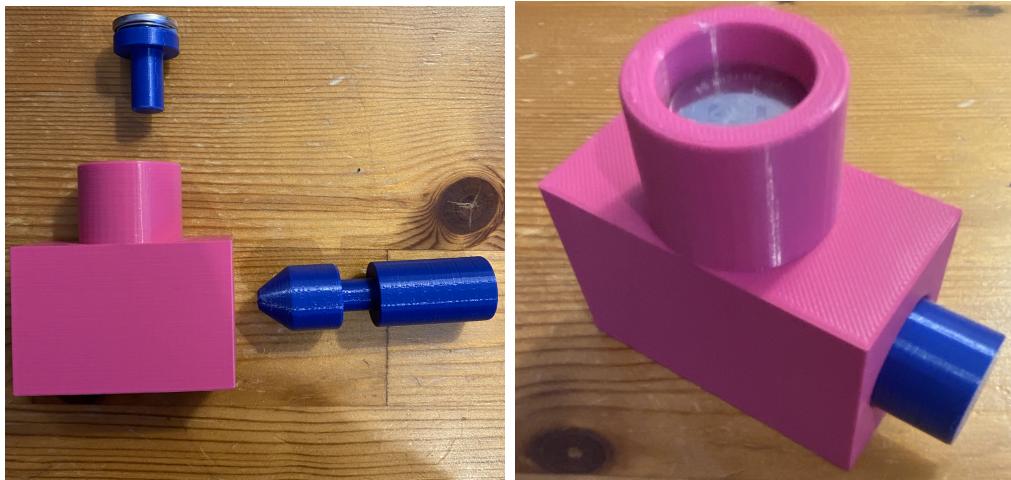


Figure 4: Locking Mechanism Components Disassembled and Assembled

13.3 Electrical Components

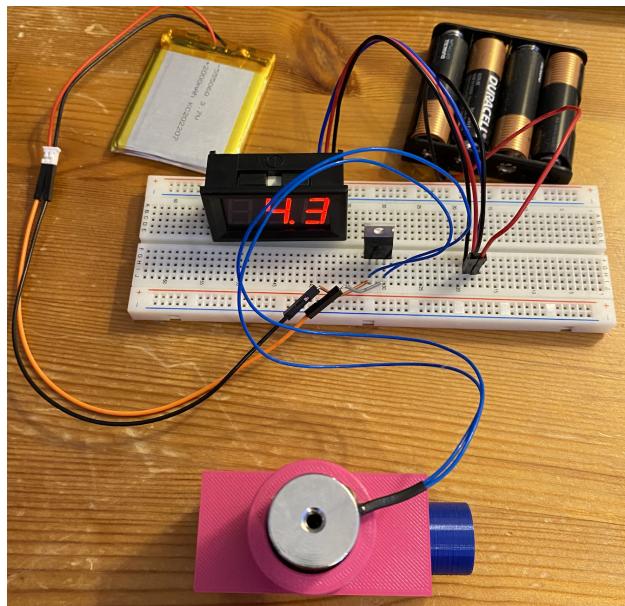


Figure 5: Circuit without the Arduino



Figure 6: Rechargeable Battery



Figure 7: Arduino

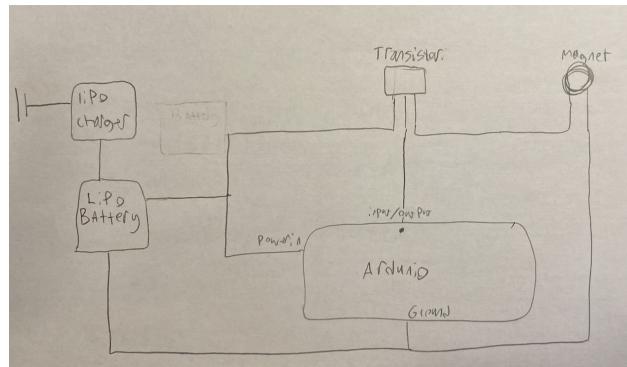


Figure 8: Circuit Diagram

13.4 Communication Protocols - Included in Section 11

13.5 Reflection

The information in this section will be used to evaluate the team members on the graduate attribute of Problem Analysis and Design. Please answer the following questions:

1. What are the limitations of your solution? Put another way, given unlimited resources, what could you do to make the project better? (LO_ProbSolutions)

Given unlimited resources our bike lock would function without any human exertion, ie. mechanics, specifically, motors would work to open and close the lock eliminating the need for the user to touch the lock at all. They would simply have to touch a button on their phone. Additionally, there would be separate components to ensure the locking of the wheels to the frame and the frame to an external secure location.

Finally, with unlimited resources, there would be a way to ensure the seat of the bike could not be removed either.

2. Give a brief overview of other design solutions you considered. What are the benefits and tradeoffs of those other designs compared with the chosen design? From all the potential options, why did you select documented design? (LO_Explores)

All of our solutions were fairly similar given that there are only so many different ways to lock a bike. The main differences came from various housing alternatives and "chain" variations. Our original idea functioned more similarly to the ideal situation described above, but in scaling our project to be realistic we had to eliminate and restructure our ideas.

Some of our other ideas were:

- A foldable chain that could be folded up small and compressed to the bike while riding
- A circular arm for locking
- An extendable arm so that you could be multiple distances
- Individual locking units for each component

We landed on our final design because it uses components that are familiar to the average bike rider, it allows for the bike to be varying distances from a frame and it only involves one locking mechanism which is simpler, less bulky and easier to use. Additionally, we opted against the motor as a way to eliminate physical intervention because it would require much more battery and replacing or charging the battery that often would be more inconvenient than using your hands to insert the lock