ELECTRONICS

VIJAY SAMBAMURTHY

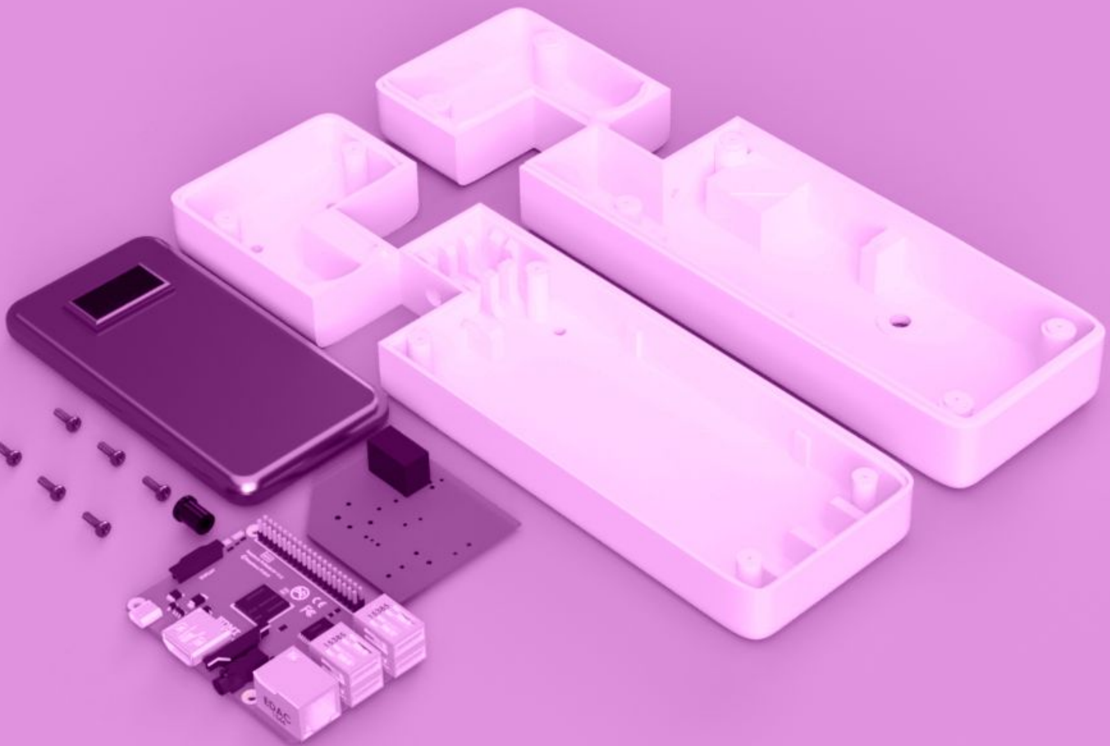
4700

CONTENTS

RESEARCH

DESIGN

CONTENTS



RESEARCH

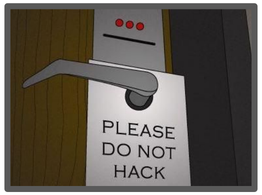
ANALYSIS OF THE DESIGN BRIEF

Introduction

In the 21st century, burglaries and break-ins are becoming more and more common by the day, with over 8 million burglaries in 2015 just in the US. Resulting in a total of 14 billion US Dollars in financial losses for the victims.

Many of these occasions were a result of a lack in security measures or improper devices for. Especially when we are in the modern world, many people may try to connect locks to the IOT, ending up in even more vulnerabilities as it does not have a secure connection or similar issues. Certain companies try to just move the vulnerabilities do a different part of the product like requiring just a Smartphone's bluetooth connection to connect to unlock the product. This is a big vulnerability as such a connection may be replicated and allow a break-in.

Some companies have successfully created electronic locks which create a viable product for consumers but the problem persists in the user interaction where those without knowledge on electronic devices may not know how to correctly set up the device thus rendering some of the key-features of the lock useless.



Design Brief

This product should be able to inform the user of any suspected intrusions in addition to **holding functionality in the case of wifi disconnection**. The product **comes with an app which should be able to fully control the functionality** of the monitoring device. The casing of the system should **have a simplistic look with the addition of durability and good build quality** but moreover must have the **correct mounting points and screw bosses to allow for mounting on doors or locker**s. The device is **aimed mostly homeowners**, but is also well suited for young adults looking for more privacy protection in dorms or rented apartments. **Ages: 14+**

The casing must be compact and well built with **no sharp edges** to prevent harm to users and **will be 3D printed** with 2 main parts and a loose wire leading to a mounted camera which would be placed on the ceiling.

Key Issues

Budget - The production costs must be within the budget and the prototype must be a fraction of the main budget.
Sustainability - The solenoid should be able to lock an **unlock approximately 15,000-20,000 times before it starts to wear out the case**. If the solenoid fails any earlier the product should be easy to disassemble and repair.
Form - The device should be firmly mounted against the door and be compact. It should look simplistic without attracting too much attention.
Manufacture Scale - The scale of manufacture for my product can **be Mass production.**
Function - The product should give **adequate protection to houses** and properties while still being affordable for most.
User requirements - It should be easy to set up and use by the user.

RESEARCH PLAN

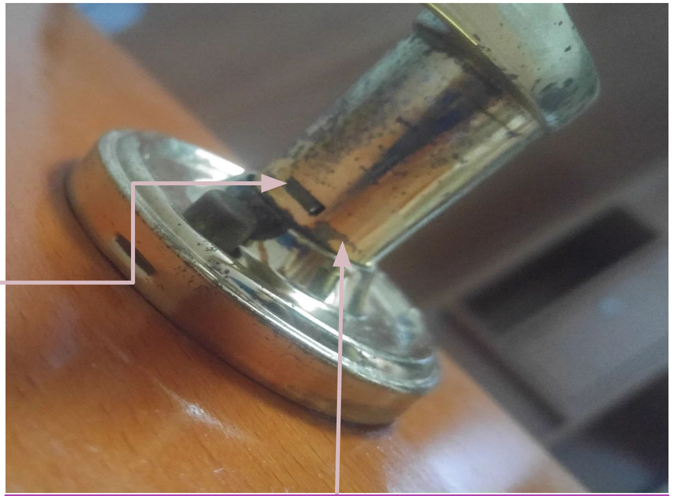
What	Why	How
Storyboard	To understand and deconstruct what happens when a break-in occurs.	By creating a fictional story about the issues the ideal user of my product will face without the product.
Persona	To understand problems with locks from homeowners and adults seeking protection.	By writing two fictional personas in the perspective of the potential users of the product.
Product Analysis	To get familiar with the pros and cons of a product in order to learn from mistakes and build using strong points of other products.	By disassembling two products I will be able to see where the vulnerabilities are and where the protected areas are in addition to understanding the design of the casings of such products.

EXISTING DOOR LOCK CRITIQUE

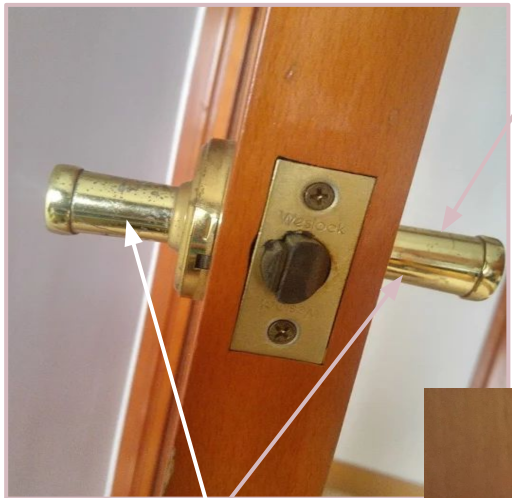
Introduction:

I have deconstructed and studied a door lock and a TV box to understand the advantages of using a mechanical door lock and the potential issues and vulnerabilities with using a mechanical door lock. The mechanical product was also only disassembled to find the potential risks, flaws and strengths with conventional door locks. On the other hand, the TV box has enriched me with the knowledge of the **compact structure of its casing and how to assemble my own product such that the casing is suitable** for the intended purpose. In addition to removing issues to the internal components and allowing the user to mount the product where required without much effort.

Brass is used in door handles as it is known for its variety of properties including: strength, machinability, ductility, wear-resistance, hardness, antimicrobial, electrical + thermal conductivity, and corrosion resistance. Brass' color is also appealing to the eye and matches well with wood.



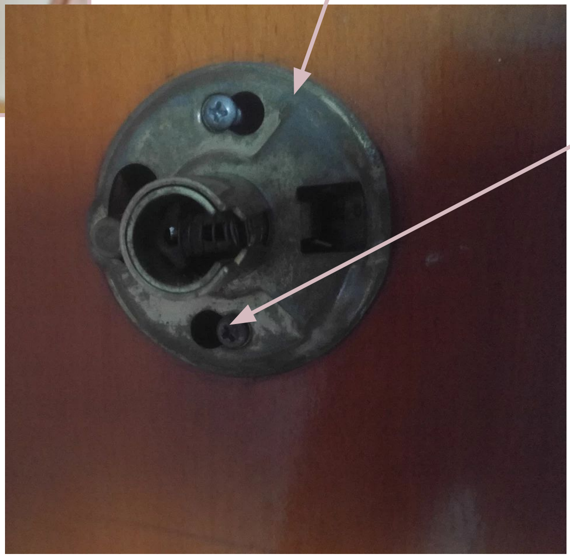
The small notch was the button that allowed removal of the handle. Was only located on the side with full control of the lock.



Ergonomic handle design for comfort and effortless opening

No sharp edges and very compact design.

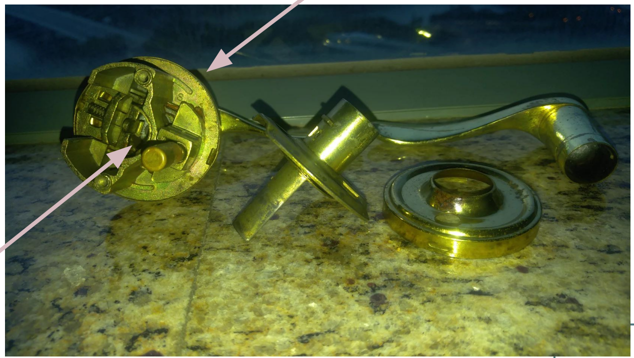
Aesthetically pleasing look of handles and no screws shown on either side. Giving a very clean and tidy look.



Overall attempting to deconstruct the handle at first was a bit of a difficulty but with a little bit of research I was able to find the notch and then it became more straightforward. This may be an issue in case the lock gets jammed or there is an problem. ***This type of security gives users no information on break ins and can be easily picked as intruders have direct physical access to them.***

The screw holes have **slight amount of excess gap** which is great when mounting as it allows **more leeway** when drilling holes and putting the parts together, making the entire process easier and require less preparation.

The metal parts **wear out very fast** due to the constant contact and because of how they rub against one another.

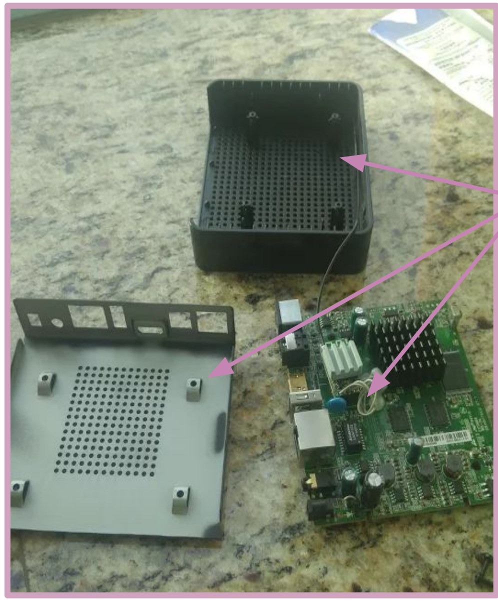


The lock button is **very simplistic** and easy to press as it is right where a user would place their thumb when using the handle.

PRODUCT ANALYSIS 2

Conclusion:

The door handle and lock were very well made and **had a perfect ergonomic design** to it, but it was **lacking in affordability due to the materials used alongside the design** of the lock which caused the brass to wear out faster. Thus requiring some sort of maintenance in the form of lubrication. The TV box on the other hand had **plenty of vents and certifications regarding safety**, moreover it was a very simple design requiring only **1 minute or so for disassembly**. The main board fit into the casing snugly and was also simple to re-assemble unlike the door handle. In my product I will incorporate the simplistic design of the TV box with the addition of the functionality of the door handle.



Simple three piece design mostly made of a durable plastic.

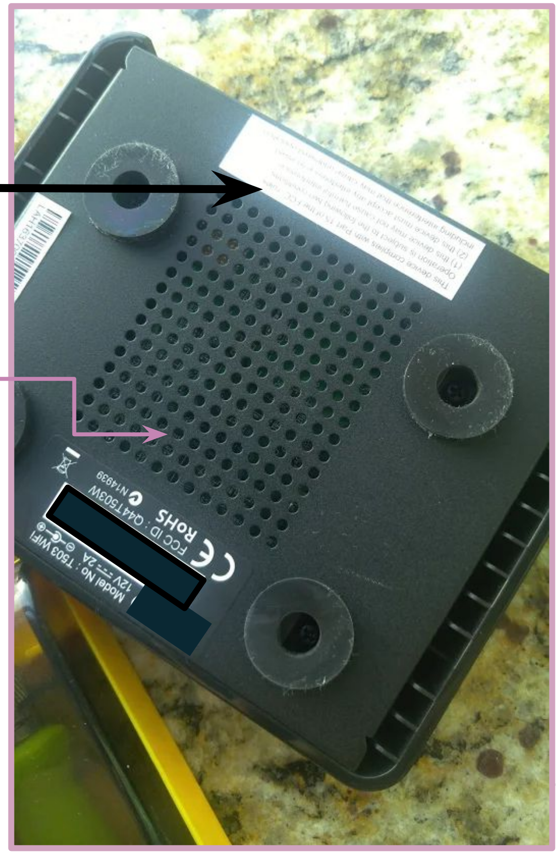
All ports in one area giving quick, obvious and easy connections to any cables that were required to be plugged in.

The low profile design of the box allows it to fit in well with most common furniture. This gives it a very low profile while still looking very sleek and minimal-istic.

RoHS compliant alongside **CE mark**, meaning there are no hazardous materials found in the electronics of this product and it meets European health and safety standards, further increasing safety of the product.

Many vents to cool down components and keep fire hazards at a minimum.

Overall the device had a very very simple look with 4 screws there was access to the main board and components which is good in case repairs are required. The device had plenty of vents to cool down the processor and components which kept electronics firesafe in addition to the casing which had no edges minimizing potential damage to the user.



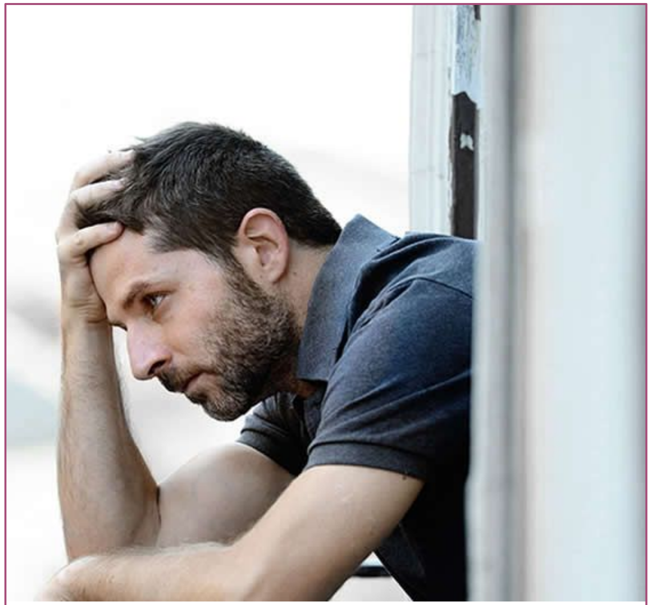
This product was made out of ABS. ABS has a strong resistance to corrosive chemicals in addition to physical impacts. It is easy to manufacture due to the low melting point making injection moulding a good choice when it comes to mass manufacture.

Persona 1:

Jim is a 38 year old landlord and lives in a very densely populated area. He has recently ended his contract with the previous tenant. He is currently in search for a new tenant and has left the responsibility of the house keys with his real estate agent who is helping him find a tenant as soon as possible. The house is fully furnished and decorated with everything from television to sofas.

The agent has gone through a few potential tenants but none have signed a contract. It has been a few months since the house was put up for lease and Jim has reported that there has been several break-ins over the past month but there are no suspects. These break-ins leave Jim paying for more furniture with a **total estimated loss of 9,000 US$.**

Jim needs a product which can **alert him of any break-ins as fast as possible** so that he can call the police and give any images for comparisons with potential suspects. This would allow Jim to feel more secure with his appartment and help catch any thieves or burglars, helping not just Jim but the community around the area.



Conclusion:

Investigating and deconstructing personas that were thought of have allowed me to find **key issues** and **points which I can address in my product**. This will be accomplished by adding features to my product.

Introduction:

These fictional characters are the type of ideal customer for the product I have created and would be the ones I am aiming to impress first.

Persona 2:

At 20 years old, Todd finds himself getting robbed every month due to his roommate leaving the dormitory room door open and giving access to anyone who passes by. Todd finds that his roommate and him are key targets for the thieves as they are known for leaving the door open and inviting unwanted guests. Todd and his roommate have lost 2 laptops, 3 phones and 1 watch in total.

Todd requires a product that can use motion capture to alert him of any robbers entering the dormitory with images and data, **allowing him to alert the school of such occurrences with evidence and keep his belongings safe even when he is away from his dormitory.**

STORYBOARD

Introduction: This page shows a story in which the target user faces issues that would arise without using the product.

1.Vulnerable House

A homeowner has **left his house unattended with a simple lock** that can be broken easily. In addition to this the house **has no cameras or alarms** to warn the homeowner of any intrusions. The house is a key target for most burglars.

2. House is robbed

The burglar **has set its target as the vulnerable house and has robbed it of all its valuables**. The owner has **not much evidence to go off** to catch the burglar as he was in a suit and the time of day at which this occurred is unknown.

3. Property is lost

The owner is now left with a dirty house, moreover the owner **has to pay to replace all the items he has lost.** In America the **average burglary costs the owner US$2,230.** Phones, laptops and other valuables have been taken.

4.Money is wasted

The owner **spends money on replacing the items lost** in the burglary **but does not invest much in more security as they find most items too expensive or pointless.** The owner decides to buy new locks that are stronger yet have no added functionality.

5. Robbed again

Once again the **owner is targeted as their house is vulnerable once again** and the cycle starts over again leaving the owner of the house losing lots of money each year.

Conclusion:

If the owner **had a device which would have notified them of such intrusions or motion** in the house at a time that it would be vacant they would be to notify security or rush back to their property in time to catch the intruder with their belongings.

SPECIFICATION

Introduction: This page contains the specifications of the electronic product based on the research I have conducted which will include all the requirements that my product should have and the reason behind each aspect.

1) Appearance

1.1) The current mode that the product is in should be clearly indicated with LEDS that are visible on the protected side.

1.2) The product should not be very vibrant but should fit in well with its surroundings and should be a suitable color like black.

2) Cost

2.1) The product should be **affordable but may be in the slightly high end range** of locks due to the added functionality while still affordable by most homeowners.

2.2) The product should be as **high as possible within the budget range in order to maximise efficiency and costs**, hence increasing longevity.

3) Customer

3.1) The product is **aimed at those wanting security** from a property owner to a teenager wanting privacy in the dormitories.

4) Environment

4.1) The product should **remain functional when there is a temporary loss of power.**

5) Ergonomics

5.1) The product should be designed in such a way that there **is less stress on the door.**

5.2) The product should be easy to **mount onto any door**.

5.3) The Skype service should have a quick response time without running into issues.

5.4) The interface on the product should be **simple and easy to understand.**

6) Size

6.1) The product should be no larger than an a tissue box.

7) Safety

7.1) The product should have **no sharp edges** to prevent any cuts and bruises to the user.

7.2) The product should have **multiple ways of unlocking in-case** one of the methods fail to function correctly.

7.3) There should be **no loose components** or cables on the PCB to prevent potential harm to the user.

8) Function

8.1) The product must be able to **take pictures of any intruders**.

8.2) The product should be **fully controllable from anywhere**, as long as there is access to skype.

8.3) The product must be able to lock the solenoid **without turning on when the door is open.**

8.4) The user should be **warned nearly instantaneously** when there is an intruder.

9) Maintenance

9.1) **Easy access to the product's components** is essential as with few screws repairing any faulty parts would be less tedious and time-consuming.

10) Materials

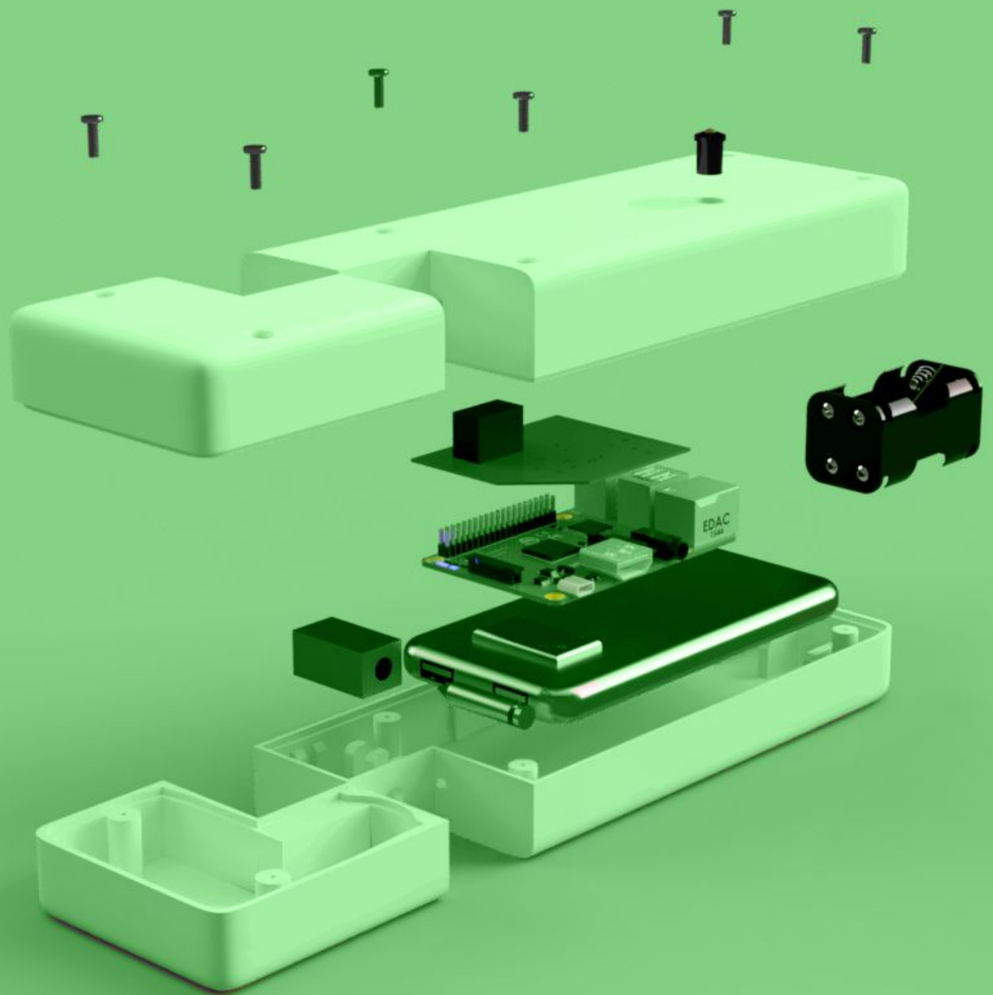
10.1) From my research I found that **ABS** or similar plastics would be very suitable, this would be made alongside metal so that it cannot be destroyed or broken easily.

10.2) The materials used to build the product should be **abundant** and recyclable to reduce costs and help the environment.

11) Manufacturing

11.1) The product should be able to be mass produced because there will be a significant amount of target users.

11.2) The raspberry pi will come built in with the correct software and code.



D E S I G N

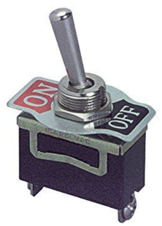
INPUTS

Push to break switch

- The push to break switch would be capable of sensing the opening of the door as it would be implanted within the two main parts of the device, thus making the circuit when the two parts are detached from one another as occurs during the opening of a door.

Single Pole Single Throw Switch (SPST)

- This switch would be of great use as a quick release for the solenoid in the case that the user was only leaving for a short period of time (e.g. take out the trash).
- In case of emergency situations, any person(s) inside the property being locked may easily turn off the device.
- While it is cheap it may be accidentally left on the off position, rendering the entire product useless.



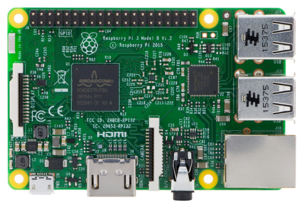
Camera

- The camera would fare really well to enable an extra layer of security, being motion capture. In the unlikely case that the main lock has been compromised and unwanted access has been given, the camera will be able to detect motion and alert the Raspberry Pi.
- In addition to motion detection, the camera would be able to take images of any people or pets that have entered the room and connect to the Raspberry Pi to send it to the owner via WhatsApp.
- Potential for false motion trigger.

PROCESS

Raspberry Pi Model 3 B

- As a very affordable microcomputer, the Pi has **GPIO pins** that allow control of power via programming, making it very suitable to operate the relay with.
- The Pi has a **wifi module** which allows the pi to monitor network traffic and detect any devices connected to the wifi.
- The Pi is also very power efficient only requiring **~2.5A@5V** with a Micro USB cable.
- With its small form factor the Pi **can fit into the main case** without affecting the size too much.
- Python 3 comes built in with Raspbian (Pi's Operating System) making setting it up less complex.



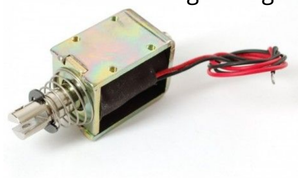
Relay:

- Used to control high voltages, the relay can be utilized to electromagnetically turn on and off the solenoid.
- Relays are used where a safe low-voltage circuit controls a high-voltage circuit.

OUTPUTS

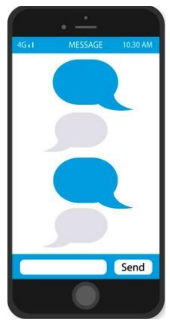
Solenoid:

- The solenoid would **act as the lock for the door**.
- In the actual product the solenoid would be a higher voltage one for added strength and durability however this would reduce the time that the Pi can last on during emergency power mode.



Raspberry Pi Model 3 B (Messaging service):

- Notifies the user of when he has a breakin.
- **Skype is used because** no need to purchase servers .
- Allows **easy access from phone**
- Instant messaging
- Can send images of potential intruders



LED:

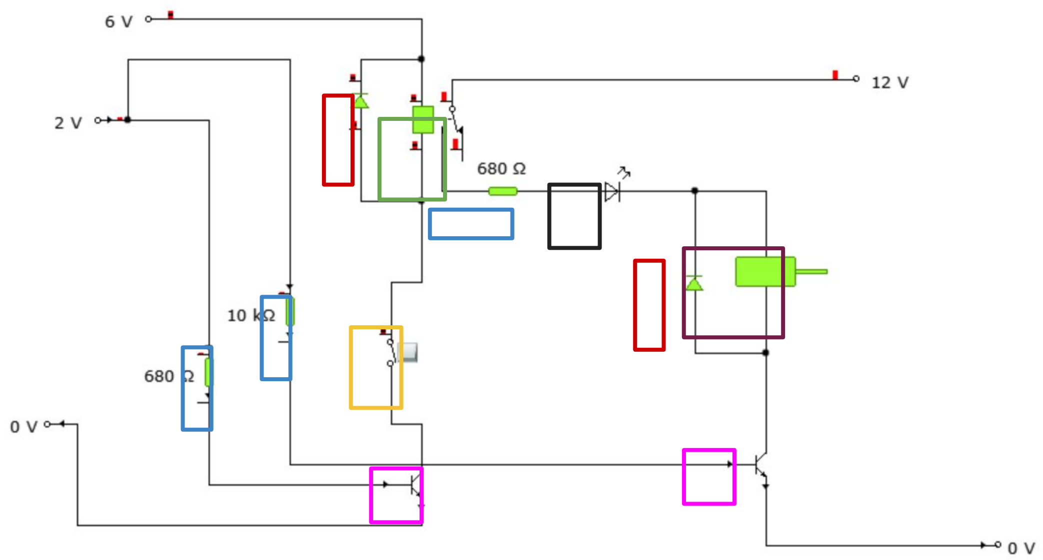
- Allows the user to know when the device **is left unlocked.**
- Raspberry Pi has leds on it to inform when the entire device is on.

Circuit Overview

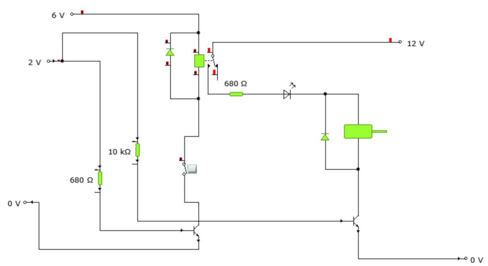
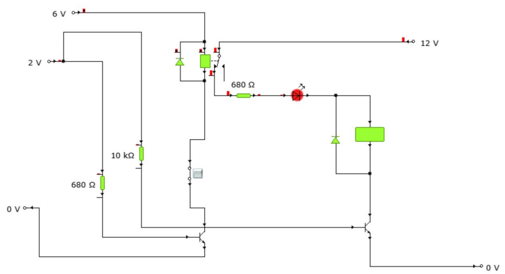
The circuit has multiple inputs which are all from the raspberry pi except the **12v input**. The camera connected to the **raspberry pi** detects motion which then causes it to send a message to the owner. On top of that, the **raspberry pi** deactivates the **solenoid** when the smartphone just connects to the **WiFi** network which turns on the **LED** too.

Reference to research

8.3) The product must be able to lock and close the solenoid without turning on when the door is not closed. - The lock is able to turn on and off, but there is no circuit or method to check if the door is closed.



Different states

Solenoid Closed	Solenoid Open
	
In this circuit the solenoid is closed which mean the door is locked and the **Pi** is checking for motion with the camera. In addition to this the **LED** is off so there as there is no need to remind the user to lock the door.	In this circuit the solenoid is open which mean the door is open and the **Pi** is waiting for the door to be closed. In addition to this the **LED** is on to warn the user that the door is currently unlocked and needs to be closed.

Parts

SPST - Relay - LED- Solenoid- Transistor-
Resistor- Diode-

Conclusion

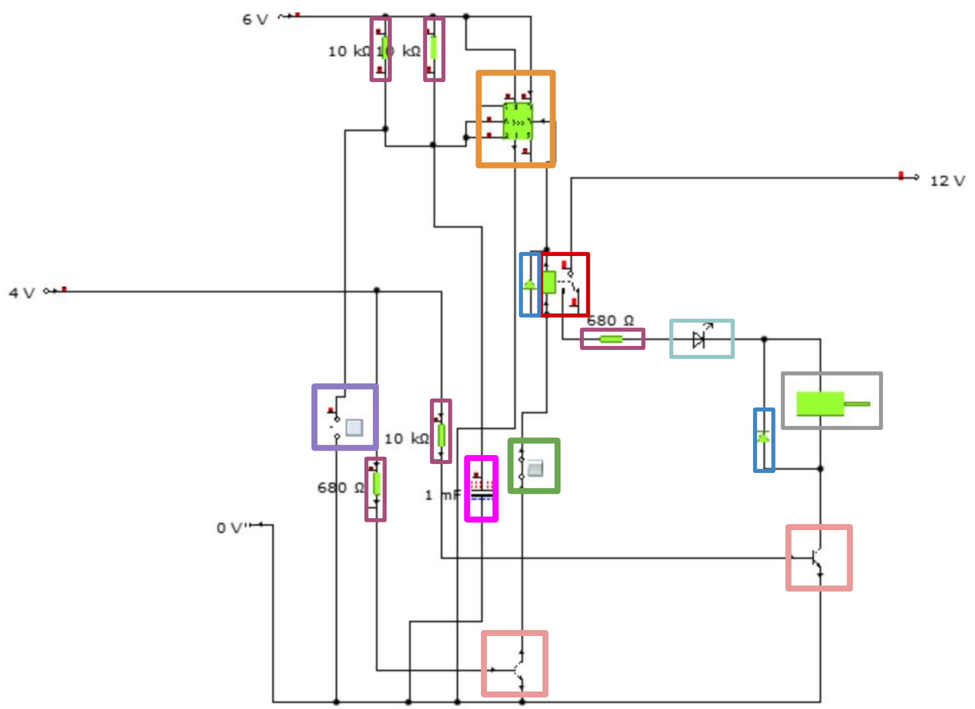
The circuit **performs its function but with such a simple design** the **coding side of the design will be much more complex.** On the other hand this might be a silver lining as there are **more aspects of the lock that the user has better access and control over**. Since the circuit is really small it can be **more compact** than the raspberry pi meaning there is **vast amount of free space for the intended batteries** for the solenoid and as backups if the house loses power.

SPECIFICATION POINT	ANALYSIS	KEY POINTS	SCORE
Cost: 2.1) The product should be affordable but may be in the slightly high end range of locks due to the added functionality while still affordable by most homeowners. 2.2) The product should be as high as possible within the budget range in order to maximise efficiency and costs, hence increasing longevity.	The components are all ones that are very cheap,good quality and obtainable from many electronics shops. The small amount of components allows the cost to be further reduced while maintaining quality. This allows more people to be able to purchase the product.	-Minimum possible components -Simple -Reasonable quality -Low cost	5/5
Erganomics: 5.4) The interface on the product should be simple and easy to understand.	Due to the fact that this product does not need complex switches and buttons, the interface is very simple and easy to understand thanks to the one LED.	-Informs user when door open -1 switch to control the lock	4/5
Size: 6.1) The product should be no larger than an a tissue box.	The small amount of components can be easily packed into a PCB with a small footprint, easily fitting into a casing smaller than a tissue box.	-Can be tightly packed -No large components	5/5
Safety: 7.3) There should be no loose components or cables on the PCB to prevent potential harm to the user.	There are no flying wires required in the design and the components, except for the LED and Switch are all on the PCB.	-No loose components -Diodes on electromagnetic components to prevent back EMF	4/5
Function: 8.3) The product must be able to lock the solenoid without turning on when the door is not closed.	The product can close and open but has no mechanism to detect if the door is open or closed. Thus failing one of the main functions and could cause an absolute disaster in the real situation.	-Closes and opens fast -Does not know if door is open or closed. (CANNOT AUTOLOCK)	1/5

INITIAL CIRCUIT DESIGN 2

Circuit Overview

The circuit has multiple inputs which are all from the **raspberry p**i except the **12v input.** The camera connected to the **raspberry p**i detects motion which then causes it to send a message to the owner. On top of that, the **raspberry p**i deactivates the **solenoid** when the smartphone just connects to the WiFi network which turns on the **LED** too. Here there is also a **555 timer** that automatically locks the door after a controlled amount of time. Ensuring that the door is locked even when the user forgets to lock.



Reference to research

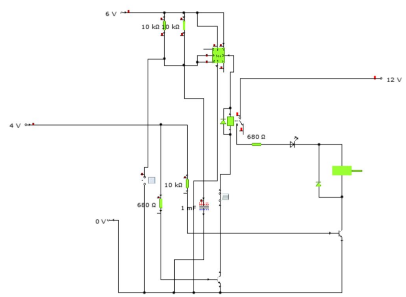
8.3) The product must be able to lock and close the solenoid without turning on when the door is not closed. - The lock is able to turn on and off in a controlled manner, but there is no circuit or method to check if the door is closed.

Parts

Transistor - Capacitor - Resistor - 555 Timer- SPST-
Reset button- Diode- Solenoid- Relay - LED-

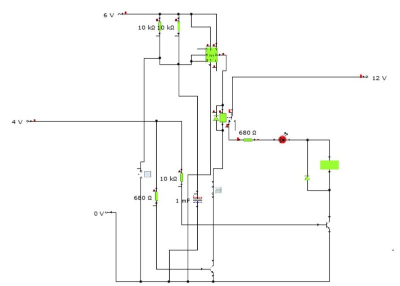
Different states

Solenoid Closed



In this circuit the solenoid is closed which mean the door is locked and the **P**i is checking for motion with the camera. In addition to this the **LED** is off so there as there is no need to remind the user to lock the door.

Solenoid Open



In this circuit the solenoid is open which mean the door is open and the **Pi** is waiting for the door to be closed. In addition to this the **LED** is on to warn the user that the door is currently unlocked and needs to be closed - it will auto lock after some time.

Conclusion

The circuit performs its function but with a **slightly complex design the coding side of the design will be slightly less work**. There are **less aspects to be controlled by the use**r. The circuit uses **bigger components like large capacitors** which may t**ake up usable space** and make the circuit harder to fit into a small casing.

SPECIFICATION POINT	ANALYSIS	KEY POINTS	SCORE
Cost: 2.1) The product should be affordable but may be in the slightly high end range of locks due to the added functionality while still affordable by most homeowners. 2.2) The product should be as high as possible within the budget range in order to maximise efficiency and costs, hence increasing longevity.	The components are all ones that are very cheap,good quality and obtainable from many electronics shops. There are much more components than required that can be replaced with a few **free** lines of code. This would also make the product harder to fix.	-Unrequired waste of components -Slightly overcomplicated -Reasonable quality -Low cost	3/5
Ergonomics: 5.4) The interface on the product should be simple and easy to understand.	Due to the fact that this product does not need complex switches and buttons, the interface is very simple and easy to understand thanks to the one LED.	-Informs user when door open -1 switch to control the lock -Automatic locking	5/5
Size: 6.1) The product should be no larger than an a tissue box.	The larger amount of components may make it a struggle to fit into a small casing in addition to the large capacitor that would take a lot of free space.	-Hard to pack tightly -1 Large component	2/5
Safety: 7.3) There should be no loose components or cables on the PCB to prevent potential harm to the user.	There are no flying wires required in the design and the components, except for the LED and Switch are all on the PCB.	-No loose components -Diodes on electromagnetic components to prevent back EMF	4/5
Function: 8.3) The product must be able to lock the solenoid without turning on when the door is not closed.	The product can close and open but has no mechanism to detect if the door is open or closed. Thus failing one of the main functions and could cause an absolute disaster in the real situation.	-Closes and opens fast -Automatically locks after some time -Does not know if door is open or closed.	2/5

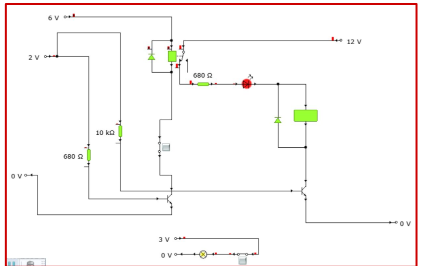
Circuit Overview

The circuit has multiple inputs which are all from the **raspberry pi** except the **12v input.** The camera connected to the **raspberry pi** detects motion which then causes it to send a message to the owner. On top of that, the **raspberry pi** deactivates the solenoid when the smartphone just connects to the **WiFi** network which turns on the **LED** too. There is also a contact pad (simulated by the bottom circuit) which detects if the door is closed before locking and opening.

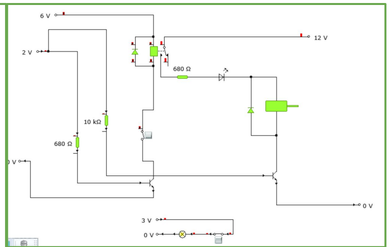
Reference to research

8.3) The product must be able to lock and close the solenoid without turning on when the door is not closed. -The lock is able to turn on and off in a controlled manner if the door is closed.

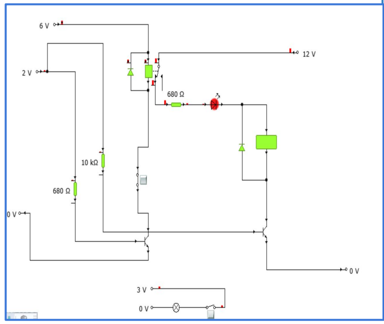
Different states



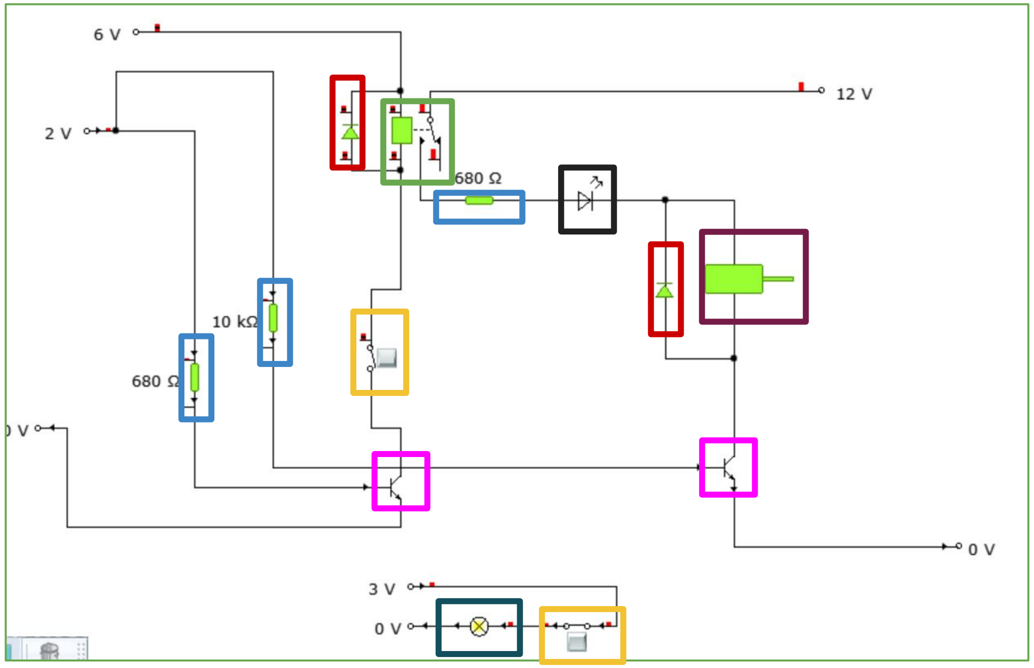
Door closed, solenoid closed
Is the default state the product would be in most of the time. Where the door is locked.



Door closed, solenoid open
This state is where the door is closed but the solenoid is open. The product must not stay in this state for long, hence why the LED remains on. The product auto locks after a few minutes in this state.



Door open, solenoid open
Here, the solenoid remains open while the door is open and will not lock or attempt to close as it is aware the door is opened. There will be warnings sent to user when door is like this for prolonged periods of time.



Parts

SPST - Relay - LED- Solenoid- Transistor-
Resistor- Diode- Lamp -

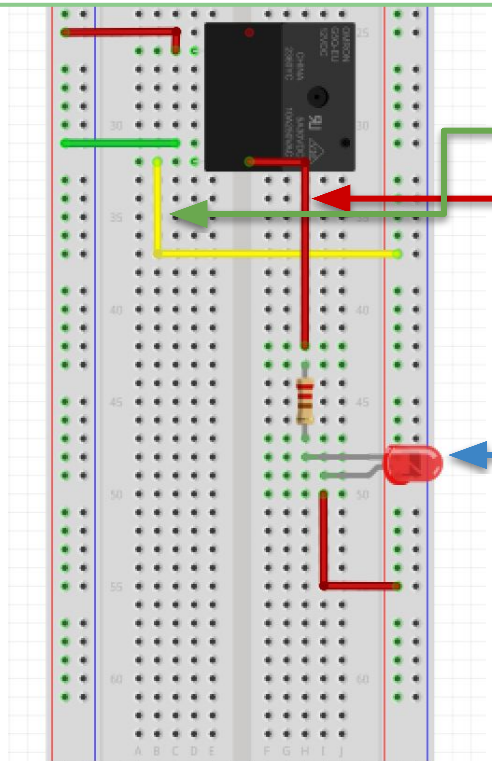
Conclusion

This circuit performs exactly the same as circuit one, **except the fact that there are contact pads (simulated by the bottom circuit)** which inform the Pi when the door is open or closed. **The lamp, in this case, simulates a GPIO pin from the raspberry pi which detects voltages.** This will make **closing and opening the door more safe.**

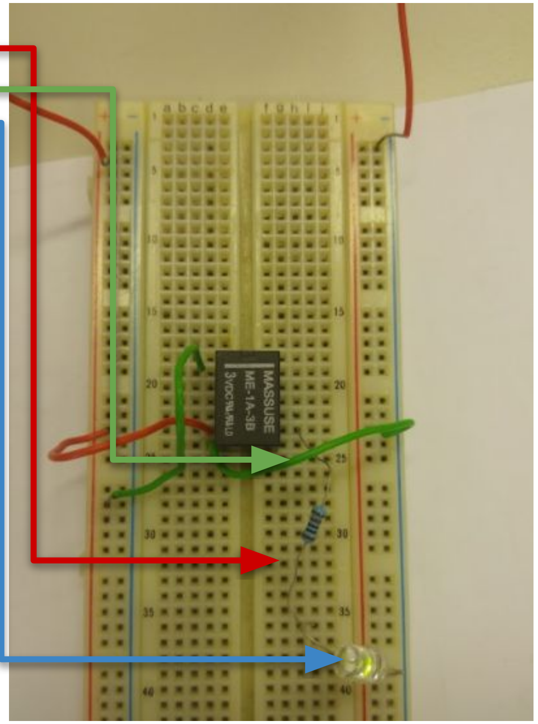
ANALYSIS OF INITIAL DESIGN 3

SPECIFICATION POINT	ANALYSIS	KEY POINTS	SCORE
Cost: 2.1) The product should be affordable but may be in the slightly high end range of locks due to the added functionality while still affordable by most homeowners. 2.2) The product should be as high as possible within the budget range in order to maximise efficiency and costs, hence increasing longevity.	The components are all ones that are very cheap,good quality and obtainable from many electronics shops. The small amount of components allows the cost to be further reduced while maintaining quality. This allows more people to be able to purchase the product. May be slightly more expensive due to the added circuit.	-Minimum possible components -Simple -Reasonable quality -Low cost -More PCB printing	4/5
Erganomics: 5.4) The interface on the product should be simple and easy to understand.	Due to the fact that this product does not need complex switches and buttons, the interface is very simple and easy to understand thanks to the one LED.	-Informs user when door open -1 switch to control the lock -Programmed with auto locking	5/5
Size: 6.1) The product should be no larger than an a tissue box.	The small amount of components can be easily packed into a PCB with a small footprint, easily fitting into a casing smaller than a tissue box.	-Can be tightly packed -No large components	5/5
Safety: 7.3) There should be no loose components or cables on the PCB to prevent potential harm to the user.	There are no flying wires required in the design and the components, except for the LED and Switch are all on the PCB.	-No loose components -Diodes on electromagnetic components to prevent back EMF	4/5
Function: 8.3) The product must be able to lock the solenoid without turning on when the door is not closed.	The product can close and open and has a mechanism to detect if the door is open or closed.This allows the product to properly carry out all required tasks without issues.	-Closes and opens fast -Can detect if door is opened or closed.	4/5

DEVELOPING AND PROTOTYPING CIRCUIT

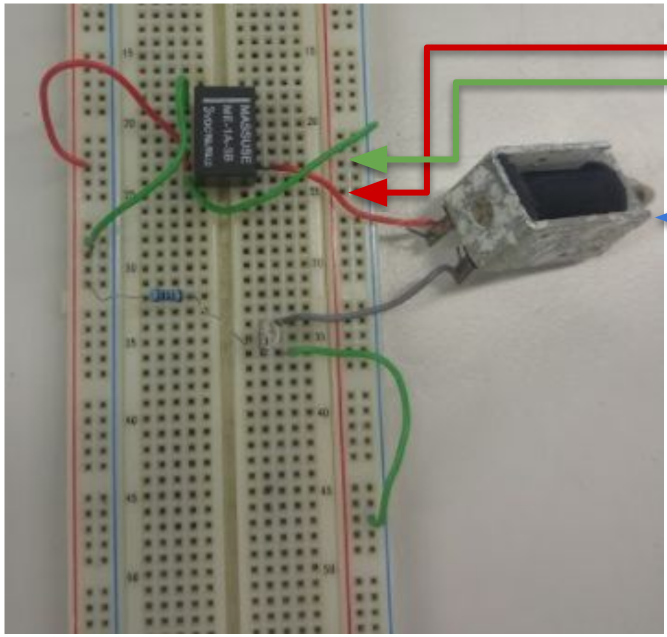


Red line - Simulated load
Green and yellow lines - controlling circuit
LED - Simulated solenoid

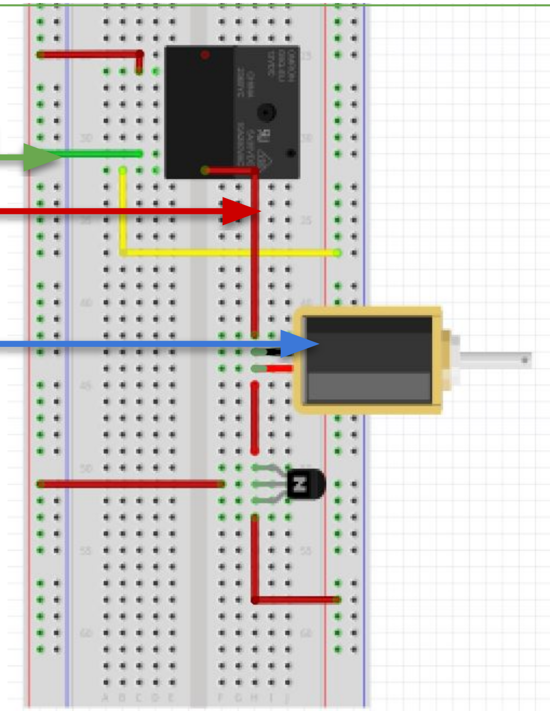


Stage 1 (Relay, Led (+resistor) - Relay setup):

Here I decided to set up the **relay pins and ensure that it was functioning correctly**. This also allowed me to work of a base design that could be widely modified to control the soon to be added components. **It was a small challenge to figure out the pin outs of the relay due to the odd pin layout on this model**, but, with a multimeter I was able to deduce the pins. Here the LED switches on and off when the yellow wire (as shown in the diagram correlating to the one on the real version) is connected and disconnected. This is because when the yellow wire is connected to the coil of the relay, it switches between normally closed and opened, changing the location of powerflow. A tick noise affirms the switching of the relay. The relay design is now complete.



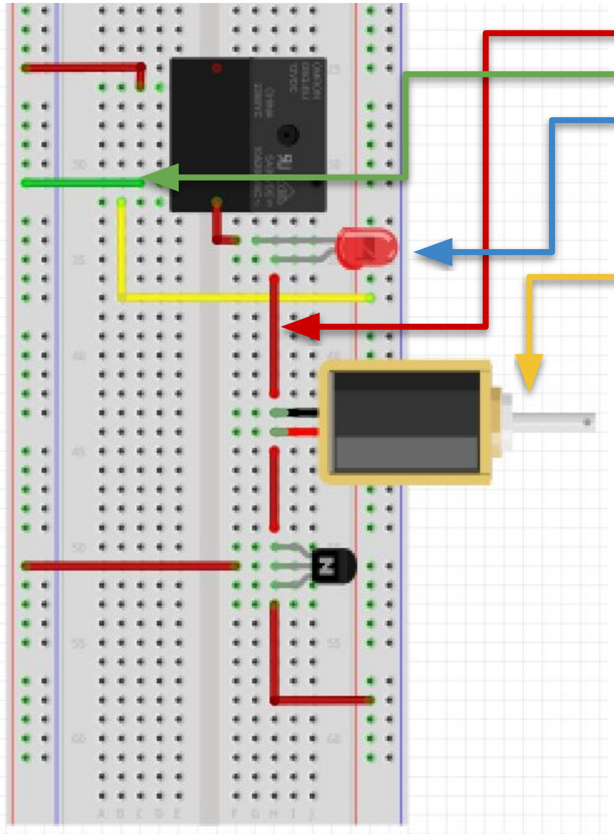
Red line - simulated load
Green and yellow lines - controlling circuit
Solenoid - Main output



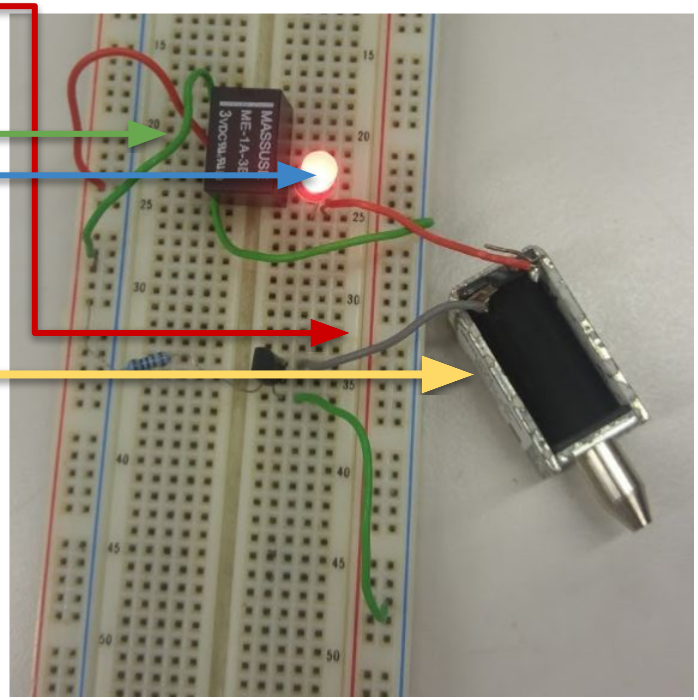
Stage 2 (Add solenoid and transistor):

I removed the LED and replaced it with a solenoid and a transistor. The transistor is used to **boost the current in the circuit.** The transistor is used in order to give the necessary amount of current required for the solenoid. Now when the same yellow wire is connected and disconnected, the solenoid turns on and off. This is the mechanism that will control the locking of the door.

DEVELOPING AND PROTOTYPING CIRCUIT

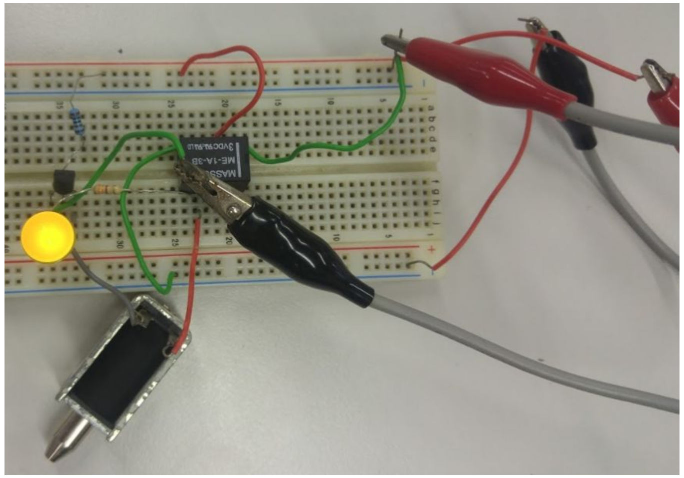


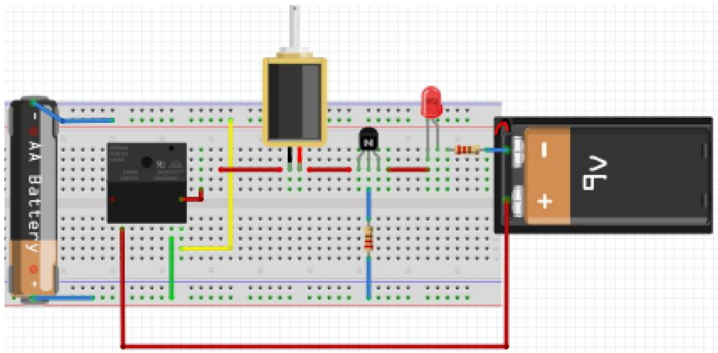
Red line - simulated load
Green and yellow lines - controlling circuit
LED - Indicator
Solenoid - Main output



Stage 3 (Add LED):

Upon making the circuit I had a few problems with the solenoid's function whenever I put in an LED with a resistor due to the lowered current. Hence why **I decided to put the LED without protection as a proof of concept**. As soon as it worked i went back to the drawing board and **re designed this unstable circuit.**



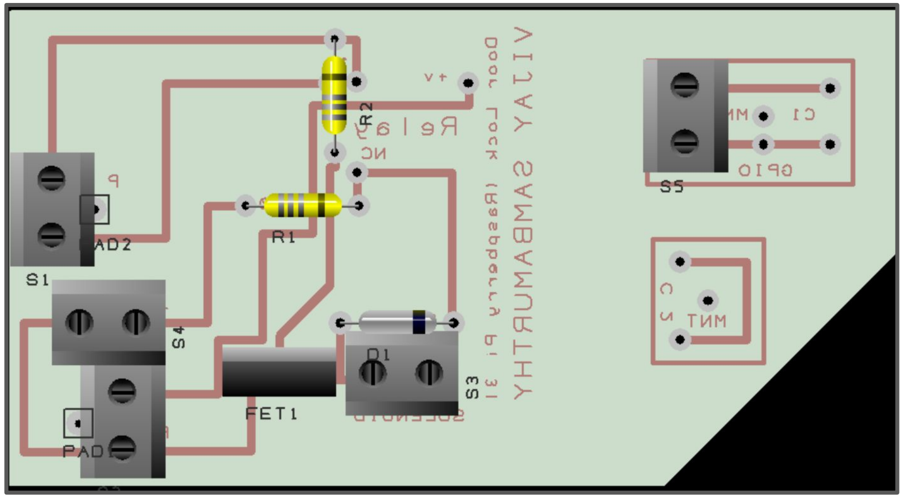


Stage 4 (Add load and fix up):

The final stage includes me **adding the 2 different power supply connectors** (simulating the pins on the raspberry pi). In addition, I protected the LED with a resistor. The smaller AA battery is a simulation of the 3v output from the **Raspberry Pi.** This is the finalized circuit design for the door lock. In this circuit, when the smaller, 3v power supply is disconnected, the solenoid turns of along with the led.

In conclusion

I built a **breadboard which replicates my intended design,** without implementing the raspberry pi yet, but rather simulated inputs. There were a few hiccups on the way to the final design in the form of failed transistors and unprotected components. The one major flaw with this circuit is the absence of diodes to protect the electromagnetic components. I intend to add this to the final PCB design. Overall, I feel as though the design was well suited for its proposed usage.



Intro

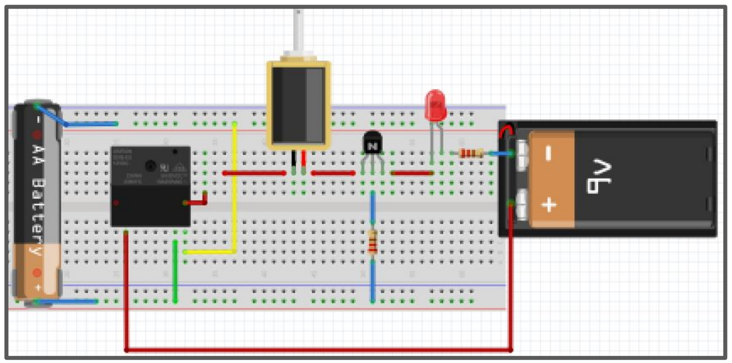
After completing the breadboarding and circuit designing phase I went ahead and built a circuit that **perfectly fits the model design**. It used **2 power supplies, one from the raspberry pi to control the relay**, and another as the heavy power supply as load for the solenoid. **Most components that are meant to be on the casing or have some sort of physical interaction are placed on flying wires which allow the components to be easily placed into the casing**. The flying wires are multi-cored wires which are more sustainable and safe.
The layout is **designed in a way that soldering is easy and does not cramp many components into a small area.**
Based of real life measurements I **designed my own relay on RealPCB as there was no preset.**

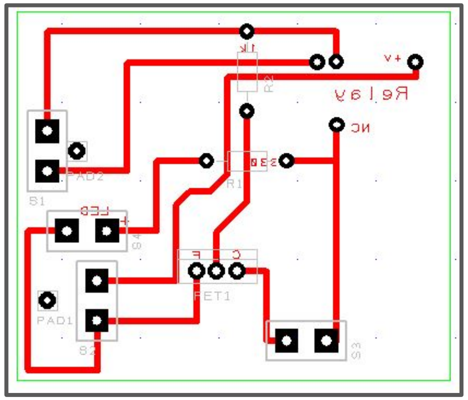
2 seperate circuit boards are designed to **detect whether or not the door is closed,** this is done by using the **raspberry pi as a input for power, sent every so often** (especially before deactivating the solenoid to re-engage the lock) which flows through the first circuit board, then, if the contact pads (which are soldered to each board) touch, a current flows through the circuit. Then, **the raspberry pi's GPIO connector is on the end of the circuit and thus relays the power to the raspberry pi, which can check if it's a full circuit.** Hence allowing the raspberry pi to check for closed doors. I used copper tracks to clearly indicate what to put where along with values for each component that needs it.

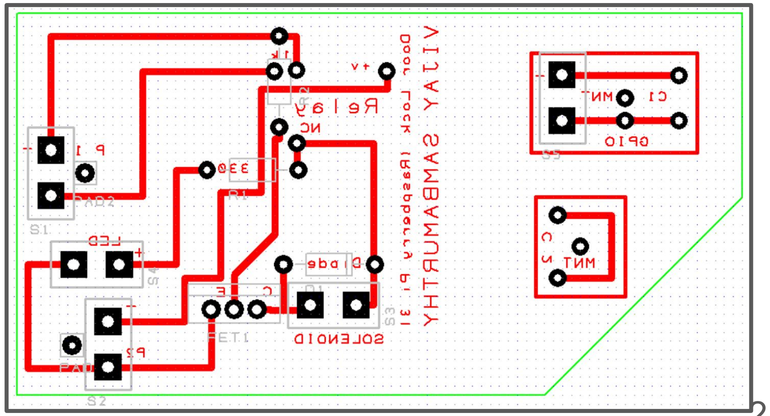
Stage 1:
Start with breadboard design and analyze how it can be transformed.

Stage 2:
Design basic circuit.

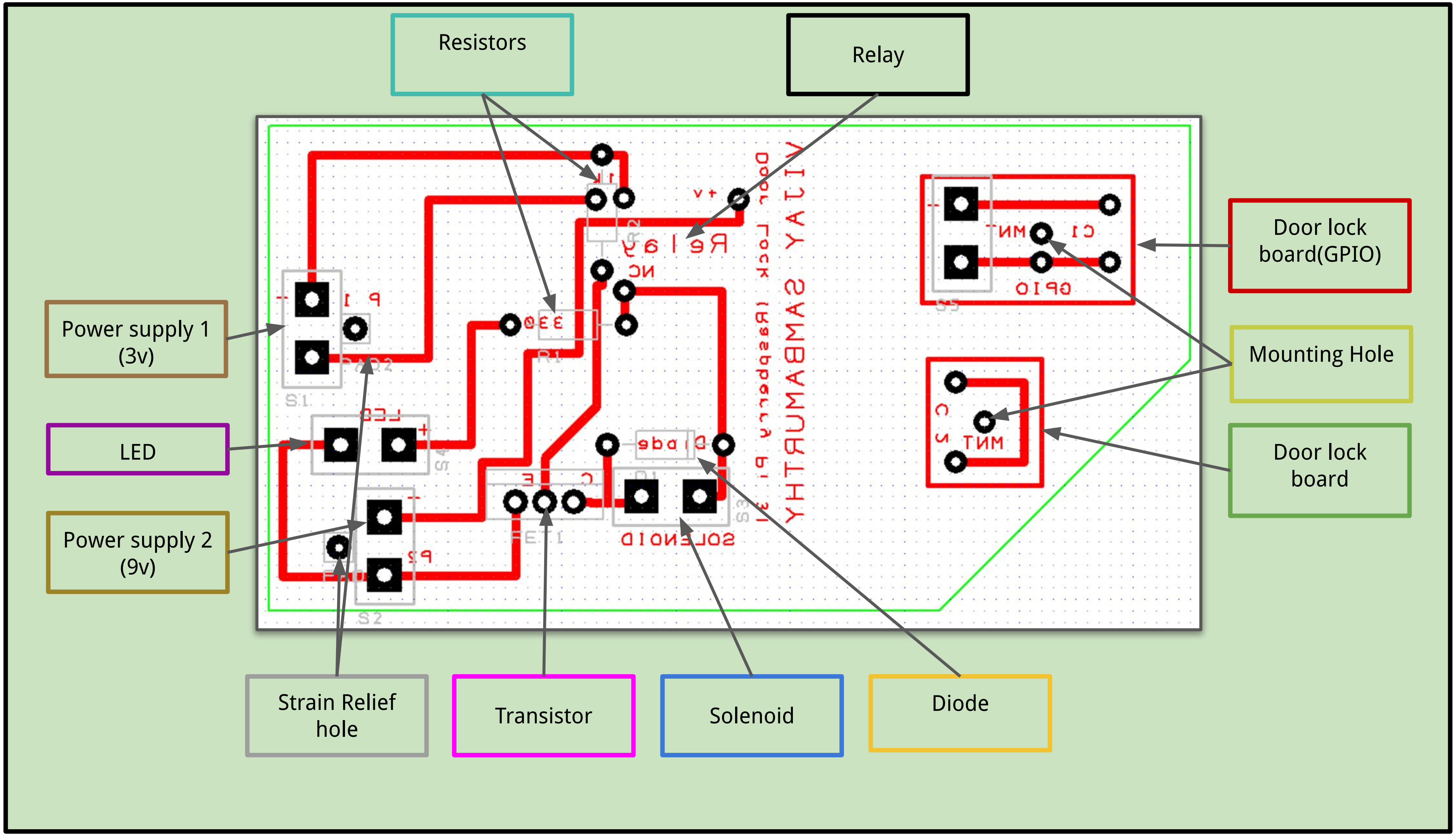
Stage 3:
Finish design by compacting circuit and adding door connectors and mounting holes with copper labels.







Since the circuit was so simple and required more coding than complex circuits, **I manually routed each connections and ensured no flying wires** by using **other components as a bypass.** I also added **mounting holes for my door boards** to ensure they stay in position (plan to use hot glue as well to assure stability), moreover, strain relief holes prevent wires from getting worn out and strained to ensure sustainability.

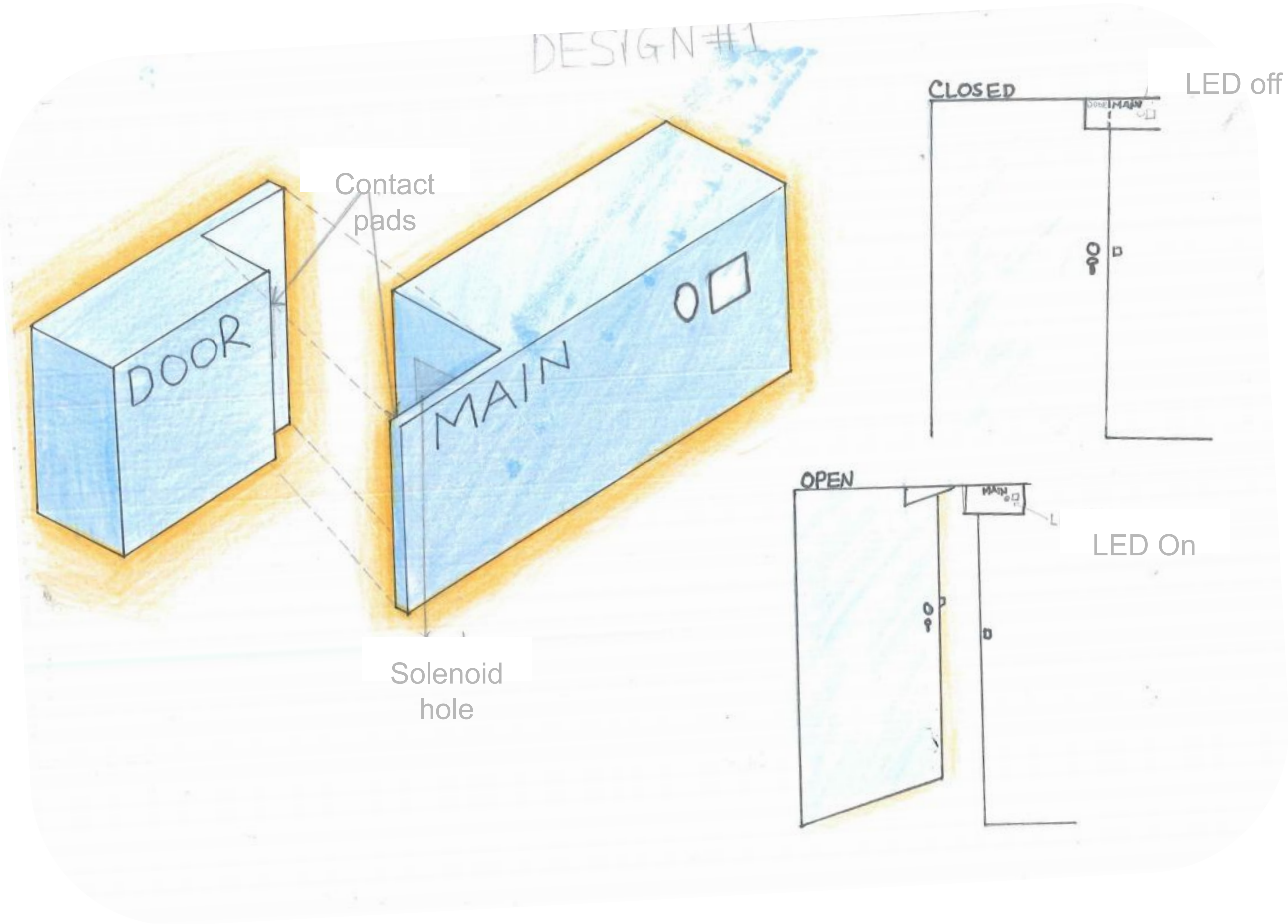


In conclusion, the circuit design seemed quite reliable and ticks all the design specifications as required. Being cheap, effective, and compact.

INITIAL CASING DESIGN 1

In the next few pages, I will be explaining the advantages and disadvantages of some designs I drew as ideas for my casing. I will analyse the 3 sketches and see which one is the best suited for my project. I compare all the designs with what I have set as the design specification from my research. **Green - Fully Satisfied, Orange - Partially Satisfied, Red - Not satisfied**

1.1)The current mode that the product is in should be clearly indicated with LEDS that are visible on the protected side.
1.2)The product should not be very vibrant but should fit in well with its surroundings and should be a suitable color like black.
5.1)The product should be designed in such a way that there is less stress on the door.
5.2)The product should be easy to mount onto any door.
6.1) The product should be no larger than an a tissue box.
7.1) The product should have no sharp edges to prevent any cuts and bruises to the user.
7.3)There should be no loose components or cables on the PCB to prevent potential harm to the user.
8.3) The product must be able to lock the solenoid without turning on when the door is open.
9.1) Easy access to the product's components is essential as with few screws repairing any faulty parts would be less tedious and time-consuming.
10.1) Must be made of ABS
11.1)The product should be able to be mass produced because there will be a significant amount of target users.



In conclusion this design **seems quite viable**, but **it's environmental impacts are quite high.** Also, if I were to use this design I would **round the corners** to ensure no potential injuries and I would try my best to compact it. In addition, it may require unmounting to be taken apart for repair or replacement of batteries. This is quite bad as it is tedious and ruins the user experience. In the real design I think it would be better to **add screw holes on the front** so that it can be easily fixed without a full disassembly.

Advantages:

- Simple design
- Output LED is visible and clear to warn the user door is open.
- Small attachment to door means less strain.
- Big enough to house multiple components

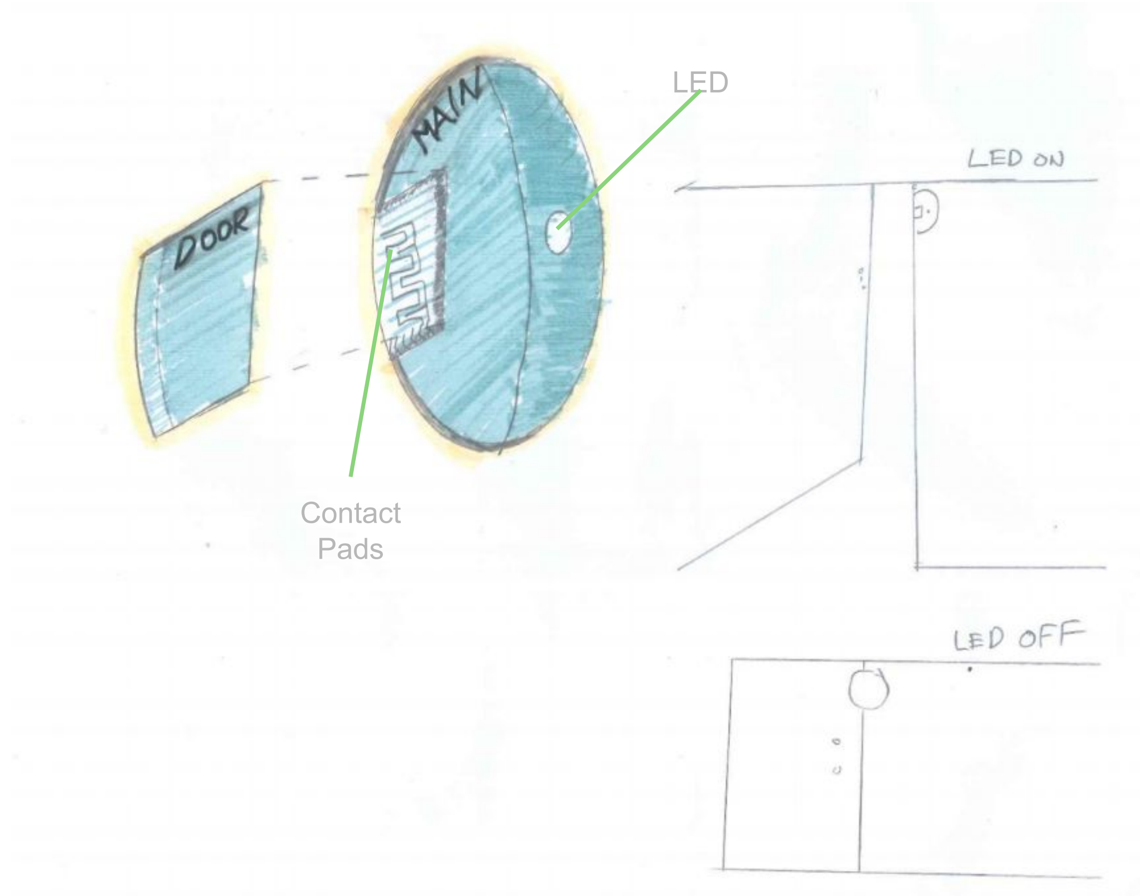
Disadvantages:

- Casing size is quite big, requires more 3D printing, harmful to environment.
- No rounded edges, can harm the user
- Due to shape, stress may be uneven and cause more wear in the long run

INITIAL CASING DESIGN 2

In the next few pages, I will be explaining the advantages and disadvantages of some designs I drew as ideas for my casing. I will analyse the 3 sketches and see which one is the best suited for my project. I compare all the designs with what I have set as the design specification from my research. **Green - Fully Satisfied, Orange - Partially Satisfied, Red - Not satisfied**

1.1)The current mode that the product is in should be clearly indicated with LEDS that are visible on the protected side.
1.2)The product should not be very vibrant but should fit in well with its surroundings and should be a suitable color like black.
5.1)The product should be designed in such a way that there is less stress on the door.
5.2)The product should be easy to mount onto any door.
6.1) The product should be no larger than an a tissue box.
7.1) The product should have no sharp edges to prevent any cuts and bruises to the user.
7.3)There should be no loose components or cables on the PCB to prevent potential harm to the user.
8.3) The product must be able to lock the solenoid without turning on when the door is open.
9.1) Easy access to the product's components is essential as with few screws repairing any faulty parts would be less tedious and time-consuming.
10.1) Must be made of ABS
11.1)The product should be able to be mass produced because there will be a significant amount of target users.



In conclusion the 2nd design is alright, however, it lacks aesthetic pleasure as it is too round and may be bigger than the first design due to lowered internal space. This lowered internal volume is due to the fact that the components are square, thus causing there to be no viable way to efficiently compact the components on the inside. In addition, it may not be easy to attach to all doors due to the weird shape taken. The shape has 1 last downside. The fact that it requires more **ABS** to print as it is a more complicated shape than the first design.

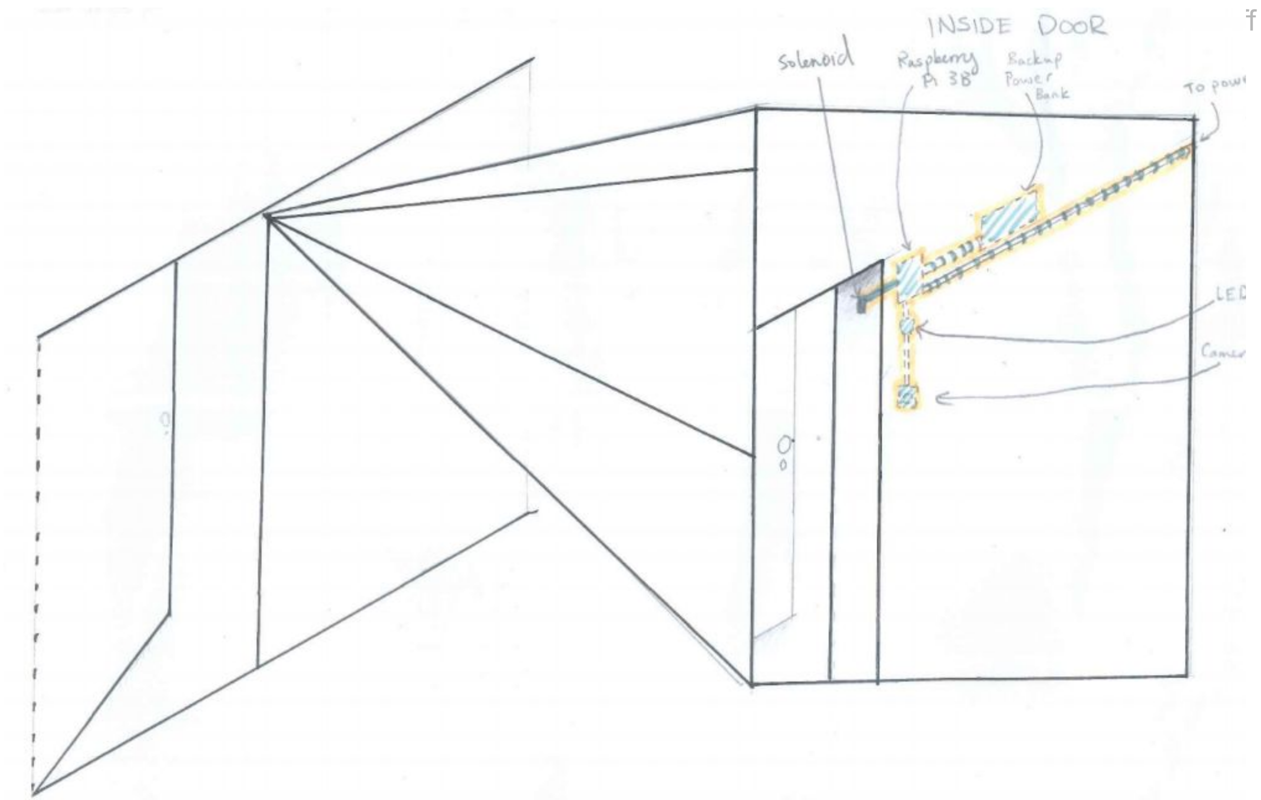
Advantages:
- Output LED is visible and clear to warn the user door is open.
- Small attachment to door means less strain.

Disadvantages:
- Casing size is quite big, requires more 3D printing, harmful to environment.
- Due to shape, stress may be uneven and cause more wear in the long run
- Difficult to mass produce

INITIAL CASING DESIGN 3

In the next few pages, I will be explaining the advantages and disadvantages of some designs I drew as ideas for my casing. I will analyse the 3 sketches and see which one is the best suited for my project. I compare all the designs with what I have set as the design specification from my research. **Green - Fully Satisfied, Orange - Partially Satisfied, Red - Not satisfied**

1.1)The current mode that the product is in should be clearly indicated with LEDS that are visible on the protected side.
1.2)The product should not be very vibrant but should fit in well with its surroundings and should be a suitable color like black.
5.1)The product should be designed in such a way that there is less stress on the door.
5.2)The product should be easy to mount onto any door.
6.1) The product should be no larger than an a tissue box.
7.1) The product should have no sharp edges to prevent any cuts and bruises to the user.
7.3)There should be no loose components or cables on the PCB to prevent potential harm to the user.
8.3) The product must be able to lock the solenoid without turning on when the door is open.
9.1) Easy access to the product's components is essential as with few screws repairing any faulty parts would be less tedious and time-consuming.
10.1) Must be made of ABS
11.1)The product should be able to be mass produced because there will be a significant amount of target users.

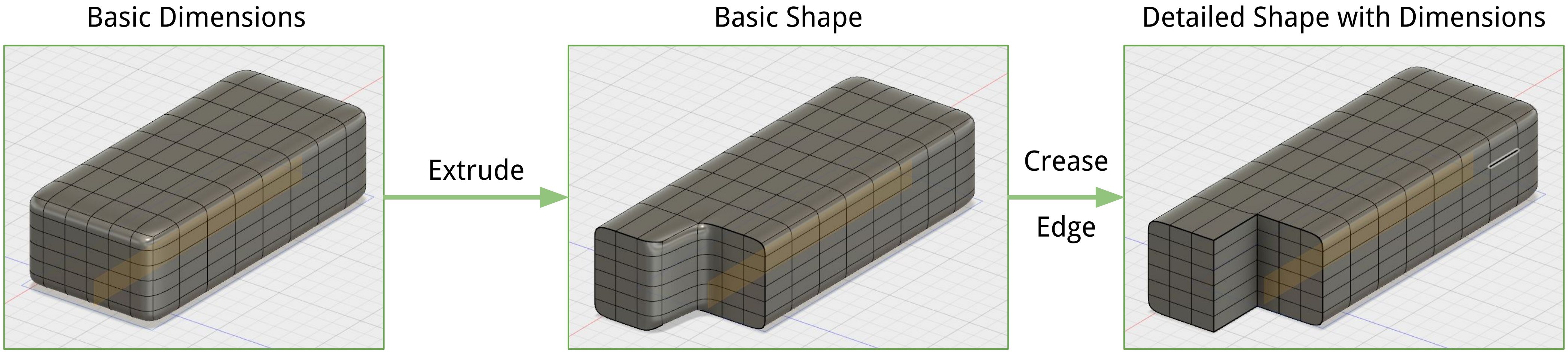


The final design, "the door design" is more complicated than the previous two. However, it has one huge advantage, there is no **ABS** required for printing as there is truly no casing. This may seem quite beneficial but causes permanent damage to the owner's property and cannot be mass manufactured as it requires people to go to install which would increase costs vastly. This is also in addition to the fact that the components are not easy to access due to the location of them inside the walls. A silver lining to this design is that the design is seamless and makes the product feel like a part of the house.

Advantages:
- Simple design
- Output LED is visible and clear to warn the user door is open.
- Small attachment to door means less strain.
- Big enough to house multiple components

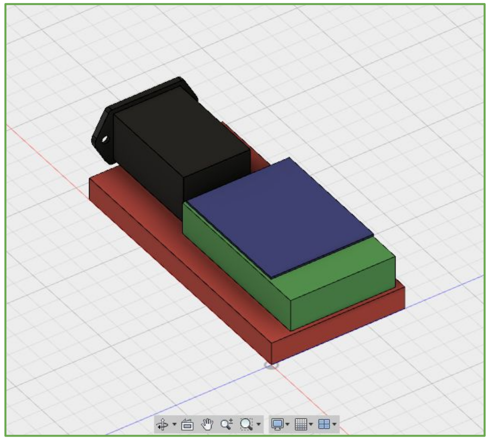
Disadvantages:
- Casing size is quite big, requires more 3D printing, harmful to environment.
- No rounded edges, can harm the user
- Due to shape, stress may be uneven and cause more wear in the long run

Stage 1 - General Form Development



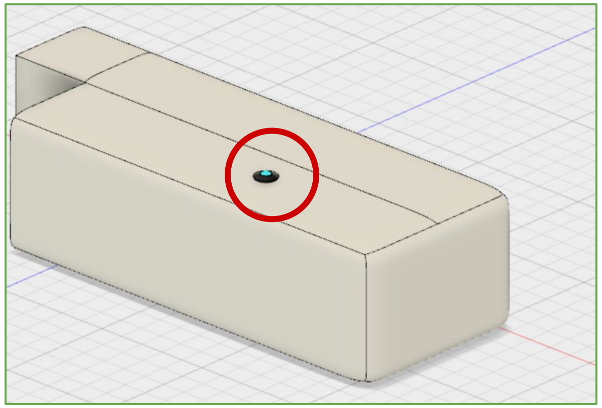
I started out with a **simple cuboid shape** with the dimensions I calculated by measuring the sizes of each individual component. Once I had the size ready, I extruded one edge, which will house the solenoid, and **used the 'crease edge' function** which causes it to become flat. I did this so that the 2 parts of my product connect well with each other without much contact. There are more length, height and width edges in order to make the design much more smooth and simple, without many curves. **I did the same for the component which needs to be attached to the door, however that one is much smaller** as there is barely anything that needs to go on the inside. While there aren't many components inside the door component, it still needs to look sleek and fit in with the large main mount so it has to match the height and width. It has been designed in such a way that the **door has no extra weight on it**. This means that the user can still use the product without any noticeable difference in the effort required to open/close the door.

Stage 2 - Component insertion

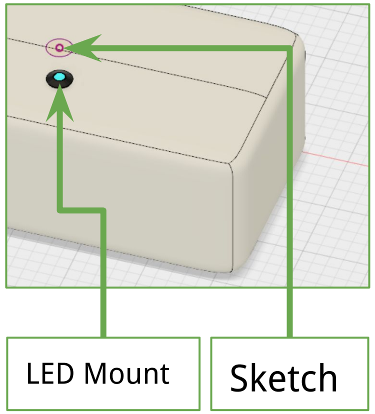


Next, **I measured all my components that I plan to use** inside the casing **using a set of digital calipers (ensuring accurate measurements).** Once I had dimensions, **I built basic cuboids to represent the maximum possible size** of each component and thus could align any new components inside my casing with ease. This also **allowed me to build ribbings (as seen in step 9)**. In addition, I soon found a high quality, detailed 3D model of the **Raspberry Pi Model 3 B** (the one I am going to use) and also implemented that into the casing. The different colors represent different components to make it clearer. Soon after designing these, I came up with more detailed designs later on for better identification.

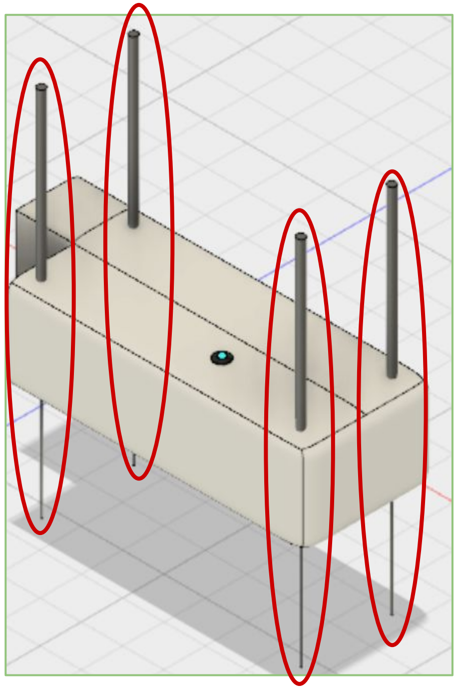
Stage 3 - LED Insertion



With the LED positioned where I would like it to be seated on my casing, I went ahead and **projected a sketch using the outline of the LED mount. With this sketch, I used extrude to cut into the casing.** However, **I did not cut all the way through** so that when shelling, Fusion **will leave a small step for the LED to be seated on without strain**. Thanks to the components I embedded in stage 2, I could position my LED mount without affecting the arrangement of components or need to cramp parts together in a unorganized fashion, avoiding scratches and damage on the inside of the casing. I also **used an LED Mount instead of just laying my LED onto the casing**, therefore I can easily remove my PCB.



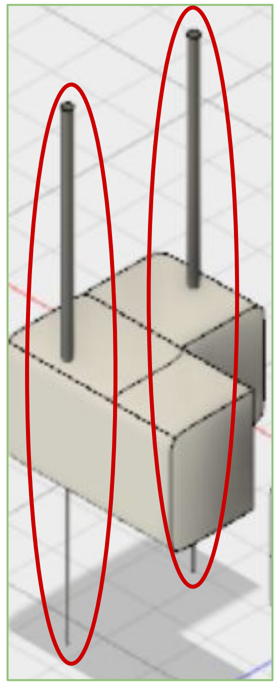
Stage 4 - Casing Screws



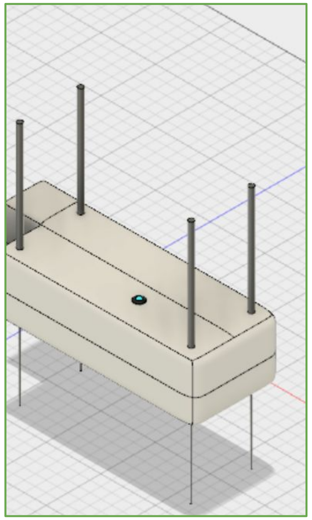
Once I had my main components in place, **I began to add pre-modeled casing screws** which allow me to once again place components wisely and get an idea of the organization inside. With the screws in place I could use **the 'combine' feature** in Fusion 360 which has 2 modes; join and cut. I used cut. This allowed my casing to have the **right size holes for the screws** and thus **not need any special drilling to fit correct size screws.** Before I accurately place my screws, I made a construction plane which allows me to get a precise position of the middle of my design.

I need to keep my screws such that the point where they become thinner is just above my middle position. This is due to the fact that the middle will be closest to where my split line is going to go. In addition, I used a sketch projected above the design to align the screws with each other such that it makes the positioning even.

These screws are inserted in order to seal the casing tight, however as seen in step 6, I add a step on the split line to vastly reduce strain on the screws, this keeps stress even along the entire product. Thus ensuring that the product can last for a longer time, saving the need to purchase new products frequently.



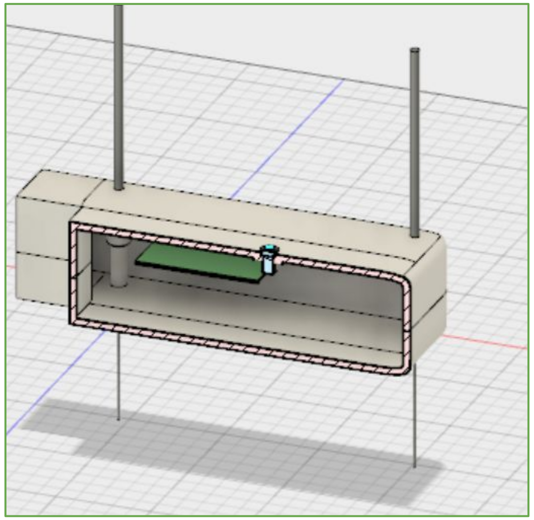
Stage 5 - Shell & Split



In Fusion 360, I used the **'shell' function that hollows out the inside of my casing with my desired thickness of a wall.** I used a **3.5mm casing wall** to keep the product lightweight and also durable. The shelling also automatically give pockets for the components that I cut out as seen in steps 3 and 4.

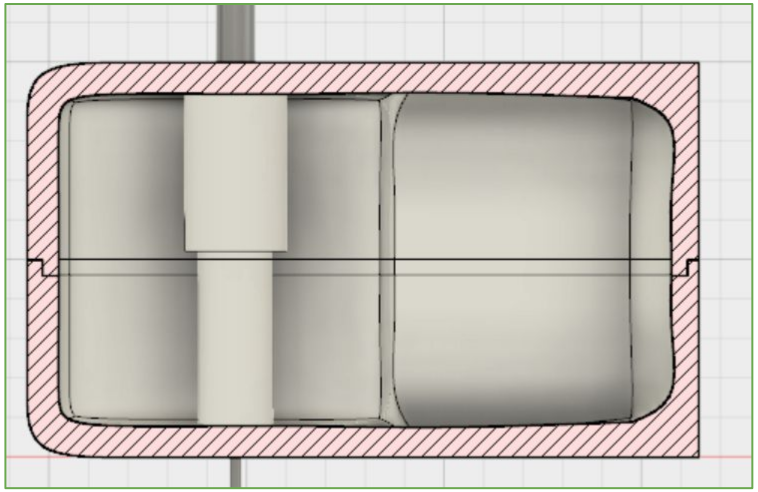
With every component that requires a small pocket in place on my product, **I was ready to split the casing in half.** This is done to make access to internal components effortless and hence parts can be replaced in the unlikely condition of a failure. In order to split, I used my previously created construction plane, which was near the middle of the casing, and cut the casing in half.

With the general outer casing in place, I realized that there is still room for movement due to the split, so my next step is to build a step up on the split line.

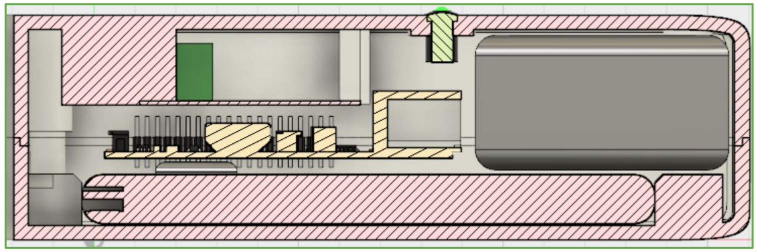


Cross Section

Stage 6 - Staggered Split Split

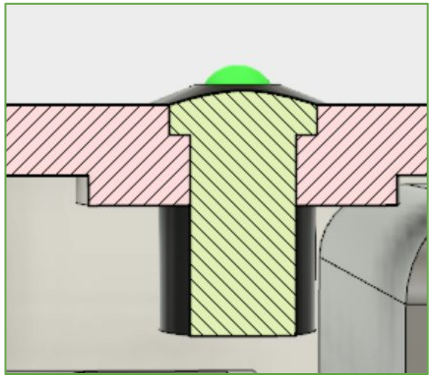


With my split complete, I began to add a staggered split line, this is easily done by creating a projected sketch of the current border of the product. Once the projection is done, shift it by half the thickness to place the offset sketch in the middle of the border. I shifted one of the faces on the bottom half of the product up by **2 mm** and did the opposite to the same face on the top side. **This not only allows the product to be more secure, but also allows the strews to carry less strain**.



Cross section. Here the split line can be seen on both sides of the product, this does not affect anything inside the product at all.

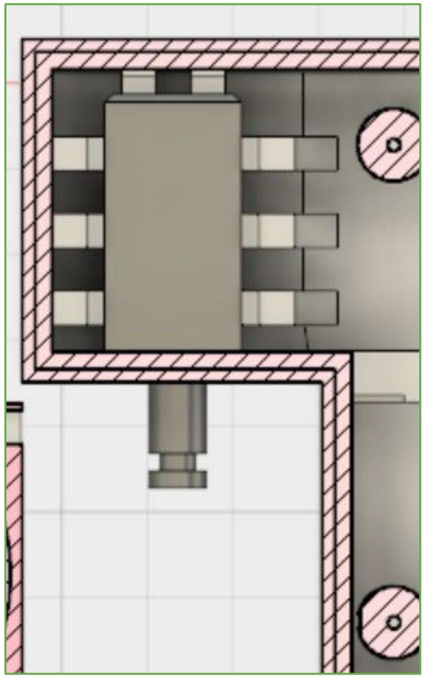
Stage 7 - Component holders



Due to the way I organized my order, the components were inserted before the splitting of the casing. This allowed me to **position them such that the Split causes there to be a small hole to house the component**. To the left there is a picture of the **LED Mount which has a small holder built into the casing**, this reduces costs as I do not require to use hot glue on top of relieving strain on the mount.

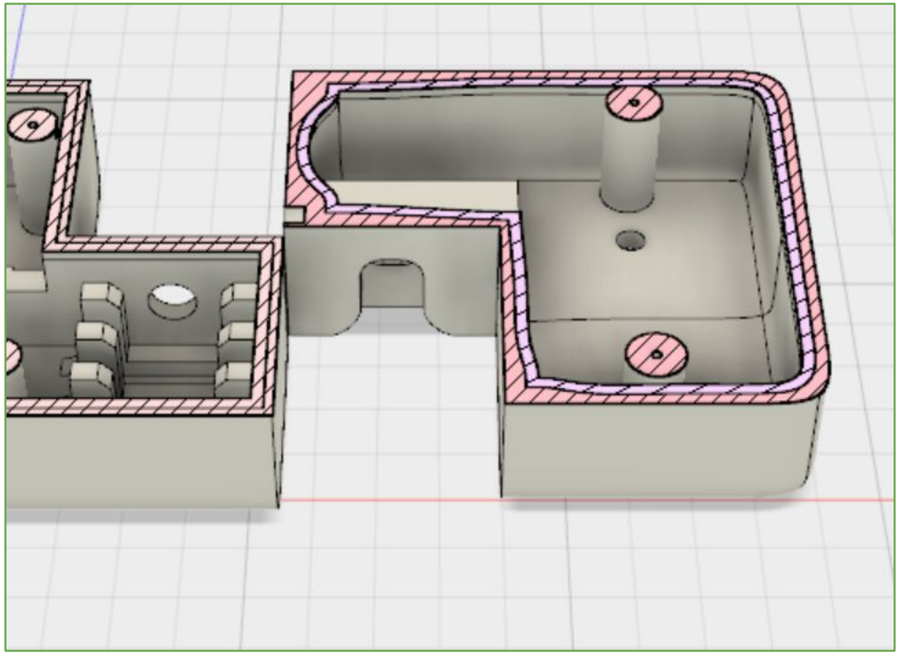
Quality check: The supporting structure is not blocking other components. Ensure the extra housing is same thickness as the original shell. This prevents issues with colliding components, gives more space inside the product, and avoids giving the product any weak spots for intruders to target.

Stage 8 - Joint and Solenoid

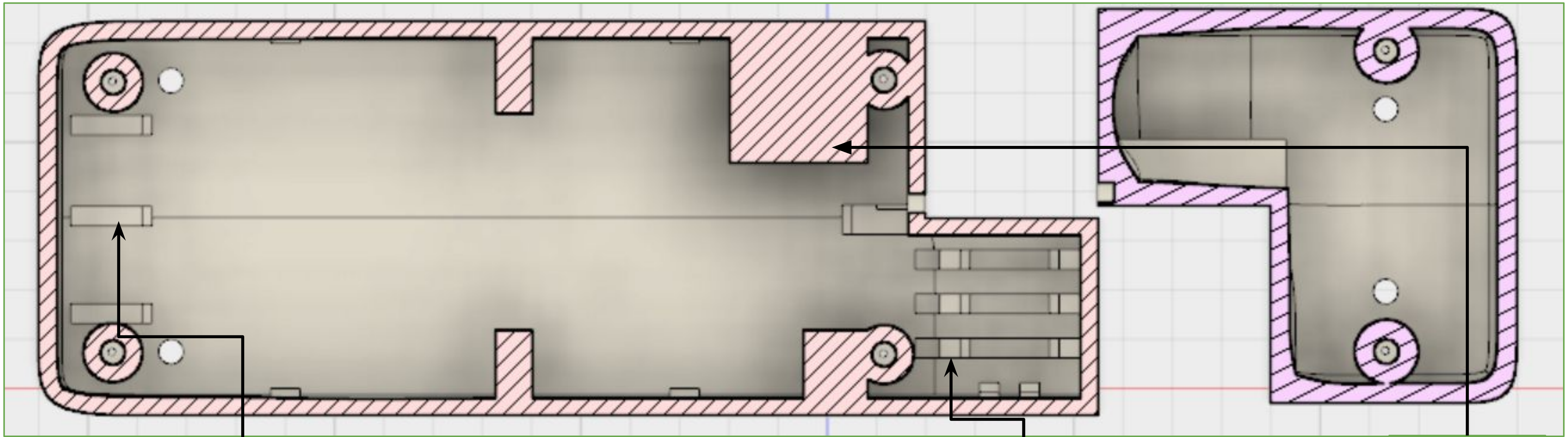


One of the final stages in the production of my product's 3D printed casing is to add holes for the solenoid and give it some freedom to move such that it can slide into the holder securely. I started off by building a joint that indicates the way in which the solenoid can move, this allowed me to visualize where I must keep the holes such that the door locks every time.

With the joint in place, I then had the hole built into the main component and added an extra extruded area to the door component. I then cut out the shapes by using sketches and press pull to house the locked solenoid. This was quite a tricky task as it did involve some planning ahead. I also did realize that it would be difficult to do all types of doors due to the different types of hinges.



Stage 9 - Ribbings



Ribbing to hold the power bank.

Ribbing to hold the Solenoid.

Ribbing to hold PCB above.

To ensure that my components sit in place **I used ribbings**. These ribbings were made **with simple extrusions from the product**. I first built sketches around each component and then drew rectangles where ever I planned to put the ribbings. Then I used Fusion's "extrude" function to give it some height. Now the **components do not move** in addition to the fact the internals are organized and simple. **This also allows people to easily diagnose problems and fix them.**

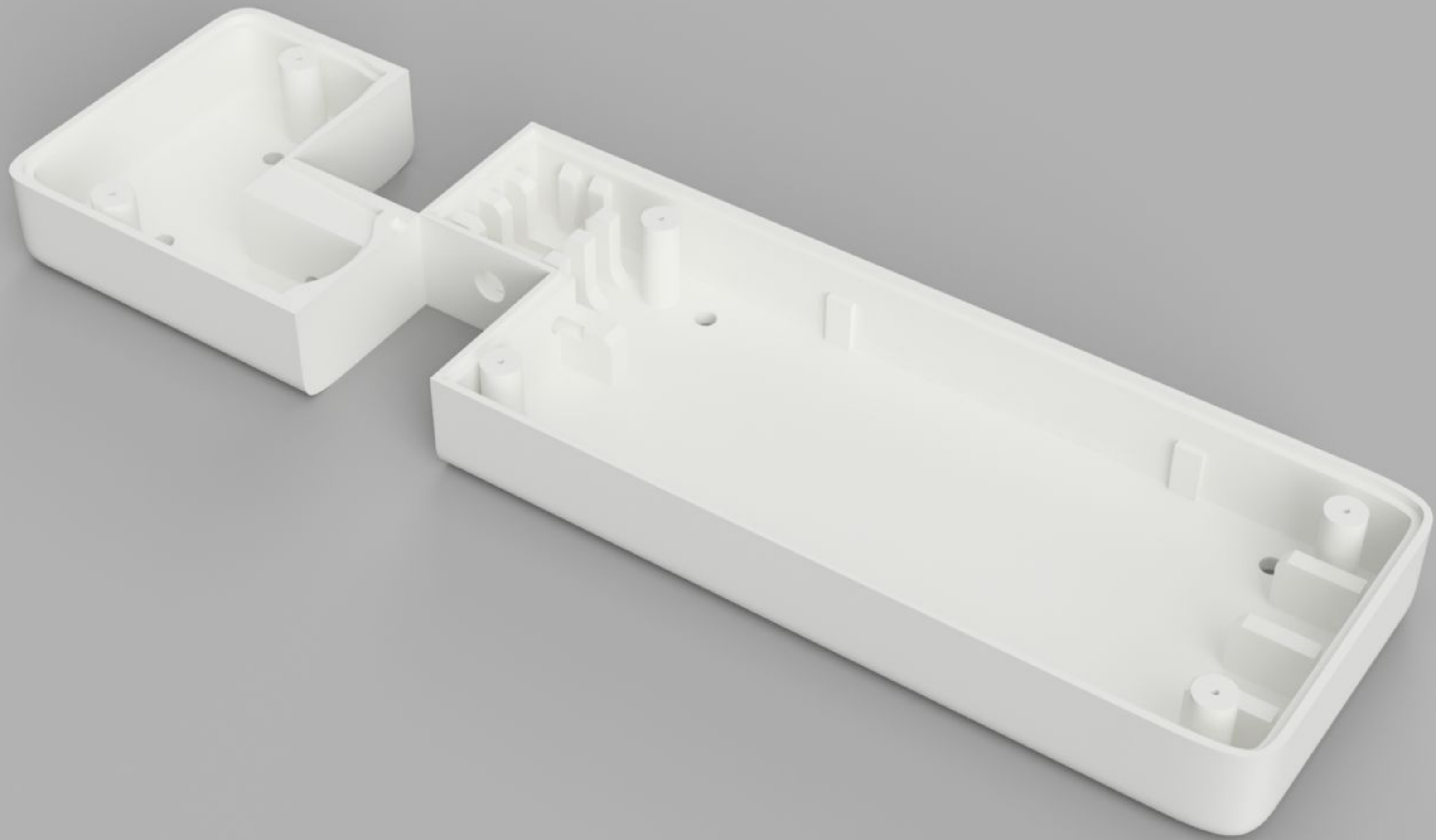
Conclusion

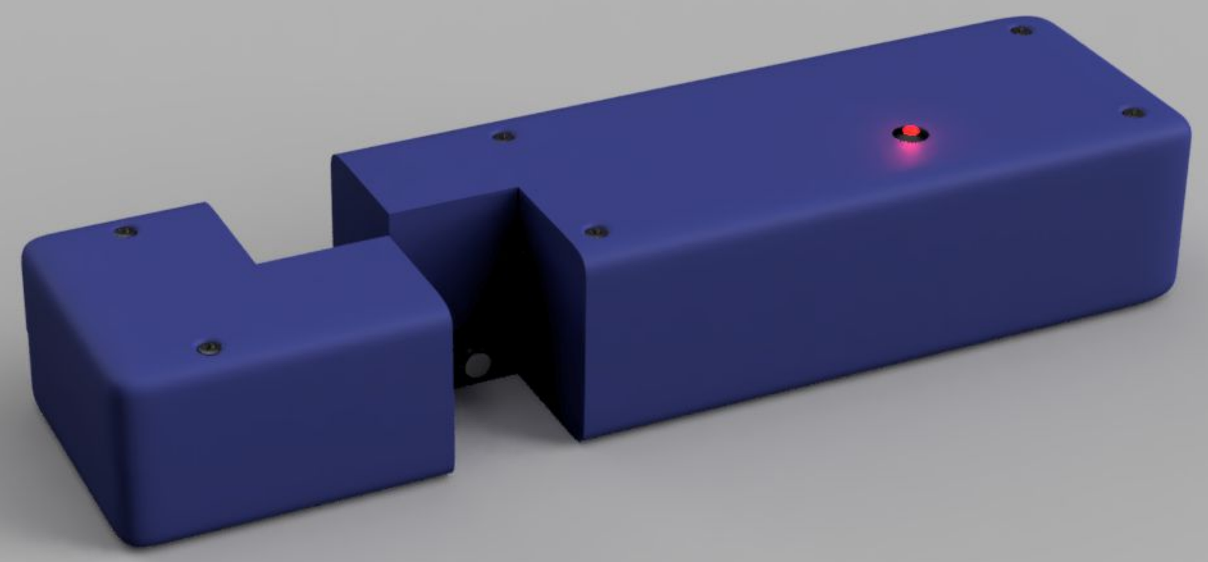
Overall, the building of my 3D product seemed very simple, quick and easy due to the software I was using, **Fusion 360.** It truly has many great features but it allowed me to add intricate details to my product such as the ribbings and holders for my internal components. In the upcoming few slides are pictures of Renderings of my 3D model in different colors to display the various colors the product comes in. The end result was virtually an exact replica of what I had in mind and ticks all the boxes it needs to in the design spec.

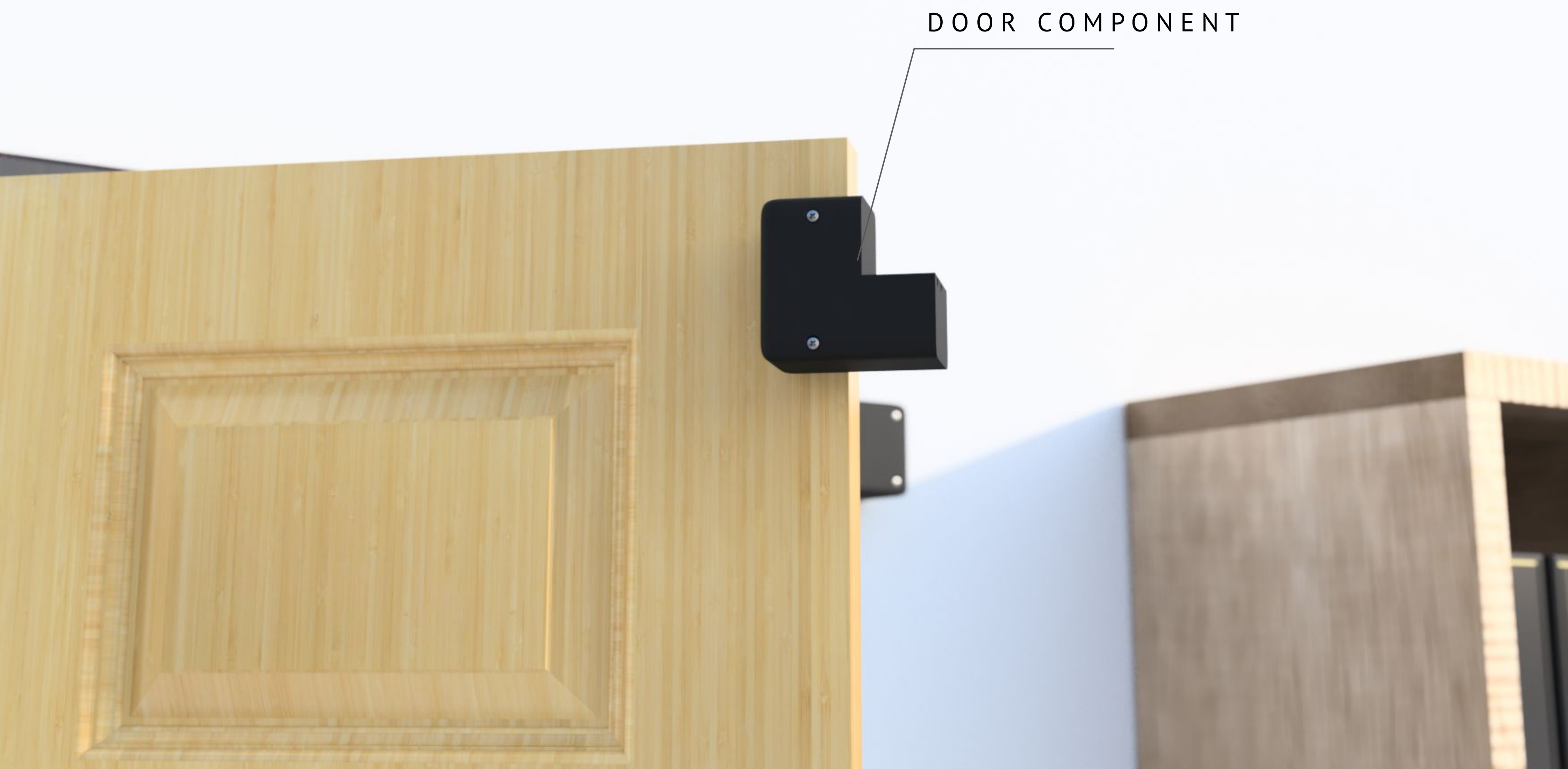
Pros: The ribbings hold components in place; the 2 parts fit together smoothly; round edges to prevent safety issues and staggered split line to reduce strain on screws.

Cons:The product had to be quite large to fit the power bank and due to restrictions, the thickness of the product was limited to 4mm.

RENDERING OF BOTTOM CASINGS

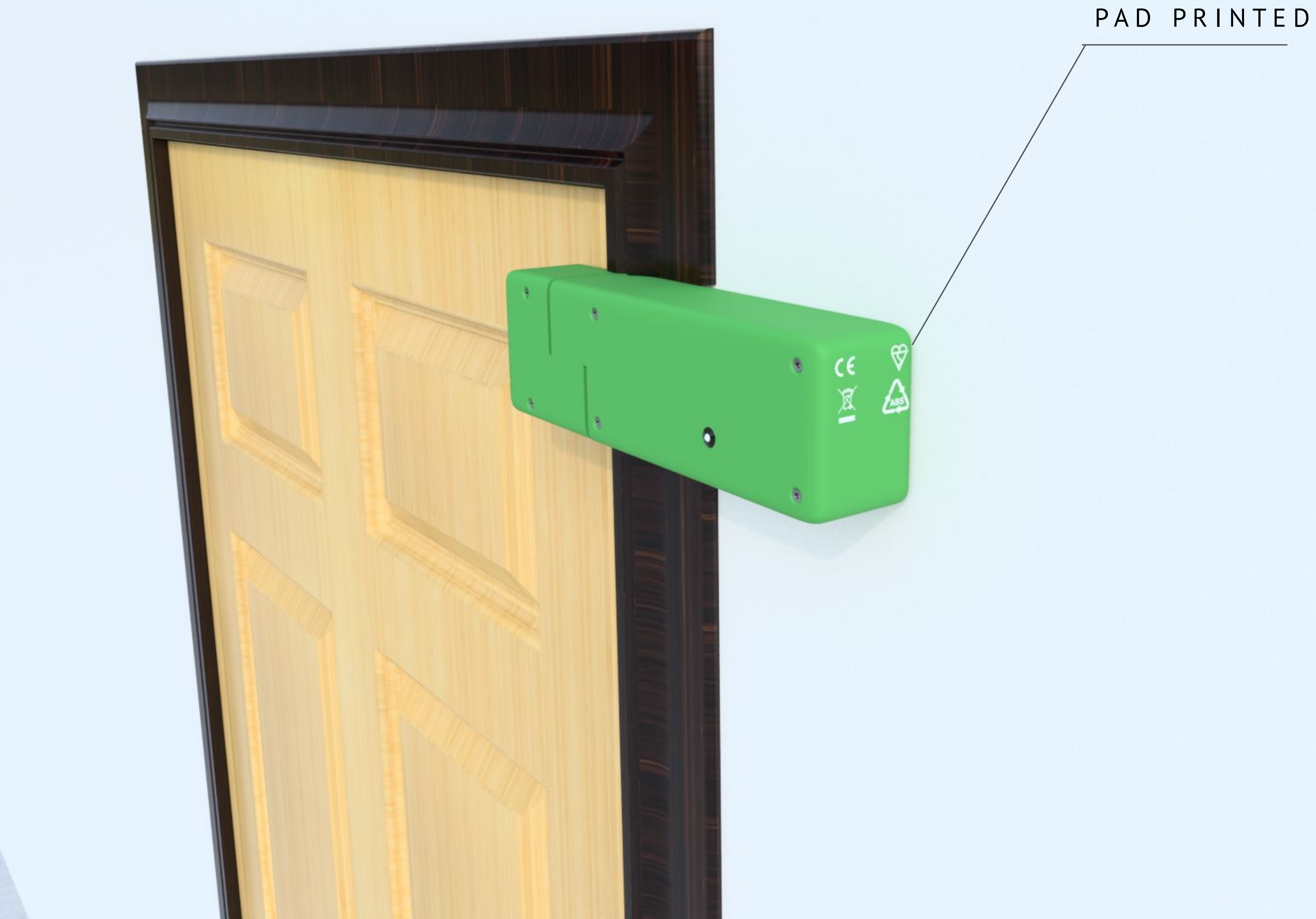


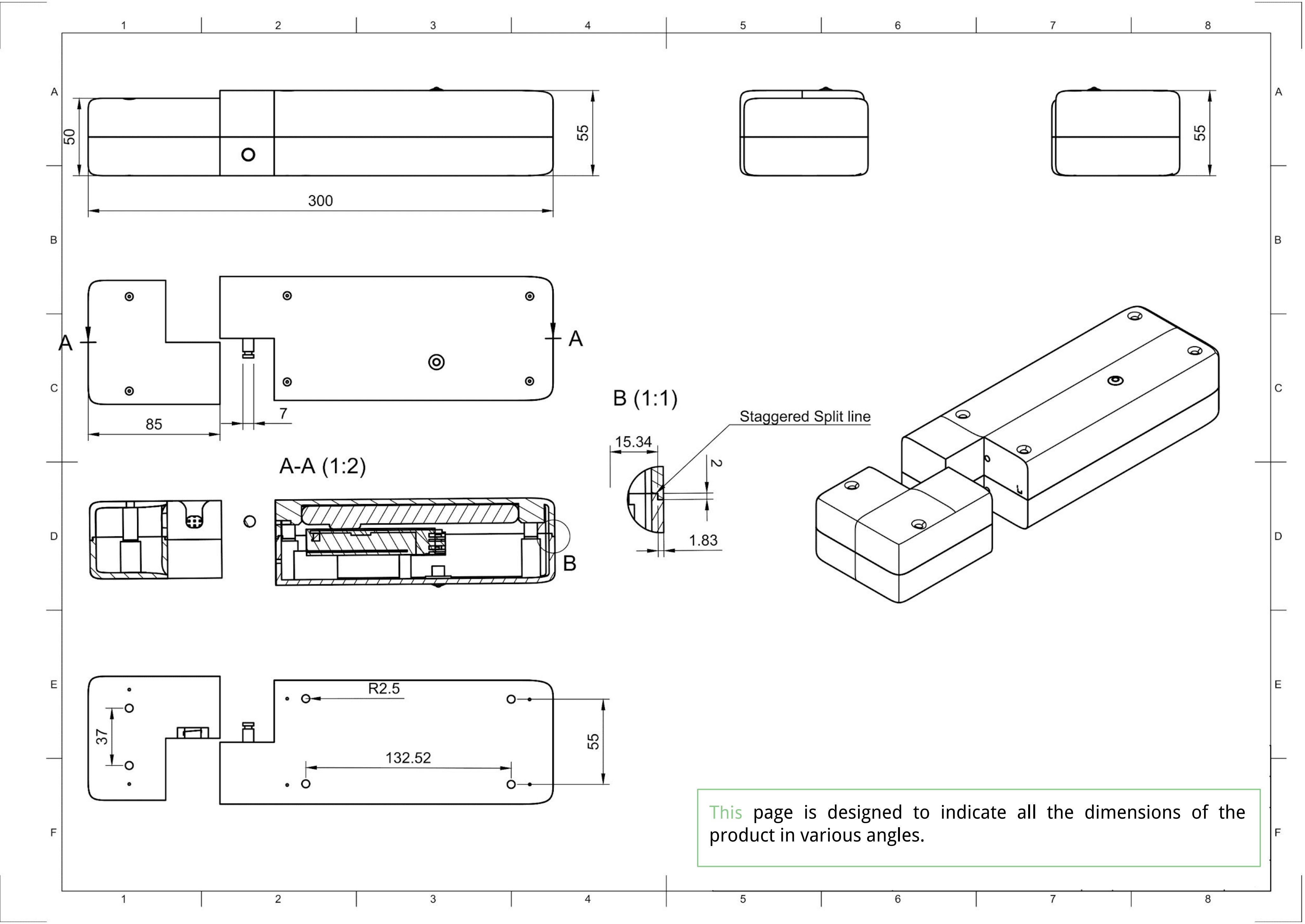



DOOR COMPONENT

RENDERING OF PRODUCT

RENDERING OF PRODUCT (GREEN)

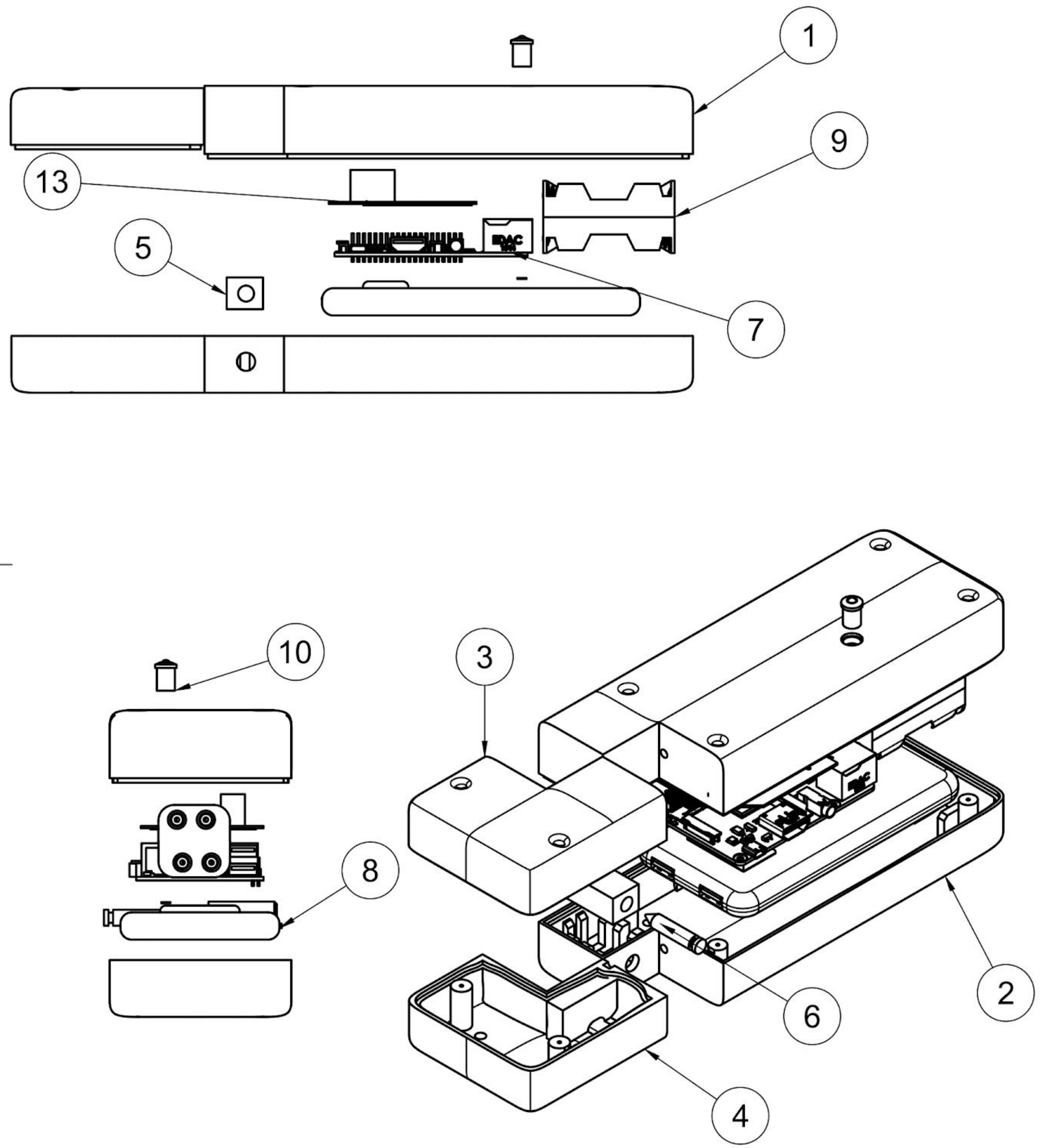




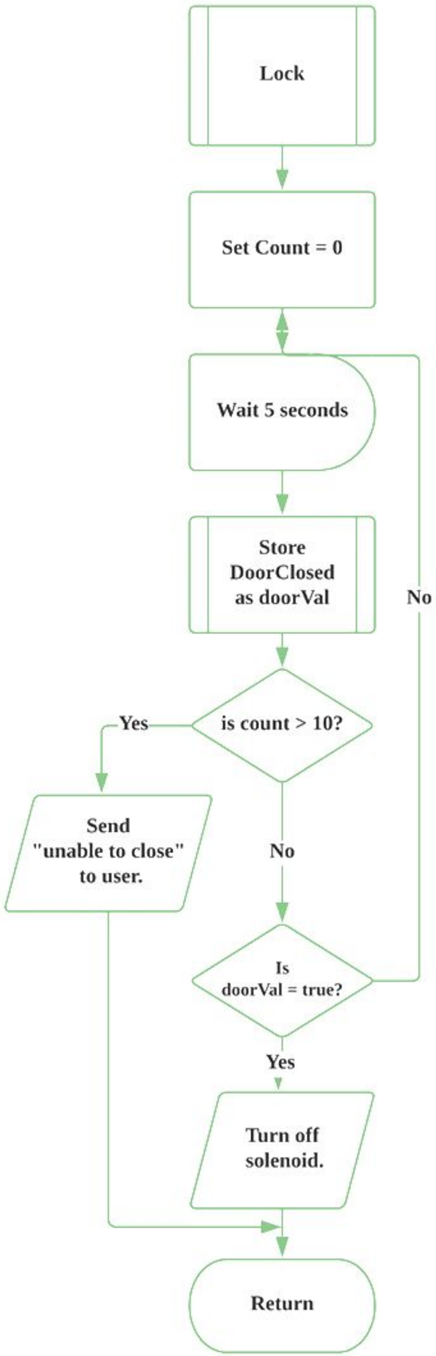
This page is designed to indicate all the dimensions of the product in various angles.

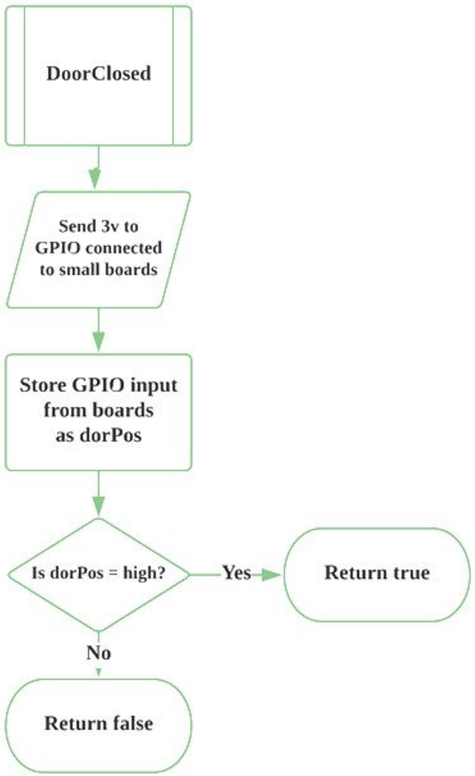
EXPLODED VIEW

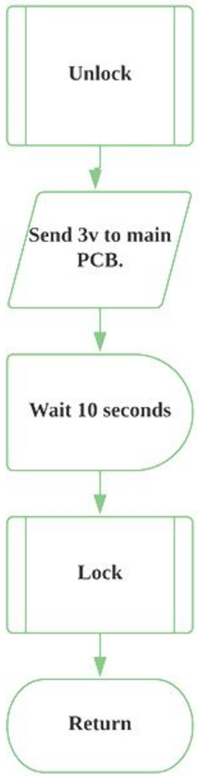
This page shows how the product is assembled and all the parts and components within using both 3D and 2D to demonstrate. It also gives details on the materials used.

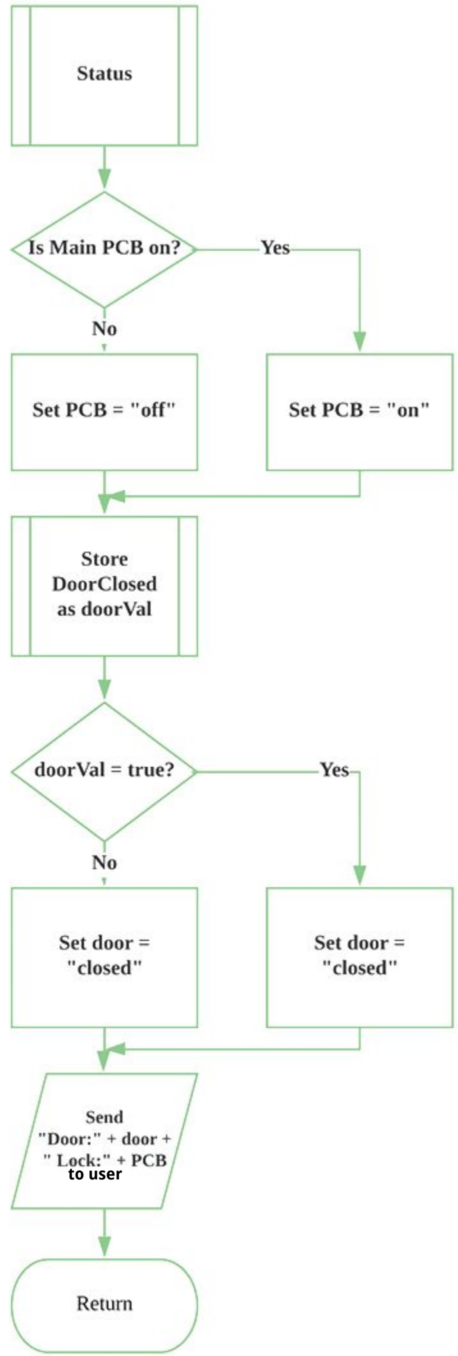


Parts List			
Item No.	Qty	Item Name	Material
1	1	Main Top	ABS Plastic
2	1	Main Bottom	ABS Plastic
3	1	Door Top	ABS Plastic
4	1	Door Bottom	ABS Plastic
5	1	Solenoid	Steel
6	1	Piston	Steel
7	1	Raspberry Pi Model 3 B	Plastic,Metals
8	1	Power Bank	Lithium Ion Batterty
9	1	AA Battery Holder	Polypropylene
10	1	LED Mount	ABS Plastic
11	1	LED	Gallium Arsenide
12	1	Casing Screws	Steel
13	1	PCB	Copper and Fiberglass



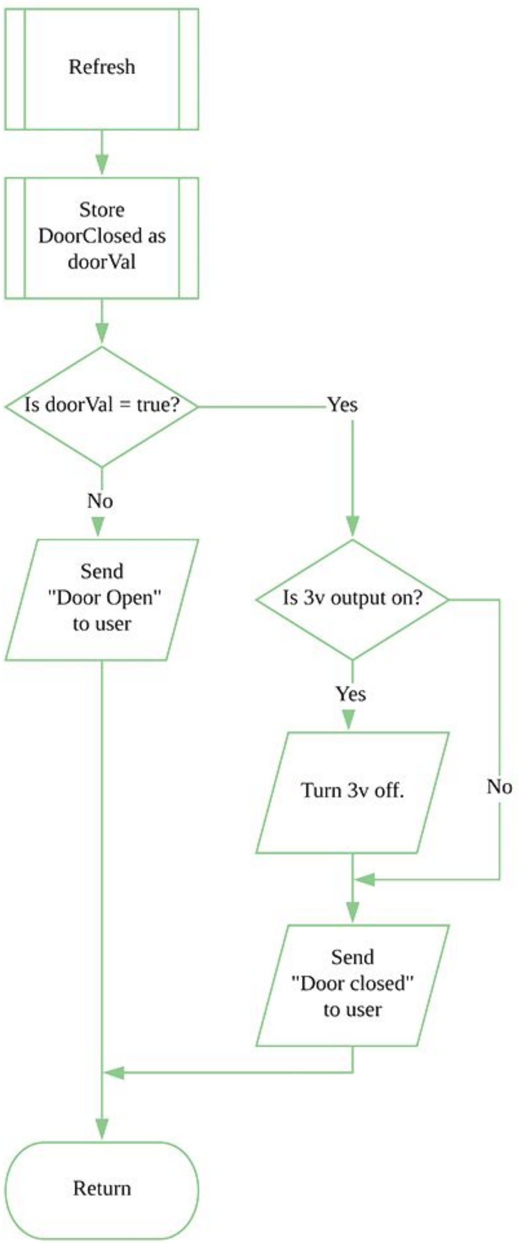


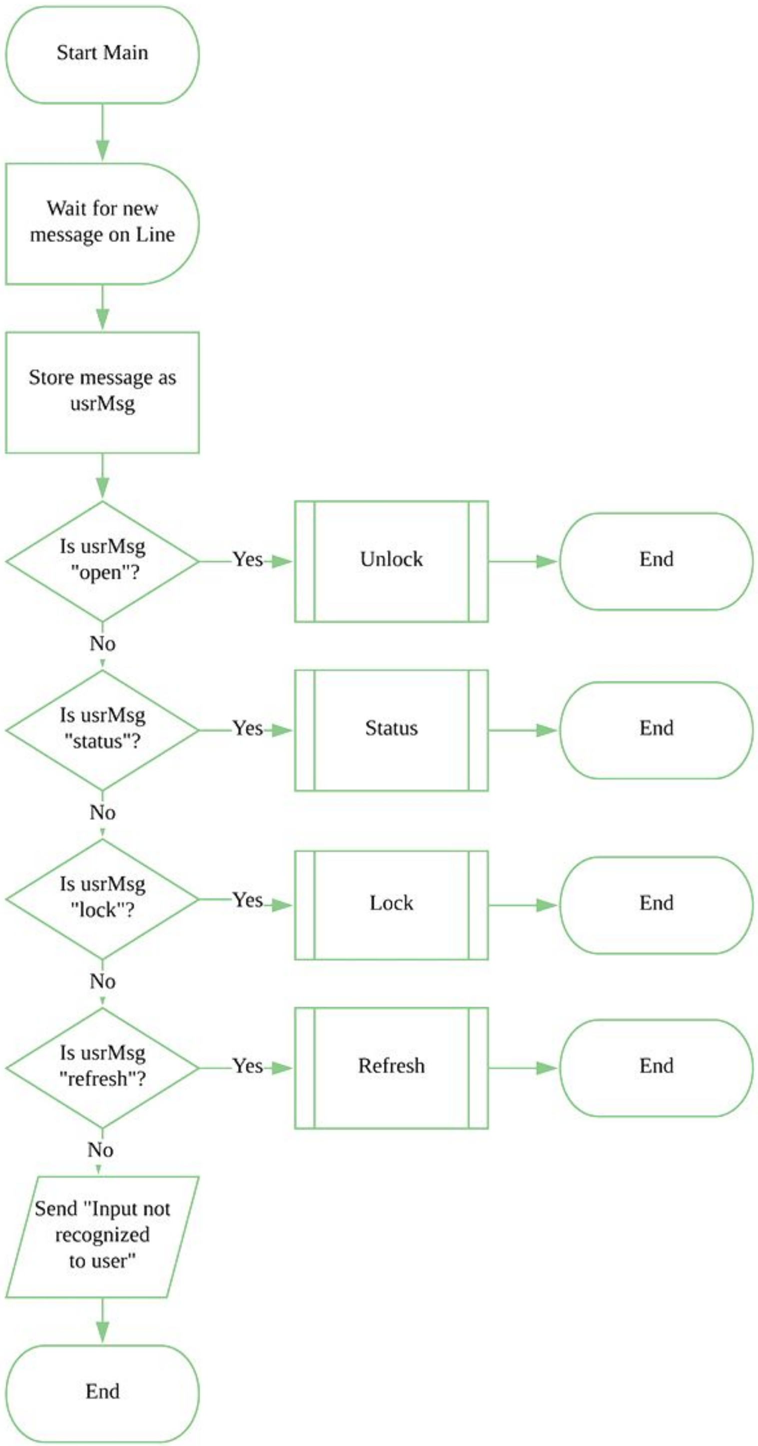




These are the **main subprograms** which will be referenced in the next page. Short and simple subprograms break down tasks into simpler ones such that programming becomes easier and more effective. In addition, more efficiency is gained as less code is needed and it makes debugging easier.

CODE FLOW CHART





In conclusion, the flow chart gives a good idea of what and how the Raspberry Pi needs to be coded. Thankfully, **Skype offers an API** which makes reading messages much easier (more detail in the manufacturing section).

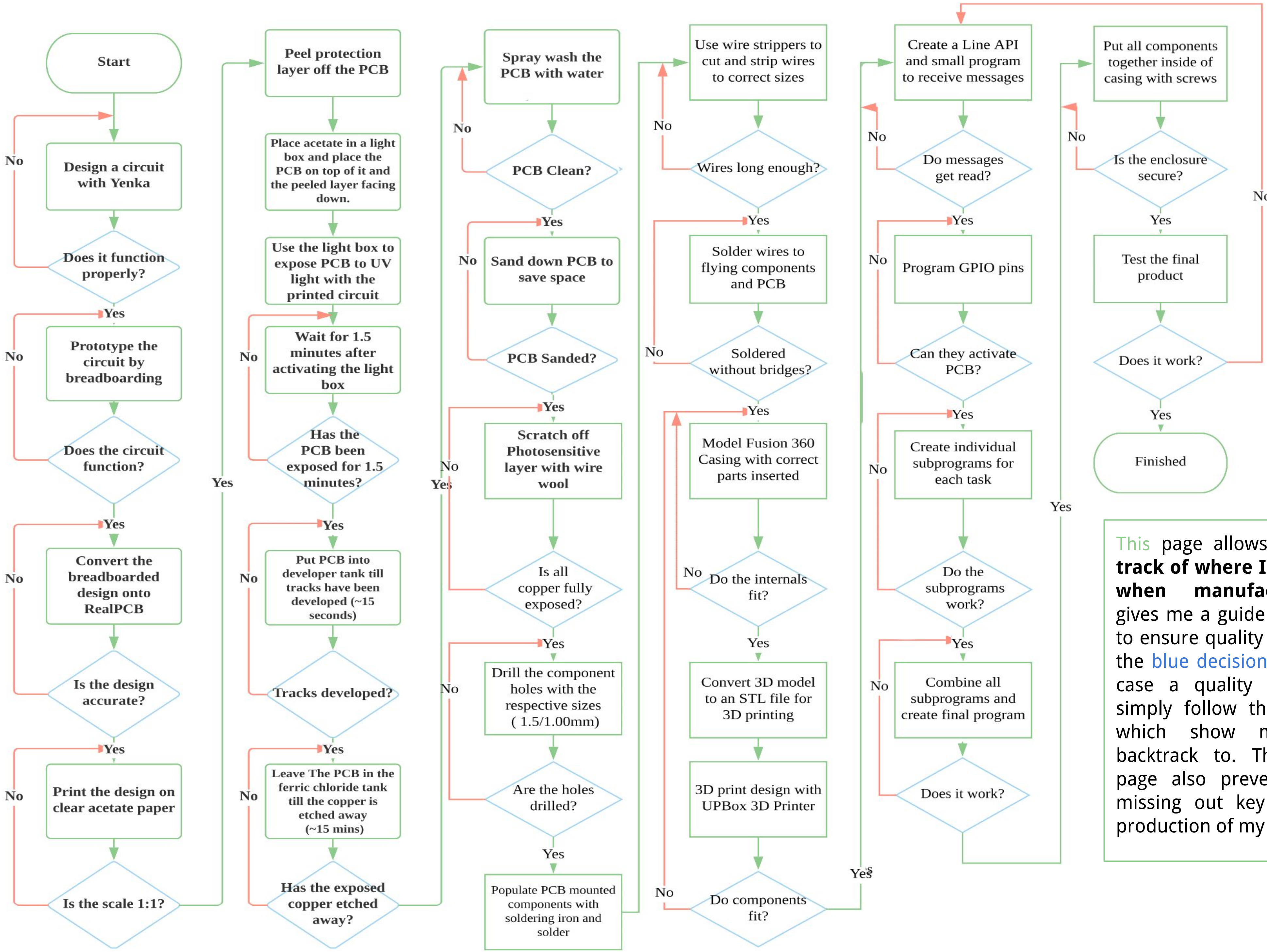
I used **many subprograms to make it easy for debugging and changes.** I also plan to use global variables to allow the delays to be easily modified with software updates.

Additionally: In the main program where the decision boxes check for a certain input, I plan to make it more of a variation, for example, **instead of just open, it would recognize "unlock", "open door", etc.** This improves quality and allows the reader to better use the lock with freedom.

This page references subprograms from the previous page denoted with the symbol:

** The camera's motion capture is more elaborate and requires equations, this shall be touched up on in the manufacture section.

PRODUCTION FLOW CHART



This page allows me to **keep track of where I currently am when manufacturing** and gives me a guide of what to do to ensure quality as denoted by the blue decision boxes. In the case a quality check fails, I simply follow the red arrows which show me what to backtrack to. Thankfully, this page also prevents me from missing out key steps in the production of my product.

GANTT CHART

Following the previous slide, I created a simple Gantt Chart to keep track of time, this ensures my product will be completed within the given time and also informs me if I am going to fast (which could reduce quality) or if I am going slow. The details inside the highlighted weeks indicate key points and details I need to remember when completing the tasks. Some may be safety points, some may be breakdowns of tasks.

TASKS	WEEK 1	WEEK 2	WEEK 3	WEEK 4	WEEK 5	WEEK 6	WEEK 7
DESIGN PCB -Using Real PCB							
DEVELOP PCB -Etching Process		Goggles and gloves.					
SOLDER PCB -Follow safety precautions		Wear goggles,apron.	Careful of gas released from solder.				
MODEL & DESIGN ENCLOSURE -Using Fusion 360			Build general Casing.	Add details (e.g. ribbings).			
PRINT ENCLOSURE/CASING -Using Fusion 360 + Up Box 3D printer					Hot right after printing.		
CODE RASPBERRY PI -Follow charts -Use Python and Skype API				Get Skype API working.	Build subprograms.	Combine subprograms.	
ASSEMBLE PRODUCT							

While my product does benefit users in many ways, there are impacts that appear in social, moral,environmental and sustainability categories if my product was mass produced. The following 2 pages evaluate such issues in detail and states a few ways in which I could reduce the negative impact of such issues.

Social Impacts

My product is designed to be used by those wanting automated security and monitoring no matter where they are. If motion is detected inside the user's property, they are almost instantaneously warned with an image captured by the camera on the device. This will allow neighborhoods to apprehend known thieves with solid evidence. In addition, this saves owner's money as they do not require to purchase newer and newer locks as this product is very modular and any part can be easily replaced with a cost less than an entire new lock. In addition, its on the inside so intruders are not able to access this physically.

Moral Impacts

Due to the camera, my product comes with many moral issues, such as the capture of video/photo without one's consent. Therefore, I shall make the camera attachment optional and give the user the ability to turn of or on the feature. This product also benefits society as it gives people a higher feeling of security and thus brings about a more relaxed situation in a community. It can also help those commonly forgetting their keys as they just need a device to access their door. My product can easily increase the required amount of security with more things such as 2 step authentication with a simple firmware update and no requirement for a new device. However, this firmware is open source so hackers may be able to exploit vulnerabilities, but at the same time people may be able to contribute and make the firmware even stronger.

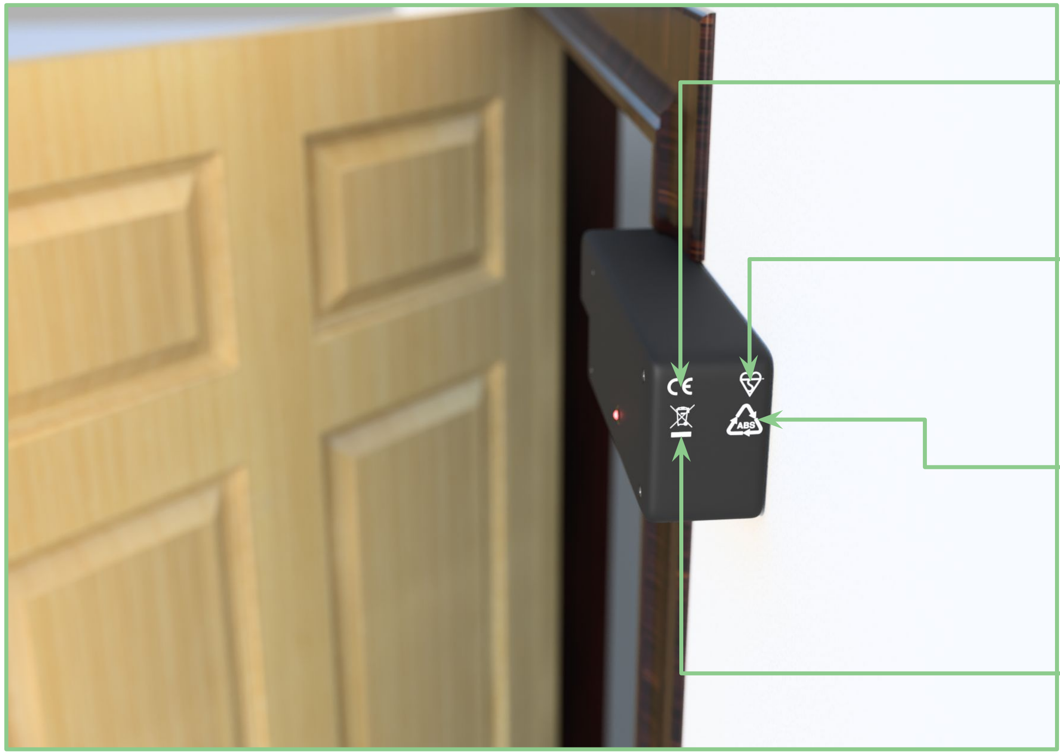
Environmental Impacts

ABS and batteries are used in my product and can cause severe environmental issues as plastics are NOT biodegradable thus they can quickly fill up landfills. ABS is a thermoplastic thus, it can be recycled and re-moulded. Batteries need to be thrown away in special bins and I need to reduce the size of my casing as much as possible in order to avoid using as much plastic. Thermoplastics need a lot of energy to be re-shaped thus it still is not environmentally friendly.

Social Impacts

Sustainability is seen in my product in various ways representing the 4 r's. Reduce , Reuse, Recycle, and repair.
Reduce - I will make the PCB as small as possible and get the best balanced power bank (size vs battery life) in order to reduce plastic required to make the casing. In addition, the shelling of the casing is balanced as well (durability vs plastic usage) .
Reuse - Thanks to the battery housing, the batteries are easily accessible and replaceable and do not require disposing the whole product.
Recycle - The casing is made of ABS, a thermoplastic, which can be re-moulded into another product instead of getting thrown in landfills. The raspberry pi is also a computer and can be easily re-programmed to do a plethora of different tasks. The PCB is not simple to re-use but components can be salvaged.
Repair - The casing can be easily opened to gain access to internal components. Each component can be replaced with ease due to the modularity of the product.

In order to indicate this to a user, below is an image which highlights the various symbols as it would be in the case that it was mass produced.

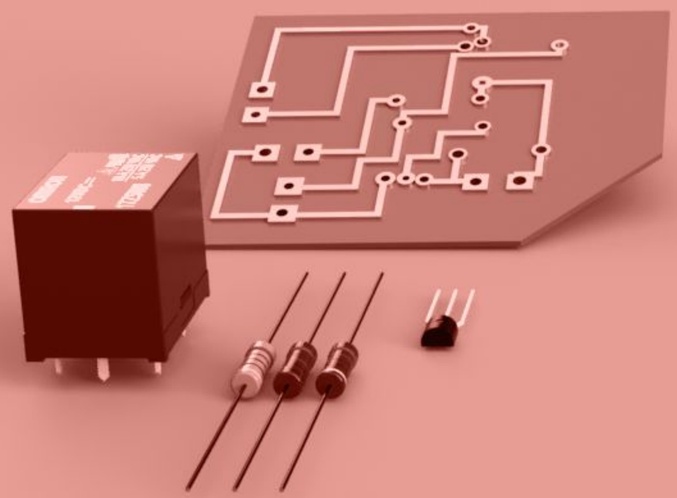


CE marking is a certification mark that indicates conformity with health, safety, and environmental protection standards for products sold in the European Economic Area (EEA).

Kitemark is a mark which shows the approval of the British Standards Institution. The kitemarks indicates the product is indicative of certain stricter safety and quality regulations stricter than the CE. This mark is designed for products for sale in UK.

Plastics mark informs the user and recycler of what type of plastic is used on the product. Here it identifies the plastic as ABS. This ensures that the plastic is reused in the correct fashion and prevents potential damage and wastage.

The WEEE (Waste Electrical and Electronic Equipment) directive aims to reduce the amount of waste electrical and electronic equipment that ends up in landfill. Regulations include: improved product design to ease dismantling, recycling and reuse.



M A N U F A C T U R E

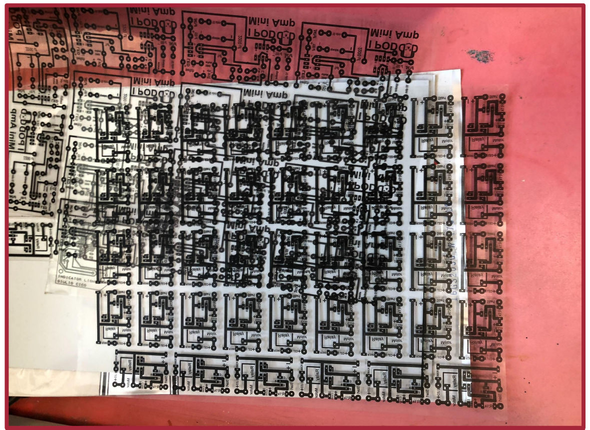
PCB MANUFACTURE

SAFETY PRECAUTIONS -

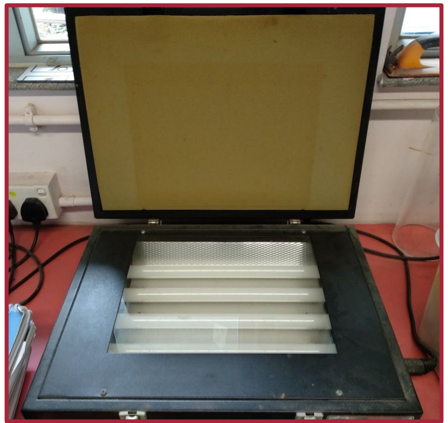
Throughout my manufacturing process, I implemented several different safety features to ensure no injuries were caused to me or anyone around during my process of manufacture. In the next few slides, I document the physical and technological manufacturing of my product with references to safety, success, failures, equipment used and quality checks.

The first stage of the manufacturing process was **using a printer** to print out my PCB design from RealPCB onto a **piece of paper**, after which I printed the design onto **acetate sheet**. Once the printing was complete, I **photo etched the the design** onto the PCB material. I accomplished this by laying the acetate sheet on the **glass layer** of the **UV light box**. **Peeling off** the the **photosensitive layer**, I then turned on the light box which **exposes the PCB to UV light**, and **creates the tracks.** Finally, I **submerged the PCB** into two seperate **chemical tanks**. The first tank was the **developing tank** to **eradicate the plastic** on the on the PCB (takes 12-15 mins). The second tank was the **ferric chloride tank** which helped to **remove unwanted excess copper** (takes 5-10 mins).

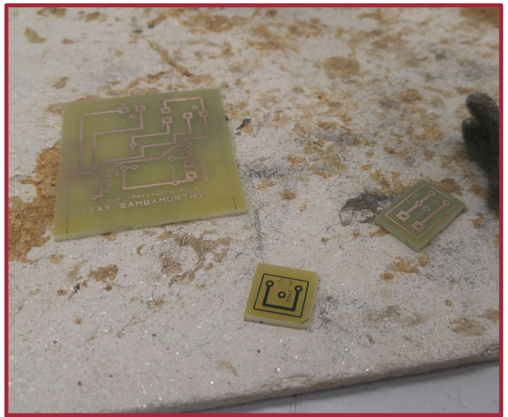
Ferric Chloride Tank



Acetate Sheet Samples



UV Light Box



Unpopulated PCB

Quality Checks
- Ensuring that the printed PCB design had a 1:1 ratio
- Making sure all components and labeling have been printed correctly
- Checking the alignment of the PCB and material (making sure it was properly placed)
- Confirming that the excess copper has been removed
- Timing and checking the exposure of the PCB to avoid overexposing or underexposing
- Check for bridges

Overall, the manufacture of the PCB went smoothly and did not require any re-designing or second attempts and everything went as planned. However, once I printed the PCB on paper, it did strike me that the PCB may be quite hard to solder and sand down for quality.

Equipment Used:
- Printer
- UV Light Box
- Ferric Chloride Tanks

Success
- High quality tracks with no bridges
- All tracks exposed at the correct level
- PCB is compact as intended on design

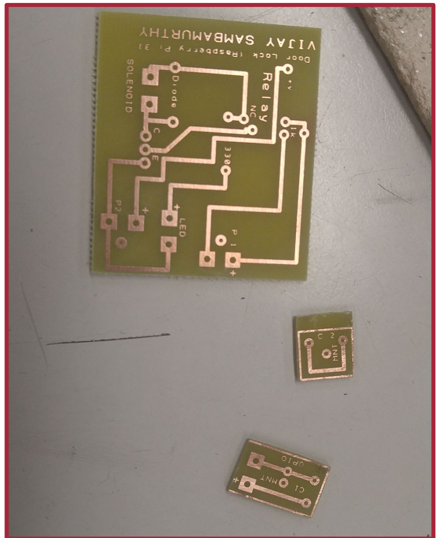
Potential Improvements
- Smaller PCBs may be too small
- Hard to solder onto due to size
- In mass manufacture, PCB may be able to get smaller to save more space.

PCB PREPERATION

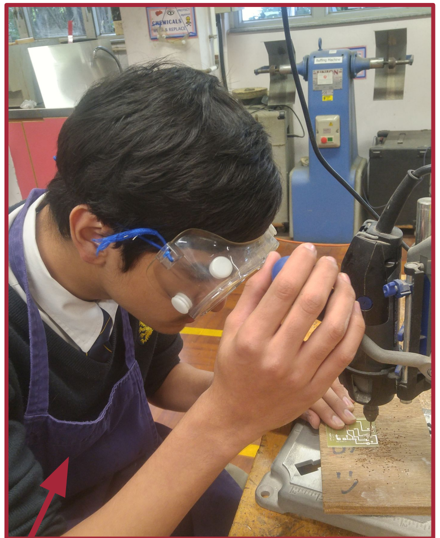
With the PCB printed and ready to prepare, I started by using wirewoll to rid the PCB of the photosensitive layer. I did this first such that using the wirewoll would be easier as it is a flat surface which is easier to clean. After that, I used the disk sander to bring my PCB down to its designated size. However, on the 2 smaller PCBs, this would be too dangerous so I had to resort to using files to trim them down. Then, I used a 1mm & 1.5mm drill on the designated contact pads of my circuit. For the components to be soldered on the board (all except the diode), I used 1 mm. On the contact pads and flying components I used 1.5mm as they are multi-cored wires which results in more thickness and hence why I require 1.5mm holes. Once I finished preparing the PCB, I used a magnifying glass to check for any hairline bridges once again. If any were spotted, I used a stanley knife to remove them.

Disk Sanding PCB

Filing Smaller PCBs



PCBs (Sanded and Wirewoll)



Drilling PCB

Fingers away from any moving parts to prevent injuries

Safety goggles protecting my eyes from dust.

Apron to protect clothes and body from dust and debris

Quality Checks

- Sanding the PCB just to the right size
- Ensuring all the photosensitive layer is removed
- Checking the size of the PCB against the desired measurements
- Making sure the holes are right in the middle of the designated areas

Equipment Used:

- Disk Sander
- Hand Vice
- Files
- Drill
- ***Dust Extractor (To prevent harmful fumes from affecting people nearby)***

Success

- PCB was right size
- Holes were in the right areas
- No bridges

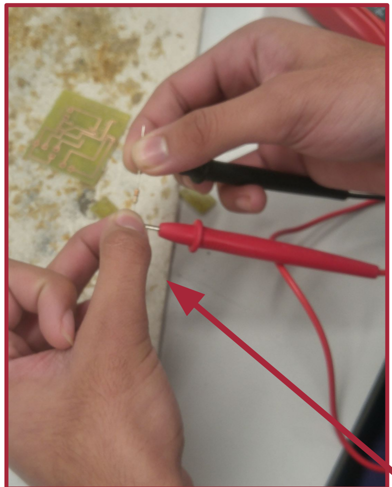
In conclusion, the PCB came out as the size that I originally intended for it to be all the component holes are matching the size that I wanted, especially my custom made relay.

Potential Improvements

- 1 strain relief hole had to be added even though it was not marked on the intended design.

PCB POPULATION

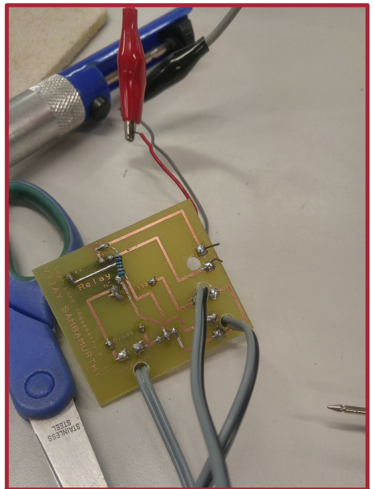
Now that the PCB was ready I began preparing the **PCB** for population by checking my component values and isolated them to test before inserting them by using a **multimeter and breadboard.** Then I began adding the components that need to be mounted directly on to the PCB with a soldering iron. Once they were in place, I stripped the dual, multi-corded cable and tinned the ends. This made it much easier to connect the components on. I then tested the PCB with **crocodile clips** to ensure it worked as needed. Before attaching the **flying components**, I added **heat shrink** (using a **heat gun** to cause the desired shrinking) to all the flying cables to prevent any possible short circuits and accidental connections. After all components and flying wires were on, I used a cutter to snip the extra cables on the PCB. Finally, I added more **flying wires** for the wires connecting to the **Raspberry Pi.**



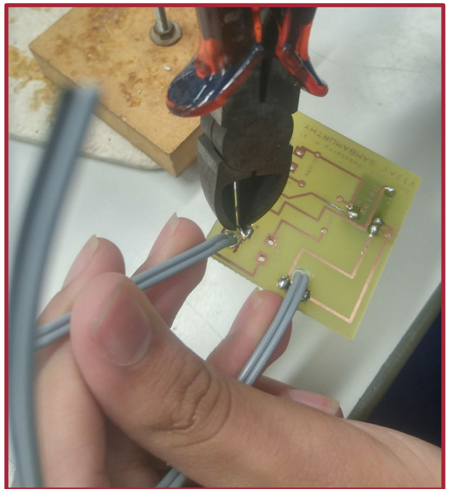
Checking values of resistors

Heat shrink protected LED

Fingers away from soldering iron tip



Testing with Bench PSU



Cutting excess wire with cutters

Heat proof mat to prevent potential fire and damage.

Wearing goggles to protect eyes from dust

Quality Checks

- Making all components soldered with similar amounts of solder to prevent bridges
- Using heat shrink to prevent unwanted contact on all exposed wires
- Checking values and polarization of components with a multimeter to ensure correct placement
- Testing PCB whenever possible to minimize chance of failure

When I completed the final test, I realized there was a flaw with my PCB, however, everything else went as planned and the quality of soldering was high quality as I wished.

MAJOR ISSUES

- **PCB failed to work when connected to 9v battery, worked with bench supply. (Fixed next page)**

Equipment Used:

- Soldering iron
- Solder
- Heat proof mat
- Crocodile clips
- Wire stripper
- Heat gun
- Desoldering Pump
- Heat Shrink
- Multimeter
- Soldering Iron Stand
- Breadboard

Success

- Components attached without causing bridges
- Values were all perfect
- Individual components worked
- No strain on any components, wires were free

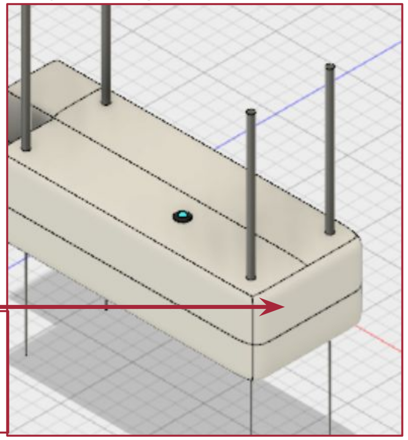
Potential Improvements

- Due to a design flaw, one of the resistors had to be placed on the bottom of the PCB as the relay took up too much space, rearranging components would help.

The complete PCB was now ready for testing. To test, I used the bench power supply and my **9v battery** on my circuit. The bench power supply was set to **3v** to simulate the **Raspberry Pi** signal. This should initiate the solenoid to pull the piston and for the LED to turn on.

First Test:
On the first try, the **LED lit up,** but the solenoid was not retracting the piston. This was a really really big issue. With a **multimeter,** I was able to deduce the fact that the current supplied to the solenoid was much too less as it still worked with the bench power supply, which has a much higher current rating. This required a few changes, first I thought that the **NPN transistor** may be too weak so I replaced it with a **BFY transistor,** which has a much higher current kick. This did not result in any change. I did realize that the current PCB was absolutely destroying 9v batteries and running them low really quick. I was quite confused so I decided to replace the intended load power supply. While a **9v battery has a higher voltage output**, a **4x1.5v (AA) battery has a higher current output**, so I **re-soldered** the entire power supply. This worked! I was now able to turn on the solenoid and the unlock mechanism was complete. All the changes I did with a temporary wire first to ensure it worked.

Required changes:
Due to 2 component changes, the 3D CAD model had to be modified. This was quite easy due to **Fusion 360**'s timeline, which allows you to delete events. Hence why I removed the event where I created the hole for the 9v battery housing. However, there was a new issue, the new housing required for the 4xAA batteries was slightly taller. I created a 3D model of the 9v battery and re-arranged the internal components. Thankfully, my 3D casing already had space inside for this modification.



No hole for 9v battery casing

3D CAD Design from Fusion 360

Quality Checks
- Use small amounts of solder to prevent any bridges
- Connect at different points of circuit to isolate issues
- Check polarization before inserting any new components
- Once a component is isolated, test on its own to see potential issues and confirm problem

Equipment Used:
- Soldering iron
- Solder
- Heat proof mat
- Crocodile clips
- Wire stripper
- Bench Power Supply
- Multimeter

MAJOR ISSUES *FIXED*
- **PCB failed to work when connected to 9v battery, worked with bench supply. (Fixed next page)**

Safety
- Goggles
- Heat Proof Mat
- Insulated testing wires
- Careful of finger placement when using soldering iron

Success
- The end result works just as required
- No mess created on PCB
- Small adjustments did not affect casing to require much re-designing

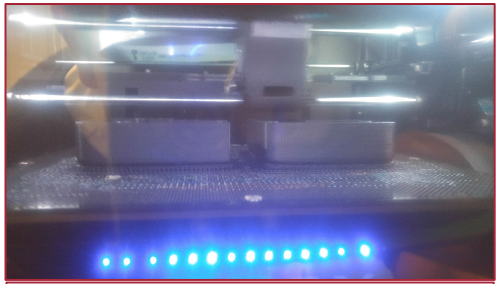
Potential Improvements
- Because of the current draw from the solenoid, the batteries now drain fast and require replacement after a few minutes of constant use - Change power supply/ solenoid type.

The completed product came out well and worked almost just as expected. The only issue was the high current draw from the solenoid which I was unable to anticipate. The **LED** was **bright and clear** for users to be notified of open door. The error fixing I ran truly required in depth analysis and probing to figure out. Thankfully, the new fix allowed adjustments without much correction required.

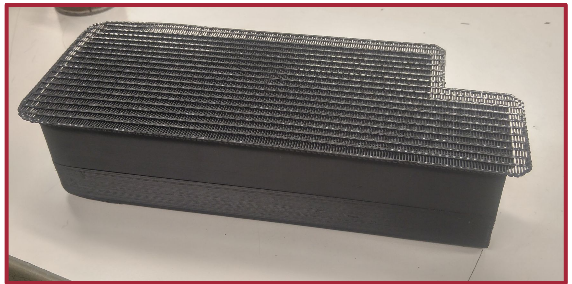
CASE PRINTING

To print the casing, I began by exporting each part of the **Fusion 360** file as an **STL file**. Then, I uploaded the STL file to the computer connected to the **UpBox 3D** printer. Then I chose my filament color and began the print. The 3D printer used has quite a high resolution and the result came with exact dimensions such that the PCB and all internal components fit snugly in the ribbings I made (Pages 25-29). Once the print was complete, I used a chisel and slowly plucked the support structure away from my casing. The support structure is automatically added from the 3D printer software in order to ensure the structure does not just drip and fall down while printing. Once the support structure was removed, some of the ABS turned white due to chipping, hence why I used spray paint to fix this.

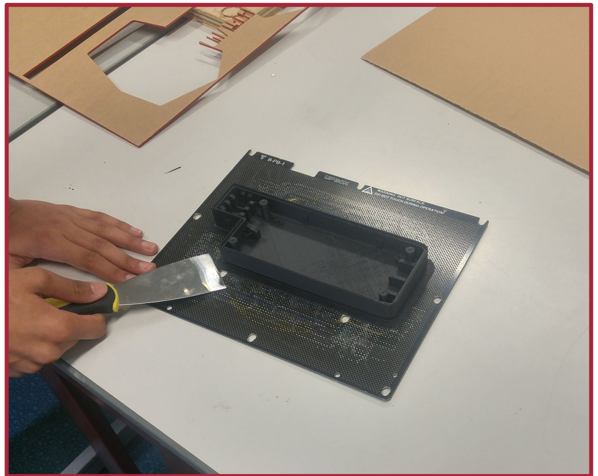
Unfortunately the ribbing for some components were a bit longer than I wished and thus required force and lots of pressure to insert the component. This was quickly fixed with a file and chisel. The main component being blocked was the power bank.



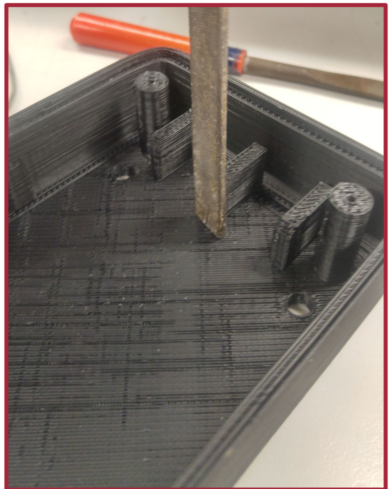
UpBoz 3D Printer, printing my design



My 3D casing, top portion has support structure still attached



Using chisel to remove support structure



Using chisel to fix ribbings

Quality Checks

- Use small amounts of solder to prevent any bridges
- Connect at different points of circuit to isolate issues
- Check polarization before inserting any new components
- Once a component is isolated, test on its own to see potential issues and confirm problem

Equipment Used:

- UpBox 3D printer
- Chisel
- File
- Screwdriver
- Drill
- Gas Mask
- Satin Black Spray Paint
- Extractor

Safety

- Finger's distance from chisel and file
- Gas mask,gloves and apron when spray painting

Success

- Casing size was perfect
- No errors in hole positions
- Support structure came off easily
- Adjusted ribbing holds components tightly
- Door component is light and thus reduces strain on door.

Potential Improvements

- Due to one part finishing early, I was unable to remove the support structure when it was hot, thus causing some damage on the top side of the casing.

To prep the 3D printed casing, I used a drill to tap the screw holes manually and a screwdriver to clear the debris from the hole. I did find that due to the rotation of the drill, the ABS would heat up and begin melting which would make tapping much easier as the ABS would wrap around the screw tightly. I tested the casing by adding all screws and closing it together. It held up very well and did not feel fragile at all! I was quite surprised in the end by how precise my design came out.

To start the programming of my code, I began by getting the **Skype API** setup and ready to use. To do this, I used **skpy, a API for skype on pytho**n. I used **Python** as it is easy to program and works well with the **Raspberry Pi GPIO** module too. I created an account for my **Raspberry Pi** and began working on the code. At first, I was easily able to send and check for messages. However, this is slow and inefficient as it requires constant checking. So, I decided to use skype's loop instead which gets automatically triggered on an event. An event can include many different things such as a call, user typing, sending contacts, editing messages, deleting messages,etc. The one I was concerned about is the **NewMessage**, This informs the program that of one of the following: The user is typing, the user has sent a message, the message has been updated or deleted. I focused on the sent message one. This made my program check every time a user sends a message and reads the contents. If the contents are "…" then it executes "…". Once I had the messaging part done. I focused on the subprograms.

GPIO pinouts were really easy to program. First, you declare whether its an input or an output pin and then you state either **ON or OFF (1/0 - true/false).** This allowed me to easily control the **PCB** i already made and thus lock or unlock the door. I used this to design each stage of my program. Lock would turn off the solenoid if the door is closed, unlock would turn on the solenoid, etc. Just as required on my flowchart.

Right now the code is at a very minimal stage, without much validation and verification for added safety. The camera motion detection program has also been designed, however I decided not to add it to my prototype due to the vastly increased power draw.

```
[SkypeTextMsg]
Id: 1523681978016
Type: RichText
Time: 2018-04-14 04:59:38.007000
ClientId: [redacted]
UserId: [redacted]
ChatId: [redacted]
Content: this is a test
```

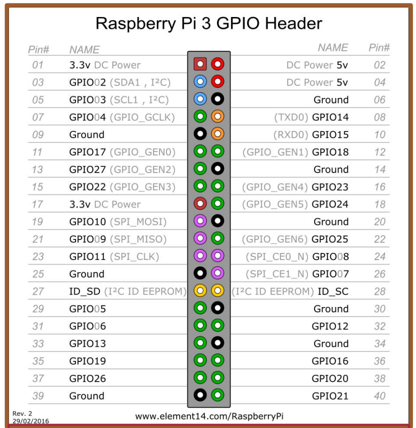
Skype Text Message RAW Data

Software & Equipment Used:

- Raspberry Pi model 3 B
- Pycharm
- Idle
- Python 3.6
- SkPy

```
[SkypeTypingEvent]
Id: 1316
Type: NewMessage
Time: 2018-04-14 04:57:43
UserId: [redacted]
ChatId: [redacted]
Active: True
[SkypeChatUpdateEvent]
Id: 1317
Type: ConversationUpdate
Time: 2018-04-14 04:57:44
ChatId: [redacted]
Horizon: None
[SkypeNewMessageEvent]
Id: 1318
Type: NewMessage
Time: 2018-04-14 04:57:46
MsgId: [redacted]
[SkypeChatUpdateEvent]
Id: 1319
Type: ConversationUpdate
Time: 2018-04-14 04:57:47
ChatId: [redacted]
Horizon: None
```

Different event types Skype


Raspberry Pi 3 GPIO Header

Pin#	NAME	NAME	Pin#
01	3.3v DC Power	DC Power 5v	02
03	GPIO02 (SDA1 , I²C)	DC Power 5v	04
05	GPIO03 (SCL1 , I²C)	Ground	06
07	GPIO04 (GPIO_GCLK)	(TXD0) GPIO14	08
09	Ground	(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)	(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)	Ground	14
15	GPIO22 (GPIO_GEN3)	(GPIO_GEN4) GPIO23	16
17	3.3v DC Power	(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)	Ground	20
21	GPIO09 (SPI_MISO)	(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)	(SPI_CE0_N) GPIO08	24
25	Ground	(SPI_CE1_N) GPIO07	26
27	ID_SD (I²C ID EEPROM)	(I²C ID EEPROM) ID_SC	28
29	GPIO05	Ground	30
31	GPIO06	GPIO12	32
33	GPIO13	Ground	34
35	GPIO19	GPIO16	36
37	GPIO26	GPIO20	38
39	Ground	GPIO21	40

Rev. 2
29/02/2016
www.element14.com/RaspberryPi

RPi Pinouts

Quality Checks

- Use subprograms to make sure the program is neat and easy to debug
- Use comments on code to remind potential source code users what they do
- Use Descriptive Variable Names
- Indent clearly

Success

- Skype API works almost instantaneously
- User can send messages all over the world
- Code runs smoothly without errors
- Door component is light and thus reduces strain on door.
- Variables allow user to be easily changed when needed

Potential Improvements

- Unfortunately, the power constraints prevented the implementation of the MAC address scanner. In the real model, a bit more of code that scans the current wifi network for an exact MAC address would allow users to have the door unlock with ease. Bluetooth would also allow for more safe connections however requires more power.

Safety

- Turn off Raspberry Pi when attaching jumper cables to PCB to prevent potential shorting and keep electronics safe.

CODE PROGRAMMING (MOTION CAPTURE)

```python
# Anything starting with "#" is a comment, python does
not execute comments ( They will be colored like this )
import numpy as np
import cv2
def diffImg(t0, t1, t2):
    d1 = cv2.absdiff(t2, t1)
    d2 = cv2.absdiff(t1, t0)
    return cv2.bitwise_and(d1, d2)
cap = cv2.VideoCapture(0)
t_minus = cv2.cvtColor(cap.read()[1], cv2.COLOR_RGB2GRAY)
t = cv2.cvtColor(cap.read()[1], cv2.COLOR_RGB2GRAY)
t_plus = cv2.cvtColor(cap.read()[1], cv2.COLOR_RGB2GRAY)
while True:
    # Capture frame-by-frame
    ret, frame = cap.read()
    #Operations on the frame come here
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
 # Display the resulting frame
    dimg=diffImg(t_minus, t, t_plus)
    threshold = 200000
    if cv2.countNonZero(dimg) > threshold:
        cv2.putText(frame, "Room Status: {}".format('movement'), (10, 20),
        cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 0, 255), 2)
    else:
        cv2.putText(frame, "Room Status: {}".format('no movement'), (10, 20),
        cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 0, 255), 2)

    cv2.imshow('Movement indicator',frame)
    t_minus = t
    t = t_plus
    t_plus = cv2.cvtColor(cap.read()[1], cv2.COLOR_RGB2GRAY)

    key = cv2.waitKey(10)
    if key == 27:
        cv2.destroyWindow('Movement indicator')
        break

# When everything done, release the capture
cap.release()
cv2.destroyAllWindows()
```

***Assumption**: The first frame of the video input will contain no motion and just background — therefore, it is possible to model the background of the video stream using only the first frame of the video.*

Delta feed
(Difference between first frame and this frame)

CODE PROGRAMMING (MAIN PROGRAM 1)

```
# Anything starting with "#" is a comment, python does not execute comments
import RPi.GPIO as GPIO  # Raspberry pi GPIO api import
import time  # Python time module for delays
from skpy import Skype  # Skype API
from skpy import SkypeEventLoop  # Skype event handler api
from datetime import datetime, time  # DateTime Api
GPIO.setmode(GPIO.BCM)  # Set the board pin type
solenoid = 21  # Solenoid + volts pin
contact = 26  # Contact pad input PIN
GPIO.setup(solenoid, GPIO.OUT)  # Set solenoid (21) as output pin
GPIO.setup(contact, GPIO.IN, pull_up_down=GPIO.PUD_DOWN) #Set contact (26)
                                                           as input pin
User = "<INSERT USER ID HERE>"  # The User's Skype ID
now = datetime.now()  # Get the current line
me = Skype("<USERNAME>", "<PASSWORD>")  # Login to skype on bot
def checkClosed():  # Subprogram to check if door is closed.
    doorVal = GPIO.input(contact)  # Is there current flowing through the
                                     contact pads?
    if (doorVal == True):
        return True  # If there is then return true
    return False  # Otherwise return false

def lock():  # Lock the door
    for i in range(0, 5):  # Loop 5 times
        if (checkClosed() == True):  # Is the door closed
            GPIO.output(solenoid, 0)  # IF so then release solenoid
            return 0  # End subprogram
        time.sleep(10)  # Wait 10 seconds
    msg(User, me,
        "Unable to close door, please check if it is closed.")Warns user if door is open.
```

```
def unlock():  # Unlock door
    GPIO.output(solenoid, 1)  # Retract Solenoid
    msg(User, me, "Unlocked.")  # Tell the user it has been unlocke
    time.sleep(10)  # Wait 10 seconds
    lock()  # Lock the solenoid (Subprogram)


def status():  # Check the status
    PCB = GPIO.input(solenoid)  # Is the solenoid output on?
    if (PCB == 1):  # If so then
        PCB = True  # Set variable PCB to true
    else:  # Otherwise
        PCB = False  # Set variable PCB to False
    door = checkClosed()  # Store the door closed value into a
variable (subprogram)
    msg(User, me, str("Lock on: " + str(PCB) + " & Door closed: " +
str(door)))  # Send message to user about info.


def msg(usr, skyp, message):  # Send a message to user via skype
    try:
        tmpid = str("8:" + usr)  # Get the user's chat id
        skyp.chats[tmpid].sendMsg(message)  # Send the message
    except:
        print("Unable to connect.")  # If error is thrown. Send
warning to console.
```

CODE PROGRAMMING (MAIN PROGRAM 2)

```
def refresh():  # Refresh the current status
    doorVal = checkClosed()  # Check the door status
    if (doorVal == True):  # if door is closed
        if (GPIO.input(solenoid) == 1):  # Is solenoid on?
            GPIO.output(solenoid, 0)  # Turn it off
        else:
            msg(User, me, "Door is closed.")  # otherwise send message to user
    Else:
        msg(User, me, "Door is currently open.")  # If door is open send message to user.
class MySkype(SkypeEventLoop):  # Skype event handler loop
    def onEvent(self, event):
        (repr(event))
        current = datetime.now()
        if ((
                current - now).seconds > 20 and event.type == "NewMessage" and
event.msg.user.id == User):  # If 20 seconds has passed since program started and the type of
event is a new message and the message is from user
            try:
                if (event.msg.content.replace(" ", "").lower() == "lock"):# If message is lock
                    print("Locking") #Send lock to console
                    lock() #Call lock subprogram
                    print("Locked!")#Send locked to console
                    msg(User, me, "Locked.") #Send locked to user.
                elif (event.msg.content.replace(" ", "").lower() == "unlock"): #If message is
unlock
                    print("Unlocking") #Send unlocking to console
                    unlock() #Call unlock subprogram
                    msg(User, me, "Locked.") #Send locked to user. (The unlock subprogram auto
locks after 10 seconds if possible)
                    print("Locked!") #Send Locked to console
```

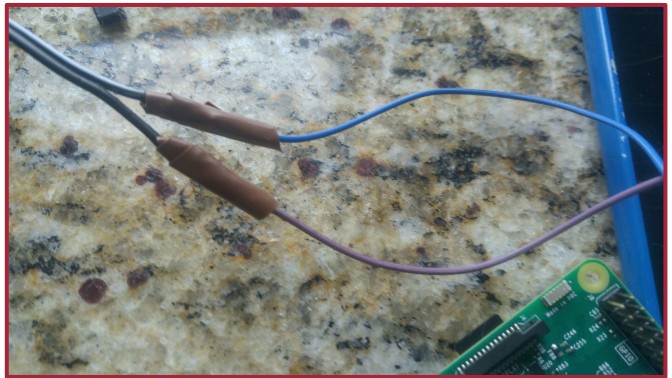
```
elif (event.msg.content.replace(" ", "").lower()
== "status"): #If message is status
                    status() #Call status
subprogram
elif (event.msg.content.replace(" ", "").lower()
== "refresh"):#If message is refresh
       refresh()#Call refresh subprogram
 except Exception as e:
 print(e) #Print any errors to console but dont
stop program if error occurs.

test = MySkype("<USERNAME>", "<PASSWORD>",
autoAck=True) #Login to skype (Bot side) This
time for event loop
test.loop() #Call loop ( Main program)
```

In conclusion the code was quite challenging to write due to the authorization and API learning that required to be done, especially since the documentation does not have as much as normally seen on such complex API's.

The program ticks all the required boxes in the specification and does it with efficiency and neatness. Unfortunately the motion capture program was more about the specifics and setting up so the comments were much less.

FINAL ASSEMBLY

Now that all the separate components were prepped, I began slowly assembling my product. The first issue I encountered was the **PCB** size. Somehow, during my **3D printing**, I got the size of the PCB slightly wrong and thus I used a small piece of card and attached my **PCB** to it, this allowed the **PCB** to sit securely in the ribbing. Then, I realized that the solenoid would fall too far if it was left just like the way it was. This was a major flaw as then, the door would not be able to open again. I used another small piece of card and cut it in the shape of the hole of the piston holder. This raised the piston up a bit such that it could actually retract. Finally, I had some issues with the smaller PCBs. The hole to attach the copper tracks to were in slightly off positions. I adjusted this with a simple bit of hand drilling and it was aligned. After that, I proceeded to connect all my components. To connect my **Raspberry Pi and PCB**, I used jumper cables for arduino / raspberry pi. They have male/female connectors that attach to the pins. I soldered 4 of them to their destinations on the PCB/ flying wires connected to the **PCB**. I already knew they worked from the code testing, however, I wanted to ensure this so I tested again and it still worked! To ensure no shorting, I used **electrical tape** around them as the available **heat shrink** was too small to fit around, this also allowed **re-use of the wires** as they can be easily taken apart. Then I put all components in, and properly wired the insides and attached the **LED** into the mount using cable ties.



Jumper Cable and PCB cables soldered and taped

Contact Pads on Wall Component

Contact Pads on Door Component

Quality Checks

- Ensure the copper contact pads actually make contact when closed (positioning)
- Neat cable management inside casing for easy repair
- Ensure ribbing sizes aren't too tight or too loose so that components can be removed and secured easily.
- Ensure screws are positioned in a position where they can hold the casing together securely

Success

- Entire product works perfectly
- Light is bright and clear when door is open
- Contact pads have 100% accuracy of informing raspberry pi if door is open or closed
-

Equipment Used:

- Soldering Iron
- Solder
- Heat Proof Mat
- Electrical Tape
- Screwdriver
- Cable Ties

In sum, the overall product worked surprisingly well. Due to the skype API, messages are sent and parsed almost instantly, thus improving product quality. Status messages are also timed very well and give the user a lot of info. Thanks to the jumper cables, the connections can be easily debugged and fix in the unlikely case of any errors. The power bank was able to sustain the Pi for quite a long time (few hours).

Potential Improvements

- Unfortunately, the power constraints prevented the implementation of the MAC address scanner. In the real model, a bit more of code that scans the current wifi network for an exact MAC address would allow users to have the door unlock with ease. Bluetooth would also allow for more safe connections however requires more power.

FINAL CASING IMAGE

EVALUATION

This page highlights how the product works in a real life situation, used by the target user group. For this test, my user will be a **middle-aged** property owner attaching it to the door of a room and **leaving the the door's lock** still useable.

In order to setup I requested the user to attempt and **mount** the product to both the door and wall. As of right now, they did **not have the drill** required to **mount** the product properly against the door, so they used an alternate method which seemed to work quite well. Then they entered their desired **skype account** which they would like to use to **unlock** and lock the door. And put everything in place. The door was a typical swing open door made of wood. The wall had much more than enough clearance to house the wall component. At first they did mention that the product **"seemed a bit big"**, however they later realized why it was as such and learnt why it was good.

The daily routine of the user was perfectly fine and **unaffected**. The room had many users enter in and out of when unlocked. When the time came, the user had to vacate the room and thus wanted to **secure** it so they used both my product and the door lock to **ensure no one could enter.** At around 9 PM , the user sent a **message** to check on the **status** and was informed that everything was fine. Day 1 was conclusive and there was no entering of any unwanted people. On day two, I decided to **simulate a intrusion**. I told the user to leave the door unlocked and see what happens, after a few minutes, the user was **constantly getting messages warning** about the unlocked door. Once again, I told them to ignore it for a test. An intruder entered the room and the **camera caught a lot of motion** even before the intruder entered as they were picking the main door lock. This is caused by the shaking of the door which **triggered the motion sensor**. The user now had a **picture** of the intruder and a time at which it occurred. But, by daytime, we had realized that the **battery ran out of power** due to us turning on the camera. This was a very bad thing to occur as it leaves vulnerabilities once again.

The user is quoted as saying the following: "On the first day, without the camera, the door was really **calm and quiet.** I had a sense of confidence and could rest myself without worry. But the second day was a whole new story. I left the door unlocked as instructed to do so, but I did leave the **designated lock for the door on**. This is what I typically do. But the newly attached product was **warning me** that it's lock was not on. First, there was the **bright LED**, then there was the **spam of messages** from the device. I do see the use in this, but what if there is someone like me who does not want the door to be locked with the product but just a simple key? Then something surprising happened, the product **sent me a message and buzzed** a lot in the **middle of the night.** I saw **images of an intruder** and was shocked! This truly would have **saved my expensive tech** if something like this were to happen in reality. Thankfully it was just a test. Unfortunately though, the product **ran out of battery** by morning. I was disappointed by the weak battery life. Otherwise, it's a **great product and is really justifiable to purchase"**

USER TESTING - SITUATION 2

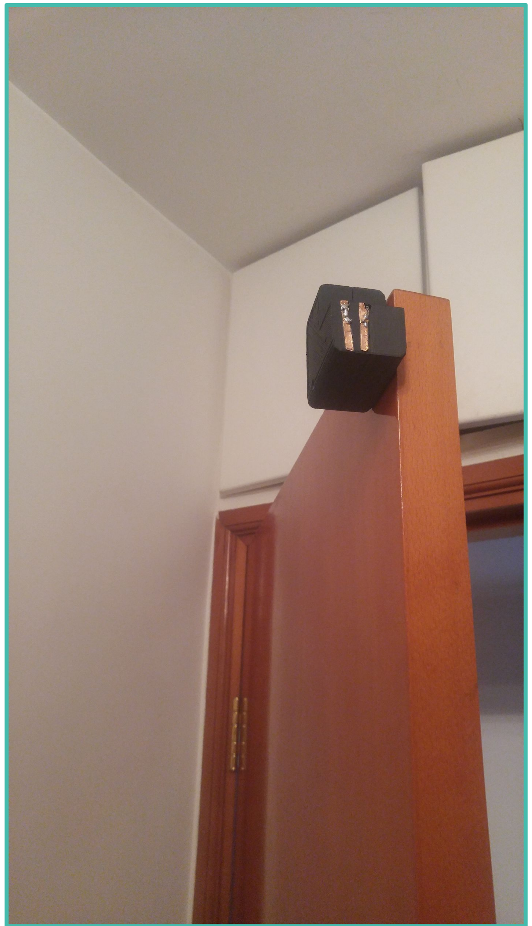
Door closed LED OFF

In the next situation, I decided to test a college student's dorm. The selected dormitory was notorious for having intruders. I began the **setup** by **attaching the product to the door and frame,** this was simple and took even less time than the previous situation. I placed the product in and set it up to accept requests from the student's skype. It took a total of **13 minutes to set up.** Already, the student was quite **impressed just off testing.**

I did find that this door, being flat, was able to hold the product much more flush and in such a way that it checked spec **(1.2 "The product should not be very vibrant but should fit in well with its surroundings and should be a suitable color like black.")** instantly upon setup.

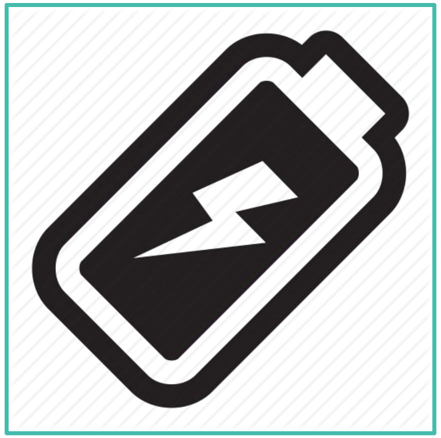
The student was outside the dorm for most of the day, hence regular checks were requested from him. This truly put the **product to its limits** as it was taking more and **more power every status update** was called. (Every hour or so). The **battery lasted longer** this time as my **power bank was swapped** out with a more dense one. I gained a **few hours** from the last test. Unfortunately, it still **ran out at 6:53 PM**. Luckily, this was just **5 minutes** after he got home. So the product was recharged and set for the night at around 9:15 PM. There were **no intruders detected** for the following 2 days and everything went as planned.

One thing that we realized is that, alongside the power of the **lithium ion battery** decreasing, the power of the **4x 1.5 V AA batteries were running out**, however this was more dependant on how long the door was left unlocked. This decrease in power was very evident by the **brightness of the LED.**



Door Open (Door component)

In conclusion I believe that the product requires several changes based of what the users say and these changes are not ones that can be easily fixed. In the pages following the spec analysis, I plan to state what changes need to be done and why. The product did function as intended in both tests and the user reported **no software errors** at all, this was great and as intended. Even if the product works as required with battery, if the battery fails, there is no point to the entire product as it won't be able to **open or close** at this point.



USER TESTING - EVALUATION

POINTS	QUESTIONS ASKED	USER 1 FEEDBACK	USER 2 FEEDBACK
1.1) The LED should clearly show the current state of the PCB 1.2) The product should fit in its surroundings	1.1) Is the LED obvious? 1.2)Does the product fit in the surroundings well?	The LED was obvious as long as the power was there, however it did dim down quite a bit when battery was running low. The surroundings did make the product stand out a little but not to that much of an obvious amount.	The LED was very obvious at max power, but every time the battery was almost out, which wasn't as much as the lithium battery, the LED would dim. I thought the product was very well suited for the given environment.
2.1) The product should be **affordable (mid range)**	Would you be willing to pay $300-$400(HKD) for this product?	Yes, this product can be quite useful and help me to save money in the long run.	No, I think a standard door lock would save me money and time in a dormitory as it is easier to use.
3.1) The product can be **anyone wanting security**	Do you think this product is suitable for use by people in similar positions to yours?	The product is definately good for me but I think that some others may not appriciate it as much.	Yes, people my age are quite lazy and love things related to tech, this would definitely pop out! The price might deter them though.
4.1) The product should **stay on when there is a temporary loss of power.**	Do you think the product is easy to use?	The skype messages were very simple, may be a bit too simple.	I found it neat and easy to use.
6.1) The product should be **no larger than an a tissue box.**	What did you think about the size of the product?	Was a bit big for my liking. I would absolutely adore this product if it was smaller.	It was bigger than I anticipated, may not fit in some situations.
7.1 The product **shouldn't have any sharp corners or edges**	How safe was the product's casing?	I did not see any potential for harm in the casing.	The casing was nice, but due to its thinness, if it breaks, sharp screws may harm someone.
8) The product should be easy to use and **controllable from anywhere with internet connection.**	How well did the product alert you? Could you hear the buzzer/see the LED well?	The alerts were too much! I think this is beneficial to some people, but I would like to control the notifications to an extent.	The notifications saved me a couple of times! Really like them the way they are.
9,1) **Opening the product when required should be easy to do.**	Is it easy to replace the battery?	It was difficult at first to figure out, but then I got the hang really quick. This is not good as that means i have to replace the battery a lot.	It was easy to open, just a simple screwdriver was enough.
10.2 The material used to make the **casing should be strong**	Did the product feel sturdy to you?	Not really, I think for a door lock it's quite easy to break!	The plastic seems to thin, but for a prototype its ok.

SPECIFICATION ANALYSIS

The next few pages will contain an analysis of the specification in relation to my product. I will compare each point and see how well the tasks have been completed.

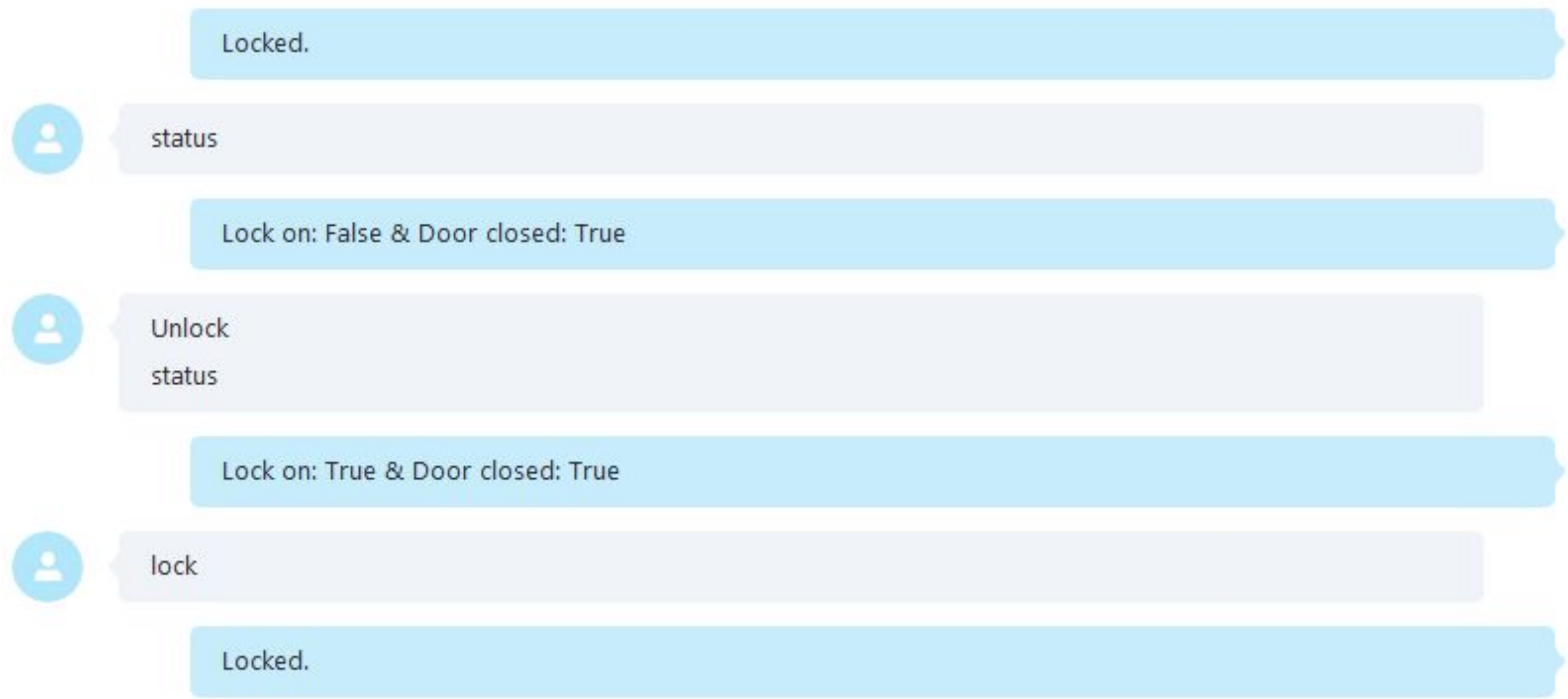
SPECIFICATION POINT	ANALYSIS
1) APPEARANCE	
1) The current mode that the product is in should be clearly indicated with LEDS that are visible on the protected side.	The product has a very clear LED to indicate whether the door is open or closed. Users say that it is visible, bright and hard to miss.
2) The product should not be very vibrant but should fit in well with its surroundings and should be a suitable color like black.	The solid black color of the prototype has been described as "matching against most surfaces" and "not obvious".
2) COST	
1) The product should be affordable but may be in the slightly high end range of locks due to the added functionality while still affordable by most homeowners.	Thanks to the medium resolution 3D printed ABS plastic and other components, the cost of the product is just as meant to be, however, the Raspberry Pi makes the cost go up slightly.
2) The product should be as high as possible within the budget range in order to maximise efficiency and costs, hence increasing longevity.	The cheaper, more readily available components saved a lot of costs in addition to the mass producible PCB which both keep the costs just below the budget. However it was not at its maximum efficiency due to the Pi.
3) CUSTOMER	
1) The product is aimed at those wanting security from a property owner to a teenager wanting privacy in the dormitories.	Most who want security may be able to purchase, but those in the slightly more poor areas may have to go for the typical door locks.
4) ENVIRONMENT	
1) The product should remain functional when there is a temporary loss of power.	The power bank allows this to be accomplished, but not for very long periods of time.

SPECIFICATION POINT	ANALYSIS
5) ERGONOMICS	
1) The product should be designed in such a way that there is less stress on the door.	Due to the small component attached to the door, there is no added strain other than a few grams. Users say "barely a noticeable difference".
2) The product should be easy to mount onto any door.	Thanks to the placement of screws, the product can be mounted anywhere where there is the space to attach it to. However, it will not function on sliding doors due to the lock type.
3) The Skype service should have a quick response time without running into issues.	The skype api allows my product to receive messages almost instantly, thus allowing the product to reply and execute commands quickly.
4) The interface on the product should be simple and easy to understand.	User interface was simple as it uses skype's app, this can be used on most devices and is already used by 560 million people around the world.
6) SIZE	
1) The product should be no larger than an a tissue box.	Unfortunately, due to the big power bank, the door component & main component (when attached) is slightly longer than an average tissue box. However, it is much thinner and shorter, therefore it is still smaller.
7)SAFETY	
1) The product should have no sharp edges to prevent any cuts and bruises to the user.	No sharp edges because of the 3D cad model, rounded corners were made in fusion 360.
2)The product should have multiple ways of unlocking in-case one of the methods fail to function correctly.	While there is only one way of actually unlocking it right now, multiple users can be registered and thus have multiple ways of unlocking. Limited due to power consumption

SPECIFICATION POINT	ANALYSIS
8) FUNCTION	
1) The product must be able to take pictures of any intruders.	Even though the camera is not attached on the prototype due to power restrictions, the code built for the camera allows pictures of intruders to be easily taken.
2) The product should be fully controllable from anywhere, as long as there is access to skype.	Raspberry Pi has wifi access, therefore, it can connect to Skype as long as there is network to connect to. Thus making the product controllable from anywhere if there is access to skype.
3) The product must be able to lock the solenoid without turning on when the door is open.	The contact pads prevent any accidental release of the solenoid as it informs if the two components are touching.
4) The user should be warned nearly instantaneously when there is an intruder.	As stated in point 8.1, the camera is not currently attached. But the code would warn the users if there was a camera.
9) MAINTENANCE	
1) Easy access to the product's components is essential as with a few screws repairing any faulty parts would be less tedious and time-consuming.	The product has only 4 screws which need to be removed in order to gain access and fix any faulty components.
10) MATERIALS	
10.1) From my research I found that ABS or similar plastics would be very suitable, this would be made alongside metal so that it cannot be destroyed or broken easily.	ABS has been used, no metal due to current size limitations. But, due to the design, reinforcement is simple and easy to accomplish.
2) The materials used to build the product should be **abundant** and recyclable to reduce costs and help the environment.	ABS is a thermoplastic and can be easily reused. ABS is not biodegradable.

SPECIFICATION POINT	ANALYSIS
11) MANUFACTURING	
1) The product should be able to be mass produced because there will be a significant amount of target users.	Outer casing can be ABS injection moulded as it can be done at an extremely fast rate compared to my prototyping 3D printer.
2) The raspberry pi will come built in with the correct software and code.	SD card has the OS for raspberry pi and the required software which runs at startup.

In conclusion, the product was made quite well. It meets most of the design specifications. Some overall improvements need to be made, however functionality seems to not require much alteration apart from slight programming changes.



To the left is an image of the bot replying to user requests. Here, it seems like I am replying, however, this is not the case. I was using my skype account for the time that one of the users were testing the product.

This also highlights how the program accepts inputs that are capitalized in any fashion. I plan to add more variation to inputs as potential improvements my product needs to make.

Another big improvement that needs to occur is battery life. These improvements will be talked about in the next page.

IMPROVEMENTS AND MODIFICATIONS

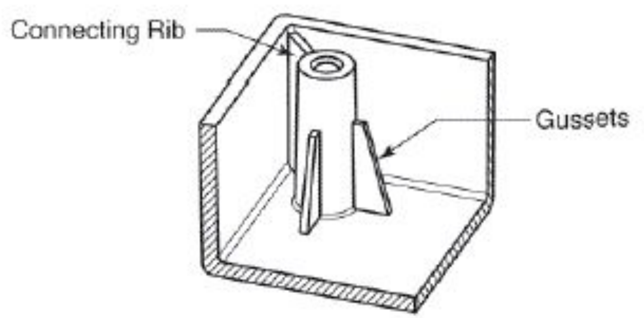
Based on my user feedback and some personal testing, I have realized there are some serious modifications that need to be done before the product becomes truly commercially viable. Below I have included the main changes my product needs to undergo for injection moulding and mass manufacture.

FUNCTION

The main change the function of my product needs is an **extension wire that plugs into the house's electrical system to draw some power**, this is based of what i learned from my user feedback sessions. **The battery ran out too quick.** This design is somewhat **based of third initial casing design (inside the door).** This allows the product to be fully charged most of the time and prevent any potential blind spots for intruders to attack. Then, in the result of a power cut, the lithium ion battery connects to both the solenoid and raspberry pi as a main power source, instead of 2 seperate power supplies.

In addition, the battery is put directly into the casing without a cover, but rather a simple shield to **prevent anything from penetrating the rechargeable lithium battery**. This saves a lot of space and allows more physical battery to be placed into the casing. Or less battery and more ABS saved. This can be sold as different products.

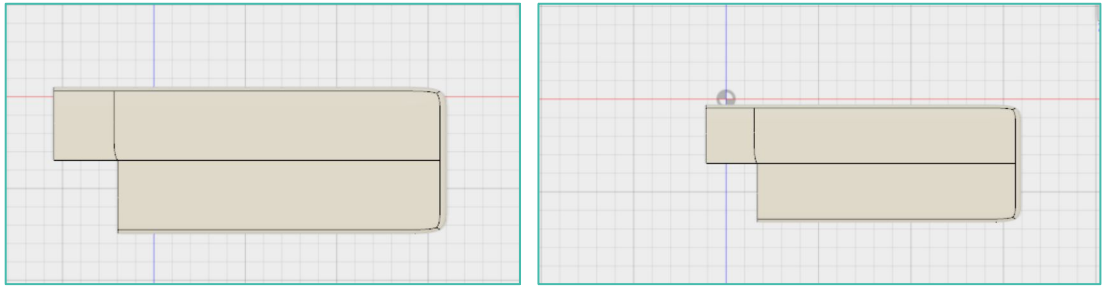
Manufacture



In order for commercial viability to occur, there needs to be some slight modifications to the components and casing. Firstly, the solenoid hole should be altered to allow more directions of input, thus vastly increasing the compatible doors for the product. Moreover, a more efficient solenoid should be used to close the door.

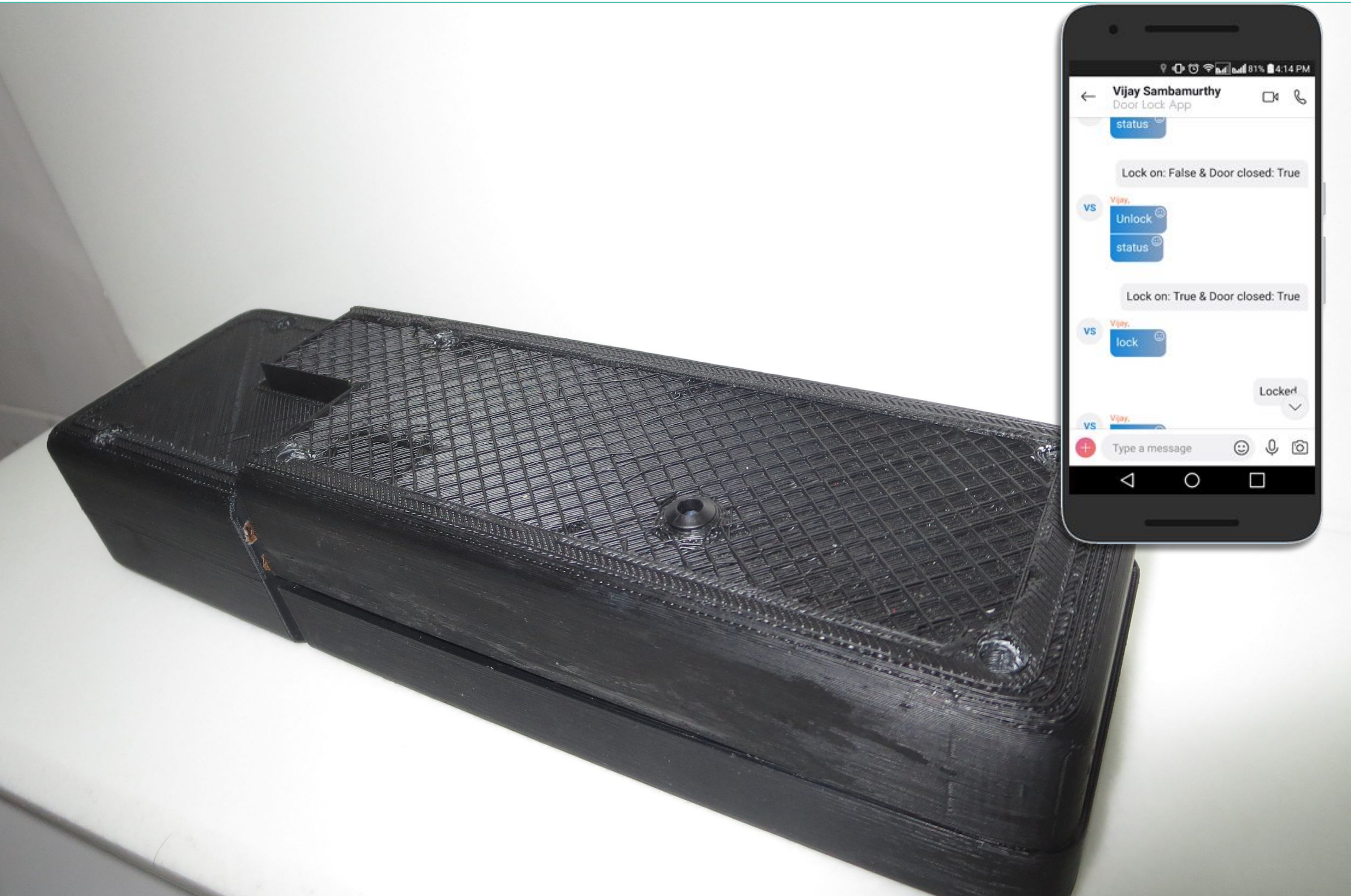
ABS is used because it **combines the strength and rigidity of the acrylonitrile and styrene polymers with the toughness of the polybutadiene rubber**. **Gussets can also be added to screw bosses for extra strength.** The biggest **change is the size,** due to the improved battery and changes to power supplies, the device **can be shrunk vastly.** Due to the shrink, the mould for injection moulding can be shrunk as well, increasing production rate. It also makes customers want the product more as requested in the user feedback.

Original | Modified



In conclusion the changes will allow the product to aim at a wider range of people and be more viable as it is more compact. It would use **ABS as material for casing and PCB manufactured with the Pick and place machine (surface mounted components) and reflow soldering (there is not much need for double sided PCB as there are not many components or tracks, but it can be used..)** as there aren't that many connections on the PCB. The raspberry pi can be pre-loaded with firmware from an SD card copied and pasted.

PRODUCT SHOWCASE



ENDING PAGE