Designing a Secure Handheld Password Manager

Ashwin Ahuja (5501) St Paul's School Extended Project 2015-2016

Table of Contents

1 Abstract

This report covers the design of a dedicated handheld password management device. The report explicitly shows the various steps through which the product has progressed, from initial research into the problem, through to sub-system research, design and development, before recombination of all systems. Largely the Electronics, Software and Mechanics are dealt with separately, as separate important parts of the product which must be defined. However, often through the document, there are references to other sections, largely in justifying why certain decisions were taken.

The first decision to be tackled is one that has largely been taken as granted through this Abstract, that of the choice of the exact project to complete. In this section, the criteria of the selection of the project is outlines and hence, various options are evaluated to these, hence reaching a considered decision.

From here, there is a discussion on the Project's Planning and Aims and how I set out to ensure that I met the aims that I set. Emphasised is the idea that good organisational practise was to be established from the very beginning of the project, and how it governed what I did. From here, there is a period of introductory research where the target market and alternatives are investigated, making use of first hand evidence, found through a productive discussion with a number of my target market ensuring that my product satisfy their requirements. This proved useful throughout the project, with a number more meetings where I ensured the solution I was designing would be successful.

Much emphasis in the entire report is the idea of 'security' (whose research is covered next), a problem that is inherent in the production of a password storage mechanism, and hence the many ways of doing this are considered throughout the project, from ensuring mechanical security, by preventing the user from opening the device, to producing a new, unique and robust cipher, which is used for data transfer.

From there, comprehensive research takes place, looking specifically at deciding certain aspects of the Electronics, Mechanics and Software, including all the necessary research to make the decisions on how my artefact would work, to solve the problem that was researched.

The development of these initial ideas is detailed at great length, investigating all aspects of the system, especially problems encountered during the initial testing of the electronics and software. From a mechanical perspective, a number of potential geometries are investigated and a final design decided, with a complete justification of these decisions.

Additionally, there is a section on how the product would be manufactured (specifically looking at when the product would be mass produced). In this section, the exact manufacturing method of the entire product is defined as well as the material.

Finally, there is a section of reflection upon the artefact and indeed the process carried out through which the artefact was created. The section references the initial aims and specifications of the process and product and judges whether the artefact is completely successful at meeting these aims.

In the appendices, the raw artefact is provided, including rendered images of the casing (and mechanical drawings), complete electronics designs and code produced, with the aim of allowing someone who reads the project be able to replicate it if they wanted.

2 Project Selection

2.1 Project options

The first task was selecting a suitable project, which I could work on for the whole year. My objective was to find a project that was in my area of interest related to computing, which provided me an opportunity to research deeper into a topic and come out with a product that solved an important current problem; all in the given time-frame. I was also keen to design and produce a product that had a market and could be produced profitably on a large scale.

I started with brainstorming a large number of ideas that I narrowed down to a few using my preliminary objectives. In fact, in the beginning, determining any ideas was challenging, however, by researching live projects on Kickstarter[1] and Indiegogo[2] (two crowd funding websites) and through observing and asking many relatives and neighbours (who in fact suggested the final proposal), I was able to define a shortlist. Through some basic research based mainly on my interest and the likelihood of succeeding in creating an effective solution, the list was reduced to four ideas. From this point, each idea was further investigated and then analysed using a Decision Matrix to arrive at the best idea to pursue. Below are details of four ideas that were short listed.

2.1.1 Smart Water Bottle

PROBLEM: Today, there is a growing trend in health and fitness to tackle the increasing problem of obesity. In order to track food intake of people, including nutrient and calories, there are a number of innovative websites and solutions, from reading the barcode of the food to be eaten[3], or even recognizing the food (an upcoming idea of LG Electronics) to a simple app to which the food eaten is manually inputted[4]. However, something yet to be truly grasped by the general population are the dangers of even minor dehydration. According to Medical Daily, 75% of the American population suffers from chronic dehydration[5], a rather incredible figure with serious health impacts. Even 1% dehydration reduces the productivity of work by 12%[6]. Remaining hydrated reportedly reduces the risk of colon cancer by 45%[7] and bladder cancer by 12%[8]. All in all, my research clearly showed a little less accepted fact that remaining hydrated is as important if not more so than maintaining a consistent diet.

There are no convenient solutions for monitoring the amount of water intake. As opposed to food, monitoring the amount of water drunk remains ignored. One has to only drink from marked bottles and write the amounts drunk on a piece of paper (or a 'Notes' app at best). Meanwhile, it is challenging to receive specific recommendations for drinking, since the amounts to be drunk should vary, based on a number of factors, including everything from weight and height to amount of exercise carried out.

SOLUTION: Design a Smart Water Bottle to monitor the amount of water drunk on a daily basis with recommendations and notification reminders for users on how much more to drink.

ADVANTAGES: The product would tie into my key interests (control technology) as well as likely to be able to be completed in the available time-frame. Additionally, it has the potential to be a product which could have large impacts on society and could in fact be highly marketable.

[1] (Kickstarter, 2016)
[2] (Indiegogo, 2016)
[3] (MyFitnessPal, 2015)
[4] (Henry, 2013)
[5] (Medical Daily, 2016)
[6] (Thorzt, 2015)
[7] (Jenkins, 2015)
[8] (Jenkins, 2015)

DISADVANTAGES: There are a number of products in the market which attempt to do this. Though they appear to not solve the entire problem, some seem to come close.

2.1.2 An alternative two step password authentication

PROBLEM: In the conventional system of entering passwords, there is a lack of security as people tend to have less than secure passwords, meaning that a simple brute force attack by a hacker often yields fast results. One possible solution to this problem is two-step authentication. This is a system used and advocated by a number of software companies (such as Apple and Google)[9], where a 4-digit code is also sent to mobile phone of the user. Then, the user requires both this code and the other password to be able to enter the account. However, today, according to a number of experts[10], the value has become diminished as phones can also be hacked into, removing the security of the second step of authentication plus the issue of remembering and securely storing the primary password remains.

SOLUTION: Produce an alternative second step of the authentication process either making use of further software or including a hardware element.

ADVANTAGES: The product fits into my area of interest of making use of complex software as well for enhanced Security concepts.

DISADVANTAGES: The product would likely be intangible, being largely made of software as opposed to having complex mechanical or even electronic elements.

2.1.3 Password Manager

PROBLEM: As people start to get more and more passwords, the challenge of storing all the passwords safely is a highly challenging one. On average, every person has 19 passwords[11], rendering the idea of readily remembering all passwords almost impossible. People use highly insecure forms of storage such as notepads, smart phone notepads or simply word files, all of which are highly insecure. Although there are a number of mobile Apps providing options[12], they all appear fatally flawed to various degrees, and hence an entirely new method of password storage is needed.

SOLUTION: Produce a new method of safely storing passwords, which is easy to use, secure and hence overcome many flaws of the current methods.

ADVANTAGES: The product fits particularly well into my area of interest – control technology as well as security. Additionally, the product would have the potential to have vast appeal reaching all corners of society, since through my research, this particular problem affects almost everyone. In fact, reviewing my own passwords storage methods, I have realised how insecure they are, and hence, the product could even help me.

DISADVANTAGES: The product would be quite challenging as it would likely have a mechanical, electronics and software elements, offering a time and complexity challenge.

2.1.4 Efficient Home

PROBLEM: There is overwhelming scientific evidence, that we must seek more carbon-free methods of living our lives, reducing greenhouse gases, simply due to the vast impacts any increases in temperature would create for the world. Hence, we are seeking innovative methods to be more energy efficient. In the recent past most of the focus was largely concentrated on the largest parts of emissions such as power plants, large factories etc., versus the small

[9] (Google, 2015)
[10] (Schneier, 2009)
[11] (Sophos, 2014)
[12] (Apple, 2016)

gains that can be easily made right in our homes. These small gains could however, quickly add up and make a large difference.

SOLUTION: Design a house with a focus on ensuring the lowest carbon footprint possible, by investigating the possibilities for every possible component, for example energy production using Solar Panels to lighting using LEDs as opposed to Incandescent Bulbs.

ADVANTAGES: It could pose real life advantages, if any true technical advance is made.

DISADVANTAGES: It would be very challenging in the time available for the project, given the scale of the requirement. Additionally, with my little knowledge of building designs and architecture, my ability to generate an innovative product design is relatively low, hence the project would require a large amount of work.

2.2 Criteria for selection

On a cursory view each project appeared to have strong attractions and issues. In order to decide which project would be most suitable, I made a list of important criteria for the selection of the project.

1. My number one criteria for project selection was my interest and how excited I felt about doing the project.
2. It was also important for me to understand my strengths and knowledge needed for the project.
3. My ability to complete the project in the time frame.
4. Whether the project provided an opportunity to further research a topic of interest to me.
5. Whether the idea and solution was innovative.
6. Whether the project carried real life benefit for the common person.
7. Scalability of my design/prototype for profitable mass manufacturing.

2.3 Decision

Based on the above criteria I came up with a decision matrix in which I ranked each of the projects on the various criteria. This exercise helped me to structure my decision making and gave me more clarity on the project that was most appropriate and I felt most excited about.

Figure 1: Decision Matrix

		Raw scores				Adjusted scores			
	IMPORTANCE	Smart Water Bottle	2-step authentication	Password Manager	Efficient Home	Smart Water Bottle	2-step authentication	Password Manager	Efficient Home
My interest and excitement	10	10	10	10	10	100	100	100	100
Time constraint	10	7	9	8	4	70	90	80	40
Strengths and knowledge	9	7	9	10	5	63	81	90	45
Opportunity for deeper research	8	6	8	10	8	48	64	80	64
Innovative product	7	10	8	9	8	70	56	63	56
Real life application	7	10	10	10	10	70	70	70	70
Scalability for large scale	6	8	10	9	7	48	60	54	42
TOTAL						469	521	537	417

Note: Importance and raw scores are rated out of 10 (10/10 being the best) and adjusted score = Importance x Raw score (maximum of 100)

In terms of criteria importance, my interest and ability to complete the project within the time available were the most important and hence got 10/10.

Not surprising all the four project options scored full marks on my interest in those projects and they had equally existing real life applications. The Password manager stood out in terms of providing me opportunity to learn more

about securities industry and its long term implications as well as my knowledge in the area. It was weaker in the time frame, but I decided with proper project management I could achieve it. The redesign of Two-Step authentication looked much more easily scalable while the smart water bottle appears to be the most innovative product.

But on the whole the Password Manager scored the highest total and hence was my first choice.

3 Project plan

3.1 Aim

After deciding on the problem I wanted to solve, I produced an initial brief which would define the project:

To produce a design for the mechanics, electronics and software of a secure handheld method to store passwords, ensuring it is not hackable by current means – and thus very secure. It should make use of biometrics to ensure that the products only responds to the intended user. It should also be especially easy to input and recover data.[13]

It was highly important for me to carefully plan the entire project to reduce the probability of omissions. Additionally, planning would allow me to more closely track my progress through the process to ensure that I was not running out of time. Finally, through thorough planning I could ensure a greater efficiency of work, for example by ensuring that all dependencies for a task were completed before the task itself was started. The first method for planning and allocating my time most effectively was simply to list all the necessary tasks that were required to be completed and writing out the dependencies and the time required to do this.

3.2 Plan and task

The first phase of the project involved background research into the problem and a decision on the target market for an appropriate product design. From here, a specification could be defined followed by advanced research. The most important concepts to research were security concepts, hacking techniques and algorithms for encryption. Additionally, in order to select the correct components, research was required on the control systems. From a mechanical perspective, there was also a need of research into possible materials and geometries.

The next phase of work was the initial ideas of development where the research was brought together to form a complete electronic design, while a number of mechanical ideas would be proposed and developed until one final idea was chosen. Meanwhile, there was a necessity of testing the electronics, using the components selected to begin to produce completed code for the product. The final phase was bringing the entire design together and designing the final artefact and justifying all decisions made, while writing a final report.[14]

Research:

a. Security
    a. Different Hacking Techniques – Brute Force, Packet Sniffing, Social Engineering and Viruses
    b. Biometric Security Systems – Iris Recognition, Sensory Recognition, Fingerprint Sensors
    c. Existing Algorithms for Encryption – Shift Ciphers, Polyalphabetic Ciphers, One-Time Pad, Diffie-Helman Key Exchange, RSA, AES (Advanced Encryption Standard)
b. Control System, including which components to use – User Interface, Input / Output Design, Password Adding Systems, Specific Component Choices
c. Mechanical, including the material and looking at geometries for the product – including looking at the usecase of the product and determining the stresses and challenges of the product from a a mechanical standpoint, and hence, the best materials.

Ideas and Development

a. Control System testing and development – making any required changes as components are tested.
b. Mechanical System – producing a number of ideas which would be evaluated.
c. Software Design – designing all the software for the product – looking at the methods used and problems encountered.

[13] Taken largely from Project Proposal Form
[14] A complete comprehensive list of tasks devised is available in the Appendices.

a. Software Design – write all the code that would be required to get the product working
b. Electronics Design – all the required information for the product to be manufactured including the design of a PCB
c. Mechanical Design – including the material, geometry and manufacturing techniques which would allow the product to be manufactured in a larger scale.

3.2.1 Milestones

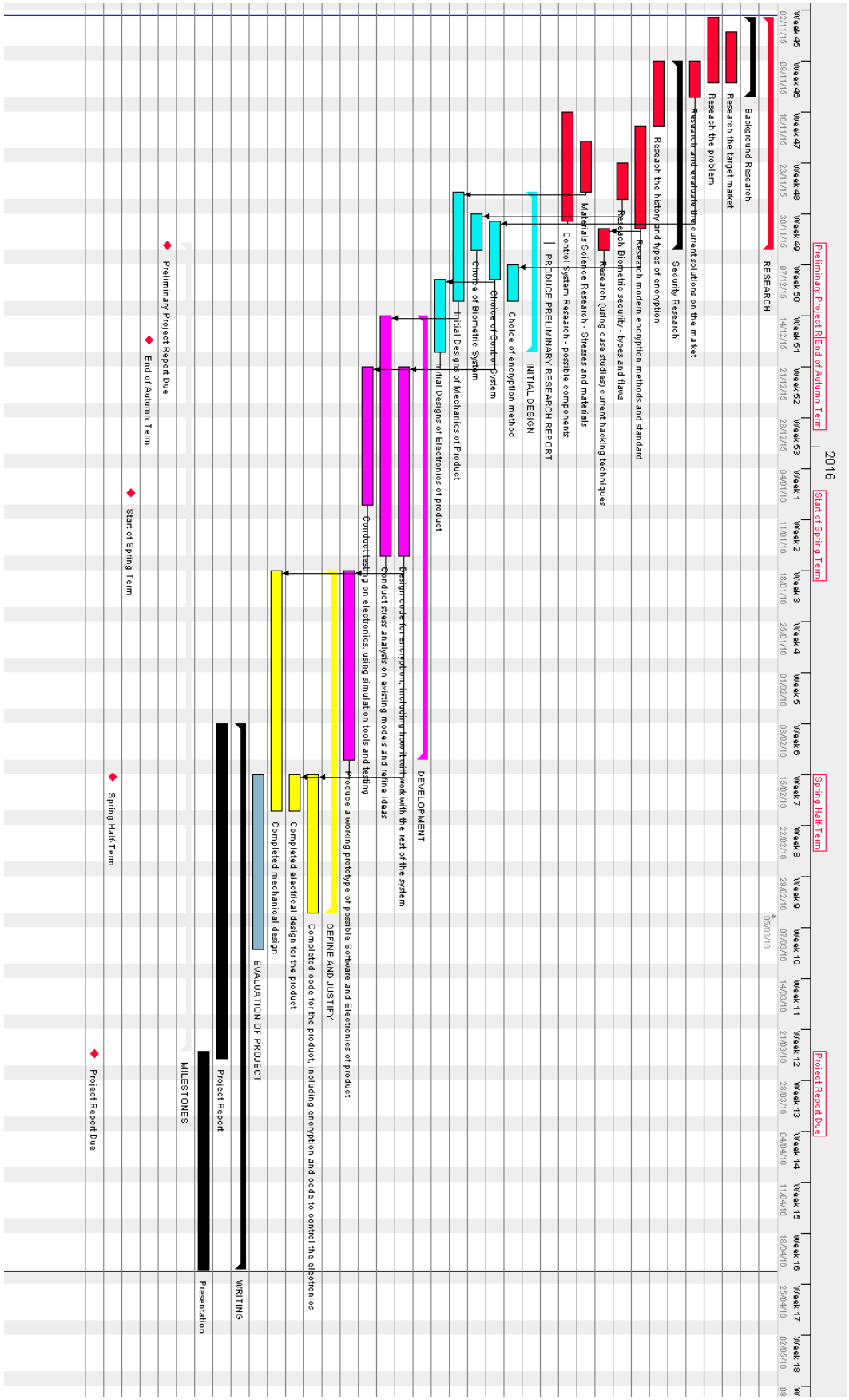
During my initial planning, in producing the first of many Gantt Charts, I established two main milestones, the 18th December 2015 and 15th February 2016 for completing the research and definitions respectively. At those points, I completed small reflections to ensure that I had not missed any important tasks, and allowing myself time to replan the next phase of work, with any changes I had found necessary during the last stage.

3.2.2 Gantt Charts

I used a Gantt Chart to more graphically track project progress, helping to ensure that I was not falling behind. The Gantt Chart also easily showed all the connections that occurred throughout the product, showing the impact of delays on other activities. The ability to track the Critical Path meant that I could give the most attention to the tasks which would impact this path. Throughout the project, the Gantt Chart changed due to some tasks proving simpler than I had expected, while others more complex, taking more time than initially estimated. The Gantt Chart proved an invaluable help allowing me to chart out more time ensuring that I could spend the required time on the complex tasks while still ensuring that I would finish the project in time.

This clearly showed the idea of specific phases, with the vast majority of research occurring before initial ideas and development, with each following on from each other. However, though effective in the beginning, this soon had to change and the lines soon blurred between the phases, as shown by the final Gantt Chart. Below I have provided Gantt Charts for beginning and end of the project. There are three in-process Gantt Charts in the Appendices.

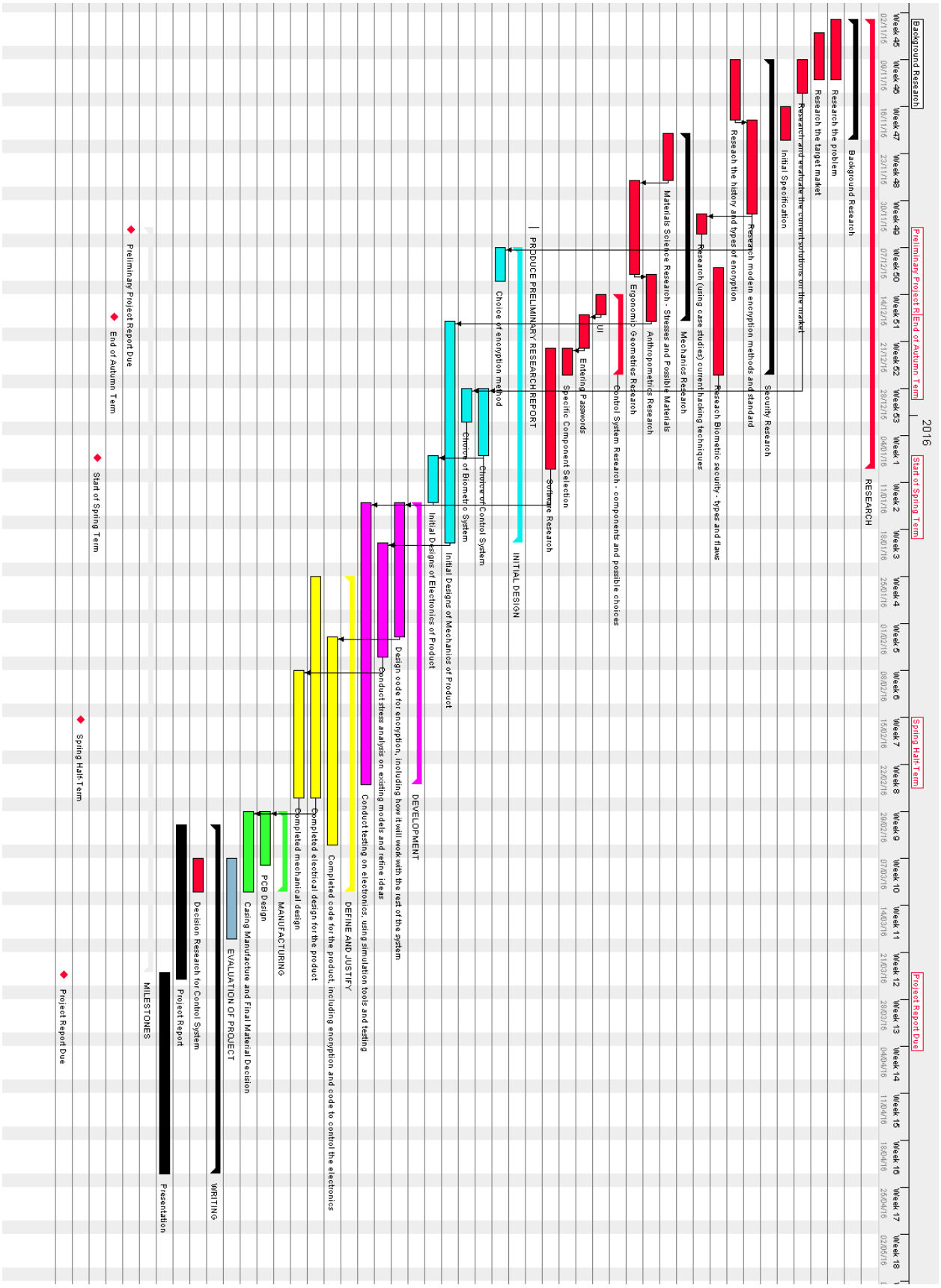
Figure 2: Gantt Chart – beginning of the project



Source: Gantt Chart produced using GanttProject[15]

[15] (GanttProject, 2016)

Figure 3: Gantt Chart – End of the project



Source: Gantt Chart produced using GanttProject[16]

[16] (GanttProject, 2016)

A comparison of the two Gantt charts shows a clear reduction of the time allocated for the Project Report as a number of the tasks took longer than I had anticipated. The Gantt Chart allowed me to ensure that they would still finish in time for the completion of the project itself, often by focussing on the tasks which were dependent on the delayed task.

3.2.3 PERT Charts

PERT charts offered a less comprehensive view of the project and was often useful to find the flow and see the next tasks which should be completed. When used in conjunction with Gantt Charts they allowed me to easily see the position through the tasks the project is. The PERT charts produced are available in the Appendices.

3.2.4 Journal

Through the process I maintained a complete journal, in which I wrote every time I did any work on my project. As well as writing any decisions I had made (as well as justifications) I also completed reflections on how I had worked. This allowed me to more effectively recall why I had made certain decisions, creating a more effective Activity Log as well as a more thorough report.

4.0 Background Research

4.1 The Problem

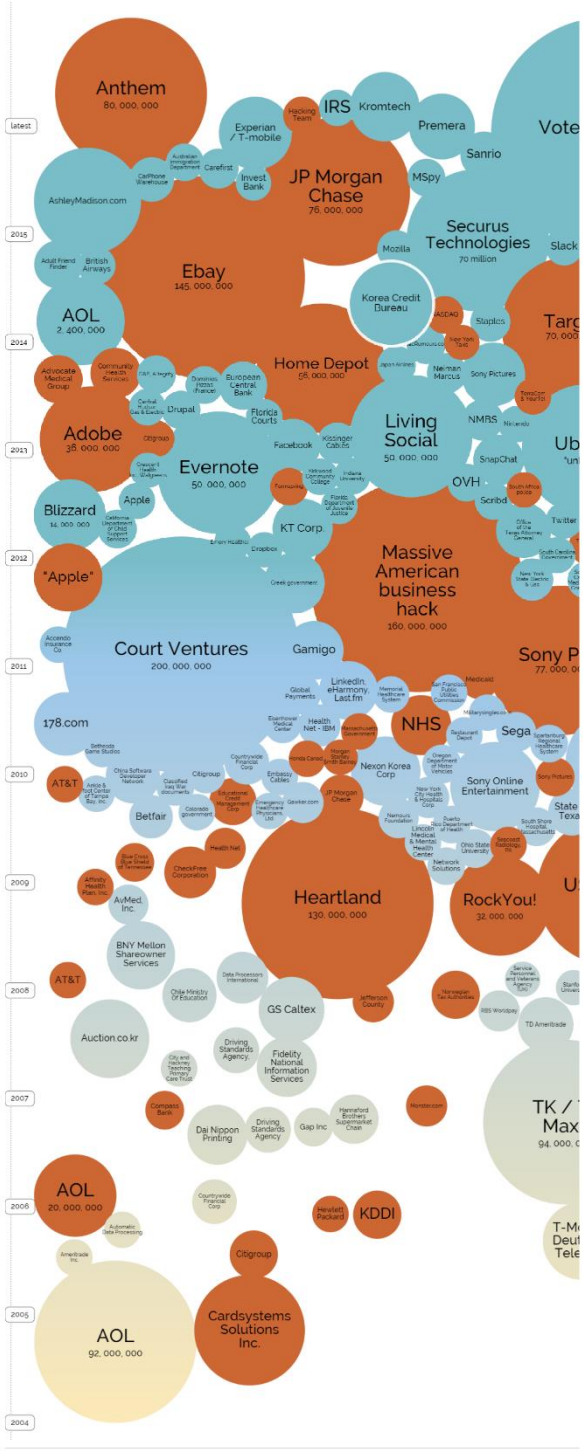
Today, the world is becoming more and more connected by the internet, as people become reliant on its services. Around the world, just under 50% of the people have access to the internet at home[18], while in the western world, over 80% of people have this access[19]. Ever since the CEO of IBM stated that the market for personal computers was around 5[20], people have attempted to prove him wrong, with around 350,000 computers sold every year today[21]. And while computer sales in the western world peaked in 2013[22], the rise of portable computers are only rising[23], with mobiles, tablets and laptops filling the void of desktop computers. With this momentous rise of the internet in every devices, the number of passwords an average person has, has vastly increased, reaching around 19 today[24]. Every single website (including some of mine!) require you to create a password, making them intensely hard to remember, especially given the list of recommendations that arise with them. While most sites require passwords of at least eight characters, most recommend including at least one capital letter, one lowercase, one number and one special character[25]. And with this comes advice to never reuse passwords[26], even parts of them, as well as changing them at least every couple of months, making remembering passwords, and password management generally, quite the challenge.

The penalty for failing to meet this advice is also quite daunting, as stories reach us about identity thefts[27], to monetary fraud, as hackers manage to extract credit and debit card details from our online shopping accounts[28]. In fact, 2/5 people have in the past year[29] been a victim of an identity theft.

To say that the problem is hidden is certainly not true, as this is highly broadcast, both over traditional and modern media. Every day we hear of another breach, including a high-profile hack into the Defence Department in USA[30], hacks into Talk-Talk[32] and streaming service Netflix[34] in the last few months. In fact, this figure showing the largest recent hacks (by date) appears to read like a directory of the largest companies in the world. If these attacks can be perpetrated upon the most secure systems in the world, with the US CyberSecurity efforts being funded to the tune of $14bn[35], imagine how easily accounts can be hacked when one's password is '123456' or 'password'.

[18](Internet Live Stats, 2016)
[19] (Internet Live Stats, 2016)
[20] (PC World, 2008)
[21] (Statista, 2015)
[22] (Statista, 2015)
[23] (Statista, 2015)
[24] (Sophos, 2014)
[25] (CERN Computer Security, 2013)
[26] (Lifehackers, 2011)
[27] (United States Department of Justice, 2015)
[28] (WIRED, 2013)
[29] (Experian, n.d.)
[30] (Reuters, 2015)
[32] (BBC, 2015)
[34] (TIME, 2016)
[35] (Financial Times, 2016)

Figure 4: Data breaches (number of records lost)



Source: Information is Beautiful[36]

Over 8/10 people are worried about their online security, yet 60% reuse the same passwords[37]. There is simply not a good method to manage all the passwords, as a result they are stored easily and insecurely. In many ways, this is a huge problem which is preventing the further maturing of the internet, as 7/10 people no longer trust their online security[38]. And yet 21% people use passwords that are over 10 years old[39], while 47% use passwords 5 years old[40].

[36] (Information is Beautiful, 2016)
[37] (Entrepreuneur, 2015)
[38] (Entrepreuneur, 2015)

Methods of better security other than passwords are slowly arriving such as two-factor authentication, which requires you to use your smartphone as well as your password, but are not replacing them. And while there is a large proportion of the population choosing poor passwords due to lack of better storage systems, there is also a part of society, largely the elderly demographic where they are yet to be educated of these needs of the passwords. Thus, a password management system could have a number of possible target markets as well as having a huge general appeal, given the scale of the problem today.

4.2 Methods of Passwords Storage

In many way, the choice of target segment that I want to attempt to reach depends on the severity of the problem for that segment. Given the flaws of each solution for password management, the needs of various groups appears to be very different. Thus, the first thing I had to research was the various available options, their markets and strengths and weaknesses.

4.2.1 Paper

From the research that I have completed, the 'low-tech' manner of storing the passwords on paper appears to be used the most by the elderly, who are most comfortable with paper, especially as they attempt to integrate themselves with the new technology, attempting to do one thing at a time. With just a brief search of the internet, one can find a number of elderly people complaining about the number of passwords. In fact, experts are even suggesting storing passwords on a physical notebook, in order to keep it as simple as possible, while the technical advice (from the Guardian) even suggests reusing 'the same password … for the less important sites'[41], which most security experts reject, given the amount of information that can be incrementally gleaned from multiple accounts. In fact, in a recent attempt to do this, Wired, the technology magazine managed to get from one solitary password (in fact the Apple account password) to all necessary information to commit identity fraud (including Social Security Numbers and Credit Card Information), a truly scary prospect[42].

Paper storage appears to shine a light upon one of the most vulnerable groups of people, that of the elderly, who are not necessarily aware of more safe methods of storing passwords, which often requires more technical knowledge. As will be shown later, my primary research shows that the vast majority of elderly in my area tend to use this method. Also, the system is clearly highly insecure. How often does one lose a book? Or worse still, if it is stolen (either accidentally or purposefully)? This is the ultimate flaw of this system, that the passwords could all be taken or lost without any warning. Your cleaner, plumber and anyone who enters your house with only a little work could easily take all passwords. One must carry around the notebook at all times, and changing passwords means the information in the notebook is quickly out of date. Finally, while finding one password from a handful is generally fine, as the number of passwords inevitably grow, into the hundreds, the simple finding the correct password is a hassle.

MARKET – *Elderly*

ADVANTAGES – *Simplicity*

DISADVANTAGES – *Irritating with particular challenge of finding the desired password, insecure as anyone who finds the paper would have the passwords, impractical as the paper would have to be carried everywhere you go.*

[39] (Entrepreuneur, 2015)
[40] (Entrepreuneur, 2015)
[41] (The Guardian, 2015)
[42] (WIRED, 2013)

4.2.3 Word Processed Files

This is the next stage up technically, with the people who tend to use this system tending to be of a younger demographic than those writing their passwords on paper, since the system requires some degree of technical abilities. This greater use of technology brings with it a raft of benefits, including the ability to find passwords from anywhere, if a cloud-storage system, such as Google Drive[43] or OneDrive[44] is used. Additionally, finding passwords is a matter of using built in systems in the software to search for the keywords. However, rather inevitably, this system is in many ways susceptible to being more flawed. Generally, the storage of passwords on a network allows for a vastly greater number of people to attempt to gain access to the passwords, as opposed to the relatively small circle for writing on paper. While some may encrypt these files, the encryption of many document formats is generally basic and easily hacked. Systems like Dropbox have famously been hacked in the past[45], meaning the assumption that the document could not be accessed is not true. Additionally, the system is clunky, with passwords poorly arranged, and with the system generally being left to the consumer. Also, for those with limited computing abilities, such as the elderly, working with word processers is as challenging, if not more so, than remembering password, thus taking an inordinate amount of time.

MARKET – *Middle aged people*

ADVANTAGES - *Relatively simple, backups can easily be kept, practical (as document can be stored on mobile device and carried around easily)*

DISADVANTAGES – *Very insecure, still hard to organise passwords*

4.2.3 Dedicated Password Management Software

The heralded option by most security experts on the internet is the Dedicated Password Management Software, such as the popular LastPass[46]. They claim to have very strong security, with passwords being unhackable[47]. Additionally, the use of the system would allow one to not be attached to a device or notebook, as one can access the passwords over the internet. Also, one only needs to remember one password, which is a truly simple way of doing this. In addition, since the software only has one purpose, it has an effective user interface which allows people to easily find the password they want as well as adding new and editing old passwords. However, this system has a number of flaws. Firstly, the system requires you to remember one password, which if you forget there is no recovery system, as there would be for cloud storage systems. Additionally, this one password could be hacked (single point of failure), and the system relies upon the security of your password and the secrecy of your username or email address. Also, although LastPass advertise themselves as having 'very strong protection'[48], they were in fact hacked as recently as last year, where a number of user passwords were stolen[49]. Systems are inevitably built to withstand the most advanced hacks that have already occurred, thus are liable to new forms of hacking which have never been exploited. Additionally, since one hack gains access to huge numbers of passwords, these pieces of software are inevitably large targets for hackers. Meanwhile, the consumer in the case of a hack stands to lose lots of information, opening up the possibilities of identity and monetary frauds. There is also a problem with the software in terms of complexity. Though they advertise being as simple as possible with simple, clean UIs (User Interfaces) when I showed LastPass to some of my elderly neighbours, they struggled to understand how to use it thereby implying they would be uncomfortable using it to store all their passwords. In fact, the number of people

[43] (Google, n.d.)
[44] (Microsoft, n.d.)
[45] (Business Insider, 2014)
[46] (LastPass, 2016)
[47] (LastPass, 2016)
[48] (LastPass, 2016)
[49] (LifeHacker, 2015)

that use a password manager according to a survey funded by Roboform (a password management system) was 8%[50], simply showing how most consumers are either unaware of it, or find it irritating to use. Additionally, a large proportion of that 8% are in the technology sector, where people are not necessarily average in terms of the number of important passwords that they have, thus accepting of the irritation such a system might cause.

MARKET – *Generally young technologically savvy people as well as people who require more security – for example handling large sums of money.*

ADVANTAGES – *Appears to be generally more secure than other methods, more tidy (duplication and arranging can be dealt with), very portable (mobile apps ensure that password management system can be carried anywhere you want).*

DISADVANTAGES – *Sense of security appears to be false (given recent hacks), requires dedicated software which is often paid for.*

4.2.4 Survey

In addition to the successes and failures of the various forms of password management, it is important to consider the relative numbers and demographics of people using various password management systems. Thus, I created a series of questions, which I asked my neighbours, parents, relatives and friends. I also created a quiz (using SurveyMonkey[51]) and put it on my Twitter[52] to encourage my followers and anyone else to fill the form.

The questions and results are provided in Figure 5 and 6.

Figure 5: Questionnaire on password use

Which age bracket do you fit into?

a. 0-20
b. 21-40
c. 41-60
d. 61-80
e. 81+

How do you store your password?

a. Word Processor or similar
b. Password Management software
c. On paper
d. Any other

Source: SurveyMonkey[53]

Figure 6: Analysis of questionnaire results

	0-20	21-40	41-60	61-80	81+
Word processor	13	2	9	1	0
Password management software	3	0	0	0	0
On paper	6	1	3	12	1

Source: Data found through SurveyMonkey[54] results systems

[50] (Roboform, 2015)
[51] (SurveyMonkey, 2016)
[52] (Ahuja, n.d.)
[53] (SurveyMonkey, 2016)

The results appear to reflect the background research I had already carried out, showing that as age changes, the majority of people store passwords differently, with the elderly tending towards storing passwords on paper, and the younger people moving towards more technologically savvy methods.

4.2.5 Conclusions

After assessing the various possible target markets, I found the elderly people best primary target market to focus on. Though the methods of word processing and password management software are not without their various flaws, many of the issues I hope to address are most relevant to elderly. Many who I talked to want to move to more technologically savvy methods, but having tried them, were unable to understand or use them easily.

Thus, one of my primary aims and specifications must be to ensure that the operation of the product is simple, such that anybody, including the elderly can use it easily. Additionally, as discussed in the introduction to the problem, the product could also meet the need of helping the elderly ensure their passwords are strong enough, educating them, as many who I talked to did not truly understand the general guidelines for password selection, for example the vast majority reused the same or very similar passwords for all accounts.

4.3 Target segment Market Research

4.3.1 Elderly and Technology

Today's elderly were termed by the BBC as the 'generation that tech forgot'[55], living during the period of great technological advance but ostracized and isolated by it, as there was and continues to be little technology specifically designed for them. Meanwhile, it is clear that the vast majority of the elderly do want to use technology, with over 70% expressing this in a YouGov poll[56], while when conducting a discussion with elderly in my street, one said that 'most of my friends and I do want to learn how to use the internet but it's very difficult'. Throughout the conversation, the theme of complexity arose many times, and hence many are finally making inroads into the new technological generation through tablets such as the iPad which have proved much simpler to understand and use. However, even when using the internet, there are a number of problems that pose a challenge. Major among them is the problem which I am setting out to solve, that of passwords, choosing and remembering them, affirming the need for my solution.

4.3.2 Medical Research

I found to better understand the need of my target audience, I needed to complete research into the problems that they face, both during normal aging and the additional problems that people suffering from ailments faced. During aging, as the reputed online doctoring website, WebMD describes, the body experiences 'monumental changes'[57] in everything from height to memory to fine motor control. For the product, it is important to note that only certain changes have any affect on the ability to use or need a certain product and hence only these will be investigated. For example, the fact that 'skin become more elastic' though perhaps highly significant to many of the elderly appears to pose little challenge to the product.

Memory: Primary to the creation of the problem, especially given the idea that the elderly appear to have the problem of remembering passwords even when there are very few, is the degradation of memory as aging occurs. As people age, there are a number of physiological changes that occur that can cause glitches in the function of the brain that means that it takes longer to learn and recall information = meaning recalling passwords becomes a larger

[54] (SurveyMonkey, 2016)
[55] (BBC, 2015)
[56] (YouGov, 2014)
[57] (WebMD, n.d.)

and larger challenge. This phenomenon is known as Age-Related Impairment[58], and it is caused by the brain pathways leading to the hippocampus becoming degraded over time, hence meaning any signals require longer to be transferred. Additionally, 850,000 people in the UK alone currently suffering from Dementia[59], with 95% of these people elderly (aged over 60)[60]. This corresponds to over one sixth of people aged over 80[61], with 80% of those in care homes suffering from some form of Dementia[62]. In 2025, according to the Alzheimer's Society the number of people in the UK suffering from Alzheimer's will be over one million people[63]! Primary symptoms of dementia are 'frequent and progressive memory loss'[64] as well as 'general confusion'[65] and 'inability to perform familiar tasks'[66], hence making the task of remembering passwords more challenging. In conclusion, given the number of problems associated with the memory in the elderly it is clear that the product could provide some help.

Vision and Hearing: With regards to vision, the main impact of aging, is the increased problems with focusing on objects that are near, known as presbyopia[67], as well as leading to cataracts[68], where the lens becomes clouded, hence leading to clouded vision of the person. While the former is relatively easily and unpainful remedied by the use of glasses of contact lenses, the latter requires an operation to replace the lenses, replacing them with artificial equivalents. Though the operation is widely completed and safe, 65% of over 70s in America suffer to some extent from cataracts[69] and many have received the operation with few issues, since most do not get the operation until the cataracts have been formed extensively, hence a large number of the elderly continue to suffer with them affecting their vision mildly, the product should be able to work for them. In order to deal with both the concerns, good brightness and the contrast of the screen are necessary.

Hearing also declines rather predictably, and generally occurs gradually, with 50% of those over the age of 70 having 'difficulty hearing'[70]. While a number of conditions including diabetes and high blood pressure can lead to hearing loss, it can also be simply be caused by the age, which leads to abnormalities in the Tympanic Membrane or changes in position of the Auditory Ossicle, in particular leading to problems hearing high frequency (pitch) sounds, meaning that these should not be used if possible.

Hand Movement: In general, even without the influence of other illnesses which affect the motor control, there are a number of issues associated with the movement of the hands. Firstly, there is generally a reduction in muscle strength of around 25-45%, often due to 'sarcopenia of old age'[71] which leads to a large reduction of muscle mass. There are 26 muscles directly linked to the function of the hand, each of which is individually weakened over time by the sarcopenia, hence, the total strength of the hand is reduced significantly. In fact, in a test of grip strength, the average 60 year old male performed 25% worse than an average 20 year old[16]. However, even worse affected is the control of individual fingers, due to the degradation of the hand tendons, which communicate the actions of the fingers muscles to the fingers themselves, and hence are very important. At birth these tendons have a very high tensile strength, but over time, given the constant changes in the tension of them, this tends to break down, by the

[58] (O'Brien, 1999)
[59] (Alzheimers Organisation, 2015)
[60] (Alzheimers Organisation, 2015)
[61] (Alzheimers Organisation, 2015)
[62] (Alzheimers Organisation, 2015)
[63] (Alzheimers Organisation, 2015)
[64] (Alzheimer's Disease International, n.d.)
[65] (Alzheimer's Disease International, n.d.)
[66] (Alzheimer's Disease International, n.d.)
[67] (WebMD, n.d.)
[68] (WebMD, n.d.)
[69] (National Eye Institute, 2011)
[70] (MedlinePlus, n.d.)
[71] (Eli Carmeli, 2013)

losing of the outer coating, the para-tendon. This reduces the ultimate tensile strength by between 30 and 50% (to around (75kg/mm)[18].

In addition, there are a vast variety of conditions (occurring more often in the elderly) which could create problems with hand movements, the list spans:

- Alzheimer's Disease
- Parkinson's Disease
- Huntingdon's Disease
- Wilson's Disease

Figure 7: Impact of diseases on Fine Motor Control and Grip

	Fine Motor Skills	Grip
Alzheimer's Disease	✓ ('Loss of Fine Motor Skills')[72]	✓ ('decreased grip strength')[73]
Parkinson's Disease	✓ ('People with Parkinson's Disease can see limitations in fine motor coordination')[74]	✓ ('loss of grip strength')[75]
Huntingdon's Disease	✓ ('changes in fine motor skills that might be noticeable in skills such as handwriting')[76]	✗
Wilson's Disease	✓ ('often experience … loss of fine motor skills')[77]	✗

4.4 Initial Product Specification

From the above background research of the elderly I arrived at a list of specification points for my password manager product. Using this list, I went on to define which items have to be part of a primary specification (green) and others that were desirable but not necessary (blue) and hence be secondary specification.

Function

1. The product must be able to store passwords.
2. The product must have an easy system through which the user could add new passwords – for example if they sign up to a new account online.
3. The product should have a system to allow the passwords to be backed up – either automatically through the cloud or manually.
4. The product should be easy to use – so that it could be used easily by elderly without much explanation .[78]
5. The product should educate the user about secure passwords – analysing inputted passwords.

[72] (Eli Carmeli, 2013)
[73] (aPlaceForMom, 2015)
[74] (Center for Movement Disorders and Neurorestoration, 2012)
[75] (Rush University Medical Center, n.d.)
[76] (Duke Department of Neurology, n.d.)
[77] (Brodsky, 2010)
[78] From the research the advantages of the dedicated password management systems and word processing was that the data was automatically backed up, ensuring that if one copy was lost, the list could still be recovered. Hence, if possible, this should occur on the designed product as well.

Control System

1. The product should have a bright, high contrast screen to ensure it could be used despite any issues that the users might have with sight.
2. Any sound used by the product should be loud and avoid making use of particularly high frequencies.
3. The product should use biometric technology to avoid user to remember a master password.[79]
4. The product should make use of encryption and other means to ensure that hackers cannot easily gain access to the product.
5. The product's control system should be designed to ensure that it could be used easily with reduced fine motor skills.

Mechanics

1. The product should be light, and easy to handle.
2. The product should be designed with reference to ergonomics and anthropometric data, hence being comfortable in a user's hands.
3. The product should be able to withstand normal, indoor use easily.
4. The product should be able to withstand being dropped.[80]

Other

1. The product should be relatively cheap, under £50.33, the average maximum price that the elderly people I asked said that they would pay for such a product.
2. The product should meet BSI, CEN and ISO Safety Standards so that the product could be sold around the world.[81]

[79] From additional research it is clear that the use of biometric technologies could be easily used to ensure that a master password would not be required. In fact, in my focus group, a number of them had new iPads with TouchID (fingerprint sensor) and found it to be a very effective mechanism of entering the device.

[80] This was another recommendation by my focus group who said that due to simple general cclumsiness the product would likely be dropped and hence I should ensure that it would withstand these drops.

[81] Despite this being highly unlikely to get certified, given the time pressures, it should be considered to ensure that if the product were to be more widely sold, it could occur easily without having to be drastically redesigned.

5.0 Further Research

5.1 Security Research

At the forefront of the specification that was defined was the need for the product to be secure. One of the major problems with the Paper system was the lack of security and for technology-related products, the greatest challenge is hacking, which evolves at a rapid pace, attempting to find holes in the latest security techniques. However, to stay ahead of the hacking, we must first look at the current hacking techniques, and see where they are ineffective and hence allow us to make our product more secure.

5.1.1 Current Hacking Techniques

5.1.1.1 Brute Force

Brute Force is a trial and error method used to crack passwords, generally cycling quickly through all the possibilities required before finding the correct password, just like a criminal might crack a safe by trying as many possible combinations. The method is infallible but highly time consuming. Generally software is used to break-in, allowing the number of passwords to be tried to be vastly increased. The best method of preventing the brute force method from being effective would be to use a longer (and more secure) password, as it would likely be further down the list of passwords the brute force algorithm would try to guess. For example, an eight letter password, with 3 letters and 2 symbols would take an average PC over a day to crack[82]. Hence, in order to prevent brute force working on the device's passwords, the user could be told how to make passwords better. To ensure that the device master password is not stolen, a larger issue, the biometric method which is to be used must be investigated to ensure that it is complex enough.

5.1.1.2 Packet Sniffing

Packet sniffing is the act of intercepting information as it moves around, largely over a network such as the internet. For example, this is the system which prevents the information from being secure if you log into online accounts when from an insecure Wi-Fi, such as the one found at local cafes. This is largely completed through spoofing other computers on the network, telling them that the computer is a router and that the traffic ought to pass through the computer. Hence, the computer can decode the traffic, allowing the computer to determine what device the request came from and the information it sent, for example the username and password of the account that the user is attempting to access. This method is also widely used by the National Security Agency in the US. It intercepts the emails as they are sent between email servers around the world. One of the best ways to ensure that there is no packet sniffing is to ensure that the product is offline, therefore ensuring there is no online transfer of data. Additionally, this would also reduce the number of people who could attempt to break in using brute-force methods, since they would have to have physical possession of the product to crack the password. Additionally, ensuring the data is encrypted at all times even if it were sent over the internet would add a layer of security.

5.1.1.3 Social Engineering

Social Engineering is a rather more personal form of hacking than the above and is largely used in very personal forms of attack, especially when working with valuable systems, such as safes. It involves getting to know the victim, studying them in order to gain an understanding for their actions. This allows a hacker to more easily gain access to accounts by making use of the people's subconscious ticks for example individual preferences which the person themselves may not know. Another form of social engineering is physically tricking the person into giving the hacker the details they want, including through the use of Phishing, where an email (looking legitimate) is received requesting the 'verification' of information, linking to a fraudulent website which requests the desired information.

[82] (Random-ize, n.d.)

5.1.1.4 Viruses

Viruses are also often methods of getting information from people. When a computer is infected it can do many things including relaying the screen or the keystrokes to a hacker. If the user were to log into an account on that computer, the hacker would now have access to this. These spread in a large number of ways. Most popular is through the spreading of an application by a computer, after a human is attracted to read or open the application (because they appear to have something entertaining – such as pictures of Russian Tennis Player Anna Kournikova[83]). In order to prevent this, one should ideally make use of consistent checking for viruses, but however since this is unlikely to be an option on the product, it would also be effective to make the product offline, since then even if the product receives a virus somehow – for example through an infected USB stick, no information could be sent anyway.

5.1.1.5 Conclusions

From the above research it appears very clear that the product should be offline, having no external communications, since it would vastly reduce the number of people who could attempt to hack it. However, this would also have some downsides. This would mean that existing systems such as iPhones and iPads could not be reused (adding simply an application or suchlike) and would necessitate an entirely new handheld device. Additionally, it would limit the mechanisms for addition new passwords to the device, meaning that methods such as a keyboard (which are very clunky) would have to be considered. However, it is very clear that the primary objective of a password security product is ensuring the greatest security, and hence the product must be offline.

5.1.2 Biometric Security Systems

Biometrics are defined as the measurement and statistical analysis of people's physical and behavioural characteristics.

5.1.2.1 Introduction

During our primary research of talking to the elderly, it became abundantly clear that one form of password which was almost universally loved was biometrics, as these offered a form of password that they could not forget, since it was an inherent part of them. More than that, it was generally easy to use, and at least as it appeared to them, quite secure. Though the experience that they had had with the idea of biometrics largely only stretched to fingerprint sensors on the newest iOS devices (Touch ID[84]), it was worth investigating other biometric technologies as they offered many of the same advantages that the target users liked.

5.1.2.2 Capacitive Touch

A capacitive touch sensor is a form of identification that the person is human, and is often used when the identification is not really necessary. It is also used as a trigger of an on-off system, so that the device automatically turns on when a person wants to use it. Alternatively, by combining a number of sensors, in an array, it can be used to form a capacitive touchscreen display, such as those used in smartphones. From here, techniques such as passcodes could be used to ensure security, though the screen adds little advantage as compared to just using a physical keypad. The capacitive touch system works through relying off the conductive properties of the human body. Because of this, when the person touches the screen, there is an electrostatic distortion in the electric field set up in the screen, allowing the device to identify the touch and specific location of it and to a certain extent it's strength.

ADVATAGES: Very simple, relatively cheap

[83] (TechTarget, n.d.)
[84] (Apple, n.d.)

DISADVANTAGES: Carries no inbuilt security (though password security could be built on systems making use of capacitive touch)

5.1.2.3 Iris Recognition

Iris recognition is a method which performs mathematical pattern recognition on video images of the iris, which appears to have complex, random patterns which seem to be unique to each person. These are generally good as they can be seen from any distance, and are largely stable since they are hard to damage or change.

ADVANTAGES: As opposed to fingerprint scanning – small injuries should have no impact, iris patterns remain stable over decades – first establishing during embryonic gestation and provides fast scanning.

DISADVANTAGES: Requires relatively high quality cameras, requires high accuracy from the person – small area of taking pictures (can be highly tedious)

5.1.2.4 Sensory Recognition

Sensory recognition includes the exact ticks of the individual person including how they would otherwise enter passwords. It is often used as another step in commercial banking to ensure that the passwords are entered in the same manner as before, sharing the same pauses, with the same gaps, and emphases in terms of force, which a person might do subconsciously. Though this is largely still under development, a leader in this, AimBrain (a start-up based in London)[85] talked to me about how it was very easy to use, with users of the basic system only remembering a 4 digit pin, while the system was able to measure over one hundred other variables to measure the user's ticks and ensure the person attempting to get into the device is the same as the one who set up the device.

ADVANTAGES: Easy to use, very secure, fast, cutting edge, can be integrated into other biometric password methods

DISADVANTAGES: Very complex, not entirely tested, very expensive

5.1.2.5 Fingerprint Sensor

Fingerprint Sensors are by far the most common form of Biometrics used in the mass market, and hence includes a number of parts which mean it might be easier to include in my product than iris and sensory recognition. The system works through the investigation of a fingerprint, under the idea that every fingerprint is unique, with a specific pattern of arches, loops and whorls found in the finger, shown by the ridges and minutia points on the fingers. Fingerprint sensors generally use an optical camera to capture the finger, before isolating the arches, loops and whorls and generating a computerized binary map of the finger, a code which represents the exact information of the finger.

ADVANTAGES: One of the simpler forms of biometrics and hence more likely to be accomplishable in the scale of the project, very accurate and generally unique, fast.

DISADVANTAGES: Injuries can interfere with the scanning of the finger by changing the structure of the finger – cuts add additional ridges; fingerprint sensors are often expensive – costing more than a capacitive touch sensor.

5.1.2.6 Voice Recognition

In many old Marvel films we see the hero making use of voice recognition, presenting it as a technology of the future, as the computer evaluates the way in which the person sounds, evaluating the specific tonalities and rhythms of a particular individual's voice. These systems in fact have already been made and are commercially available, though the systems are not quite as sophisticated and it is often used as simply the first of many barriers of security.

ADVANTAGES: Easy to use

[85] (AimBrain, 2015)

DISADVANTAGES: Possible to forge, possible to steal other people's voices, still requires the user to remember a password, expensive.

5.1.2.7 Conclusion

It is clear that the best biometric technology that could be used is a fingerprint sensor as it would be the most achievable in a handheld system. Additionally, the only main problem, that of the unreliability of the system – and the lack of it working if one were to injure their finger could easily be resolved by ensuring the person had a number of fingers registered – so if one was injured, the person could use one of the other fingers that they had enrolled.

5.1.3 Encryption (Software Security)

A very important part of the security research was encryption technology. Encryption is a method that ensures that a hacker not gain access to secure information, even if they had access to the product. In order to investigate the various merits of encryption techniques, I travelled through the literature on advances in encryption, moving from the most primitive encryption, and how it has evolved into highly secure current technology.

5.1.3.1 Preliminary Definitions

Code – The set of letters, number, symbols etc., that are used to encrypt messages between two devices.

Cipher – A way of changing a message in order to keep it secret.

Cryptography – The process of writing or reading secret messages or codes.

Encrypt – To change information from one form to another, in order to disguise the true meaning of the message.

5.1.3.2 Origins of Ciphers

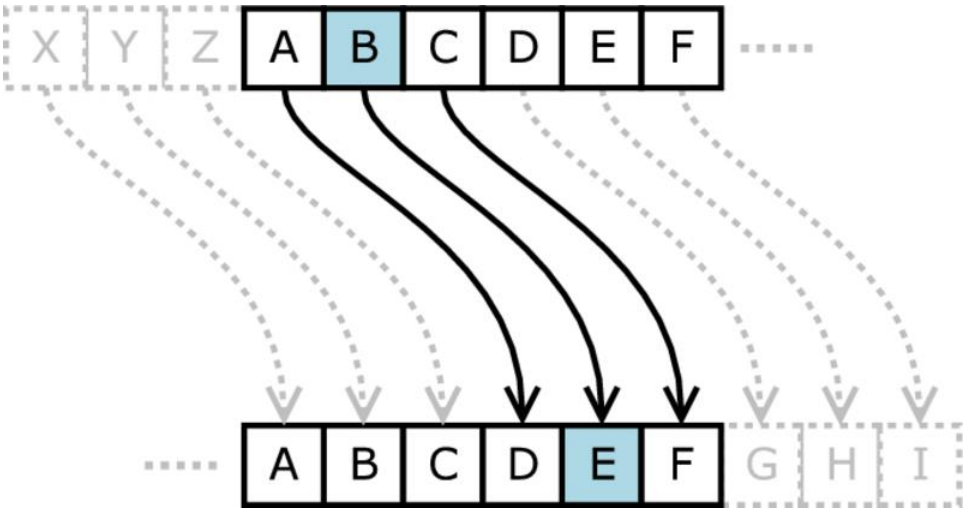
The first ciphers that we know of today originated in Ancient Egypt, created over 4,000 years ago in Menet Khufu, as an adaptation of hieroglyphics, in the tomb of Khnumhotep. Though generally agreed of as a cipher, in many ways, its purpose was in fact much different to what we might term as a cipher today: Designed to impart dignity and authority, it was intended to impress the reader, as they would understand the text, yet it was interesting and novel. Though this inspired the development of such ciphers, simple adaptations of the other Cyrillic scripts, as Professor Owen Lattimore of the University of Leeds states, 'literacy was always restricted to such a small minority that the mere act of putting something into writing was to a certain extent equivalent to putting it into code'. In many ways, this was a constant for the ancient world where messages would be hand delivered, and few could read it, even without a cipher applied. In many ways, the first true attempts to produce secretive codes were in India, where Arthashastra described the ways of the espionage services of India in the years prior to 500BC, where the tasks and reports were provided to superiors using code writing. In fact, Lalitavistara Sutra, which tells of the life of Gautama Buddha, describes the now god-like figure's prestigious skill of encryption, while the Karma Sutra, the infamous text describing how one reaches a virtuous life, but today is known for its graphic descriptions of human sexual behaviours, describes 'secret writing' (Miecchita-Vikalpa) as the 45th skill that all women should be proficient in. At this point, the ciphers were largely restricted to substitution ciphers, where consonants and vowels were swapped, or there were random reciprocations.

5.1.3.3 Types of Ciphers

As I intend to make use of at least one cipher to ensure that the passwords stored in the system are protected, I intend to make use of at least one form of encryption. Thus, I must look at the various ciphers that exist, and the various advantages and flaws of these ciphers, including the feasibility of coding and how hackable they are, both by humans and by software. The complexity and thereby the general security of the ciphers largely also progress chronologically, reflecting the advancements in cryptography.

Shift Ciphers – Caesar Ciphers

Figure 8 Diagram showing a Shift Cipher



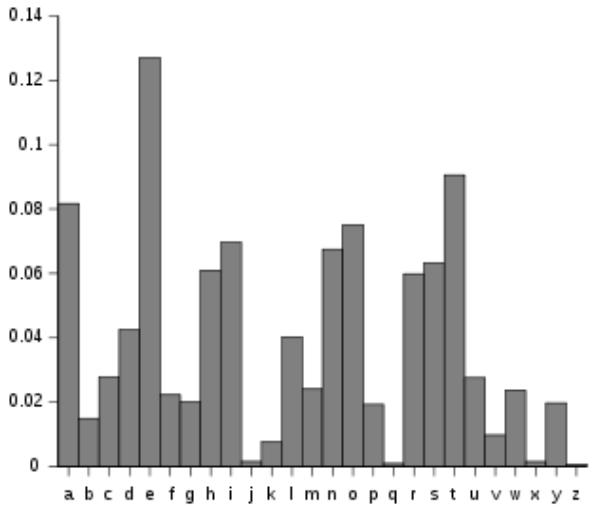
Source: Wikimedia Common Image[86]

One of the earliest form of ciphers (as expressed by Figure 8), as used famously by Julius Caesar, to ensure secrecy in messages was a shift cipher where each letter of the message was moved a certain number of characters up or down the alphabet, such that a shift encryption with a 3-shift of CAT is FDX. This is in essence a very simple cipher, and relied mostly upon the reader not spending the time to break the number of possibilities, clearly 26, the number of possible shifts that could occur, each one producing a different result.

Frequency Analysis

Over time, a more sophisticated form of breaking shift ciphers and in fact a number of other ciphers, was developed by the Arab polymath, Al Kindi, in his paper, 'A Manuscript on Deciphering Cryptographic Messages'[88], by investigating the text of the Qu'ran.

Figure 9 Graph of frequency of characters in English



Source: Wikimedia Common Images[89]

[86] (Wikimedia Common Images, n.d.)
[88] (Singh, n.d.)
[89] (Images, n.d.)

The process relies upon the fact that in certain languages the frequency of certain letters is characteristic, with Figure 9 showing the graph for English. Thus, by investigating the frequency of letters in the encrypted text, one can easily see the code, as the graph would likely be shifted to the side, such that the peaks and troughs would be moved a certain amount. Therefore, one can easily find the likely shifts, reducing the number of attempts required to find the exact shift used.

Substitution Ciphers

A substitution cipher is an expansion on the shift cipher, where a predesigned code (such as DFHU…) is used to define where the letters of the message to be encrypted are transposed to, such that in this example: CAB becomes HDF. The number of attempts that would be required to find the specific code in this is much higher, such that it is much harder to break by hand, in fact this is 26! (26x25x24x23x22x21…x3x2x1) = 403291461126605635584000000, since A could be replaced by 26 characters (A through Z), B by 25 (A through Z, except the replacement for A) and so forth. Though in essence the code is easy to implement, one must ensure that both parties and no one else knows the code, generally a challenge, given it should be a random string of 26 characters. Additionally, this is possible to break, using a variation of Frequency Analysis, since one can tend to find the most frequent character (likely to be an E in English), and then next most frequent character etc. Additionally, most systems would also make use of the fact that English and most other languages include a number of repeated words, which can be isolated to find the specific substitutions that are used in that case. Finally, it must be noted that once a few substitutions have been found, the number of possibilities to check reduces very quickly, since for many things to be readable, the entire code is unrequired. For example, it is trivial to work out that 'T?E TI?E IS ?ID?IGH?' is likely to be 'THE TIME IS MIDNIGHT' at which point the entire substitution code is found.

Polyalphabetic Ciphers – Vigenere Ciphers

The polyalphabetic cipher was the next step, after the Shift Cipher, where each character was shifted by a different number of position. It included a way in which the code (as a word) itself could be memorised, thus making the process of being the coder and the decipherer much easier. The process works in the following manner, let's say the code is RABBIT, and the message to be coded is HELLOMYNAMEISASHWIN:

Figure 10 Table showing example encryption using a Polyalphabetic Cipher

	H	E	L	L	O	M	Y	N	A	M	E	I	S	A	S	H	W	I	N
	R	**A**	**B**	**B**	**I**	**T**	**R**	**A**	**B**	**B**	**I**	**T**	**R**	**A**	**B**	**B**	**U**	**T**	**R**
	8	5	12	12	15	13	25	14	1	13	5	9	19	1	19	8	24	9	14
+	18	1	2	2	9	20	18	1	2	2	9	20	18	1	2	2	9	20	18
	26	6	14	14	24	33	43	15	3	15	14	29	37	2	21	10	33	29	32
=	26	6	14	14	24	7	17	15	3	15	14	3	11	2	21	10	7	3	6
=	**Z**	**F**	**N**	**N**	**X**	**G**	**Q**	**O**	**C**	**O**	**N**	**C**	**K**	**B**	**U**	**J**	**G**	**C**	**F**

Thus, the message, when encrypted becomes: ZFNNXGQOCONCKBUJGCF, which is entirely incoherent, except to someone who is aware that the code is RABBIT, and so can easily reverse the encryption by reversing the change. Though generally simple, the cipher's security relies upon the key remaining private. Additionally, the cipher is still vulnerable to the method of Frequency Analysis, by investigating the letters at different intervals, such that the graph of expected frequency distribution will be eventually found, thus allowing us to know the length of the code word. From here, the task is as simple as breaking a number of shift ciphers, a painful, but doable task for a person, while a trivial task for a computer. Despite the apparent and clear flaw in the system which was generally well known, the use of polyalphabetic ciphers continued for a long period, only truly being superseded at the end of the first world war, by more sophisticated systems such as Enigma. In this period, ciphers such the Viginere's cipher

(used by the French in WW1), were made more complex by utilising a number of polyalphabetic ciphers, each with different code words, thus drastically increasing the number of calculations that an intercepting enemies would need to do, to break the message.

One-Time Pad

According to many, Claude Shannon is the father of modern cryptography whose concepts are still used today, when designing new ciphers. Shannon's work addressed the 'problems of cryptography'[91]. He largely separated the types of cryptography into two categories, one where the cipher was designed to protect against hackers who have infinite resources, now known as unconditional secrecy and a second, where the cipher protects against hackers with finite amount of resources. He also began to define the idea of 'Perfect Secrecy' the cipher text conveys 'no information about the content of the plaintext'[92]. The manner in which this can occur is using the One-Time Pad, where frequency analysis and other cryptanalytic techniques would have no effect upon the encrypted text. The One-Time Pad is similar to the Polyalphabetic and Substitution Cipher except that the code (key) is entirely random and as long as the text to encoded, such that there is no repetition of the entire code, which leaves the polyalphabetic code vulnerable. Shannon proved that perfect secrecy was only possible if this was true.

An example of the use of a One-Time Pad:

Text to be encrypted: EVEYOUCANTHEARMENOW; Code to be used: 23 19 14 4 26 16 8 8 10 13 10 26 14 20 2 13 15 4 15

Figure 11 Table showing an example use of the One Time Pad for encryption

	E	*V*	*E*	*Y*	*O*	*U*	*C*	*A*	*N*	*T*	*H*	*E*	*A*	*R*	*M*	*E*	*N*	*O*	*W*
	23	*19*	*14*	*4*	*26*	*16*	*8*	*8*	*10*	*13*	*10*	*26*	*14*	*20*	*2*	*13*	*15*	*4*	*15*
+	*5*	*22*	*5*	*25*	*15*	*21*	*3*	*1*	*14*	*20*	*8*	*5*	*1*	*18*	*13*	*5*	*14*	*15*	*23*
=	*28*	*41*	*19*	*29*	*41*	*37*	*11*	*9*	*24*	*33*	*18*	*31*	*15*	*38*	*15*	*18*	*29*	*19*	*38*
=	*2*	*15*	*19*	*3*	*15*	*11*	*11*	*9*	*24*	*7*	*18*	*5*	*15*	*12*	*15*	*18*	*3*	*19*	*12*
=	*B*	*O*	*S*	*C*	*O*	*K*	*K*	*I*	*X*	*G*	*R*	*E*	*O*	*L*	*O*	*R*	*C*	*S*	*23*

Though the One-Time Pad is by far the most sophisticated type of cipher that we have encountered so far, there are a few issues that it introduces. Firstly, the length of the code must be at least as long if not longer than the message that the people wish to encrypt, thus taking up a lot of length. This makes it very difficult to remember and use effectively. Additionally, the question becomes how to ensure that both parties have the same code, to decrypt and encrypt the message, since much of the communication is not in person and rely upon inherently insecure systems such as the internet. The mechanisms of fixing these issues were fundamentally fixed in RSA and the Diffie-Helman Key Exchange System.

Diffie-Helman Key Exchange – Asymmetric Key Exchange

The next big breakthrough came in 1976, produced by Whitfield Diffie and Martin Helman, who they designed a manner in which keys could easily be exchanged, solving one of the theoretical problems of the One-Time Pad. For the first time, the two parties (Alice and Bob) never needed to come into contact for the message to be secure. It established a method of key exchange called Asymmetric Key Exchange, where both parties have both a 'Private Key' and a 'Public Key'. Ensuring that Eve does not have any access to the key uses a situation known as the Discrete Logarithm Problem.

[91] (University of California San Diego, 2008)
[92] (University of California San Diego, 2008)

How it works:

First, we must establish the concept of Modular Arithmetic, where we alter the base of a number. For example, 3 = 1 (mod 2). This is the same as calculating the division of 3/2 and then finding the remainder of this. This is an integral part of the method, since the modular function is very easy to calculate, but very hard to reverse.

So, to start off, both Alice and Bob decide (publically) on two prime numbers, where p is a prime, and q is a generator of p. A generator is a number that when raised to whole number (integer) powers less than p, it never produces the same result mod p (every remainder is equally likely). These numbers can be distributed over the internet, thus, Eve is now aware of these numbers. From here, the pair each create their own personal key (a and b) and find another number using the following formula:

$$a' = q^a \, mod(p)$$

$$b' = q^b \, mod(p)$$

From here, they transfer a' and b' over the insecure network to each other (and thereby Eve), thus allowing them to communicate using the following key for encryption:

$$Key\ (Bob) = a'^b \, mod(p)$$

$$Key\ (Alice) = \ b'^a \, mod(p)$$

As it turns out, these two are identical, since, by substituting how these keys were produced:

$$Key\ (Bob) = q^{a^b} mod(p) = q^{ab} \, mod(p)$$

$$Key\ (Alice) = q^{b^a} mod(p) = q^{ab} \, mod(p)$$

However, despite Alice knowing a', b', p and q, finding a and b, as would be required to find the key is a challenge which requires a large amount of computing power, with computers normally taking at least a year to solve this. This is due to a problem known as the Discrete Logarithm Problem. From here, the key can be used as the key to encrypt and decrypt messages (using ciphers such as Shift Ciphers or more likely AES) as required without Eve being able to decrypt the message, since she does not know the key.

Discrete Logarithm Problem

Let g be a generator of two integers; x and p (where p is a prime);

$$Answer = \ g^x \, mod(p)$$

For example:

$$3^{29} \, mod(17) = 12$$

However, knowing this, it is hard to find 29 given all the rest of the information, in fact relying upon trial and error to solve the reverse function:

$$Answer \, mod(17) = \ log_3(12)$$

This method is clearly very useful when the transfer of the information is required, especially over the internet. However, it does not have any impact on us for the product, since it largely deals with creating a secure key which the encryption can use as opposed to discussing the algorithm with which this key would be used.

RSA

From there, the next largest development was the RSA system, so named as it was created by Ron Rivest, Adi Shamir and Leonard Adleman in 1977. Though the algorithm could be used for encryption, the system is a relatively slow form of encryption and so is generally only used to verify identities of users and suchlike, for example for passwords sent over the internet. The system relies upon two more mathematical functions:

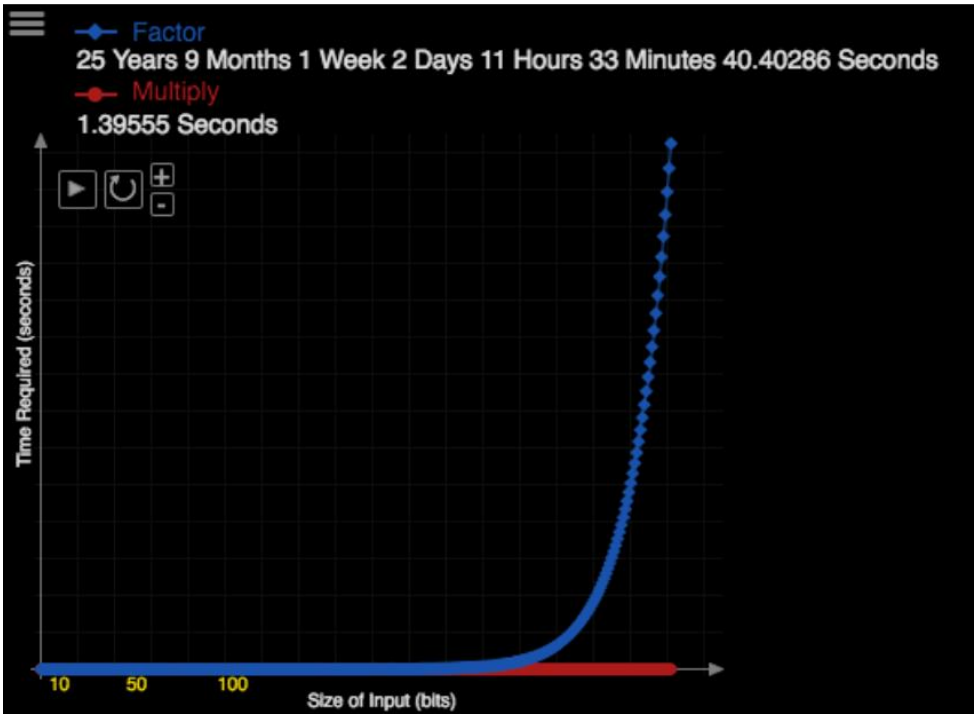
Prime Factorisation

In his 'Elements'[107], Euclid stated the following: 'if q (any integer) is not prime, then some prime factor p divides q'. This allows us to create a system through which every number can be written as a multiplication of prime factors; eg

$$24 = 2 \times 2 \times 2 \times 3$$

Though this is simple to do for low numbers, this is a fundamentally hard problem. Though we have all factorised low numbers in our childhood, this becomes significantly harder as the number increase in length. In fact, in order to factorise a 232-digit number, researchers at the Swiss Federal Institute of Technology took over two years[108], using multiple computer machines and the most efficient algorithms that we know. Ultimately, according to Dirichlet's Theorem, this is due to the random distribution of primes, which prevents prime factorization occurring over a simple iterative cycle time, (i.e. polynomial p^n time). Though there are some mathematicians today who suggest that P=NP (all problems that would otherwise run in non-polynomial time could be run in polynomial time using quantum computing) it is yet to be solved, despite carrying with it a million-dollar prize, as one of the Millennium Prize Problems. As a representation of this, figure 12 compares the time taken for multiplication vs the time taken for prime factorisation as the numbers increase. RSA relies upon this, to ensure that the function is very hard to reverse.

Figure 12: Comparison of time taken to multiply versus factorise numbers as they increase in size.



Source: Khan Academy[110]

[107] (Clarke University, 2015)
[108] (Scientific American, 2015)
[110] (Khan Academy, 2014)

Phi (Φ) Function (Euler's Totient Function):

This is a function which counts the number of positive integers below the number which do not share any common factors with it, thereby are relatively prime to n.

$$\phi(n)\ is\ the\ number\ of\ integers\ i\ in\ the\ range\ 1 \leq i\ \leq n\ where\ GCD(i,k) = 1$$

$$eg.\ \Phi(8)\text{: } 1,\cancel{2},3,\cancel{4},5,\cancel{6},7,\cancel{8} \Rightarrow\ \Phi(8) = 4$$

An important fact about this is that the Phi Function is multiplicative, therefore if Phi(n) = a, and Phi(m) = b, Phi(mn) = Phi(m) x Phi(n) = a x b = ab, presuming that A and B are coprime (thereby A and B do not share any factors bar 1).

Another important fact to note, which is relatively trivial is that Phi(p) where p is prime = p – 1, since by its very definition, p does not share any common factors with any number less than it except one.

How it works

First, Alice (the person receiving data at this point) creates a private key, which is not distributed over the internet, instead remaining private such that only she knows what it is. She makes use of the phi function; using the fact that Phi(pq) where p and q are primes => $\phi(pq)= (p-1)(q-1)$. However, finding Phi(n) where n=pq is very hard (especially when the numbers are high) since one does not know the prime factorisation of n, and doing this would be a fundamentally hard problem.

Alice chooses two large primes (p and q) and calculates n = pq. Then she chooses a small number e (often 65537 unless n is a multiple of this), where e and n are coprime. Then she sends n and e to Bob, while also calculating for herself, the private key (d):

$$e\ \times\ d = 1\ mod\ \phi(n)$$

{ d can be found easily using the Extended Euclidean Algorithm }

In order to calculate the message to send back, Bob does the following, where m is his message:

$$Ciphertext\ (c) = m^e\ mod\ n$$

Alice can find out the message that was sent by doing the following:

$$c^d\ mod\ n = m$$

Thus, Alice is able to get the message Bob has sent. Meanwhile, Eve, who has n, c and e cannot get m.

Rijndael Cipher (Advanced Encryption Standard - AES)

The Rijndael Cipher was chosen (in 2002) as the new encryption standard by the National Institute of Science and Technology (NIST) of America to replace the DES (Data Encryption Standard) as a cipher which could be widely used for both personal and commercial uses ensuring it is entirely unhackable. The cipher attempts to use a key (16 bytes long) and can encrypt a 16-byte message using an iterative process which is still relatively easy to reverse, given the key is known. Essentially, it makes it harder for a hacker to find the message given the cipher text, when they do not know the key. AES is a block cipher, meaning that the number of bytes that it encrypts is fixed. For the purposes of this, I will look at the 16-byte block length, though theoretically the block length could be 32 or 64 bytes, though the encryption process varies between the block length and so is not necessary to mention.

Prerequisites

XOR

XOR is a function which operates on the individual bits in a byte in the same way a 2 pin XOR-Logic Gate would, thereby having the following table of inputs -> outputs:

Figure 13 XOR table

```
0 XOR 0 = 0
1 XOR 0 = 1
1 XOR 1 = 0
0 XOR 1 = 1
```

Source: AES Simplified[112]

HEX

HEX is an alternate definition of numbers in base 16, thus a single digit could have a value of upto 15 (represented by F). This allows us to refer to a single byte as two hex characters as opposed to 8 bits, thus saving lots of time and paper.

Figure 14 Hexadecimal to Decimal conversion table

First Hex Digit → [e.g. $64_{16}=100_{10}$] $HEX_{16} \leftrightarrow DECIMAL_{10}$

© 2001 Roderic A Davis, 2nd	0x	1x	2x	3x	4x	5x	6x	7x	8x	9x	Ax	Bx	Cx	Dx	Ex	Fx
x0	0	16	32	48	64	80	96	112	128	144	160	176	192	208	224	240
x1	1	17	33	49	65	81	97	113	129	145	161	177	193	209	225	241
x2	2	18	34	50	66	82	98	114	130	146	162	178	194	210	226	242
x3	3	19	35	51	67	83	99	115	131	147	163	179	195	211	227	243
x4	4	20	36	52	68	84	100	116	132	148	164	180	196	212	228	244
x5	5	21	37	53	69	85	101	117	133	149	165	181	197	213	229	245
x6	6	22	38	54	70	86	102	118	134	150	166	182	198	214	230	246
x7	7	23	39	55	71	87	103	119	135	151	167	183	199	215	231	247
x8	8	24	40	56	72	88	104	120	136	152	168	184	200	216	232	248
x9	9	25	41	57	73	89	105	121	137	153	169	185	201	217	233	249
xA	10	26	42	58	74	90	106	122	138	154	170	186	202	218	234	250
xB	11	27	43	59	75	91	107	123	139	155	171	187	203	219	235	251
xC	12	28	44	60	76	92	108	124	140	156	172	188	204	220	236	252
xD	13	29	45	61	77	93	109	125	141	157	173	189	205	221	237	253
xE	14	30	46	62	78	94	110	126	142	158	174	190	206	222	238	254
xF	15	31	47	63	79	95	111	127	143	159	175	191	207	223	239	255

Second Hex Digit ↓

Source: Anonymous – Genealogy Website[113]

General Steps

The first section of the process is to expand the key; this entire process moves from a 16-byte key to a 176-byte key which can be used in the main encryption system. This is made by completing the following operations:

Rot Word; Sub Word; Rcon; EK; K

[112] (Berent, n.d.)
[113] (Anon., 2012)

These are completed in the following order (with each one acting like a function, taking a certain input and giving a certain output). The inner functions are therefore completed first when there are multiple in one round, followed by the outer one progressively.

Figure 15 Method of expanding key

Round	Expanded Key Bytes				Function
0	0	1	2	3	**K**(0)
1	4	5	6	7	**K**(4)
2	8	9	10	11	**K**(8)
3	12	13	14	15	**K**(12)
4	16	17	18	19	**Sub Word**(**Rot Word**(**EK**((4-1)*4))) XOR **Rcon**((4/4)-1) XOR **EK**((4-4)*4)
5	20	21	22	23	**EK**((5-1)*4)XOR **EK**((5-4)*4)
6	24	25	26	27	**EK**((6-1)*4)XOR **EK**((6-4)*4)
7	28	29	30	31	**EK**((7-1)*4)XOR **EK**((7-4)*4)
8	32	33	34	35	**Sub Word**(**Rot Word**(**EK**((8-4)*4))) XOR **Rcon**((8/4)-1) XOR **EK**((8-4)*4)
9	36	37	38	39	**EK**((8-1)*4)XOR **EK**((9-4)*4)
10	40	41	42	43	**EK**((10-1)*4)XOR **EK**((10-4)*4)
11	44	45	46	47	**EK**((11-1)*4)XOR **EK**((11-4)*4)
12	48	49	50	51	**Sub Word**(**Rot Word**(**EK**((12-4)*4))) XOR **Rcon**((12/4)-1) XOR **EK**((12-4)*4)
13	52	53	54	55	**EK**((13-1)*4)XOR **EK**((13-4)*4)
14	56	57	58	59	**EK**((14-1)*4)XOR **EK**((14-4)*4)
15	60	61	62	63	**EK**((15-1)*4)XOR **EK**((15-4)*4)
16	64	65	66	67	**Sub Word**(**Rot Word**(**EK**((16-4)*4))) XOR **Rcon**((16/4)-1) XOR **EK**((16-4)*4)
17	68	69	70	71	**EK**((17-1)*4)XOR **EK**((17-4)*4)
18	72	73	74	75	**EK**((18-1)*4)XOR **EK**((18-4)*4)
19	76	77	78	79	**EK**((19-1)*4)XOR **EK**((19-4)*4)
20	80	81	82	83	**Sub Word**(**Rot Word**(**EK**((20-4)*4))) XOR **Rcon**((20/4)-1) XOR **EK**((20-4)*4)
21	84	85	86	87	**EK**((21-1)*4)XOR **EK**((21-4)*4)
22	88	89	90	91	**EK**((22-1)*4)XOR **EK**((22-4)*4)
23	92	93	94	95	**EK**((23-1)*4)XOR **EK**((23-4)*4)
24	96	97	98	99	**Sub Word**(**Rot Word**(**EK**((24-4)*4))) XOR **Rcon**((24/4)-1) XOR **EK**((24-4)*4)
25	100	101	102	103	**EK**((25-1)*4)XOR **EK**((25-4)*4)
26	104	105	106	107	**EK**((26-1)*4)XOR **EK**((26-4)*4)
27	108	109	110	111	**EK**((27-1)*4)XOR **EK**((27-4)*4)
28	112	113	114	115	**Sub Word**(**Rot Word**(**EK**((28-4)*4))) XOR **Rcon**((28/4)-1) XOR **EK**((28-4)*4)
29	116	117	118	119	**EK**((29-1)*4)XOR **EK**((29-4)*4)
30	120	121	122	123	**EK**((30-1)*4)XOR **EK**((30-4)*4)
31	124	125	126	127	**EK**((31-1)*4)XOR **EK**((31-4)*4)
32	128	129	130	131	**Sub Word**(**Rot Word**(**EK**((32-4)*4))) XOR **Rcon**((32/4)-1) XOR **EK**((32-4)*4)
33	132	133	134	135	**EK**((33-1)*4)XOR **EK**((33-4)*4)
34	136	137	138	139	**EK**((34-1)*4)XOR **EK**((34-4)*4)
35	140	141	142	143	**EK**((35-1)*4)XOR **EK**((35-4)*4)
36	144	145	146	147	**Sub Word**(**Rot Word**(**EK**((36-4)*4))) XOR **Rcon**((36/4)-1) XOR **EK**((36-4)*4)
37	148	149	150	151	**EK**((37-1)*4)XOR **EK**((37-4)*4)
38	152	153	154	155	**EK**((38-1)*4)XOR **EK**((38-4)*4)
39	156	157	158	159	**EK**((39-1)*4)XOR **EK**((39-4)*4)
40	160	161	162	163	**Sub Word**(**Rot Word**(EK((40-4)*4))) XOR **Rcon**((40/4)-1) XOR **EK**((40-4)*4)
41	164	165	166	167	**EK**((41-1)*4)XOR **EK**((41-4)*4)
42	168	169	170	171	**EK**((42-1)*4)XOR **EK**((42-4)*4)
43	172	173	174	175	**EK**((43-1)*4)XOR **EK**((43-4)*4)

Source: AES Simplified[115]

[115] (Berent, n.d.)

The other section is the main encryption, which is also composed of a number of functions, acting in a specific order. The list and order of use of functions are as follows:

Add Round Key; Byte Sub; Shift Row; Mix Column

Figure 16 Encryption Routine for AES 16 Byte Encryption

Round	Function
–	Add Round Key(State)
0	Mix Column(Add Round Key(Byte Sub(Shift Row(State))))
1	Mix Column(Add Round Key(Byte Sub(Shift Row(State))))
2	Mix Column(Add Round Key(Byte Sub(Shift Row(State))))
3	Mix Column(Add Round Key(Byte Sub(Shift Row(State))))
4	Mix Column(Add Round Key(Byte Sub(Shift Row(State))))
5	Mix Column(Add Round Key(Byte Sub(Shift Row(State))))
6	Mix Column(Add Round Key(Byte Sub(Shift Row(State))))
7	Mix Column(Add Round Key(Byte Sub(Shift Row(State))))
8	Mix Column(Add Round Key(Byte Sub(Shift Row(State))))
9	Add Round Key(Byte Sub(Shift Row(State)))

Source: AES Simplified[117]

At the end of this, the encryption produces a 16-byte encrypted message, which is theoretically unhackable, unless you have the key, in which case it is very easy to reverse, simply by completing exactly the same steps in reverse.

Definitions of each step

Key Expansion

Rot Word

This function takes an input of four bytes, circularly shifting them around, for example, moving:

$(a1\ a2\ a3\ a4)\ to\ (a2\ a3\ a4\ a1)$

The offset (1, 2, 3 or 4) depends on the round, using:

$offset = round\ mod(4)$

Sub Word

This function inputs four bytes, and converts them using the below table, dealing with each byte individually, to produce a different byte for each of them and merging it again to form a block of four bytes.

[117] (Berent, n.d.)

Figure 17 Substitution Table for AES 16 Byte Encryption

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

Source: AES Simplified[119]

Correspondingly, when the step is required during decryption, this table is used:

Figure 18 Substitution Table for AES 16 Byte Decryption

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	52	09	6A	D5	30	36	A5	38	BF	40	A3	9E	81	F3	D7	FB
1	7C	E3	39	82	9B	2F	FF	87	34	8E	43	44	C4	DE	E9	CB
2	54	7B	94	32	A6	C2	23	3D	EE	4C	95	0B	42	FA	C3	4E
3	08	2E	A1	66	28	D9	24	B2	76	5B	A2	49	6D	8B	D1	25
4	72	F8	F6	64	86	68	98	16	D4	A4	5C	CC	5D	65	B6	92
5	6C	70	48	50	FD	ED	B9	DA	5E	15	46	57	A7	8D	9D	84
6	90	D8	AB	00	8C	BC	D3	0A	F7	E4	58	05	B8	B3	45	06
7	D0	2C	1E	8F	CA	3F	0F	02	C1	AF	BD	03	01	13	8A	6B
8	3A	91	11	41	4F	67	DC	EA	97	F2	CF	CE	F0	B4	E6	73
9	96	AC	74	22	E7	AD	35	85	E2	F9	37	E8	1C	75	DF	6E
A	47	F1	1A	71	1D	29	C5	89	6F	B7	62	0E	AA	18	BE	1B
B	FC	56	3E	4B	C6	D2	79	20	9A	DB	C0	FE	78	CD	5A	F4
C	1F	DD	A8	33	88	07	C7	31	B1	12	10	59	27	80	EC	5F
D	60	51	7F	A9	19	B5	4A	0D	2D	E5	7A	9F	93	C9	9C	EF
E	A0	E0	3B	4D	AE	2A	F5	B0	C8	EB	BB	3C	83	53	99	61
F	17	2B	04	7E	BA	77	D6	26	E1	69	14	63	55	21	0C	7D

Source: AES Simplified[121]

Rcon

This takes an input of which round it is and the keysize (always 16 in the case of a 16-byte key) finds: (Round/(KeySize/4)-1 and returns the following value according to the table:

[119] (Berent, n.d.)
[121] (Berent, n.d.)

Figure 19 RCon conversion table for AES 16 Byte Encryption

```
Rcon(0)     = 01000000
Rcon(1)     = 02000000
Rcon(2)     = 04000000
Rcon(3)     = 08000000
Rcon(4)     = 10000000
Rcon(5)     = 20000000
Rcon(6)     = 40000000
Rcon(7)     = 80000000
Rcon(8)     = 1B000000
Rcon(9)     = 36000000
Rcon(10)    = 6C000000
Rcon(11)    = D8000000
Rcon(12)    = AB000000
Rcon(13)    = 4D000000
Rcon(14)    = 9A000000
```

Source: AES Simplified[123]

EK

The EK function will return 4 bytes of the Expanded Key after the specified offset (which is the input). For example, if the offset is 4, the EK function will return bytes 4, 5, 6 and 7 of the Expanded Key.

K

The K function will return 4 bytes of the original key after the specified offset (in the same way as EK but for the original key).

Encryption

Add Round Key

The function XORs each byte of the state (the ciphertext as it stands) with 16 bytes of the key, such that every bit of the state is compared against a bit of the key. Additionally, for the 11 times it is run, the part of the key (given it is now 176 bytes long) that the state is compared against is changed, such that the first time, it is compared against bytes 0-15, the second 16-31 etc.

Byte Sub

Byte Sub replaces the values of each byte using the values in the table below, such that the value of every byte is changed. For example, a byte with value HEX 19 would be replaced with a byte of value D4.

[123] (Berent, n.d.)

Figure 20 SBOX Substitution Table for 16 Byte AES Encryption

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

Source: AES Simplified[125]

To reverse this process, the value is replaced with the corresponding inverses of the SBOX (the specific name for the table) thus:

Figure 21 SBOX Substitution for AES Decryption

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	52	09	6A	D5	30	36	A5	38	BF	40	A3	9E	81	F3	D7	FB
1	7C	E3	39	82	9B	2F	FF	87	34	8E	43	44	C4	DE	E9	CB
2	54	7B	94	32	A6	C2	23	3D	EE	4C	95	0B	42	FA	C3	4E
3	08	2E	A1	66	28	D9	24	B2	76	5B	A2	49	6D	8B	D1	25
4	72	F8	F6	64	86	68	98	16	D4	A4	5C	CC	5D	65	B6	92
5	6C	70	48	50	FD	ED	B9	DA	5E	15	46	57	A7	8D	9D	84
6	90	D8	AB	00	8C	BC	D3	0A	F7	E4	58	05	B8	B3	45	06
7	D0	2C	1E	8F	CA	3F	0F	02	C1	AF	BD	03	01	13	8A	6B
8	3A	91	11	41	4F	67	DC	EA	97	F2	CF	CE	F0	B4	E6	73
9	96	AC	74	22	E7	AD	35	85	E2	F9	37	E8	1C	75	DF	6E
A	47	F1	1A	71	1D	29	C5	89	6F	B7	62	0E	AA	18	BE	1B
B	FC	56	3E	4B	C6	D2	79	20	9A	DB	C0	FE	78	CD	5A	F4
C	1F	DD	A8	33	88	07	C7	31	B1	12	10	59	27	80	EC	5F
D	60	51	7F	A9	19	B5	4A	0D	2D	E5	7A	9F	93	C9	9C	EF
E	A0	E0	3B	4D	AE	2A	F5	B0	C8	EB	BB	3C	83	53	99	61
F	17	2B	04	7E	BA	77	D6	26	E1	69	14	63	55	21	0C	7D

Source: AES Simplified[127]

Shift Row

First, the state is converted into a 4x4 matrix, with one byte per element of the matrix, thus, with the elements formed vertically:

[125] (Berent, n.d.)
[127] (Berent, n.d.)

$$\begin{matrix} 0 & 4 & 8 & 12 \\ 1 & 5 & 9 & 13 \\ 2 & 6 & 10 & 14 \\ 3 & 7 & 11 & 15 \end{matrix}$$

Then the rows are circularly shifted by a certain number of elements, according to which row it. The top row is shifted 0 elements along, the second row shifted 1 element, third 2 elements and bottom one 3 elements along, thus forming this:

$$\begin{matrix} 0 & 4 & 8 & 12 \\ 5 & 9 & 13 & 1 \\ 10 & 14 & 2 & 6 \\ 15 & 3 & 7 & 11 \end{matrix}$$

During decryption, the same process is reversed and all the rows are shifted by the same amounts to the left.

Mix Column

This step involves multiplication of the state and the multiplication matrix thusly:

Multiplication Matrix

$$\begin{matrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{matrix}$$

State

$$\begin{matrix} b1 & b5 & b9 & b13 \\ b2 & b6 & b10 & b14 \\ b3 & b7 & b11 & b15 \\ b4 & b8 & b12 & b16 \end{matrix}$$

The process is completed in the usual manner for matrix cross-multiplication, except that instead of addition of terms, the function XOR is completed instead. For example:

b'1 = (b1 x 2) XOR (b2 x 3) XOR (b3 x 1) XOR (b4 x 1)

Unfortunately, however, another complexity is added by the fact that the multiplication is completed using a Galois Field.

Galois Field Multiplication in Practise

In practise, this involves the manner in a x b (where both a and b are bytes) is completed. Firstly, one must look up the values for a' and b' in the L table. Then these values are arithmetically added together and then the result is looked for in the E Table.

Figure 22 L Table for AES 16 Byte Encryption

L Table

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0		00	19	01	32	02	1A	C6	4B	C7	1B	68	33	EE	DF	03
1	64	04	E0	0E	34	8D	81	EF	4C	71	08	C8	F8	69	1C	C1
2	7D	C2	1D	B5	F9	B9	27	6A	4D	E4	A6	72	9A	C9	09	78
3	65	2F	8A	05	21	0F	E1	24	12	F0	82	45	35	93	DA	8E
4	96	8F	DB	BD	36	D0	CE	94	13	5C	D2	F1	40	46	83	38
5	66	DD	FD	30	BF	06	8B	62	B3	25	E2	98	22	88	91	10
6	7E	6E	48	C3	A3	B6	1E	42	3A	6B	28	54	FA	85	3D	BA
7	2B	79	0A	15	9B	9F	5E	CA	4E	D4	AC	E5	F3	73	A7	57
8	AF	58	A8	50	F4	EA	D6	74	4F	AE	E9	D5	E7	E6	AD	E8
9	2C	D7	75	7A	EB	16	0B	F5	59	CB	5F	B0	9C	A9	51	A0
A	7F	0C	F6	6F	17	C4	49	EC	D8	43	1F	2D	A4	76	7B	B7
B	CC	BB	3E	5A	FB	60	B1	86	3B	52	A1	6C	AA	55	29	9D
C	97	B2	87	90	61	BE	DC	FC	BC	95	CF	CD	37	3F	5B	D1
D	53	39	84	3C	41	A2	6D	47	14	2A	9E	5D	56	F2	D3	AB
E	44	11	92	D9	23	20	2E	89	B4	7C	B8	26	77	99	E3	A5
F	67	4A	ED	DE	C5	31	FE	18	0D	63	8C	80	C0	F7	70	07

Source: AES Simplified[129]

Figure 23 E Table for 16 Byte AES Encryption

E Table

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	01	03	05	0F	11	33	55	FF	1A	2E	72	96	A1	F8	13	35
1	5F	E1	38	48	D8	73	95	A4	F7	02	06	0A	1E	22	66	AA
2	E5	34	5C	E4	37	59	EB	26	6A	BE	D9	70	90	AB	E6	31
3	53	F5	04	0C	14	3C	44	CC	4F	D1	68	B8	D3	6E	B2	CD
4	4C	D4	67	A9	E0	3B	4D	D7	62	A6	F1	08	18	28	78	88
5	83	9E	B9	D0	6B	BD	DC	7F	81	98	B3	CE	49	DB	76	9A
6	B5	C4	57	F9	10	30	50	F0	0B	1D	27	69	BB	D6	61	A3
7	FE	19	2B	7D	87	92	AD	EC	2F	71	93	AE	E9	20	60	A0
8	FB	16	3A	4E	D2	6D	B7	C2	5D	E7	32	56	FA	15	3F	41
9	C3	5E	E2	3D	47	C9	40	C0	5B	ED	2C	74	9C	BF	DA	75
A	9F	BA	D5	64	AC	EF	2A	7E	82	9D	BC	DF	7A	8E	89	80
B	9B	B6	C1	58	E8	23	65	AF	EA	25	6F	B1	C8	43	C5	54
C	FC	1F	21	63	A5	F4	07	09	1B	2D	77	99	B0	CB	46	CA
D	45	CF	4A	DE	79	8B	86	91	A8	E3	3E	42	C6	51	F3	0E
E	12	36	5A	EE	29	7B	8D	8C	8F	8A	85	94	A7	F2	0D	17
F	39	4B	DD	7C	84	97	A2	FD	1C	24	6C	B4	C7	52	F6	01

Source: AES Simplified[131]

[129] (Berent, n.d.)

5.1.3.4 Conclusion

Figure 24 Table showing the summary of all encryption methods

Cipher	Applicability	Effectiveness
Shift Cipher	The shift cipher could be used for encrypting the passwords.	The Shift Cipher is easily cracked, as there are a finite number of possibilities for the original. Additionally, frequency analysis could be used to further quicken the process.
Substitution Cipher	The substitution cipher could theoretically be used for encrypting the passwords – though given passwords can make use of special characters, the system would be far more complex, and therefore may not be very simple.	Substitution ciphers can easily be cracked when using a frequency analysis tactic. Despite the fact that passwords may not contain ordinary sentences, they are likely to contain similar distributions, allowing a hacker to develop a frequency distribution for passwords and therefore use this on the list which is encrypted using a substitution cipher.
Polyalphabetic Cipher	The polyalphabetic cipher could be used for encrypting the passwords.	A polyalphabetic cipher could again, given a certain amount of trial and error, experimenting for the length of the key, be cracked using a frequency analysis.
Diffie-Helman Key Exchange	The Diffie-Helman system, though far more secure and sophisticated than any of the above systems is not applicable for the encryption of passwords, instead dealing with digital handshaking, ensuring two people on an insecure network can gain a secure key which they could use for encrypted conversation. If passwords were to be transferred over a network (though this was earlier ruled out due to a number of side doors which would mean encryption could be made to be useless) Diffie-Helman and RSA could be exceedingly useful to allow for the creation of a secure key for the two devices which could be	The Diffie-Helman Key Exchange is relatively secure, though relies upon the good selection of initial values for the prime and the generator.

[131] (Berent, n.d.)

	used with secure algorithms such as AES or DES (it's predecessor).	
RSA	RSA though useable for encryption of data is a very slow algorithm and therefore is more commonly used for key-exchange, similar to Diffie-Helman.	RSA (due to the problem with prime factorisation) is very hard to reverse.
Advanced Encryption Standard	AES, the successor to algorithms like DES is highly applicable for the use of encrypting passwords.	AES is generally acclaimed to be impossible to hack, requiring huge amounts of computing power over hundreds of years, because it seems impossible to reduce the combinations in AES to more simple changes. This means that anyone attempting to hack a system must go through all the combinations of keys at every step.

In the attempt to find the most secure algorithm, it is clear that the newest algorithm, Advanced Encryption Standard is by far the most secure algorithm, simply because it is by far the most complex, employing a number of functions which would be hard to reversed without the presence of the key. It is because of this that the NSA brands the AES algorithm 'unbreakable', therefore recommending everyone uses it as their algorithm of choice. Hence, the product should make use of the AES algorithm, to securely save the passwords, ensuring that the passwords could not easily be retrieved without the key.

5.2 Control System Research

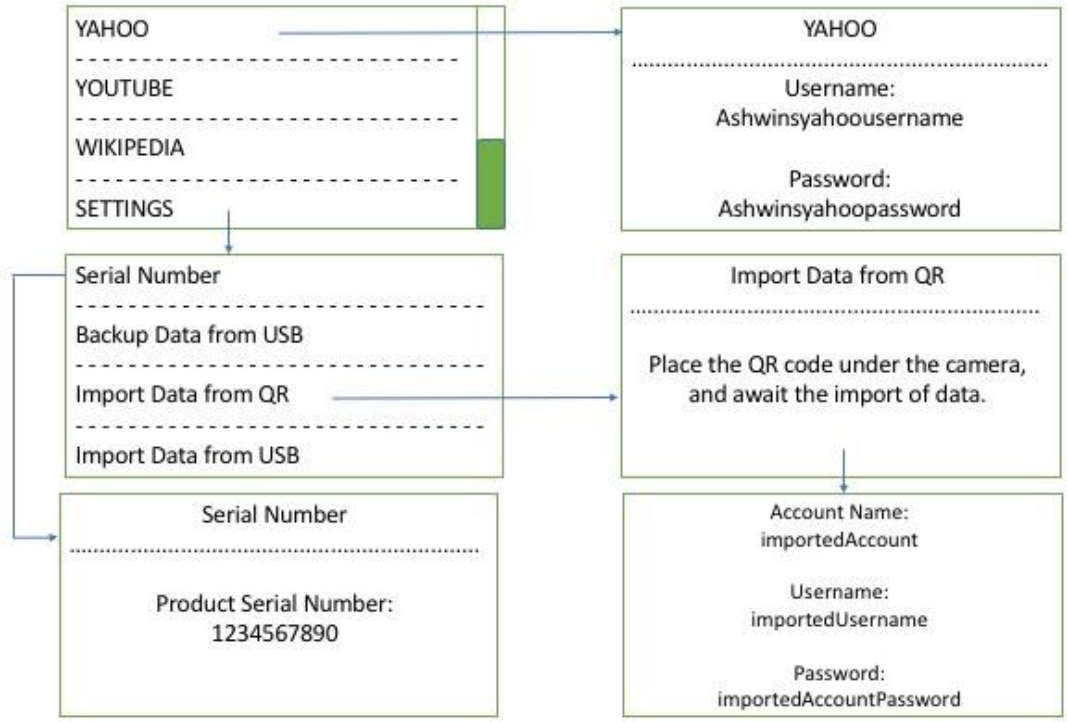
The control system is a large part of the project – since it governs the way in which a person interacts with the product. To meet the product specification a user friendly interface with proper choice of components such as Microcontrollers, sensors, screens and battery systems are necessary.

5.2.1 User interface (end user experience)

In order to define what the user interface would be, it is important to decide what all the product needs to do for the user. Firstly, it must allow the user to enter the product by placing their finger on the fingerprint sensor, allowing the product to scan the finger, then allowing it into the restricted section and finally decrypt the passwords so that they could not be viewed by another person. From here, the user must see a list of account names – since the usernames may not tell people what account the password corresponds to. One of the problems with paper systems that is being improved upon by our devices is that passwords will be ordered, allowing the person to scroll through the list and choose the one that they want. From here, they should be able to click on the one they want and then be able to see the exact information about that account – therefore seeing the username and password as required, hence allowing them to access the account they wanted.

From these initial ideas, a few drawings were created which show the interface in action. Additionally, it adds in the idea of the settings bar, showing all the required settings, including the ability to backup information as discussed in the specification and import data from backed up data from other devices.

Figure 26 User Interface plans



In order to use all of this for the product, a number of things are needed:

1. A screen – preferably a graphic display which would allow text to be resized as in the bottom right to fit it into an area.
2. A scrolling method – such as a potentiometer which would allow the user to scroll through the list.
3. A button to enter the selected menu item.

In addition to these, the following would also be required for the product:

1. A fingerprint sensor to allow the person to enter the product
2. A main On/Off toggle switch to allow the person to turn the product on and off to conserve battery.

5.2.2 Adding passwords to the device

There are a number of ways to add passwords securely to the device. One way which had to be removed from consideration early on was any method that made use of online communications, such as Bluetooth and Wi-Fi. Though these technologies in many ways are the easiest to implement, and very popular, they clearly offer another pathway allowing a hacker to gain access to the product. Even if the device's Wi-Fi chip were turned off during periods of no communications, it has been shown (by hackers hired by The Verge)[133] that they could enable the chip and turn it on remotely. Bluetooth, though short-range, could be gained access to using a number of other nearby Bluetooth (and Wi-Fi and mobile networks) enabled as the intermediary, which could be hacked over the network.

KEYBOARD: Many microcontrollers, such as the Raspberry Pi, have an inbuilt support for a keyboard, meaning a keyboard can easily be connected directly to the product to enter passwords. This keyboard could come in a number of different forms, since it could be custom-designed for the job. Firstly, there is the question of whether the keyboard should be removable or continuously connected to the product. Additionally, there would be a number of possibilities of the design, from making use of an existing large USB keyboard, with a full chicklet-style ergonomic keyboard, to making use of fewer keys, such as a numeric keyboard with a number of characters available from each key.

Advantages: Simple and easy to use, relatively simple to make, could make use of existing keyboards.

Disadvantages: Relatively large, taking up quite a lot of space.

CONNECTION TO COMPUTER: Another method that could be used would be connecting the device directly to a computer with a serial connection, which would allow information to be seamlessly sent from the computer to the device. From here, the passwords could easily be sent. The system would use a custom designed application for the computer which would allow for the required communications. Additionally, there might be the possibility of the communications through Android and iOS, through the production of mobile applications and specific wires to connect to each device.

Advantages: Requires almost no electronics on the side of the product – hence does not make it bulkier, very simple for the user

Disadvantages: Quite complex, require the production of a desktop (and / or mobile applications), could be insecure – the safety is only as strong as the security of the computer since the serial connection would allow a port into the device which could otherwise be disabled.

QR CODE SYSTEM: This system would allow a user to create a QR code on mobile, desktop and web-apps which would have all the information that they want to transfer to the device, though encryption must be considered for this display of information. Then, a camera on the device would scan this QR code, interpreting it to generate all the information from the code, allowing it to be used.

Advantages: Highly innovative, requires little electronics on the side of the product – hence does not add much bulk to the product, quite secure (if QR creation is completed using a cipher)

[133] (TheVerge, 2015)

Disadvantages: Very complex, requires the use of at least a web-app, with the possibilities of mobile and desktop applications.

To make the decision, I made use of a decision matrix, which summarised the reasons a decision would be made.

Figure 25 Decision matrix for password entering

	Complexity		Ease of use		Security		TOTAL
	WEIGHT	MARK	WEIGHT	MARK	WEIGHT	MARK	
QR CODE	6	10	7	5	10	7	112
KEYBOARD	6	4	7	7	10	8	91
CONNECTION TO COMPUTER	6	7	7	7	10	4	105

From this it is clear that the best system to use would be the QR code system, since it is better on security than the connection to the computer, which was by far the most important criteria (it was most heavily weighted).

5.2.3 Specific Component Selection

5.2.3.1 Microcontroller

The microcontroller to be used is a critical decision, since this in many ways determines the options for many other component decisions. The decision largely boils down to that between two ecosystems, the ecosystem of Linux PCs, with the Intel Edison and Raspberry Pi versus the hobbyist electronics ICs, with the Arduino platform and PICAXE (and associated BASIC). Immediately it is easy to see that if the latter were to be used the Arduino platform would be used since it is very highly supported, with a vast array of microcontrollers which are compatible with it, while it would also be programmed using Arduino C, a language many degrees more advanced and better than BASIC. Additionally, the chips running Arduino tend to be of a similar cost as well as far more powerful, having far more powerful CPUs and more memory! Hence the question becomes Linux PCs vs Arduino.

Figure 37 Micro-controller comparison

	Linux PCs	Arduino
Power	The Linux PCs are far more powerful in some respect as they contain full power PC chips. For example, the Raspberry Pi 3 B contains a 1.2GHz 64-bit quad-core ARM CPU, one which could easily have been used a few years ago as the guts behind a desktop computer. This is in fact very similar to the processor inside many flagship smartphones, which carry out a number of very complex tasks including encryption and reading from fingerprint sensors (two of the most power intensive things my product must do). Additionally, the Pi 3 also contains 1GB of dedicated RAM a far cry from the few KBs available on most Arduino's.	Arduino chips tend to have far less power than the Raspberry Pi. For example, the largest Atmel chip generally used, the Mega 2560 contains a single-core 16MHz, 8-bit CPU with 256kB of flash memory. This means it is likely to struggle with tasks such as the complex maths associated with encryption using the Advanced Encryption Standard, as well as definitely struggling with reading QR codes. In fact, Arduino's due to their very small memory could not work with live video at all, instead at a maximum being able to look at individual images. Furthermore, these images would have to be of a very poor quality, since they must have a size less than 256kB – most 5MP cameras would take images of around 3MB in size.

Ease of Use	Linux PCs tend to be slightly more complex to use and require a lot of setup to ensure that the person can communicate with the GPIO and have access to all required tools. Additionally, they require a lot of additional parts to set them up, including peripherals such as a monitor, keyboard and mouse. However most contain an ability to enable networking, which would allow the person to connect via another computer (via secure shell – SSH) and hence forgo the external items. However, this lack of ease of use is also in many ways determined by the amount of things that can be done with a Linux PC. For example, for a Pi, it is sometimes required to download and flash Raspbian (the most common distribution of Linux used), since the person could in fact use one of many distributions, or make their own. Additionally, once the system has been setup, a Linux PC can be used very similar to an Arduino, by only making use of certain tools on a command line interface.	Arduinos are very simple to use, being plug and play, requiring them to be plugged into a computer and then programmed using the Open Source application which is available for Windows, Mac and Linux computers.
Programming	Programming on a Linux PC could happen on almost anything. In fact, if wanted, a library known as WiringPi could be used and the GPIO could be coded for using Arduino C, the same as Arduino. However, the most common language to use on Linux PCs (and especially Raspberry PIs) is Python, since it is very simple and easy to use, and provides a number of libraries for many components which could be used.	Arduino programming must occur in Arduino C, though the chips themselves could theoretically without the Arduino bootloader be programmed in any language desired, such as C. Arduino C is generally a well supported language, with a large community supporting it. It also has the ability to use the vast array of C++ libraries written for the components, and there is a library for most components that one could buy.
Availability of components	Linux PCs support almost all components, whether they use protocols such as I2C and SPI which would connect through the GPIO or through the USB connections which are available through all of the possible boards such as Intel Edison and Raspberry Pi. However, it is often true that components do not have libraries for the communications between the microcontroller and the sensor	Only a select number of components can work with Arduino, ones which work using the protocols, such as I2C and SPI which the Arduino supports. Notably, there is a lack of cameras, only some low quality SPI ones, from which it struggle to manage to successfully scan a QR code.

	and so I would have to design this myself.	
Cost	Linux PCs tend to be more expensive than Arduino boards or chips.	Arduino board (or chips) tend to be of a much lower cost than Linux PC boards.

In order to make this decision, I used a decision matrix :

Figure 38 Decision matrix for micro controller

	Power		**Ease of Use**		**Programming**		**Availability of Components**		**Cost**		
	Weight	Mark	Weight	Mark	Weight	Mark	Weight	Mark	Weight	Mark	**TOTAL**
Arduino	8	4	7	8	5	6	7	6	6	9	214
Linux PC	8	8	7	6	5	9	7	9	6	5	244

Next, we need to decide which Linux PC to use, from among a number of options including Raspberry Pi, BeagleBone Black and the Intel Edison.

Raspberry Pi

Advantages: Very highly supported, with active online forums. Also it contains a number of ports such as USB ports and CSI (Camera Port) built in. Additionally, the GPIO connections are nicely broken out into header pins (or nice 0.1" holes on the Raspberry Pi Zero)

Disadvantage: Generally, quite large, though the Raspberry Pi Zero, a slightly less powerful board contains enough computing power with a much smaller footprint. It also has a higher power consumption than the BeagleBone and the Edison.

Intel Edison

Advantages: Contains a very powerful Intel Atom Dual-Core processor – a step up from the processors in the BeagleBone Black and the Raspberry Pi. It is also the smallest of them, having a very tiny footprint. Finally, it has a very low power consumption, which would mean the same battery would last longer powering the Intel Edison.

Disadvantages: The ports – connector shown – are very hard to access without making a custom PCB with the necessary connector. Additionally, the Edison contains Wi-Fi and Bluetooth built-in. Though this would normally be an advantage, in this case it offers another backdoor into the product. Even if the Wi-Fi and Bluetooth systems are off it is sometimes possible for a hacker to communicate to the product through their antennas, thereby allowing them to attempt to find a backdoor into the encryption and attempt to find the passwords. Finally, since it is so new, the support that is available online is slightly lower than support available for the Pi platform.

Figure 39 Back of the Intel Edison

Source: Arduino.cc[134]

BeagleBone Black

Advantages: Even more ports available than the Raspberry Pi, therefore would not suffer problems like the lack of Serial Ports that I would have with the Raspberry Pi. It is also very simple to set-up being almost plug-and-play.

Disadvantages: The BeagleBone Black is far larger than the Raspberry Pi Zero (smallest of the Pis) and Intel Edison, meaning the product would have to be larger and indeed heavier.

Due to the security flaw that the Intel Edison would create, the size problems of the BeagleBone Black, as well as the online support offered by the Raspberry Pi and my own prior experience in using them, I chose to use the Raspberry Pi. However, I now had to choose between the larger and more powerful Raspberry Pi 2 (the newest larger brother at the point of choosing) and the Raspberry Pi Zero, which was cheaper and much smaller, but far less powerful.

Figure 40 – Comparison between Raspberry Pi 2 and Raspberry Pi Zero

	Pi 2	Pi Zero
Price:	£30	£5
Chip:	Broadcom BCM2836	Broadcom BCM2835
Processor:	ARMv7 Quad Core	ARM11
Processor Speed:	900MHz	1GHz
Voltage and Power Draw:	700mA @ 5V	300mA @ 5V
GPU:	Dual Core VideoCore IV Multimedia	Dual Core VideoCore IV Multimedia
Size:	85x65mm	65x30mm
Memory:	1GB SDRAM @ 400MHz	512MB SDRAM
Storage	Micro SD	Micro SD
GPIO:	40	40
USBs:	4 + 1 Micro-USB for power	2 Micro-USB

[134] (Arduino CC, n.d.)

From the above it is clear that the Pi 2 is significantly more powerful than the Pi Zero, however, there is the question of purpose. It is clear that in fact, the power is largely unneeded. The only power intensive tasks are communicating with a camera as well as the complex maths associated with using the Advanced Encryption Standard. It is clear that while an Arduino would have struggled to do these tasks, either of the Pis would find both of these tasks relatively easy. Hence, given the significant size savings that the board offers, I will use the Raspberry Pi Zero. Additionally, while the Pi 2 would use a large amount of the budget I specified in my original specification (£50.33), the Pi Zero, with it being very well priced at £5 would make the cost significantly lower.

5.2.3.2 Input / Output systems for users

The initial research into the user interface suggested three main input output systems for the user, a scrolling system, a button system and a power toggle system. For each of these systems, there are a number of possibilities which could be used.

Gaming Joystick

Figure 27 Gaming Joystick

Source: Sparkfun Electronics[136]

A gaming joystick would allow for double axis movement (up and down as well as side to side) and also contain a button which would allow for the system to register a press. Additionally, with this system it would be relatively easy to read any changes, since it simply uses two potentiometers, one for each axis and a normally-open push switch for the joystick push. While there was no original plan to use the third axis of movement it could be used for the menu system allowing the user to move through the levels of screens, getting more information about the account by sliding right and sliding left to return to the main passwords, doing the same when going through layers of the settings. Also, this saves the main joystick push to be used for something else, such as a quick access to the settings.

Advantages: The joystick allows for easy movements and requires very little effort for the elderly. The joystick was designed with ergonomics in mind and hence should fit the thumb of a person very easily and comfortably. The electronics are relatively simple and would enable easy connectivity to the microcontroller. It also contains a switch which would mean there would not be a need for another push button switch.

Disadvantages: The joystick is rather large and would make the product larger than necessary. Also, pushing the button of a gaming joystick's such as this is fairly challenging and requires quite a lot of pressure.

Rotary Potentiometer

[136] (Sparkfun, n.d.)

Figure 28 Rotary Potentiometer

Source: Sparkfun Electronics[137]

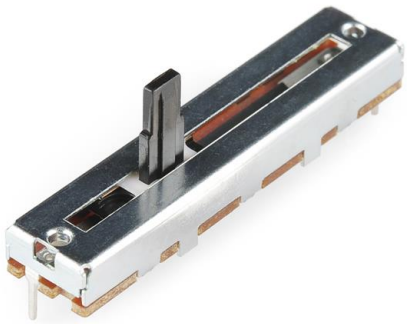
A rotary potentiometer was my first thought as it would allow for scrolling and is simple electronically, simply producing an analog value for the position which could be read by the microcontroller. In fact they also come with an integrated push switch which could be used instead of an external push button switch as originally envisioned.

Advantages: Very simple electronically, it is also very intuitive and hence would make it very easy for the elderly to use.

Disadvantages: The potentiometer has a set degree of motion, normally around 270 degrees and hence the potentiometer can only rotate through this degree as a maximum. Therefore, if there were lots and lots of passwords, the amount of change required to turn through the passwords might be very small and hence easy to skip through the password that the user wanted.

Sliding Potentiometer

Figure 29 Sliding Potentiometer



Source: Sparkfun Electronics[138]

A sliding potentiometer uses the same electronics as the rotary potentiometer (basically a variable resistor) meaning that it is still very easy to use. However, it's usage is even more intuitive, with the system including a sliding tab which could be used to move up and down the list of the passwords.

[137] (Sparkfun Electronics, n.d.)
[138] (Sparkfun Electronics, n.d.)

Advantages: Very simple to use both for me and for the elderly, since it corresponds exactly to what is on the screen.

Disadvantages: In the same way as the rotary potentiometer, the area of movement is very limited and therefore if the list of passwords is long, the user might have to complete very small slides to move through the passwords. Additionally, this system doesn't contain a built-in push switch. If a micro-switch or push switch is not custom built into the back of the unit (which would be highly time consuming), an external switch would be necessitated which would take up extra space. Finally, this system would take up a lot of space.

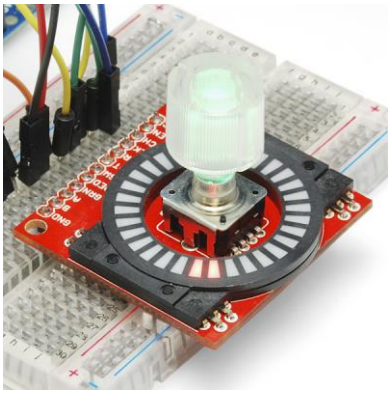
Rotary Encoder

Figure 30 Rotary Encoder

Source: Sparkfun Electronics[139]

A rotary encoder would allow for continuous rotation around a circle, while providing digital input to a microcontroller. The code provided would be in the binary format, similar to gray code, to show how much and in which direction the rotary encode has turned. Therefore, the encoder can be used to move along the list wherein the amount of rotation taken to move up or down the list varies or changes as desired. Additionally, the unit contains a built-in RGB LED which could be used to show the status of the product, forgoing the need for a power LED. Finally, it also contains a Normally-Open Switch which can be used as the select or back menu switch.

Figure 31 Illuminated Rotary Encoder



Source: Sparkfun Electronics[140]

[139] (Sparkfun Electronics, n.d.)
[140] (Sparkfun Electronics, n.d.)

Advantages: We would not need any external power or status LEDs or another button. Additionally, as opposed to the potentiometers it would provide very intuitive controls for an elderly person whilst not having the issues associated to the length of lists as with both of the potentiometer systems. Finally, because the steps associated with the turning provides tactile feedback, it would be easier for many elderly people to use.

Disadvantages: Relatively complex to use (relative to the other possibilities)

To facilitate a decision I made use of a decision matrix which rated each of the systems for the various criteria (each of which was weighted).

Figure 32 Decision Matrix for Input Device

	Ease of Use		Intuitiveness		Cost		Features		
	Weight	Mark	Weight	Mark	Weight	Mark	Weight	Mark	Total
Gaming Joystick	7	8	10	6	7	7	8	8	229
Rotary Potentiometer	7	9	10	8	7	7	8	7	248
Sliding Potentiometer	7	9	10	10	7	6	8	7	261
Rotary Encoder	7	7	10	8	7	8	8	10	265

Though the Rotary Encoder was far more complex than any of the other input systems, it's suitability in terms of continuous rotation as well as tactile feedback which would be attractive to elderly people meant it was the best choice.

For the power switch there were also a few options:

Toggle Switch

Figure 33 Toggle Switch

Source: Sparkfun Electronics[141]

Advantages: Very simple and easy to use, even for the elderly.

Disadvantages: Very large, taking up a lot of space, also sticking out from the product, since it would likely be mounted on the outside of the casing.

Rocker Switch

[141] (Sparkfun Electronics, n.d.)

Figure 34 Rocker Switch

Source: Sparkfun Electronics[142]

Advantages: Takes up less space than the toggle switch and looks slightly slicker. Requires less force to turn on and off – making it easier for those with hand issues.

Disadvantages: Sometimes easy to press the switch by mistake as it provides little tactile feedback.

Slide Switch

Figure 35 Slide Switch

Source: Sparkfun Electronics[143]

Advantages: Takes up less space than either of the others. Provides quite a lot of tactile feedback.

Disadvantages: Somewhat too small, could easily be missed by an elderly person with less than perfect vision.

In order to decide on the power switch I again used a decision matrix:

Figure 36 Decision Matrix for Power Switch

	Ease of use		**Cost**		**Aesthetic Appeal**		**Size**		
	Weight	Mark	Weight	Mark	Weight	Mark	Weight	Mark	Total
Toggle Switch	9	10	7	7	8	4	8	4	203
Rocker	9	8	7	7	8	8	8	7	241

[142] (Sparkfun Electronics, n.d.)
[143] (Sparkfun Electronics, n.d.)

Switch									
Slide Switch	9	5	7	7	8	7	8	9	222

The matrix shows that the Rocker Switch was the best option since it met all criteria (though not being the best for each individual criteria) without falling short in aesthetic appeal.

5.2.3.3 Fingerprint Sensor

For the fingerprint sensors, there aren't many options that exist, especially when one is looking based on proven experience of having being used with the Raspberry Pi. There are two basic options, going for USB fingerprint sensors which were made for external use, working back through the various applications they used and reverse engineering them – since they largely did not provide datasheets about how they worked or to take a Serial Fingerprint Sensor which would use up the one precious Serial Port on the Pi, and using the datasheet and creating a library that would work with this. It is clear that the latter would be better. Hence a couple of options were determined for these.

Grove Fingerprint Sensor

Figure 41 Grove Fingerprint Sensor

Source: Seeed Studio[144]

Advantages: Handles the image capture and rendering and calculations on board. Also the system has the ability to store up to 162 fingerprints on board the device, meaning the user can easily have multiple scans on-board one fingerprint scanner in case their finger gets injured – one of the major problems that were found with Fingerprint Scanning.

Disadvantage: The device is rather large and would take up quite a large space. Additionally, the design of the sensor, with the large bend where the optical camera itself is, might be quite challenging to implement mechanically, though it might be possible to open the device to reduce the size and eradicate the bend by re-soldering the boards at different angles. This would also mean that I would be able to lose the large casing which appears to take a large proportion of the space. Finally, I could find no example online of anyone managing to use this sensor with a Raspberry Pi, so it might require some experimentation to get it working.

GT511C3

[144] (Seeed Studio, n.d.)

Figure 42: GT511C3

Source: Sparkfun Electronics[145]

Advantages: It is much smaller than the other sensor as well as not having a bend. It has similar impressive abilities, also completing all the required functions, capturing images, and processing them on-board. The sensor has an ARM Cortex M3 processor on-board – a fairly impressive processor in itself! Additionally, like the Grove, it has space for a number of fingerprints, in this case 200. Meanwhile, while the False Acceptance Rate is similarly impressive (<0.001%), the false rejection rate was 10 times better (<0.1%). This implies that the resolution of this (450dpi) is higher than that of the Grove (which does not have a published resolution). Finally, though there is no working library available, there are a number of people who claim to have got this sensor working on the Raspberry Pi, implying that the task is not impossible. Additionally, there is a complete Arduino library which could be used to make a Python library.

Disadvantages: The sensor uses a rather fragile connector to breakout the Serial connection, which requires another wire. In addition, it is very expensive at £35, though the Grove is of a similar cost.

It is clear that the GT511 is the better option, since it has better all-round specifications, as well as better support, with the promise of a few people who had got the sensor working on a Raspberry Pi. However, it is important to note the option which was not considered, that of making the fingerprint sensor on my own. Though electrically relatively simple with an optical camera and a microcontroller, it is clear that algorithmically it is beyond the scope of the project. However, if the project were to be mass–produced it would be important to reconsider this decision, as it would be much cheaper to make the system myself, as it would only require the cost of a camera (and a low quality one at that) while the processing could be done on the Pi.

5.2.3.4 Screen

Though I had expected the screen to be a relatively easy choice, since there would be many available, the choice of Fingerprint Sensor had made this much harder, as the GT511C3 would use up the only Serial Port available on the device. Hence, I had to look to other transmission protocols, such as SPI and I2C to connect to the screen.

16x2 I2C Character (Backlit) LCD

Figure 43 Winstar 16x2 Backlit LCD

Source: Rapid Electronics[146]

[145] (Sparkfun Electronics, n.d.)

Advantages: By using a monochrome display – white on black as well as a backlit system, it would ensure the maximum brightness and contrast as required by my research into the elderly market. Additionally, the use of a character display ensured a large character spacing and indeed a relatively large line spacing, meaning things would be easy to read. Also, the system would be relatively easy to use, as well as being inexpensively priced at £6 for one, with large discounts for bulk purchases. The screen is also large, which means the font size is quite big, and readable by the elderly. In fact, a different Character LCD of the same size was branded by one reviewer as the 'perfect size for everyone'.

Disadvantages: The 16x2 size has a huge number of drawbacks including the lack of ability to show anything longer than 16 characters on the screen necessitating automatic scrolling or another system. Additionally, the two lines means that scrolling through a long list of passwords would take along time.

N.B. This option was originally a 20x4 Backlit Serial LCD which would have fixed a number of these disadvantages, but the lack of a second serial port has meant this would not longer work.

Adafruit 128x64 Graphical OLED

Figure 44 Adafruit 128x64 Graphical OLED

Source: Adafruit[147]

Advantages: The screen, largely due to the OLED screen, would have a very high contrast ratio, as well as having a very low current requirement – 40mA. Additionally, the screen can communicate using both SPI and I2C, with Adafruit having produced a Raspberry Pi Python Library to work with both, making the job of getting them working easier. In addition, Adafruit even has a basic guide, showing how to install the software as well as wiring the screen. Despite it's small size 35mm diagonal, the screen has a large number of pixels, which would allow for a large number more data. The exact font size could also be set, using trial and error, to ensure that the data vs readability is ideal.

Disadvantages: The very small size screen means that to ensure the readability of information for the elderly, that very little information could be held on the screen at one time. Additionally, for it's size, it is very expensive, costing £25. This together with the cost of the Fingerprint Sensor would already be over the budget that was set out in the specification

PiTFT 2.8" Touchscreen (320x240)

Figure 45 PiTFT Touchscreen

[146] (Rapid Electronics, n.d.)
[147] (Adafruit, n.d.)

Source: Adafruit[148]

Advantages: The touchscreen would add another way of interacting with the product, with the user being able to scroll to move through the list of passwords and tapping the password to see further information. Additionally, the large size of the screen mean that the original ideas for a complete graphical interface would be possible. The display connects over SPI meaning it should be relatively easy to wire.

Disadvantages: Despite the existence of libraries designed to work with the display, this would make rest of the software far more complex, as a full graphical forms interface would have to be designed. Additionally, there have been a few comments on the internet which appear to suggest that apart from displaying the Pi screen it is very challenging to display specific things. Since displaying the screen is not a good idea, as it would allow for the potential of backend problems to interrupt the main program, this would pose quite a large challenge. Meanwhile, the screen is rather large taking up a lot of space and possibly making the device excessively large. Finally, if the touchscreen were to be used and the manual rotary encoder not, it could be challenging for a number of elderly people who have issues with Fine Motor Skills. In fact, the touch screen is one of the most complained about things when I was talking to a few elderly people in my neighbourhood, as they found it rather irritating. Finally, the screen is very expensive, at £50, meaning it would use my entire budget in one go.

In order to make my decision, I created a decision matrix.

Figure 46 Decision matrix for screens

	Ease of use		Cost		Ease of seeing for elderly		Size		
	Weight	**Mark**	**Weight**	**Mark**	**Weight**	**Mark**	**Weight**	**Mark**	**Total**
16x2 Character LCD	**9**	**10**	**7**	**10**	**9**	**8**	**8**	**4**	**264**
128x64 Graphical OLED	**9**	**9**	**7**	**6**	**9**	**8**	**8**	**5**	**235**
320x240 Touchscreen	**9**	**6**	**7**	**3**	**9**	**9**	**8**	**8**	**220**

From the decision matrix it is clear that the 16x2 Character LCD is the best screen, despite the small number of characters / lines of data it can display at one time.

[148] (Adafruit, n.d.)

5.2.3.5 Battery System

In order to power the power hungry Raspberry Pi and ensure that the product would last enough time before the product was to be recharged or the batteries replaced (another decision to made in this section), the selection of batteries was important. For this, there were two main decisions, the choice of Battery Chemistry and the exact battery to use.

Figure 47 Battery options

	Advantages	**Disadvantages**
Lithium Ion	Relatively high energy density. Very low self-discharge – they do not lose charge quickly over time when left alone Low maintenance. Relatively safe – require large issues to produce battery explosions.	Despite relative stability they still require some protection during both charging (to prevent overcharging) and discharge (to prevent too fast discharge). However, today, these protections are often included in the batteries or can be very simple to add (simply an IC or an array of Zener Diodes). Despite normally being cheaper than Lithium Polymer batteries, they tend to be around 40% more expensive than Nickel Cadmium batteries. High internal resistance to the batteries, so tends to be inefficient in high current situations.
Lithium Polymer	Can provide higher current than Lithium Ion batteries. Very high energy densities, meaning a smaller battery can contain a larger amount of potential energy – powering the device for a longer period of time.	Far more volatile than Lithium Ion and Nickel Cadmium batteries, hence require very careful usage of them, as well as sophisticated protection, which is relatively complex. In the event of issues, damage is far more catastrophic than Lithium Ion and Nickel Cadmium Batteries. High maintenance – they should not just be left in a product that is slowly draining current.
Nickel Cadmium	Very inexpensive. Relatively reliable. Tend to have relatively high output current capabilities.	Low energy density. Have a tendency to lose battery quickly in low current situations.

When making the decision it was important to consider given the various drains caused by the components, and hence the current requirements of the batteries.

Figure 48 Current Consumption

Component	Current Requirement
Raspberry Pi Zero	c. 400mA
GT511C3	c. 100mA
Rotary Encoder (with RGB LED)	c. 40mA
LCD Screen	c. 120mA
Camera	c. 100mA
TOTAL	c. 760mA

It is clear therefore that the issues associates with high current properties of the batteries will have little effect on the product, and then we must consider the importance of other specifications. It is clear that most important would be a high safety, to ensure that the probability of any problems with battery explosions be minimised. Hence, we must remove Lithium Polymer batteries due to the easy possibilities of misusing them. Additionally, if the product were to be dropped and the lithium polymer battery were to come into contact with a pin of a PCB, puncturing it, it could easily lead to an explosion which could have potentially horrific safety implications. From NiCa and Li-Ion batteries, the two choices left, which both have similarly good safety, the next most important characteristic must be a high energy density to ensure that with a small mass (and indeed size), the highest capacity could be gained to ensure the product could be left on. Furthermore, the low current efficiency of the Li-Ion batteries is useful, as it ensures that the product would not lose too much power when the product is in a standby mode and only the Pi is on.

From the many choices available for Li-Ion cells, we must also choose the correct one of these:

Figure 49 Battery Sizes

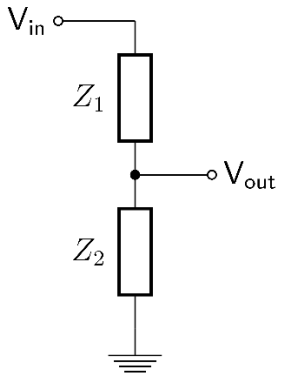
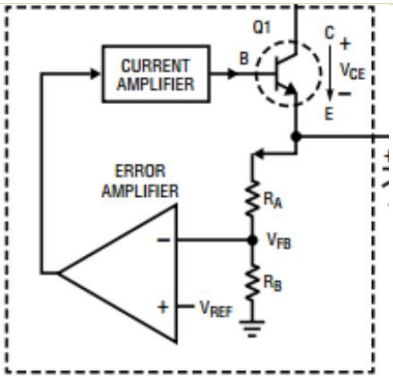
Source: AlphaRubicon[149]

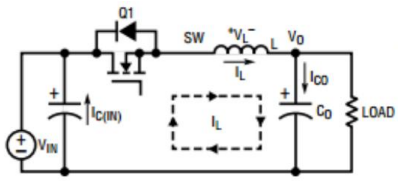
With the exception of the 9V battery, the vast majority of Li-Ion batteries are of 1.5V sizes (one cell) and this poses the challenge of combining them to provide enough voltage. This could be done either by using more batteries (which would increase the size required for them) or using a boost regulator. This is necessary since all the components run off 5V or 3.3V. While the Raspberry Pi helpfully provides a 3.3V voltage regulator on-board, we still have to provide the main 5V supply to the Raspberry Pi. While this could be done by combining at least four AA, AAA or AAAA and regulating the voltage down to 5V this would take quite a lot of space, and hence make the product bulkier than it needs to be. Additionally, the main advantage provided by these batteries, the provision of higher currents is largely unnecessary in my product, which would draw a peak current of around 760mA as shown in the table. Hence, by far the best choice of battery is the 9V battery, which deals well with the relatively low current draw, with very low self-currents meaning that it could also be left off for long period of time while remaining

[149] (AlphaRubicon, n.d.)

charged. However, this does mean that a 5V Voltage Regulator would be required to generate the correct voltage for the system and for this there are a few options.

Figure 50 Methods of regulating voltage

	How it works	Advantages	Disadvantages
Voltage Divider	 By altering the resistances of Z1 and Z2 (hence changing the voltages around each resister), the exact value of Vout can be changed, following the equation: $$Vout = \frac{z_2}{z_1 + z_2} \times Vin$$	The system is very simple to setup and use.	Does not provide a very constant voltage, hence creating problems in the Raspberry Pi. This is due to problems largely in the load, since the inputted voltage will not remain constant – instead dropping as the battery gets depleted. Additionally, if there are other loads sharing Vin, there would be a reduction in this Vout. Very inefficient.
Linear Voltage Regulator	 A linear voltage regulator essentially uses a voltage divider with a feedback loop that alters the inputted voltage into the voltage divider through a Field Effect Power Transistor (FET). It means that when Vbus decreases (hence causing Vcc to reduce), Vfb decreases as well. This means that the error amplifier voltage increases and the current passing through the FET increases meaning the voltage drop Vce reduces and hence Vcc increases.	Output voltage is regulated, meaning a constant input voltage to the Raspberry Pi. Very simple to use.	Generally, more expensive than a voltage divider. Still relatively inefficient, producing a lot of heat – efficiency is generally in the region of 25-30%.

	When the Vcc increases too much, the opposite happens, meaning the Vcc is always regulated to a specific value.		
Switching Voltage Regulator	Switching Mode Voltage Regulators are more complex, using a number of transistors in switching mode (enabling it or disabling it) instead of using the transistor in a linear fashion.  	Switching Mode Voltage Regulators are far more efficient than other voltage regulators producing far less heat than the other forms – in fact they have around a 90% efficiency as opposed to 30% (the maximum produced by Linear Regulators) Can produce much higher current more easily.	Much more expensive than Linear Voltage Regulators. Much more complex to use – having specific requirements for input voltages and currents.4

Decision: Despite the greater efficiency that is displayed by the Switching Voltage Regulator, due to the simplicity as well as the far reduced cost associated with Linear Voltage Regulators, it is clear that they are the better choice for me. In addition, it would allow me to get LDO (Low Dropout Regulator) which would continue to work very close to 5V ensuring that the greatest amount of energy could be got out of one charge before the battery must be replaced or recharged.

Hence, the statistics of the regulator to be obtained is: 5V, Low-Dropout Linear Regulator. Additionally, since 5V is the only voltage that is desired, and the additional circuitry (though rather simple) which would be required to use an adjustable regulator, a fixed regulator would be best. Finally, since the peak current of the product would be around 760mA, the regulator must be capable of providing at least that current without breaking. In fact, it would be advisable to leave a small window such as 250mA so that if there were any increases in power requirements by components such as the Raspberry Pi that this could be provided. Hence, the common 1A regulator would probably be sufficient, providing enough of a leeway.

The other question is that of Through Hole components versus Surface Mount. Though Through hole components would be much easier to hand solder and use for prototyping, it would make any PCBs much larger, and would also be more expensive as the circuit board manufacture were to be automated. Hence, planning for the automation of components, an SMD version would be preferable.

Hence, through setting the required information on Farnell, I found the cheapest product which would meet these specifications looking also at the dropout voltages, ensuring it was as close to 5V as possible. Hence the following product was found.

On SemiConductor NCP1117ST50T3G

This uses the SOT2233 package which though SMD is actually relatively easy to solder by hand, and therefore means that making a prototype might be possible. It has a minimum input voltage of 6.4V and a maximum of 20V (far higher than the maximum of the batteries of around 9V).

Recharge or Replace

Given the choice of batteries, it is also important to begin to consider how the user might go about using the product once the batteries have run out. For this there are two main options – replace them, or recharge them – which would require the building in a re-charging system into the electronics.

Figure 51 Table comparing the merits and flaws of recharging or replacing batteries

	Advantages	Disadvantages
Recharging	Would be cheaper – the user would not have to replace the batteries after they run out once. Relatively easy for the user to plug in a USB or suchlike to charge the product	Would have to build in an IC to deal with the charging as well as considering how the user would connect a power source to the product. Limits the system to using slightly lower capacity rechargeable batteries – so the product would have to be recharged more often than being replaced. Product would sometimes have to have batteries replaced anyway, as the batteries have a max number of charge-cycles.
Replacing	Allows the system to use higher capacity batteries which are not rechargeable.	Mechanical system would need to allow the user to be able to open the product Painful for the user – especially considering the elderly target market it poses a large challenge as they may struggle to understand this.

In making this decision, it was clearly important to consider the target market and the pain and irritation the replacement of batteries would cause. Additionally, it was important to consider how long the product might last with a specific battery, hence calculating how long the batteries would last and hence how often the batteries would have to be replaced or recharged. I used statistics from batteryshowdown.com to show that the average 9V battery had a capacity of around 2500mAh. Given the voltage of 9V, this translates to around 22,500mWh[150]. Assuming a running voltage of all components of 5V, this produces a capacity of 4500mAh. Assuming an idle current of around 300mA (using measured currents of a Raspberry Pi Zero) and a running current of 760mA, the battery would last for fifteen hours when idle and 5.9 hours when being used. This means that even leaving the product on for a day would mean it would run out of power. It also reinforces the necessity of adding in a simple SPST power switch to allow the user to entirely turn off the system, more than just allowing the Raspberry Pi to go into a 'sleep' mode.

The frequency with which the battery might have to be charged if it is mistakenly left on, means that a recharging system should likely be built in, to ensure that it could easily be recharged. This necessitates the usage of a charging IC. A number of possibilities were considered to be used, including the MAX1737 and the BQ24259, both of which would comfortably charge the 9V battery, however, due to it's cleverer technology in charging, the MAX1555 was chosen. It means that the chip will continue to read the voltage from the battery during charging and turn off the

[150] This figure is slightly high since batteries reduce in voltage as they discharge.

charging once it has received a preset limit, which is set by passives connected to the MAX1555. Additionally, it includes a thermal limiter, so if the system gets too hot during heating it will turn off the charging. Finally, the MAX1555 also has a specific pin for 5V, USB input which could be connected directly to a Micro-USB female port – as shown – and a normal Android Phone charging wire could be used to charge the battery.

5.2.3.6 Camera

QR codes, such as the one shown below, is a form of matrix barcode barcode, (two-dimensional barcodes) which were first designed for the automotive industry in Japan. A QR code uses four standardized encoding modes to store data the most efficiently: numeric, alphanumeric, binary and kanji (one of the main Japanese / Chinese scripts). The QR consists of black squares on a white grid background, taking the binary data (black and white) from the vertical and horizontal patterns that exist in the image.

The system makes use of 3 set patterns for orientation, so a QR code shown in the wrong orientation could be corrected for by the computer, ensuring the data cannot be wrong through this. Additionally, QR codes take advantage of Reed Solomon error correction, with a preset error correction percentage – which means despite a specific percentage of failures in scanning the system would be able to return the correct data. This is normally 15%, though it sometimes can be placed at 30% if this higher error correction is required. QR codes carry a number of advantages over their one dimensional UPC cousins, such as those found on all supermarkets purchases one makes. Firstly, they carry characters as well as numbers, a large advantage in this case, as this would prevent the length of the 1D barcode being excessively long. Additionally, they can be scanned in any direction, with the 3 bar system meaning they would be encoded in the correct orientation. Thirdly, in addition to the ability to store characters, QR codes can store far more data than standard 1D barcodes, with the maximum amount of data being able to be stored in a version 40 QR code being in the region of 4KB.

Figure 52 Sample QR code

Source: QRStuff[151]

For my system, using a number of different phones, with varying qualities of camera (with them being used instead of buying a number of different camera) I found the size of QR code that I could can and from what distance, to allow me to find the correct size and the quality of camera I would need.

Figure 53 Experiment results for scanning QR codes with different quality cameras

	2 bytes		**10 bytes**		**100 bytes**		**1KB**		**2KB**	
2MP - Nokia N70	0.2m	Medium	0.2m	Medium	/		/		/	

[151] (QRStuff, n.d.)

5MP - Samsung Galaxy SIII Mini	0.5m	Fast	0.5m	Fast	0.75m	Fast	/		/	
10MP - iPhone 5	1m	Fast	1m	Fast	1m	Fast	1.5m	Slow	/	
16MP - OnePlus One	1m	Very Fast	1m	Very Fast	1m	Very Fast	1.25m	Very Fast	1.5m	Medium

Hence, in choosing a camera, it would be important to consider at least a 5MP camera, and 10 if possible, since they appear to relatively easily cope with 100 bytes. Since this would convert to 100 alphanumeric and most basic special characters, this appears more than enough to store all the basic information about one account, such as the 'Account name' – by which the user could remember what the account is for, 'username' and 'password'. This however, does restrict the system to dealing with a single password at a time, while a 16MP camera would appear to relatively easily cope with ten or twenty passwords, which would save a lot of time of scanning and inputting for the user. Hence, a list of possible camera for the system were devised.

Raspberry Pi Camera – 5MP, £20

Advantages: The Raspberry Pi offers an option which has already been used by a number of people successfully, by using the approved camera made by the Raspberry Pi foundation. They also produce the PiCamera library, which would hopefully allow the camera to work easily with the Raspberry Pi. It also connects to the CSI port, which means that the camera will not take up one of the precious USB ports on the Pi Zero, or indeed connecting over GPIO.

Disadvantages: It has a fixed focus lens which means that the QR code would have to be held at a specific distance from the camera for the code to be in focus. According to some online research, this distance would be around a metre, which is reasonable, but the lack of automatic focus might make the task of scanning the code more challenging for some.

Logitech C270 HD Webcam – 3MP, £15

Advantages: It is cheaper than the Raspberry Pi camera, hence allows me to more easily meet my price specification. Additionally, as opposed to the Chinese cameras (the next two) they are well supported by Logitech, and would be reliable. Finally, the camera also implements autofocus, which means that the QR code could be held at a greater range of distances and the system would be able to deal and adapt to this.

Disadvantages: Low quality sensor means that the camera might struggle with QR codes with large amounts of data. Additionally, since the camera connects over USB, the very well supported PiCamera library could no longer be used, however, there are a number of other libraries such as fswebcam, which could be used.

5MP Micro Webcam - £12

Advantages: This camera offers the greatest quality to price ratio, offering a very low cost for a decent 5MP sensor. Additionally, the sensor even comes with autofocus.

Disadvantages: The camera has to be bought from dodgy Chinese manufacturers which a number of users appear to have had a number of problems. In fact, when I went to the Chinese website, and looked at some of the comments, (after translating them) I found that a number of people had had problems with the reliability of the camera, one

saying it 'worked once, then not again'. Additionally, as opposed to the PiCamera, the camera plugs in over USB, and has similar irritations as described for the Logitech C270 HD

8mp ov5640 mini hd cmos senor (sic) webcam camera module - £30

Advantages: The camera is of the best quality and so should be easily able to deal with the QR code scanning. Additionally, the camera again contains autofocus built in.

Disadvantages: The camera again must be purchased from AliExpress – hence bringing the same reliability issues as for the 5MP micro webcam. In fact, here, there was a few complaints about the quality of the camera, with a couple of users complaining that the sensor had a much worse quality than they had expected.

To make a decision, I made a weighted decision matrix, weighing up important points such as the quality of the sensor, reliability and additional features.

Figure 54 Decision matrix for camera

	Quality of Camera		**Ease of Use**		**Reliability**		**Cost**		**Additional Features**		
	Weight	Mark	Weight	Mark	Weight	Mark	Weight	Mark	Weight	Mark	**TOTAL**
Raspberry Pi Camera	8	6	8	10	9	9	8	7	5	2	275
Logitech C270HD	8	3	8	7	9	7	8	5	5	4	203
5MP Micro Webcam	8	6	8	7	9	2	8	9	5	4	214
8MP OV5640	8	9	8	7	9	2	8	7	5	4	222

From the decision matrix it is clear that the best choice is the Raspberry Pi Camera, due to it's superior reliability and ease of use.

5.2.4 Power Loss Protection

Much of the challenge when producing a 'secure' device is ensuring that during all phases of the product there is no way for an unauthenticated user from gaining access to the list of unencrypted passwords. As it transpires this poses most of the challenge during the phase of power-off, since during the final checks before turning off, or indeed sleeping if so designed, the device must prevent anyone from seeing the passwords, locking them again. In this case, the Raspberry Pi must encrypt the file again, using the Advanced Encryption Standard. However, if the power is interrupted at this point, there is a possibility that the final locking commands would not be sent (by the Raspberry Pi), and so the passwords would remain unlocked. In fact, even if the product were to be interrupted during specific periods of the normal operation, the file would be unencrypted, since it is being read from, and hence it would be liable to a similar attack. A hacker would be easily able to open the device, remove the SD card from the Raspberry Pi and find the passwords in plaintext. Though there might be a few ways of resolving this in software, in particular concentrating on how the files are encrypted and unencrypted so that the unencrypted file is not open for excess time, there would be other issues. For example, consistent power cycling would likely damage the Raspberry Pi as well as the other components, since they may not have the opportunities to shut down safely. This could cause malfunctions which could affect the security of the passwords in unpredictable ways. Hence, it is generally worth finding methods to ensure that there are no sudden power-offs. Additionally, given the choice of the fingerprint sensors (the GT511C3), where the bitmap template of the registered fingers would be stored onboard the fingerprint

sensor itself, it means that one might be able to open the device and access these IDs, which could then be used to break into the device (especially since they end up being chosen to be the key for the encryption of the passwords). Hence, at the likelihood of any break-in it would be a high priority to ensure that the IDs are deleted from the fingerprint sensor, requiring an additional signal sent to the sensor.

5.2.4.1 Ultra capacitor uninterruptable power supply

One method for creating an uninterruptable power supply is using an supercapacitor, to provide power, when the battery runs out. This should then power the Raspberry Pi for the matter of a few seconds while it ensures that all the files are encrypted and powers down safely. This would also be able to send the delete ID signal to the fingerprint sensor to ensure that in the case of the sudden power-off – indicating a hacking attempt – the IDs and hence, the key to the encrypted passwords would be safe. However, it might be far more efficient to use a different IC (such as the ATTiny line of chips) to send the required signals to the GT511C3 (fingerprint sensor), since they would require far less current (requiring a much smaller capacitor), so this needs to be considered. To help to determine the exact figures involved with the system, I performed a current test on the Raspberry Pi, in two different manners, when idle, and under stress, which would help me determine whether powering the Raspberry Pi would be possible (as it was the simpler and better option). Throughout the process, the Pi was plugged into the regulated 5V supply of a Desktop PC, and the current drawn from the socket measured. The following results were found:

Figure 55 Results from current consumption test of Raspberry Pi Zero

	Test One	Test Two	Test Three	Average
Idle	346mA	330mA	362mA	346mA
Stress Test (CPU)[152]	440mA	554mA	490mA	495mA

This is as compared to the current requirements of other usable ICs hence:

Figure 56 Current Requirements of a number of Microcontroller ICs

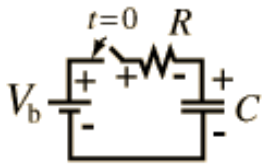
ATMega328	ATTiny85	ATTiny45	PIC20M2	PIC18M2	ARM Cortex M3
16.43mA	10mA	8mA	28mA	14mA	51mA

From the data, there are clearly a number of IC's which would meet the requirement of consuming less power, and hence could be more efficient. Also, the Atmel chips and the ARM chips could relatively easily be made to work with the Arduino bootloader and hence could programmed using Arduino to send the relevant commands to the fingerprint sensor, using existing libraries for the device hence requiring little additional software. However, it is important to note that it would require large amounts of external circuitry to allow the other chips to work and to program them. However, most importantly, the external IC would not be able to deal with the encryption of the passwords (not communicating with the SD card of the Pi), and hence we must instead concentrate on ensuring the Raspberry Pi can be kept on with the ultracapacitor.

The general charging circuit of a capacitor is hence:

[152] The choice of the stress test was largely centered around the CPU since this was being tested during the actual powering off procedures, as there would be no monitor or touchscreen connected which would test the GPU.

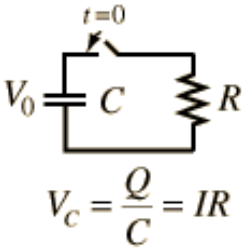
Figure 57 Capacitor Charging Circuit



Source: HyperPhysics[153]

and discharge circuit is hence:

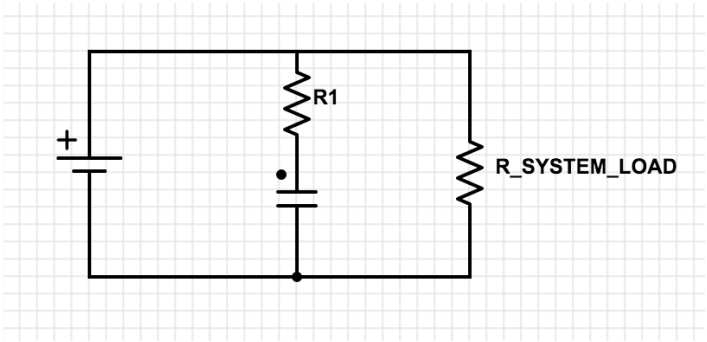
Figure 58 General Capacitor Discharge Circuit



Source: HyperPhysics[154]

By combining the two circuits, we can form a circuit which would charge a capacitor when the battery is connected and not at too low a voltage and would switch to being powered by the capacitor when the battery is disconnected.

Figure 59 General Circuit Diagram for Capacitor System



Source: Designed using CircuitLab[155]

To help to determine the exact size of the capacitor that we want, we must look at the physics of a capacitor and the exact discharge curve. A capacitor follows an exponential discharge curve, with the following equation of the voltage of the capacitor:

$$V_C = V_0 e^{-\frac{t}{rc}}$$

- Vc is the current voltage

[153] (HyperPhysics, n.d.)
[154] (HyperPhysics, n.d.)
[155] (CircuitLab, n.d.)

- Vo is the starting voltage of the capacitor – for our purposes, since we want this to be 5.5V as this value needs to be above the required powering voltage of the Raspberry Pi (4.5 – 5.5V) – as this ensures that we would not need another voltage regulator, which would make the system far more inefficient.
- t is time
- r is the load resistance
    - By V = IR, R = V/I
    - Calculating the total current required for the complete power off system:

Figure 60 Total Current requirement for the system

Raspberry Pi	**495mA**
GT511C3	130mA
PiCamera	50mA
TOTAL	675mA

    - V = Vc, and I = 675mA
- c is the capacitance of the capacitor

Subbing in for r:

$$V_c = V_0 e^{-\frac{t}{\left(\frac{V_C}{I}\right) \times c}}$$

Subbing in for I:

$$V_c = V_0 e^{-\frac{t}{\left(\frac{V_C}{0.675}\right) \times c}}$$

Subbing in for V_0:

$$V_c = 5.5 e^{-\frac{t}{\left(\frac{V_C}{0.675}\right) \times c}}$$

Given that we want the voltage of the capacitor to be above 4.5V for the system to still be working, hence Vc must remain above 4.5V. Despite the time for the final commands to be sent probably requiring around a micro-second, it would be highly useful if it could also have time to receive information back (acknowledgement) from the GT511C3 and resend if any error. Hence, also adding on the few seconds required for the safe bootdown of the Raspberry Pi, the capacitor should hit the minimum working voltage (4.5V) around after a least a couple of seconds. Hence, we are looking for the value for c from the equation, given the following:

Vc = 4.5 Volts

T = 2 seconds

So, first we must rearrange the earlier equation:

$$\ln(V_C) = \ln(5.5) + \ln(e^{-\frac{t}{\left(\frac{V_C}{0.675}\right) \times c}})$$

$$\ln(V_C) = \ln(5.5) - \frac{t}{\frac{V_C}{0.675} \times c}$$

$$\frac{V_C \ln(V_C)\, c}{0.675} = \ln(5.5)\left(\frac{cV_c}{0.675}\right) - t$$

$$\frac{cV_c \ln(V_C)}{0.675} - \frac{cV_C \ln(5.5)}{0.675} = -t$$

$$c\left(\frac{V_c \ln(V_C)}{0.675} - \frac{V_C \ln(5.5)}{0.675}\right) = -t$$

$$c = -\frac{t}{\left(\frac{V_c \ln(V_C)}{0.675} - \frac{V_C \ln(5.5)}{0.675}\right)}$$

Sub in the values:

$$c = -\frac{2}{\left(\frac{4.5 \ln(4.5)}{0.675} - \frac{4.5 \ln(5.5)}{0.675}\right)}$$

Calculating this, c = 1.49F (3sf)

Therefore, we must look for 1.5F, 5.5V Capacitor. With a cursory search online on Farnell, there are not many options, so I decided to go for the cheapest option – the Panasonic EECF5R5U155 – a 1.5F +80% -20% 5.5V capacitor. Despite the very high tolerance, this appears to have little difference, since the two seconds allotted for should be far more than necessary. Additionally, the 80% upwards tolerance indicates it is more likely that the capacitance is higher than the 1.5F, and hence would mean the product would have even more time to send the final commands rom the Raspberry Pi and turn off.

5.2.4.2 Mechanical latch

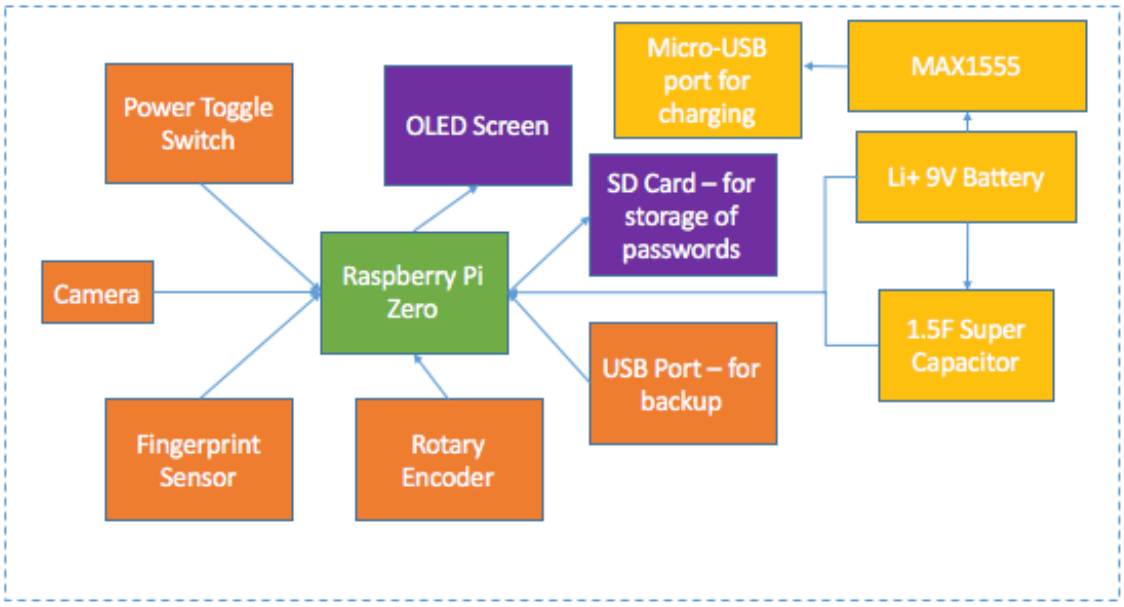
In addition to the power being able to be removed, the ability for the person to be able to open the device and investigate the internals of the computer would more easily allow them to find a loophole around the security. For example, since the information about fingerprints is stored on the GT511C3, if one were to gain access to the product, they could write their own code to retrieve the information from the product, then using the information retrieved from there to decrypt the file, since the key of the fingerprint will likely be used as the key for the AES encryption for the passwords. One way to prevent the user from gaining entry to the product would be making the product of impenetrable materials, such as thick lead. However, this would be very heavy as well as very expensive. Hence, other methods other than making the shell itself impenetrable must be investigated. One other way that could occur is creating a sealed loop of conductor around the entire product (in the inside) which could be used to check that the product is still together. Then, when the product is opened, the Raspberry Pi could read this change, as the loop would now be broken, and hence could respond, deleting the ID information from the fingerprint sensor and erasing the Raspberry Pi code itself, ensuring the person cannot gain access to the coding of the product. However, given the battery might need to be changed, especially if someone is on a long journey with little access to power supplies to recharge the battery, another system must be added to allow the user to inform the Pi that they are about to break the inner conductive loop, and the product is not being hacked. This should be only possible once the product has verified the identity of the user.

The electronics of the system would work much like a switch, using a potential divider system with the wire around the product connected in place of the switch. The output of the potential divider system is connected to a Raspberry Pi GPIO pin. Meanwhile, it means that the main power switch, as described above must be connected to the Raspberry Pi rather than directly across the power connections since the Pi must remain in a state of sleep to allow it to read when the case is opened even if the product itself is powered down.

5.2.5 Conclusion

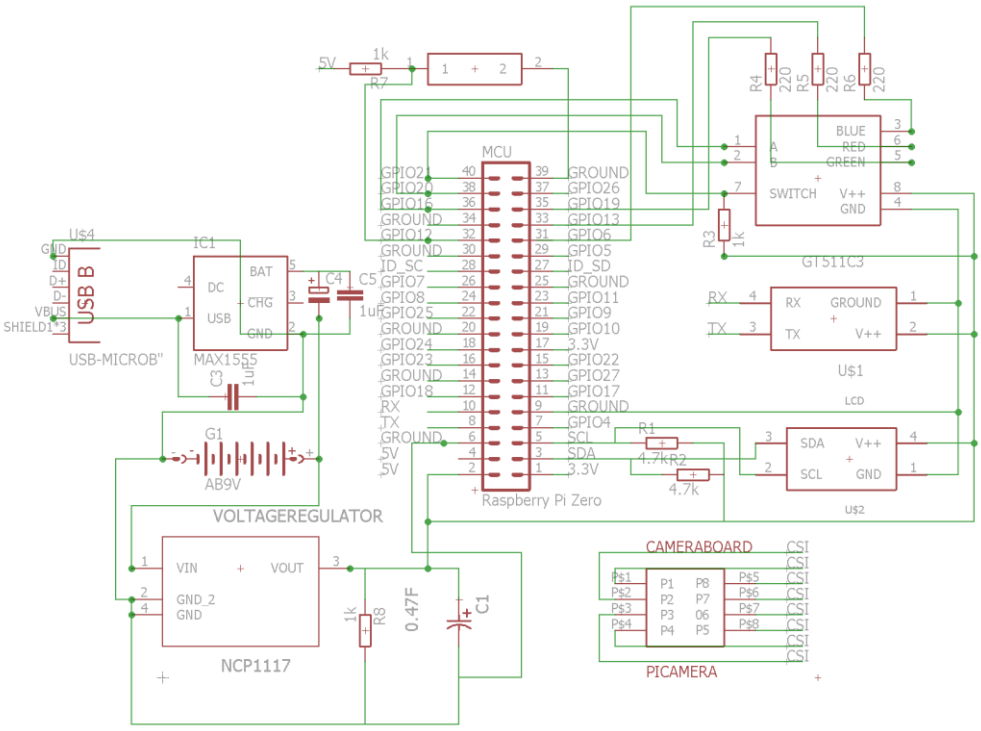
At the end of the Control System Research and the initial decisions, the electronics design stood hence:

Figure 61 Basic Diagram of Electronics



From here, looking at the various datasheets, I went about making a full circuit diagram, which would incorporate all the required passive components and also provided n opportunity to ensure that the selected components would meet the needs of the system. This eventually led to the following diagram made in CadSoft Eagle CAD[156] – which includes all the connections required.

Figure 62 Initial Circuit Diagram



Source: Produced using CadSoft USA Eagle CAD

[156] (CadSoft USA, n.d.)

5.3 Mechanical Research

5.3.1 Usecase Discussion and Mechanical Requirements

The mechanical design in many ways is held back by the need to carry specific components, ensuring that they are held comfortably into the product. It attempts to create an ergonomic system for the user, with materials which would be strong enough to withstand both being dropped and the basics stresses of being placed in a packed bag. As the target market is the elderly, we must think about the special stresses that the product might encounter. Despite our earlier research showing that the main skill which degrades over time is fine motor skills, there is also some degradation in grip and so the product might be dropped a few times. In order to minimise damage, the geometry of the product must be investigated, including looking at the size of the average hand, and beginning to think of ideas for handles etc. which could provide the user a larger area to hold the product. Similarly, in the first section, we must look at the material chosen, to see how it could withstand falls and other stresses it would encounter over time: both looking at it mathematically and (given the problems encountered with this method) qualitatively.

5.3.2 Materials

Material property is important to ensure that it would not suffer any problems during its usage. Though foremost in this is the resistance to fracture when dropped, it is also important to consider the other various stresses the product might be under and ensuring that any material chosen is capable of withstanding these.

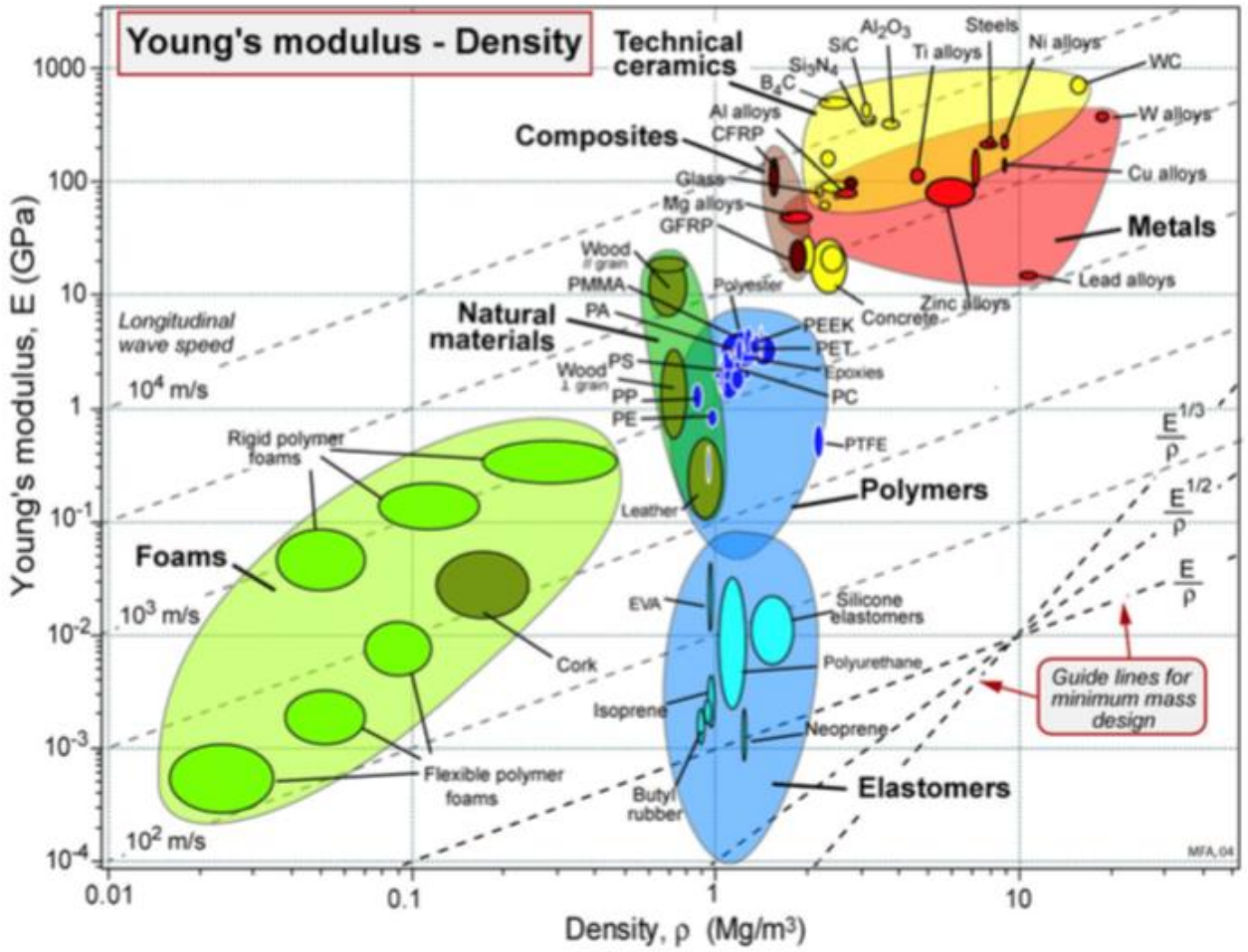
Of the various stresses which could occur, it is clear that tension and compression are more likely to have an impact on the product than shear forces. Though shear forces could occur by mistake when handling the product, these would likely be minimal in magnitude and would be unlikely to test the material properties of the material chosen. An ergonomic design, intended to be held in both hands, in a tight position would likely result in increases in tension stress, while the use of handles – a possible idea – would results in the necessity of both tension and compression stresses being applied on the casing. Hence, it is important to consider the material's resistance to deformation given tension and compression forces. Hence, we must consider the two equations.

$$\boldsymbol{Tension}: Stress = E\ (Young's\ Modulus)\ \epsilon$$

$$\boldsymbol{Compression}: Stress = K\ (Bulk\ Modulus)\ \epsilon$$

Hence, we must attempt to maximise the K (Bulk Modulus) and E (Young's Modulus). In order to find materials which would have suitable bulk moduli, we can look at an Ashby Plot of Bulk Modulus. However, it appears that since Bulk Modulus is rarely used, these have not been created. According to the CES EduPack Resource Booklet however, there is a conversion between the Young's Modulus and Density, which would allow us to estimate the Bulk Modulus of materials. It suggests that E ≈ K for metals, ceramics glasses and glassy polymers, while K ≈ 10E for elastomers and rubbery polymers. However, (despite their very high Bulk Moduli) due to the very high cost of getting the materials and manufacturing with these, it would be almost impossible to use them. Hence, for the purposes of this report E ≈ K.

Figure 64 Young's Modulus vs Density Ashby Plot

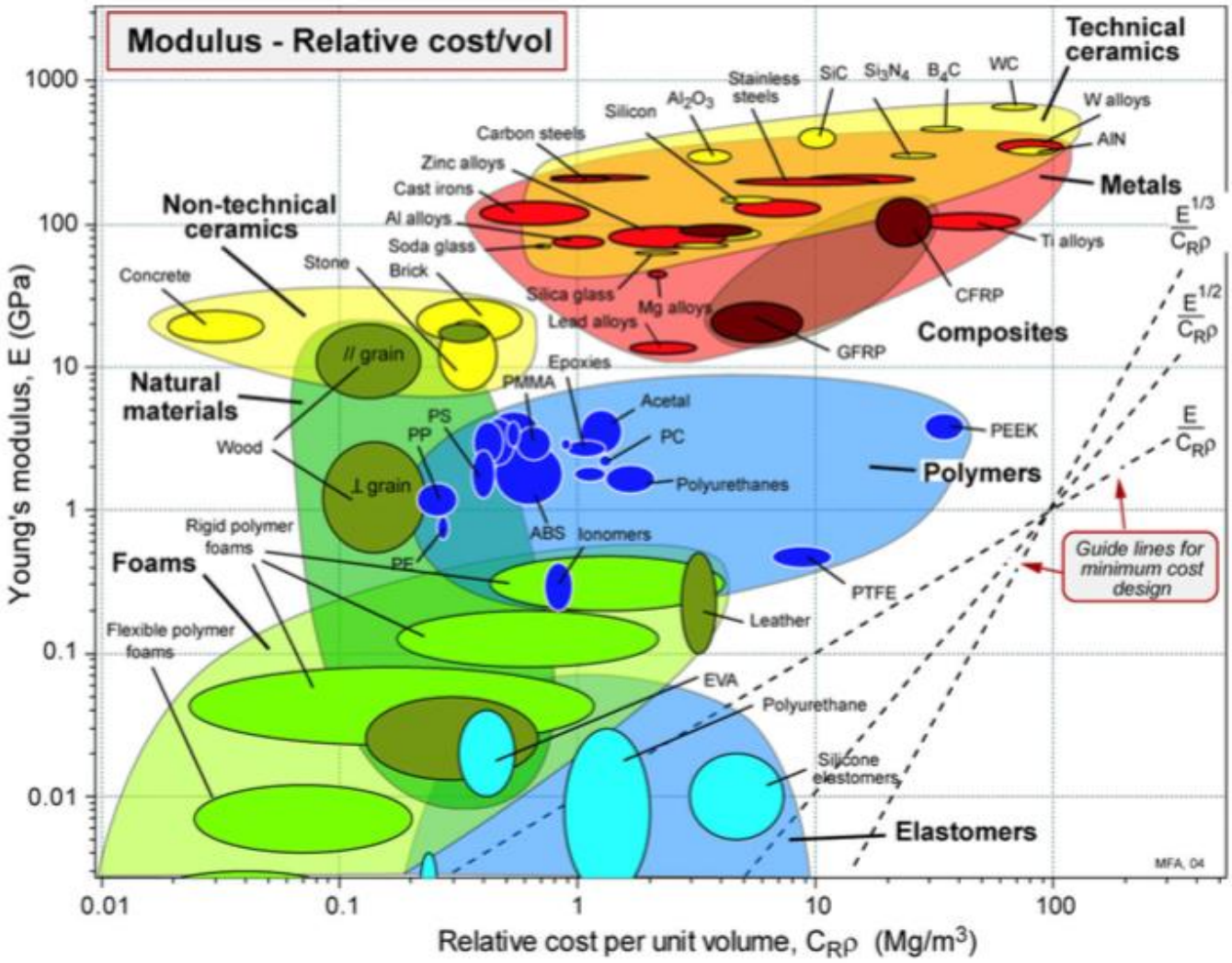


Source: VeloSpace[157]

The graph shows a general trend that as the density of the material increases, making the product more heavy, the Young's Modulus becomes higher (hence better). In order to draw a good compromise between the two, the ideal material would likely come from the Natural Materials or Polymers section of the graph. However, given prior knowledge about the difficulties of manufacturing wood to the correct shape it is obvious that the product offers a much lower versatility in producing precise and cheap geometries, with milling and routing being the best for these – however, both producing excessive wastage and being quite expensive. Meanwhile, polymers allow a number of manufacturing techniques, from 3D printing to Injection Moulding, which would allow one to producte the elegant ergonomic casing more effectively.

[157] (VeloSpace Forums, n.d.)

Figure 65 Young's Modulus vs Relative Cost per Unit Volume



Source: Cambridge University[158]

Toughness

As we have previously shown, during the aging process, there is a significant degredation in hand skills – including fine motor control and grip. This means that it would be highly likely that the product might be dropped and hence, it would be important to ensure that a drop would not fracture the product, rendering it useless. In order to withstand these drops, we must maximise the toughness of the material used. Toughness is the ability of a material to absorb energy and plastically deform without fracturing. Toughness is often calculating the area under a stress strain curve during a tensile test, taken at the point before the material fractures. This ensures the key to toughness is the combination of strength and ductility, with both having an impact on the energy that would be absorbed before fracturing.

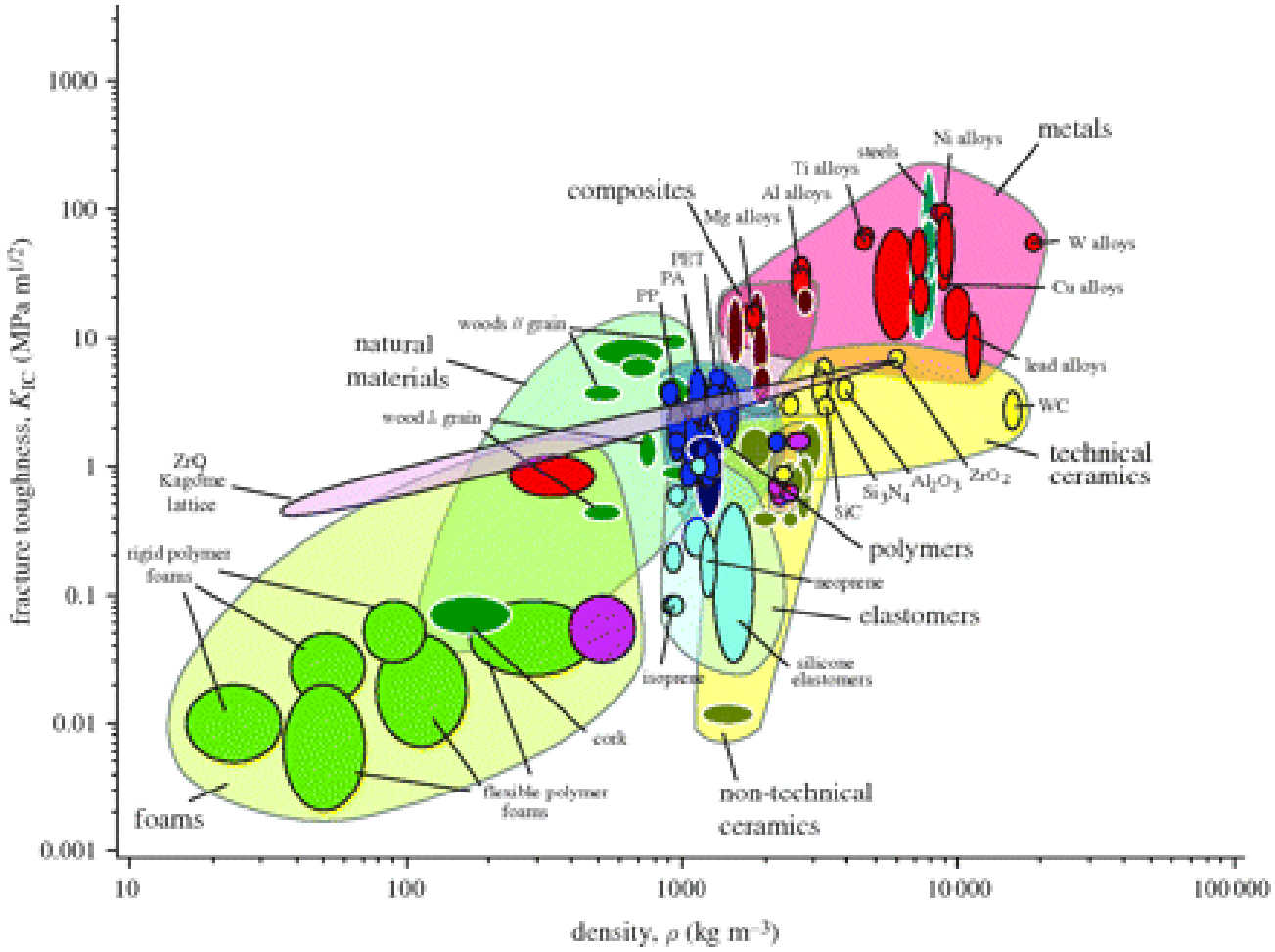
The toughness can be measured using a number of standard tests, with common tests including the Charpy Impact Test, which uses a notched piece of material (10mmx10mmx55mm) and a pendulum of known mass. Through varying the mass of the pendulum, the energy imparted on the material when the pendulum is released can be varied. The exact amount of energy imparted can be found through finding the height to which the pendulum

[158] (University of Cambridge, 2009)

returns, indicating the loss in gravitational potential energy. Hence, toughness is measured in J/m^3, showing the amount of energy one m^3 of material could absorb before fracturing. This is clearly the property of materials which would be tested if the product was dropped.

We can look at an Ashby Plot to determine which materials have high toughness's compared to density, a highly important property of any material chosen, as it will govern the weight of the product, therefore whether it would be easy to carry around or not.

Figure 66 Ashby Plot of Fracture Toughness vs Density



Source: University of Cambridge[159]

The Ashby Plot above shows Fracture Toughness vs Density. Fracture Toughness is in fact slightly different to absolute toughness, indicating the amount of stress required to 'propagate a pre-existing flaw'. This is however, highly comparably for the purposes of maximising how much energy the product can absorb before fracturing. This is also anyway important as any manufacturing technique would have the impact of producing imperfections such as cracks or voids, which can easily propagate.

The Plot shows that generally as density increases, so does fracture toughness, such that materials such as metals have very high fracture toughness's. It also identifies Composites and many polymers as above the trend showing higher toughness's compared to the expected value from their density. However, since the plot is also using a

[159] (University of Cambridge, 2009)

logarithmic scale it is also showing that the small difference shown between polymers and metals may correspond to a statistically large change in toughness.

In order to better understand this, we can use the raw data, available through the Cambridge website – which shows that while the toughness of metals is generally far higher than the toughness of polymers, there are some polymers which have relatively high toughness's, even near the values of toughness of Lead (the least tough of the metals). In particular PET has a very high toughness relative to many of the other plastics, while ABS, Nylons, PC and PEEK also have relatively high toughness's.

Figure 69 Fracture Toughness Statistics

13

II.5 FRACTURE TOUGHNESS (PLANE STRAIN), K_{IC}

		K_{IC} (MPa√m)
Metals		
Ferrous	Cast Irons	22 - 54
	High Carbon Steels	27 - 92
	Medium Carbon Steels	12 - 92
	Low Carbon Steels	41 - 82
	Low Alloy Steels	14 - 200
	Stainless Steels	62 - 280
Non-ferrous	Aluminium Alloys	22 - 35
	Copper Alloys	30 - 90
	Lead Alloys	5 - 15
	Magnesium Alloys	12 - 18
	Nickel Alloys	80 - 110
	Titanium Alloys	14 - 120
	Zinc Alloys	10 - 100
Ceramics		
Glasses	Borosilicate Glass	0.5 - 0.7
	Glass Ceramic	1.4 - 1.7
	Silica Glass	0.6 - 0.8
	Soda-Lime Glass	0.55 - 0.7
Porous	Brick	1 - 2
	Concrete, typical	0.35 - 0.45
	Stone	0.7 - 1.5
Technical	Alumina	3.3 - 4.8
	Aluminium Nitride	2.5 - 3.4
	Boron Carbide	2.5 - 3.5
	Silicon	0.83 - 0.94
	Silicon Carbide	2.5 - 5
	Silicon Nitride	4 - 6
	Tungsten Carbide	2 - 3.8
Composites		
Metal	Aluminium/Silicon Carbide	15 - 24
Polymer	CFRP	6.1 - 88
	GFRP	7 - 23
Natural		
	Bamboo	5 - 7
	Cork	0.05 - 0.1
	Leather	3 - 5
	Wood, typical (Longitudinal)	5 - 9
	Wood, typical (Transverse)	0.5 - 0.8

(Data courtesy of Granta Design Ltd)

		K_{IC} (MPa√m)
Polymers [1]		
Elastomer	Butyl Rubber	0.07 - 0.1
	EVA	0.5 - 0.7
	Isoprene (IR)	0.07 - 0.1
	Natural Rubber (NR)	0.15 - 0.25
	Neoprene (CR)	0.1 - 0.3
	Polyurethane Elastomers (elPU)	0.2 - 0.4
	Silicone Elastomers	0.03 - 0.5
Thermoplastic	ABS	1.19 - 4.30
	Cellulose Polymers (CA)	1 - 2.5
	Ionomer (I)	1.14 - 3.43
	Nylons (PA)	2.22 - 5.62
	Polycarbonate (PC)	2.1 - 4.60
	PEEK	2.73 - 4.30
	Polyethylene (PE)	1.44 - 1.72
	PET	4.5 - 5.5
	Acrylic (PMMA)	0.7 - 1.6
	Acetal (POM)	1.71 - 4.2
	Polypropylene (PP)	3 - 4.5
	Polystyrene (PS)	0.7 - 1.1
	Polyurethane Thermoplastics (tpPU)	1.84 - 4.97
	PVC	1.46 - 5.12
	Teflon (PTFE)	1.32 - 1.8
Thermoset	Epoxies	0.4 - 2.22
	Phenolics	0.79 - 1.21
	Polyester	1.09 - 1.70
Polymer Foams		
	Flexible Polymer Foam (VLD)	0.005 - 0.02
	Flexible Polymer Foam (LD)	0.015 - 0.05
	Flexible Polymer Foam (MD)	0.03 - 0.09
	Rigid Polymer Foam (LD)	0.002 - 0.02
	Rigid Polymer Foam (MD)	0.007 - 0.049
	Rigid Polymer Foam (HD)	0.024 - 0.091

[1] For full names and acronyms of polymers – see Section V.

Note: K_{IC} only valid for conditions of linear elastic fracture mechanics (see I. Formulae & Definitions). Plane Strain Toughness, G_{IC}, may be estimated from $K_{IC}^2 = EG_{IC}/(1-\nu^2) \approx EG_{IC}$ (as $\nu^2 \approx 0.1$).

Source: Cambridge University Engineering Department[160]

Though it would be very challenging to find a required the required Fracture Toughness due to the number of variables which are there, such as the quality and choice of manufacturing technique, we can look at the raw figures derived from tests such as the Charpy Test which returns a value of 28 kJ/m^2 (according to Chapman and Hall in 1979). The Charpy Test is unique in that we can use its value quantitatively to find whether the absorbed energy is sufficient for our material due to the very small sample size it makes use of. Hence, we can find a minimum value which we would need for our product, assuming some worst-case values. However, these are generally less common and less accurate than the calculation of Kic, which though numerical is very hard to interpret to finding mathematically whether a material would survive upon an impact, instead only really being useful to show the relative strength of materials against one another.

[160] (Cambridge University Engineering Department, 2003)

Let us assume the product weighs 1kg (rather higher than we would expect it to), has an exposed surface area of 0.001m^2 (falling on a medium sized face) and fell from 2m.

The GPE which would need to be absorbed would be:

$$Gravitational\ Potential\ Energy = mgh$$

$$= 1 \times 9.81 \times 2$$

$$= 19.62J$$

Hence, assuming the 0.001m^2 exposed area, the required toughness would be:

$$\frac{GPE}{Surface\ Area} = \frac{19.62}{0.001} = 19.62kJm^{-2}$$

Despite taking worst case scenarios, we reached a solution that was far below the measured toughness of even ABS. Therefore, it is clear that the majority of the tough polymers would easily satisfy the restrictions here, while metals would offer unnecessary additional toughness for large sacrifices in higher densities, which would lead to much higher masses of the product, making it harder for the elderly to carry them.

Figure 70 Toughness, Density and Cost table of Polymers

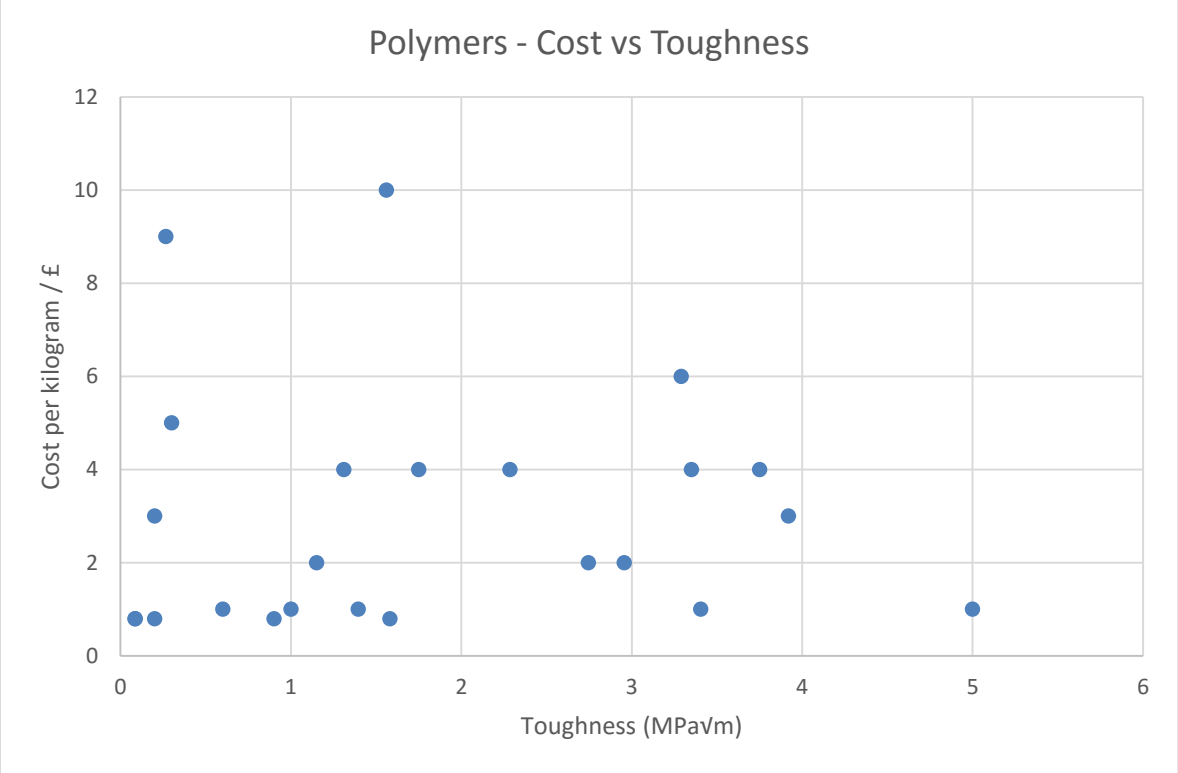
	Average Fracture Toughness / MPa√m	Density Average / gcm^(-3)	Cost per Kilogram / £
Butyl Rubber	0.085	0.91	0.8
EVA	0.6	0.95	1
Isoprene	0.085	0.935	0.8
Natural Rubber	0.2	0.925	0.8
Neoprene	0.2	1.24	3
eIPU	0.3	1.135	5
Silicon Elastemers	0.265	1.55	9
ABS	2.745	1.11	2
CA	1.75	1.14	4
Ionemers	2.285	0.945	4
Nylons	3.92	1.13	3
PC	3.35	1.175	4
PEEK	3.515	1.31	90
PE	1.58	0.9495	0.8
PET	5	1.345	1
Acrylic	1.15	1.19	2
Acetal	2.955	1.41	2
PP	3.75	0.9	4
PS	0.9	1.045	0.8
tpPU	3.405	1.18	1
PVC	3.29	1.44	6
PTFE	1.56	2.17	10
Epoxies	1.31	1.255	4
Phenolics	1	1.225	1
Polyester	1.395	1.22	1

Source: Statistics from Cambridge University Engineering Department[161]

[161] (Cambridge University Engineering Department, 2003)

In addition, we must also consider the cost of the material, with this table showing the costs per kilogram, average density and average toughness according to information from the Cambridge Materials Data Book. It shows that despite PEEK showing relatively high toughness and relatively low densities, the very high cost of it means it would be infeasible to use it. Excluding PEEK, graphing the toughness vs the cost, we find one exceptional outlier, PET, which has a very low cost, yet a very high toughness.

Figure 71 Graph showing Cost vs Toughness of Polymers



Source: Statistics from Cambridge University Engineering Department[162]

Hence, while other materials would also definitely be usable such as ABS, PET would be the best option according to its fracture toughness, meaning it would be the least likely to break if it were dropped.

Conclusion

From both the analysis of stresses and of toughness, it is clear that the best material would likely come from the polymers section as they provide the ideal material characteristics. Additionally, they offer a number of possible manufacturing methods which could be used. In particular, the toughness investigation found that PET might be a particularly effective material choice, though we must also consider the manufacturing possibilities, ensuring that it would be possible to manufacture the product easily.

5.3.3 Geometry

Target Market Research

To show what geometries appear to be successful with products for the elderly, it is important to investigate a few products specifically made for the elderly, and look at why they are successful and how this product could learn from them. Among the relevant products, it is most pertinent to look at handheld ones, and specifically at the manner in which they deal with the issues associated with elderly and how they attempt to make the product ergonomic.

[162] (Cambridge University Engineering Department, 2003)

Figure 72 Soft Pencil Grips

Source: Posturite[163]

These are intended to surround pens and pencils providing a larger and more ergonomic surface which the elderly can hold onto. The addition also has indentations on two sides, which mean that they fit the thumb and finger more comfortably, providing support and increasing the surface area with which they can grip the pencil.

Successes: Having personal experience using similar products (though I used one intended for young children) these were successful in making using a pencil more comfortable, with a soft touch polyurethane material meaning that the person could easily grip the pencil.

Failures: I found that the grips were rather specific in terms of finger size and if people had too large or too small fingers the pencil would be useless. This was because they only produced one size. Additionally, some people on the internet have expressed a concern that they don't always fit on the pencil well, instead sliding off it too easily.

AARP Real Pad

Figure 73 AARP Real Pad

Source: AARP[164]

[163] (Posturite, n.d.)

The AARP Real Pad was a tablet designed by the AARP (American Association of Retired Persons) in association with Intel as they attempted to produce a product which would be more effective for elderly people. It has a number of features of the user interface to ensure that the product is earlier for the elderly to use, including large graphics, large interfaces and preloaded useful apps. They have also designed a novel RealHelp suite which provides step-by-step videos to solve any problems that you might be having, as well as providing 24x7 help over the phone. Finally, the product is designed to be lightweight and a small as possible, with a reduced bezel size ensuring that the product would be easy to hold. Though there are no obvious ergonomic features, the manufacturers have gone to a lot of trouble to ensure that all edges are highly filleted allowing a curved surface that an elderly person can hold onto.

Advantages: The product has a number of software features which enhances usability for the target audience, large graphics and provide better help facilities. The product is relative lightweight and is made to be as small and thin as possible without compromising usability.

Disadvantages: The product does not offer any easy manner to hold onto the product, in fact with a slippery curved casing throughout the product. The product could easily make use of indents to make it more easy to grip. The fact that the RealPad has discussed that it should withstand a drop implies that they have failed to design the product to be gripped effectively. Additionally, the product, though claiming to be lightweight is much heavier than many of its rivals. In fact, the iPad Mini, with a slightly larger screen (7.9” vs 7.8”) weighs 10 ounces, almost 30% less than the RealPad.

BIGtrack Mouse

Figure 74 BIGtrack Mouse

Source: Amazon[165]

This mouse appears to provide many advantages for elderly with fine motor skills issues or with poor vision by no longer needing the person to move a mouse around, instead moving a tracking ball, which requires far less force, in fact only needing the movement of one finger. Additionally, the product has particularly large buttons, meaning that people would have an increased accuracy when clicking, with fewer mistaken clicks.

Advantages: The product makes use of lots of bright colours to allow people with vision problems to use the mouse easily. Also, it keeps a large gap between buttons to press, ensuring the person does not mistakenly click the wrong button.

Disadvantages: The product is very bulky, taking up a lot of space.

Learning from other products:

[164] (AARP RealPad, n.d.)
[165] (Amazon, n.d.)

Each of the products appear to cater to different problems the elderly face, the first attempting to counter griping problems by the use of finger inserts and grooves to provide greater surface area. The second product uses rounded edges which may not be too effective as a mechanism for holding the product, while the last one tackles the problems related to fine motor skills, making use of bright colours and separation between buttons.

We can easily learn from all three, utilising the idea of grooves for better grip, as well as bright colours and separation of buttons for people with challenges related to vision and fine motor skills. Finally, the usage of curves on the RealPad shows how these are clearly preferable to straight edges and so curves should be added wherever possible.

Looking at products designed for ergonomics

In addition to looking at products intended for the elderly, and learning from this, it is also important to look at other handheld products which have been particularly successful in being very comfortable to hold, and hence have excelled in their ergonomics. From here, we can investigate a number of methods of ensuring the product is easy to hold, including the use of handles, grooves etc.

Powergrip Utility Knife

Figure 75 Powergrip Utility Knife

Source: Amazon[166]

Features: The bend in the product as well as the bottom loop clearly provides a large area to grip the product, hence ensuring a good grip for cutting, as well as ensuring that fingers are unlikely to be in a position to be injured during cutting. Particularly interesting is the solid top bulge which offers some resistance to the grip allowing the person with average hand size to loop their hand entirely around the knife for complete safety.

Berkley Knife and Scissor Sharpener

Figure 76 Berkely Knife and Scissor Sharpener

Source: Berkely-Fishing[167]

[166] (Amazon, n.d.)

Features: This has a number of the features of the utility knife also introducing the grooves in the face to allow the fingers to fit comfortably inside. Furthermore, they have a curve of the fingers with the middle finger having the highest groove and the other fingers having lower grooves to attempt to keep the fingers in a natural position. Additionally, the product has a small groove in the top face for the thumb to move around and sit in place away from the blade of the sharpener.

Quentons 8" Ceramic Chef Knife

Figure 77 Quentons 8" Ceramic Chef Knife

Source: Quentons[168]

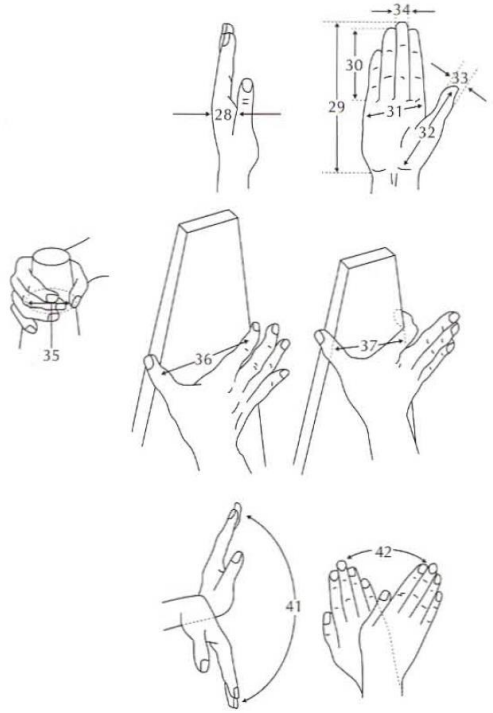
Features: As opposed to the Berkley Sharpener this knife does not have grooves, instead just having the curve to help the finger to optimally grip (since the middle fingers are longer than other ones).

Anthropometric Research

In order to effectively produce a product which would be comfortable for an elderly person to hold in their hands, we must complete some anthropometric research to look at the sizes of people's hands and indeed how this changes over time. This would almost certainly be necessary whether handles, grooves or a curve (the three manners shown above) were to be used. From our original research, we know that though grip does degrade over time, it is often the best characteristic to rely upon, due to further degradations in fine motor skills. In 'The Aging Hand' Eli Carmeli[169] precisely describes the various effects of aging to the hand, showing how it largely affects the muscles in the hands and hence generally does not have much of an impact in the exact dimensions of the hand itself – since the muscles of the hand are largely in the wrist. In 'Hand Anthropometry: A Descriptive Analysis on Elderly Malaysians'[170], the authors completed a large scale survey, looking at the sizes of people's hands, as compared to the measurements taken of adults by Wickens in 1998.

[167] (Berkley Fishing, n.d.)
[168] (Quentons, n.d.)
[169] (Eli Carmeli, 2013)
[170] (Poh Kiat Ng, 2014)

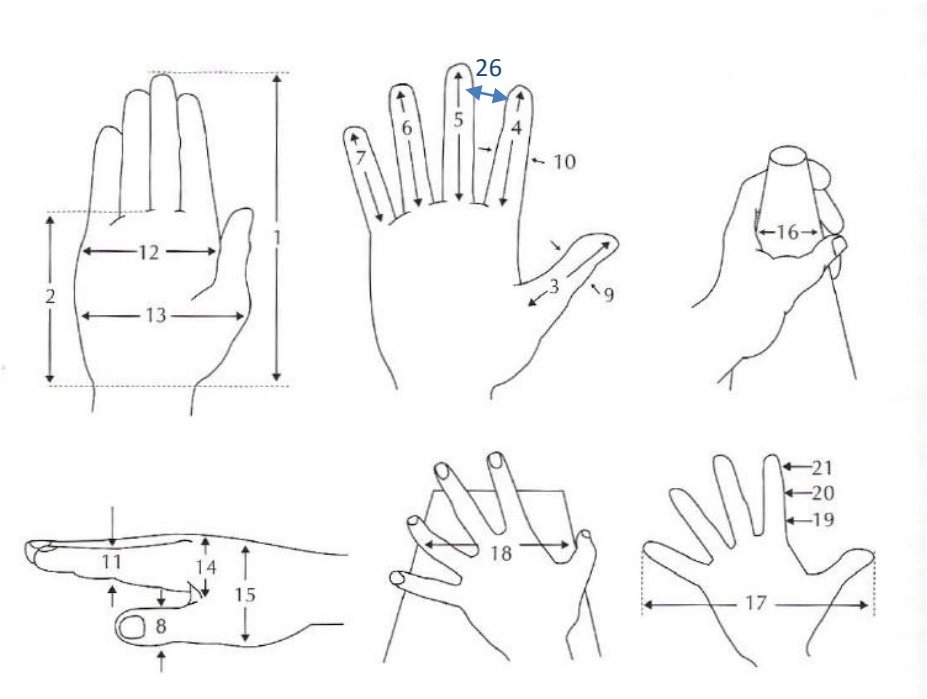
Figure 78 Measurements of the Hand Taken in "Hand Anthropometry: A Descriptive Analysis on Elderly Malaysians"



Source: "Hand Anthropometry: A Descriptive Analysis on Elderly Malaysians"[171]

Hence, they took a sample of 250 elderly individuals (half male and half female) taking measurements of all their hand features and then tabulating them.

Figure 79 Hand measurements taken



Source: "Hand Anthropometry: A Descriptive Analysis on Elderly Malaysians"[172]

[171] (Poh Kiat Ng, 2014)

Figure 80 Various different hand measurements

No.	Hand Measurements	Mean	Max	Min	Range	Std
1	Hand length	17.44	22.30	14.00	8.30	1.51
2	Palm length	10.28	13.50	8.20	5.30	1.03
3	Thumb length	5.79	7.00	4.30	2.70	0.58
4	Index finger length	6.48	8.60	4.90	3.70	0.58
5	Middle finger length	7.15	8.80	5.80	3.00	0.61
6	Ring finger length	6.45	7.50	5.10	2.40	0.52
7	Little finger length	5.46	6.70	4.30	2.40	0.51
8	Thumb breadth	1.95	2.70	1.10	1.60	0.31
9	Thumb thickness	1.70	2.30	0.90	1.40	0.29
10	Index finger breadth	1.69	2.30	0.70	1.60	0.35
11	Index finger thickness	1.59	2.12	0.70	1.42	0.35
12	Hand breadth (metacarpal)	8.99	11.50	6.70	4.80	1.13
13	Hand breadth (across thumb)	10.83	13.60	8.10	5.50	1.20
14	Hand thickness (metacarpal)	2.54	3.60	1.50	2.10	0.54
15	Hand thickness (including thumb)	3.81	5.80	2.20	3.60	1.00
16	Maximum grip diameter	4.06	5.00	3.00	2.00	0.40
17	Maximum spread	16.65	25.00	11.40	13.60	3.25
18	Maximum functional spread	13.87	18.80	9.10	9.70	2.69
19	Proximal phalanx (thumb)	2.88	3.80	2.00	1.80	0.41
20	Distal phalanx (thumb)	2.89	4.10	2.10	2.00	0.37
21	Proximal phalanx (index finger)	2.38	3.20	1.50	1.70	0.39
22	Middle phalanx (index finger)	1.87	2.70	1.10	1.60	0.38
23	Distal phalanx (index finger)	2.24	3.10	1.60	1.50	0.30
24	Proximal phalanx (middle finger)	2.54	3.30	2.00	1.30	0.30
25	Middle phalanx (middle finger)	2.24	2.80	1.40	1.40	0.28

Source: "Hand Anthropometry: A Descriptive Analysis on Elderly Malaysians"[173]

This table offers a large amount of information which could be used in order to produce an ideal product which would take into account the size of the hands. In addition, if as many ergonomic products have shown, I should make use of grooves, then this data of specifically elderly people would be very useful in showing the sizes of fingers hence allowing me to produce correctly sized grooves, to best fit the vast majority of the population. In fact, the only piece of missing information is the distance between the fingers when at rest (labelled 26 on Figure 79), requiring me to measure this myself. Under the assumption that this did not change with age, I measured this piece of information in myself and a few other people, including members of my family and friends and found this:

[172] (Poh Kiat Ng, 2014)
[173] (Poh Kiat Ng, 2014)

Figure 81 Additional measurement results

Measurement	Number of people	Mean /mm	Maximum /mm	Minimum /mm	Range /mm	Standard deviation /mm
Between index finger and middle finger at rest – marked 26 on the diagram.	6	8.00	9.00	7.00	2.00	0.65

The low standard deviation, despite the large range of ages and the presence of both genders appears to show that this value is relatively reliable and hence could be used when designing the product as the distance between fingers.

Conclusion

In order to make the product as ergonomic as possible, it appears there are a few tactics, using grooves, handles and curves. Each appears to provide a specific amount of comfort and support, while also each having a drawback, including amount of material needed and how much it would expand the size of the product. Throughout the generation of ideas for possible geometries, the problem of the RealPad remained in my mind, with the main aim of reducing the size of the product appearing to take away from the ability to add ergonomic features.

5.4 Software Design Research

5.4.1 Programming Languages Choices

For programming the majority of the product there were a number of programming languages that could be used, however, the native support of Python (in the Raspberry Pi) as well as the existence of libraries like the PiCamera library as well as PyCrypto (which provided inbuilt support to Advanced Encryption Standard, which would allow me to encrypt chunks of data simply by passing them through the function) meant that it was the obvious choice. Though I also considered C, using WiringPi for GPIO support, which contained a similar cryptographic library in the end, the lack of libraries for Camera management and QR decoding meant that the task would be more challenging.

For the mobile apps, though I could have used Java and Cordova for the Android app and Swift for the iOS app, this appeared unnecessary and useless since the app was very simple and ought to be identical for both devices. Hence, I chose to use Xamarin.Forms[176], a method of app development which allows you to do simple Storyboarding for both apps together as well as doing large parts of the backend in C#, hence saving a lot of time.

Finally, for the Web-Apps I chose to use HTML and CSS to produce the website, while making use of JavaScript for the QR code production. This was a rather simple decision, as basic website development (without resorting to PHP) allows you to do little else. Additionally, in order to save time, as well as it resulting in perfect results, I chose to make use of the Google Charts API to produce the QR codes, by its return to a specific GET request in the JavaScript.

5.4.2 Encryption for transmission

While the security risk of the QR code production and transfer by taking the picture was very small as it would occur directly between the device producing the QR code and the product, not involving any networks, it was clearly still important to lightly encrypt it to ensure that someone stumbling upon the QR code on a computer would not have

[176] (Xamarin, n.d.)

access to one account because it said 'YAHOO ASHWINYAHOO ASHWIN'SYAHOOPASSWORD' when the person scanned the QR through a QR scanner on their phone.

There were a couple of options for this encryption. Though I could again have made of use of the Advanced Encryption Standard, this would have required a lot of computing power on the phones and might have taken a while. Additionally, especially for the apps, where there were no existing AES APIs for either iOS or Android, this would have had to be coded for by hand, and this would have been very tedious and likely very challenging. Hence, I chose to look at other possible ciphers that could have been used, especially simple ciphers, since these would easily serve our purposes here. However, I found clearly that the simplest ciphers were fatally flawed and hence I chose to design my own simple cipher that would meet the needs such as these, where a secure simple cipher is required.

5.4.3 Encryption for passwords

For the encryption for the passwords, the most important decision to be made was how to get the key, which would determine the security of the passwords. If someone gained access to the key, then they could decrypt the passwords themselves. Hence, there were a couple of options for this key.

Firstly, there could have been a master key, which would decrypt the passwords for every device, while only being known by the people designing the product, therefore not allowing hackers to find it out. However, this would have to be coded into the code of the product and hence, the hackers could not gain access to this code. However, in the case of the Raspberry Pi, this would be impossible as they could simply open the SD Card and look at the .py file inside which would contain the code that the product is running.

Secondly, there could be a specific key for each device, entirely unrelated to the person who is using it. Instead there would be a number, specific to each device, which must be predetermined and this will be the key to the product's passwords. While being slightly safer in that the programmer no longer has access to all passwords, the safety is still reliant on the safety of the files inside the Pi, since this key must be stored inside the code, or indeed another file which could be accessed by the hacker.

Finally, the last option would be to use the specific fingerprint data as the key for the encryption to the passwords. By each time reading the exact data (392 kB in length) from the fingerprint when the person using the product has been verified, the key could be securely stored on the fingerprint sensor which could be automatically erased when the product is being opened, or when the supercapacitor cuts on (indicating the power being cut or running out). However, one last matter must be considered, since the length of fingerprint data is far longer than the amount of data required for a key for AES encryption, which has a maximum fixed length of 256 bits. Therefore, this must be passed through a hashing algorithm to shorten it. There are a few possible hashing algorithms which could be used:

Hashing algorithms are functions which allow you to map data of arbitrary size to data of a fixed size, without being able to return to the original data.

SHA256 is by far the most secure hashing algorithm – though it only produces a 256 bit output (this is perfect for my requirements). However, since it is the most secure, containing the most complex algorithm it also takes the most time, creating speeds of around 111MB/s, around a tenth the speed of MD6.

Whirpool is another hashing algorithm which can produce an output of any length by altering the algorithm though it is largely used to output a 512-bit word. It is vastly less secure than SHA256, since produces a large number of collisions (where one input goes to multiple outputs), however it is very fast.

MD6 was an algorithm designed precisely for speed, used specifically for hashes of very long inputs, claiming it can complete hashing using 28 clock cycles per inputted byte, while still maintaining cryptographic resistance, with speeds of over 1GB/s having been measured. Over the years a number of issues have been found in the

implementations of MD6, however, these have been quickly resolved. It is generally thought to be about as secure as SH1 – the predecessor to SHA2 (of which SHA256 is a member).

Despite SHA256 clearly being much slower than MD6 this appears to be irrelevant given the relatively small file which is being dealt with (392KB). At 111MB/s this should take 0.0035 seconds, a rather tiny amount. Therefore, the additional security it provides is useful as well as it producing fixed lengths of 256 bit outputs being extremely helpful. This makes it much simpler to use, since the output from the hashing algorithm could be used straight as the key for the AES encryption of the passwords.

6.0 Development

6.1 Control System Testing and Development

Throughout testing, I followed the idea of testing each component and subsystem separately before combining them, as this helped to find any possible problems early and made sure they could be remedied as they occurred.

6.1.1 Microcontroller Issues

The first issue that was encountered when testing and developing the system was the lack of availability of the Pi Zero, since it was backordered and so would not be received until the beginning of 2016. Therefore, as a very similar product, though slightly more powerful, the Raspberry Pi 2 was used. However, this created some problems with camera system integration which will be discussed later.

6.1.2 Fingerprint Sensor

In testing the fingerprint sensor, a number of problems were encountered, especially due to the oddities in the manner in which the module communicated with the Pi. However, once the basic communication was established, there was a process of producing the various functions required. Though I had found a third party Python library available on the internet, during the testing I found that the library was not functional. This still proved useful for investigating the various structures that I needed to produce, correcting the large number of errors as I produced my library. The Arduino library produced by the manufacturer also proved useful, as I was able to transpose it in part to Python. However, majorly, I looked at the datasheet for the module, producing the most efficient library including the various functions I needed and excluding the functions which would be unnecessary.

The first function added was getting the backlight on and off, as this was relatively simple as well offering a tangible method of showing that the Fingerprint Sensor was working. From here, the functions to test whether there was a finger on the fingerprint sensor as well as the capture of a finger was written. This allowed an enrol function (to add new fingers to the database of the module) to be written, which monitored the success of three enrolment functions (each of which required separate commands to be sent to the sensor) before reporting the success of the entire enrolment. From here, after enrolling a couple of fingers, I produced a function to compare the finger being placed on the fingerprint sensor with those available on the module's database to gain entry to the device. The final function I needed was the GetTemplate function, which would return the raw data of the ID requested, hence allowing this to be used to lock and unlock the encrypted file, as decided during the previous section. This appeared to pose quite a large challenge, as it required the Pi to download this template in four goes, each time taking 98 bytes, finally needing the Pi to concatenate the four to get the single 392-byte template. Additionally, I had a number of problems due to latency in the module transmission, since it would take time for the data to be sent to be collated.

6.1.3 Camera

Before the issues related to the camera interacting with the microcontroller, the entire system was made to work independently. Though I had originally hoped to get the system working with OpenCV, continuously scanning for a QR code in the camera view, I failed to get this working, so I attempted to continuously take a picture attempt to find a QR code, and override the old picture with a new picture if it failed until it was able. This was much simpler as it allowed me to use the very simple QRTools library as well as the well documented PiCamera one. This appeared very successful managing to easily scan QR codes from a large variety of distances, even managing a 1KB barcode from nearby with comfort, far better than I had expected. This success meant that I could revisit the idea of transmitting multiple passwords in one go.

6.1.4 Change of Camera

After testing the system with the Raspberry Pi 2 and looking into difference between the Pi 2 and the Pi Zero, an error was found with the early decision of choosing a Pi Camera. Apparently, a number of websites had been misleading in saying that the camera was compatible with all new Raspberry Pi boards. In fact, the Pi Zero, in its attempts to make it as small as possible, had forgone the port.

Hence, another camera was required to be found, and hence revisit the decision matrix. Thankfully all options use the USB port which is definitely available on the Raspberry Pi Zero. Additionally, another option was discovered, the GPIO camera will also be investigated.

Logitech C270 HD Webcam – 3MP, £15

Advantages: It is cheaper than the Raspberry Pi camera, hence allows me to more easily meet my price specification. Additionally, as opposed to the Chinese cameras (the next two) they are well supported by Logitech, and would be reliable. Finally, the camera also implements autofocus, which means that the QR code could be held at a greater distance and the system would be able to deal with it.

Disadvantages: Low quality sensor means that the camera might struggle with QR codes with large amounts of data. Additionally, since the camera connects over USB, the very well supported PiCamera library could no longer be used, however, there are a number of other libraries such as fswebcam, which could be used.

5MP Micro Webcam - £12

Advantages: On paper, this camera offers the greatest quality to price, offering a very low cost for a decent 5MP sensor. Additionally, the sensor even comes with autofocus.

Disadvantages: The camera has to be bought from a reliable Chinese manufacturer which a number of users appear to have had many problems in the past. In fact, when I went to the Chinese website, and looked at some of the comments, (after translating them) I found that a number of people had had problems with the reliability of the camera, one saying it 'worked once, then not again'. Additionally, as opposed to the PiCamera, the camera plugs in over USB, and has similar irritations as described for the Logitech C270 HD

8mp ov5640 mini hd cmos senor (sic) webcam camera module - £30

Advantages: The camera is of the best quality and so should be easily able to deal with the QR code scanning. Additionally, the camera again contains autofocus built in.

Disadvantages: The camera again must be purchased from AliExpress – hence bringing the same reliability issues as for the 5MP micro webcam. In fact, here, there was a few complaints about the quality of the camera, with a couple of users complaining that the sensor had a much worse quality than they had expected.

CMOS Camera Module – 722 x 488, £22

Advantages: The camera is largely well supported, with a number of blog discussions on how to use it. However, from the discussions it appears clear that this is necessary because it's usage is rather complex with no nice library such as 'fswebcam' which could be used.

Disadvantages: The resolution is very low – 722x488 = total of 352,336 pixels = 0.352 Megapixels. From my testing using a 2MP camera, it was relatively challenging to scan a QR code, hence with a camera of such a low resolution, it would be very challenging.

In order to make this decision, I made use of the same decision matrix that was originally used to decide the camera with the information updated.

Figure 82 Decision Matrix for Camera

	Quality of Camera		Ease of Use		Reliability		Cost		Additional Features		
	Weight	Mark	Weight	Mark	Weight	Mark	Weight	Mark	Weight	Mark	**TOTAL**
CMOS Camera	8	1	8	4	9	8	8	3	5	1	141
Logitech C270HD	8	3	8	7	9	7	8	5	5	4	203
5MP Micro Webcam	8	6	8	7	9	2	8	9	5	4	214
8MP OV5640	8	9	8	7	9	2	8	7	5	4	222

Despite the additional cost and possible lack or reliability due to the product coming from an unknown Chinese manufacturer, the 8MP camera provided a very high quality (in theory) image, hence it was selected.

6.1.5 Change of Charger IC

After more investigation of the MAX1555, which was originally selected as the Charger IC, I found that it only charged one or two cell Li-Ion batteries. This necessitated a switch to an IC which could deal with multiple cell batteries, such as a 6 cell Lithium Ion one. This yielded very few results, and the only one which was sufficiently documented, was the MAX745EAP+-ND. This appeared to do everything the MAX1555 as well as meeting the criteria of being able to charge multiple cells. However, it is much more complex to use, and hence, required a lot of further investigation to help to implement it in the electronics, with two power MOSFETs and a number of Schottsky Diodes being required to get the chip working. Additionally, it uses a TSOP package, which would be much harder to solder by hand, making the possibility of producing a complete prototype much lower. Finally, it was much more expensive, with a cost price of £5 per unit, though the cost price would reduce rather quickly if the quantities were to be increased.

6.1.7 Rotary Encoder

Despite using the Arduino example sketches and electronics directly into Python and onto the Pi, when I first began to test the Rotary Encoder, the Pi failed to recognise the encoder. However, when I realised the product had a common anode as opposed to a common cathode (it appears the guide got it wrong) I was able to fix the problem and the Pi began to recognise the rotation of the Rotary Encoder. However, even then the Rotary Encoder only seemed to recognise the movement in one direction, assuming it went in that direction even if the Rotary Encoder were turned the other way. After constantly checking the code and checking it on an Arduino to check if the Raspberry Pi was the problem, I assumed that it was a broken Rotary Encoder and I tried the spare that I had acquired, this time finding it working perfectly. In the end I found (when testing the broken one) that there was a short between two of the encoder pins (inside the encoder) which meant that the computer would only register it moving one direction.

6.1.8 New Cipher Design

In completing the cipher design, a lot of time was spent considering what would make a cipher stronger than any other. I studied the work of Claude Shannon, and his various ideas for unbreakable ciphers, culminating in the theory

behind the One-Time Pad. Ultimately, I decided on a cipher based on the simple idea of the Polyalphabetic Shift Cipher where a specific key would shift the item to be encrypted along a specific amount. However, I had created a new innovative idea which would change the length of the key, ensuring that a basic frequency analysis on a specific length of characters would yield no results.

The cipher would work hence:
Assume the item to be encrypted is: HELLOMYNAMEISASHWIN
Assume the single key is: 123456789

In the conventional system this would occur adding 1, 2, 3… to the numeric values of the letters of 'H', 'E' 'L',... hence creating IGOP… Once the system reaches the 10th character of the input to be encrypted, it starts going through the key again, starting 123456789. Herein is the flaw in the system, as someone who carried out a frequency analysis on the encrypted text, looking at the characters every 9 places would find a shifted English language frequency analysis graph, hence allowing them to easily solve each of these 9 shifts, hence finding the complete key and being able to find the input.

In my system however, after every 9 goes, the key would shorten, removing the first character of the key, making it 23456789, then 3456789 etc. For a key originally of length 9, this would mean there would be a new key of 9+8+7+6+5+4+3+2+1 = 39 numbers long. This means for the issues associated with transferring a 9-digit key instead of a longer one, a 39 number key has been found. (After this length, the key would then wrap around). This means that frequency analysis would have to be carried out every 39 characters for the hacker to find anything. While for most things this would not be particularly useful, this is very good for the password manager, as the total data (in terms of number of characters) to be transferred would rarely be very long, in fact 39 characters being among the longest the Account Name + Username + Password is likely to be. Hence, it serves the purposes effectively.

However, as I began to test the new cipher, and saw it successfully pass the hacking test which I completed, by attempting to write a program to hack the cipher, I learnt that the success of the cipher relied upon the security of the design of the cipher, since someone who had gained the knowledge of how the cipher worked could then easily perform frequency analysis on the specific characters which would use the same number of the key. Hence I improved it, so that the key would also change as it went through the encryption process, meaning that it would be impossible to complete a frequency analysis, even given knowledge of the system by which the encrypted text was made.

This involves using the result of the previous encrypted character, using the numerical value of that in addition to the key character to produce the encrypted key. Hence, it would work like this:

Key = 123456789

Item to be encrypted = HELLOMYNAMEISASHWINANDTHISISANEWCIPHER

	H	E	L	L	O	M	Y	N	A	M	E	I	S	A	S	H	W	I	N	A	N	D	T	H	I	S	I	S	A	N	E	W	C	I	P	H	E	R
1	8	5	1	1	1	1	2	1	1	1	5	9	1	1	1	8	2	9	1	1	1	4	2	8	9	1	9	1	1	1	5	2	3	9	1	8	5	1
			2	2	5	3	5	4		3			9		9		3		4		4		0			9		9		4		3			6			8
2		8	5	1	1	1	1	1	1	1	1	5	9	1	1	1	8	2	9	1	1	1	4	2	8	9	1	9	1	1	1	5	2	3	9	1	8	5	
				2	2	5	3	5	4		3			9		9		3		4		4		0			9		9		4		3			6			
3	1	2	3	4	5	6	7	8	9	2	3	4	5	6	7	8	9	3	4	5	6	7	8	9	4	5	6	7	8	9	5	6	7	8	9	6	7	8	
4	9	1	2	2	3	3	4	3	2	1	2	2	3	2	2	3	5	4	2	2	2	2	3	3	2	3	3	3	2	2	2	3	3	2	3	3	2	3	
		5	0	8	2	4	5	7	4	6	1	8	3	6	7	5	0	5	7	0	1	5	2	7	1	3	4	5	8	4	4	4	3	0	4	0	0	1	
5	9	1	2	2	6	8	1	1	2	1	2	2	7	2	1	9	2	1	1	2	2	2	6	1	2	7	8	9	2	2	2	8	7	2	8	4	2	5	
		5	0				9	1	4	6	1			6			4	9		0	1	5		1	1					4	4			0			0		
6	I	O	T	B	F	H	S	K	X	P	U	B	G	Z	A	I	X	S	A	T	U	Y	F	K	U	G	H	I	B	X	X	H	G	T	H	D	T	E

In 1), the numerical values for each of the characters are found. In 2), these values are transferred one along onto the next character as this will be part of the value added to get the encrypted value. In 3), the values for the key are added, one by one, removing the first value every time we finish the current key. In 4) the numbers are added together to find the numeric value for the encrypted equivalent to that letter. In order to find the actual numerical equivalent however, some of them require being 26 subtracted from it. In fact, they would require being modularly divided by 26, as the value could be over 52. In 6) the number is converted into the final encrypted character which will be transferred to the device.

In this case the new cipher design creates a system which forces any hacker to attempt to move through the hacking one character after another, as part of the information of the previous unencrypted character will provide information about the value of the next character, or to brute force through the possible keys (10^9 in this case) which will require a lot of computing power and take a considerable period of time.

Now that the cipher had been decided there were couple of practical considerations for how the product will provide certain information to the user. Firstly, the cipher requires a key which would be known by the product and the computer of phone producing the QR code and no one else could know it. In this case the key was 9 characters, but the greater the length, the greater the strength, in case of particularly long transmissions being required. Hence, a Serial Number must be added to the software, which could be viewed by the users going to settings, and looking at the value listed under the Serial Number section. Though I had considered engraving the Serial Number on the product, I changed my mind to make it more secure since if the user had left a QR on a screen when transferring a password, they were likely to have left the product in the vicinity, and this would then be possible for the person spotting both to break, and gain access to the account that was being transferred to the device.

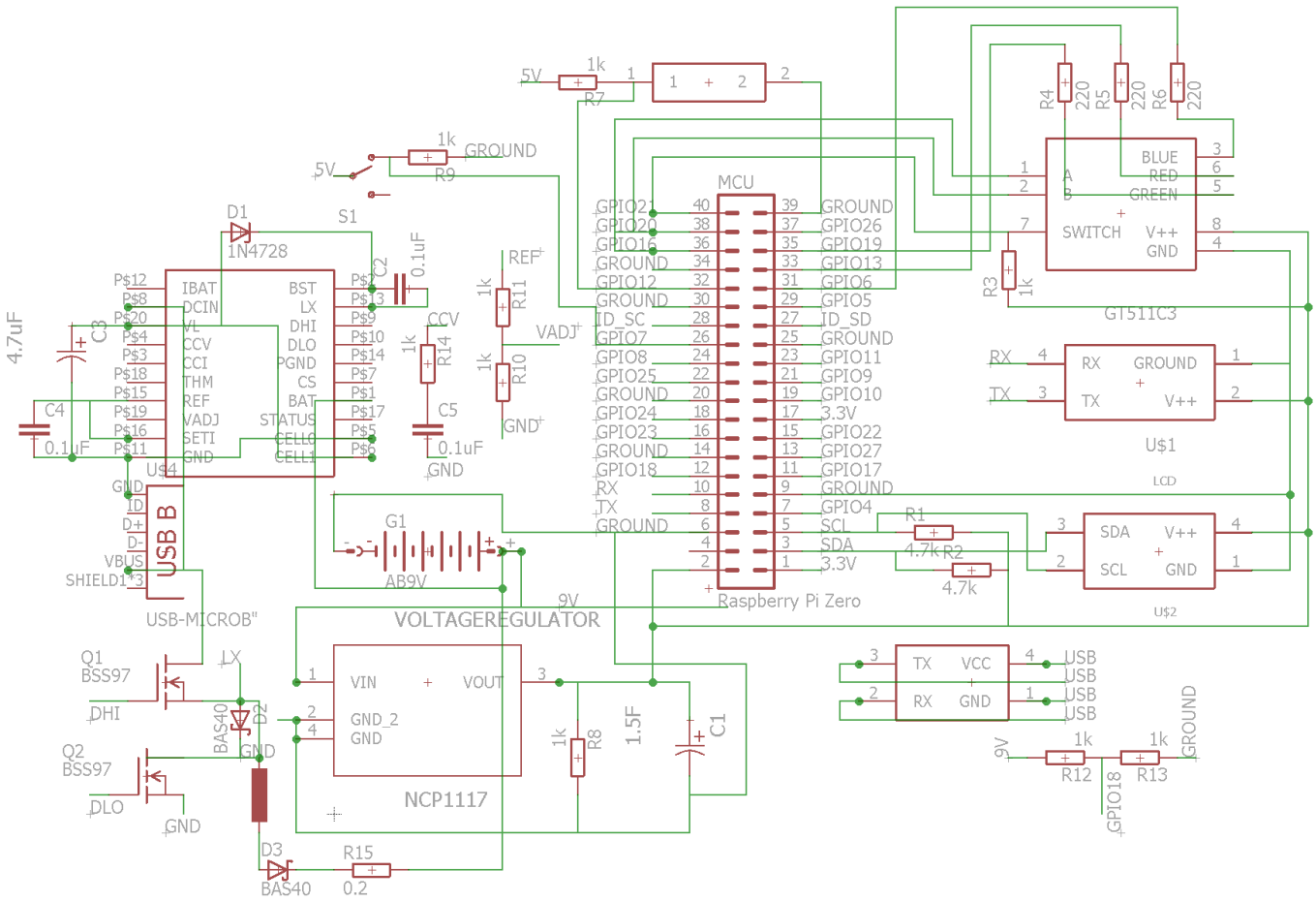
Though this would be rather irritating for the user as they would have to look up the Serial Number every time to create a transfer QR code on the web-app or the mobile app, I found that in fact this could be partially simplified by the use of storing the Serial Number as a cookie in the Web-App and in storage on the iPhone and Android devices. However, I soon realised that someone with access to a device could easily find the Serial Number even if they did not have direct access as with known inputs and a known output (they could read the QR code with any Smartphone to find the raw encrypted value) they could find the key. Hence, I moved back to a system which would require the user to enter the Serial Number of the product, every time they wished to add a new password.

6.1.9 Conclusion

During the initial testing and development, there were a number of setbacks as I encountered problems with components failing to work, however through perseverance and large amounts of research into the specifics of each component, the vast majority of these problems were quickly resolved. However, this debugging took quite a long period of time, placing me behind in my original plan, meaning that I had to rethink my plan, reallocating time on the Gantt chart.

At the end of this section, and the end of Control System Testing and Development the Electronics design stood as thus:

Figure 84 Updated Circuit Diagram



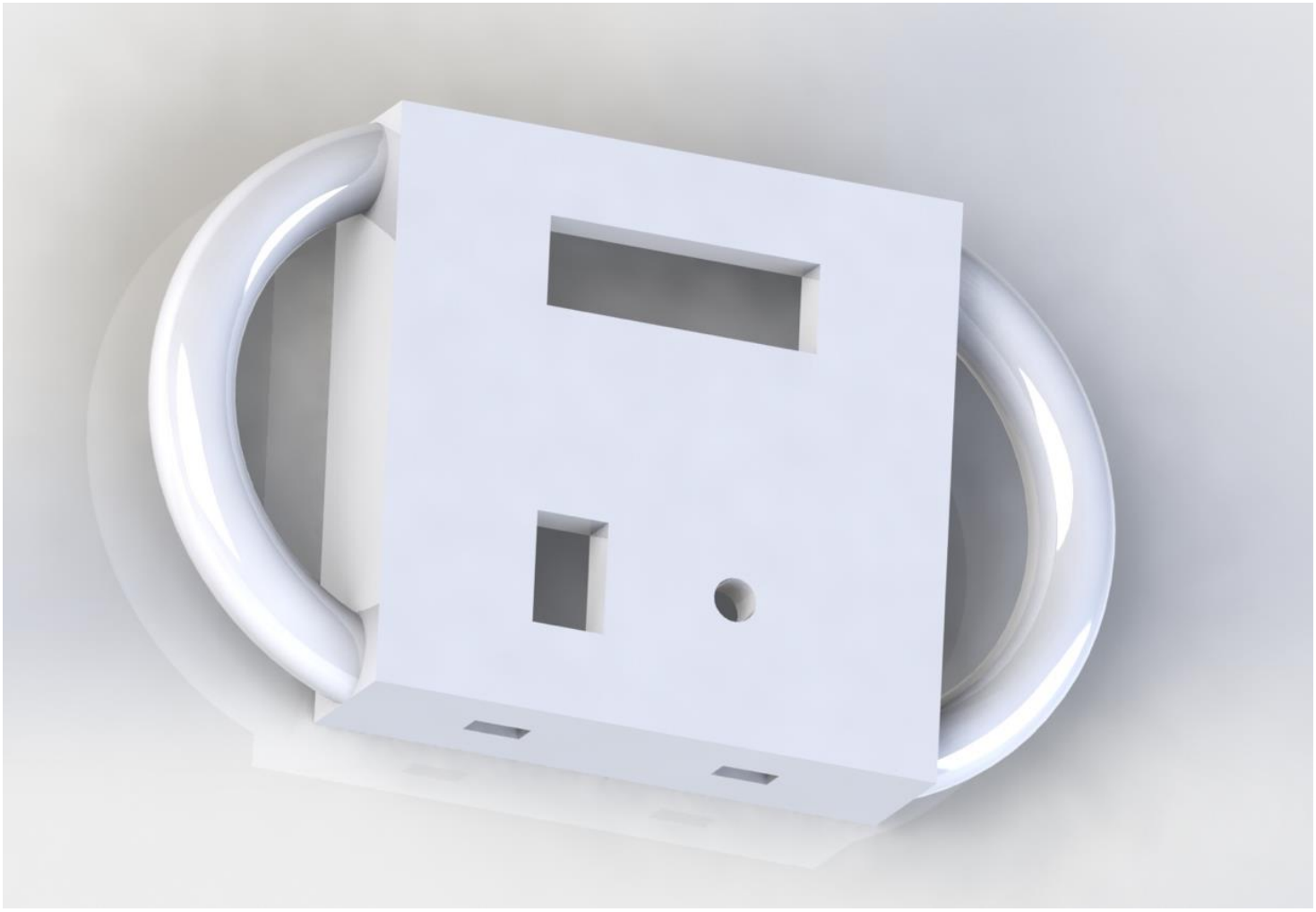
Source: Made using EagleCAD[177]

[177] (CadSoft USA, n.d.)

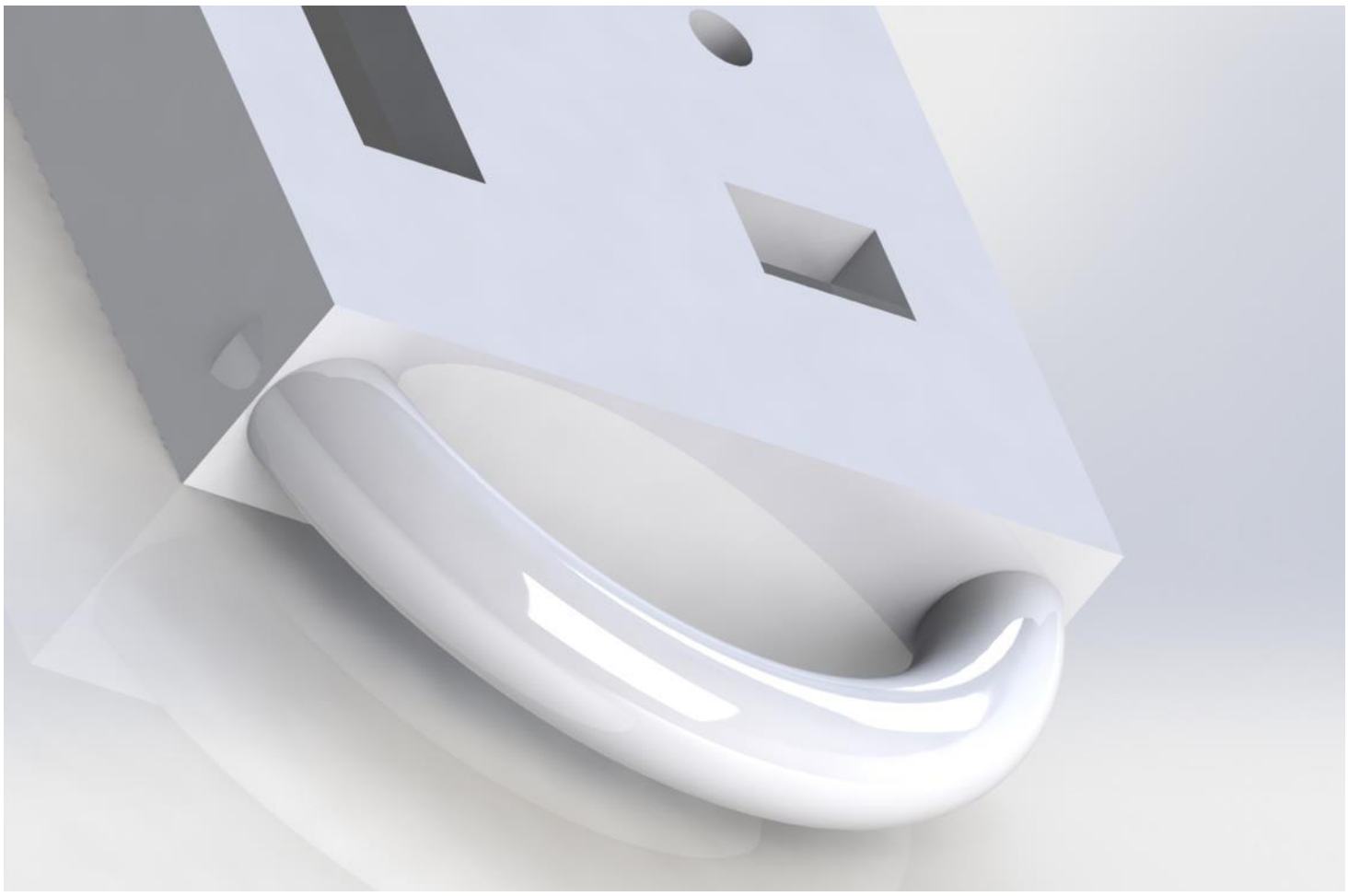
6.2 Mechanical System Ideas and Development

6.2.1 Idea One

Figures 85, 86 and 87 Renders of the First Idea – made using Dassault Systèmes SolidWorks 2014[178]



[178] (SolidWorks, n.d.)



Features of the design: The design features a solid handle which would allow the person to hold around and hence reduce the chances of them dropping the item, taking inspiration from the PowerGrip Utility Knife. The product also incorporated part of the BIGtrack mouse, by ensuring large amounts of space between the rotary encoder and the fingerprint sensor. The design takes into account all the components, ensuring they are best positioned so that the screen is easily visible and to ensure that the inputs and the two Micro-USB ports at the bottom of the device are available.

Successes: The design appears relatively slick and aesthetically looks good, with nice curves and uses of straight lines. Additionally, the placement of all the components is well thought through and all of them appear successful in ensuring they could be used. The handles appear to be successful in allowing an elderly person to more easily hold onto the product.

Failures: The design generally is very large and the components themselves do not take up much of the space, and hence, this could be rethought to ensure that the wastage is minimized and that the size of the product is reduced. (This is due to the necessity of allowing the user to put all their fingers inside the handle). Additionally, the smooth design of the handles may have a few problems depending on the exact manufacturing technique as it may in fact be slippery, hence may have the counter effect. Finally, the necessity of two handles should be considered since one hand would be required at all times to use the product. In fact, the only time when a hand would not be needed to scroll through a list or something of the sort would be when new data is attempted to be added to the device, and hence the device must be held in a precise location in order to be on top of the QR code.

6.2.2 Idea Two

Figures 88 and 89 Renders of the Second Idea – made using Dassault Systèmes SolidWorks 2014[179]

[179] (SolidWorks, n.d.)

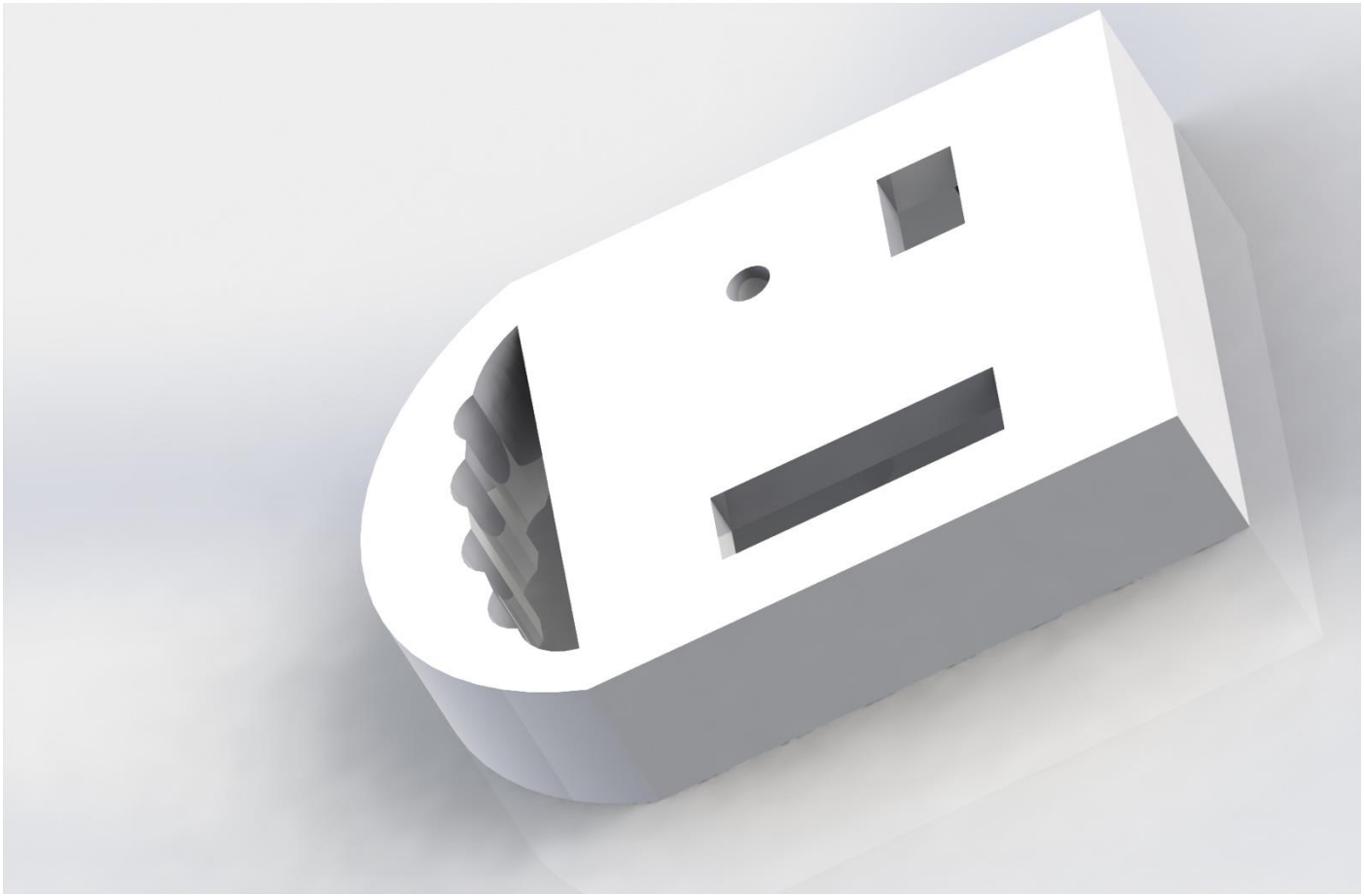
Features of Design: This design takes inspiration from the RealPad and iPhones and iPads, as it attempts to create the entire product in the smallest package possible, using filleted edges to allow the user to grip around the product. It is much smaller than other ideas, and hence should be lighter and easier to carry. However, like the first idea, it contains complete plans to contain all components, hence ensuring they fit and are securely held inside the product. Additionally, despite the reduced space, the Fingerprint Sensor and Rotary Encoder are kept a decent distance apart, ensuring that it is not easy to press one by mistake.

Advantages: The product looks the slickest of the three, with a large aesthetic appeal similar to new technology products. It contains all the required components (with complex plans to ensure they could easily fit and be held in place) with a much more compact shape. Additionally, the thin and light frame would mean it would be very easy to carry around in a pocket or a small bag. Finally, the regular cuboid shape means it would easily fit into a pile as opposed to the more complex shapes of the other two ideas.

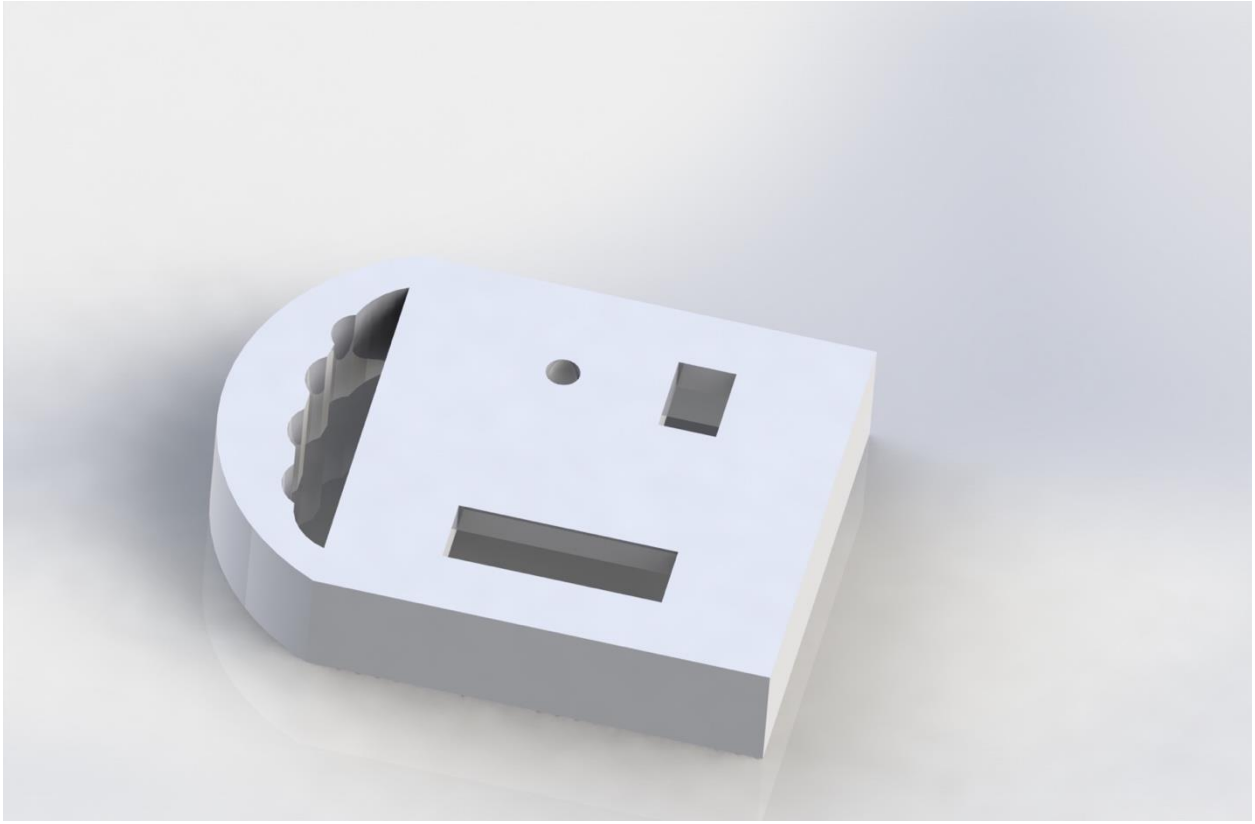
Disadvantages: The fingerprint sensor and rotary encoder are close enough to the sides that it would be possible to accidently press the buttons when simply holding the product on the sides. Additionally, to conserve space, the fingerprint sensor was placed at a 90-degree angle which would mean that the user would have to turn the product 90 degrees vertical to be able to operate and place their finger on the scanner. Finally, this idea has the worst ergonomics of the three with no specific way of holding the product instead, relying upon the user gripping around the product and having to make use of the friction of the casing.

6.2.3 Idea Three

Figures 90 and 91 Renders of the Third Idea – made using Dassault Systèmes SolidWorks 2014[180]



[180] (SolidWorks, n.d.)

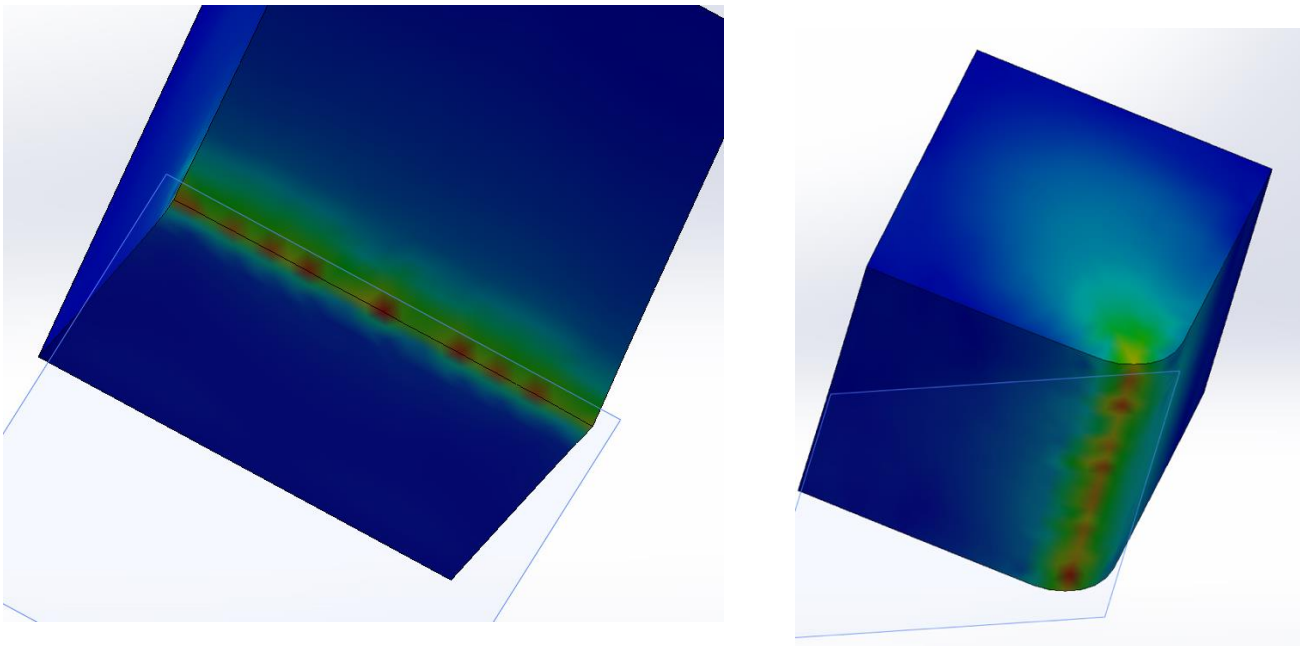


Features of Design: The third idea is highly unique with the idea of the single handle, appearing to be a better idea in this case, since once hand would likely be required at all times to operate the electronics of the product. It again contains all the required cut-outs to allow the components to easily fit into the casing with lots of space to spare. Again, the casing has been forced to be larger than it otherwise had to be, since the handle had to fit four fingers. As can be seen in figure above, there is a notch on the side of the handle, with a small flattening to allow the thumb to easily fit on the outside adding even more grip to the structure. However, for future revisions, it might be useful for the product to have a proper space for the user's thumb with a track available around the top. Through the middle of the handle, reminiscent of the Bentley Knife and Scissor Sharpener there are spaces for the fingers with the middle fingers being slightly closer than the side fingers hence allowing the hand to be in a relaxed natural gripping position.

Advantages: The product has the best ergonomic positioning, allowing the user to easily place fingers around the handle of the product, and hence grip it firmly. Due to the non-cylindrical shape of the handle, it is not slippery. The product is also much smaller than the first idea, as it no longer requires the extra unused second handle.

Disadvantages: This design would restrict the user to using the product with their left hand, as the right hand would be being used to grip the product. Additionally, the bottom of the product, next to the grills would benefit from being filleted to ensure the fingers does not have to enter the paths at right angles to the path, as this would likely be uncomfortable. In fact, though the right, straight corners add some aesthetic value they create a number of problems such as safety, since they may be sharp. Additionally, during drops they would create an area of vulnerability, since the area exposed would be very low, meaning the product might be more likely to fracture. The handle still adds bulk as opposed to the second idea, in fact the desire to have all four fingers fit in the handle makes the product much larger than it would otherwise be.

Figures 92 and 93 Drop Test showing von Mises Stress differences when dropped on a square edge vs a filleted edge



Source: Made using SolidWorks 2016 and SolidWorks Simulation[181]

SolidWorks Drop Tests (shows the von Mises stress encountered) of a cube with an edge squared and then filleted – the squared edge shows how the edge encounters the vast majority of the stress and hence is more likely to fracture as opposed to the filleted edge where the stress is spread over a much larger area meaning a lower stress at the edge, therefore a lower chance of fractures.

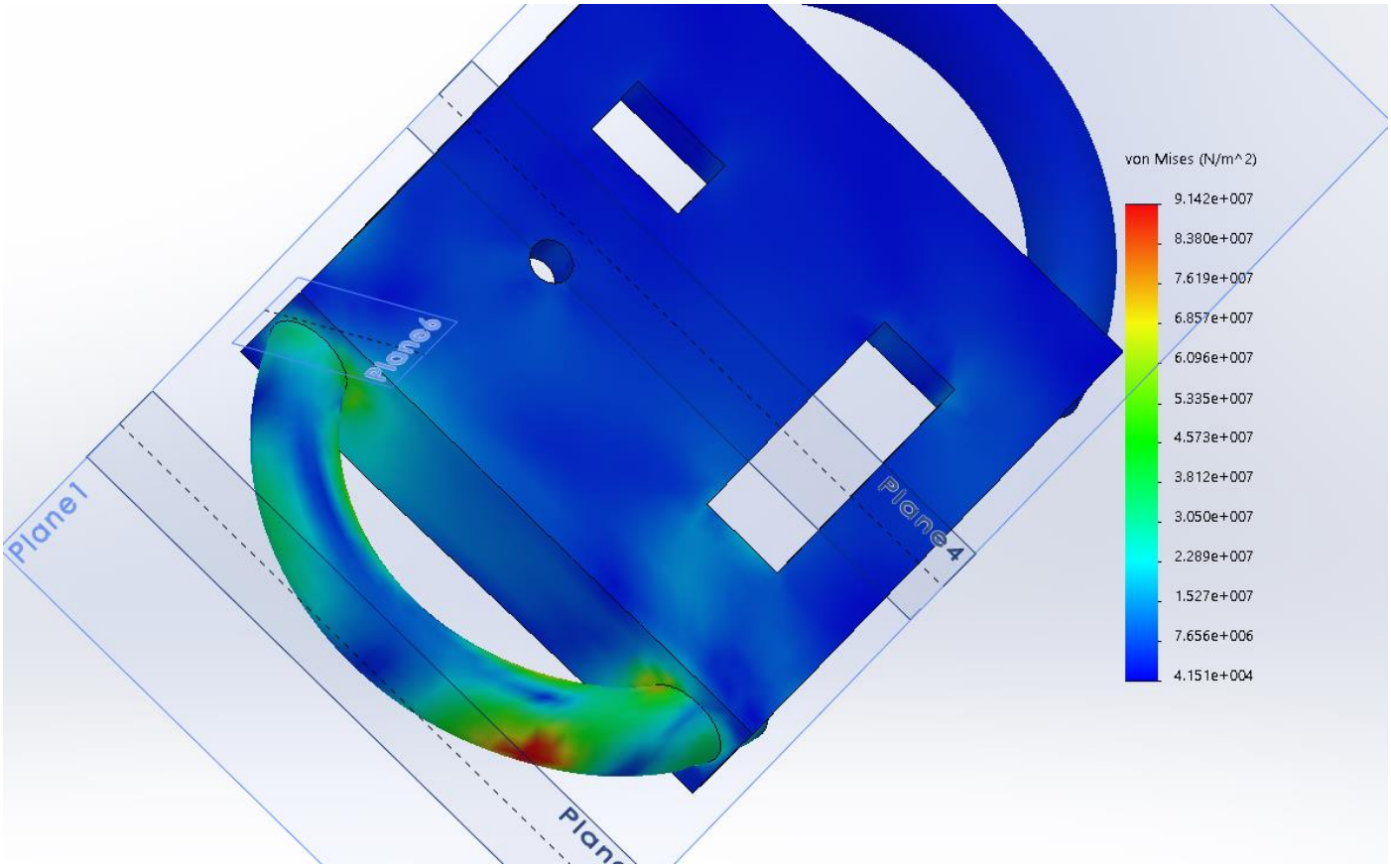
6.2.4 Evaluation of Ideas

Each idea clearly has some benefits and some failures, hence in order to produce a final design, it was important to take the successes of each of the designs and consolidate them into one better design. Firstly, I liked the compactness of the second design, and I hoped to make the product as small as possible, hence easy to carry around. However, I also liked the handles aspects of idea one and idea three as they clearly offered greater ergonomics making it easier for an elderly person to hold. Additionally, I liked the grooved in particular of the third design as they offered even more ergonomic support, hence even further reducing the likelihood of the product falling. Though there were problems associated with the single handle, since they would best cater one population, the right-handed or left-handed people, I found that the space conservation that it provided was indeed beneficial and chose that it would be a good idea for the final design, though a few small changes were necessary. Finally, before choosing a particular design, I completed drop tests on all the designs, looking at the various flaws of the designs, hence allowing me to resolve these problems if the part was integrated into the final design.

6.2.5 Stress Testing and Evaluation

[181] (SolidWorks, n.d.)

Figure 94 Stresses encountered on Drop Test of First Idea

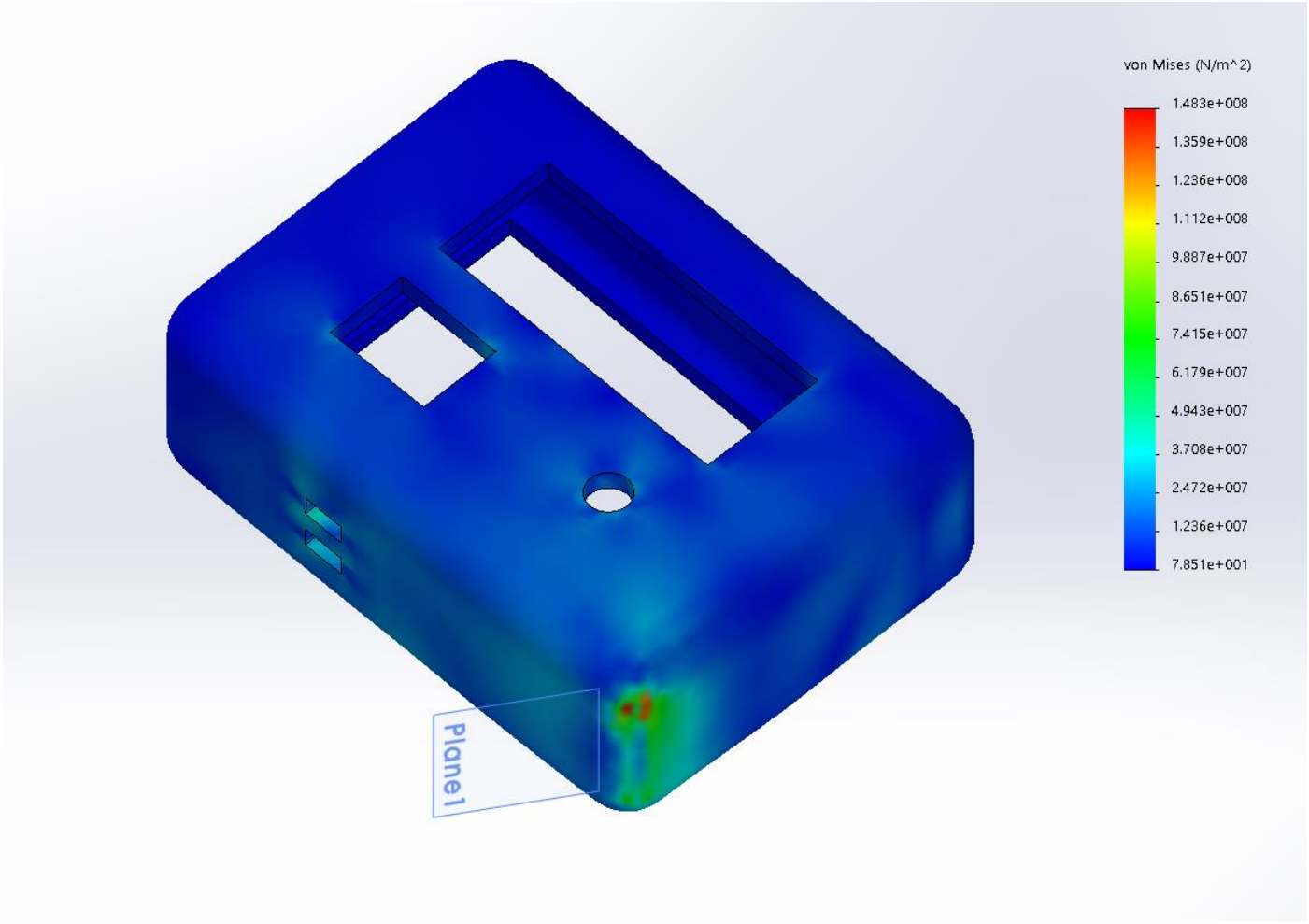


Source: Made using SolidWorks 2016 with SolidWorks Simulation[182]

For the first idea, the stress testing (dropping the product from 2 metres) showed that the handle was relatively effective as it managed to transmit the stress equally along the structure of the product. The height of 2 metres was chosen as it was slightly higher than the average height from which the product would be dropped therefore providing a margin for error. However, there was some unnecessary stress (in the sense that it would be relatively easy to reduce) at the point at which the handle re-entered the product showing that this point could be redesigned to increase the surface area of contact. This would further increase the transfer of stresses between the handle and the main product therefore further reducing the chances of fractures.

[182] (SolidWorks, n.d.)

Figure 95 Stresses encountered on Drop Test of Second Idea

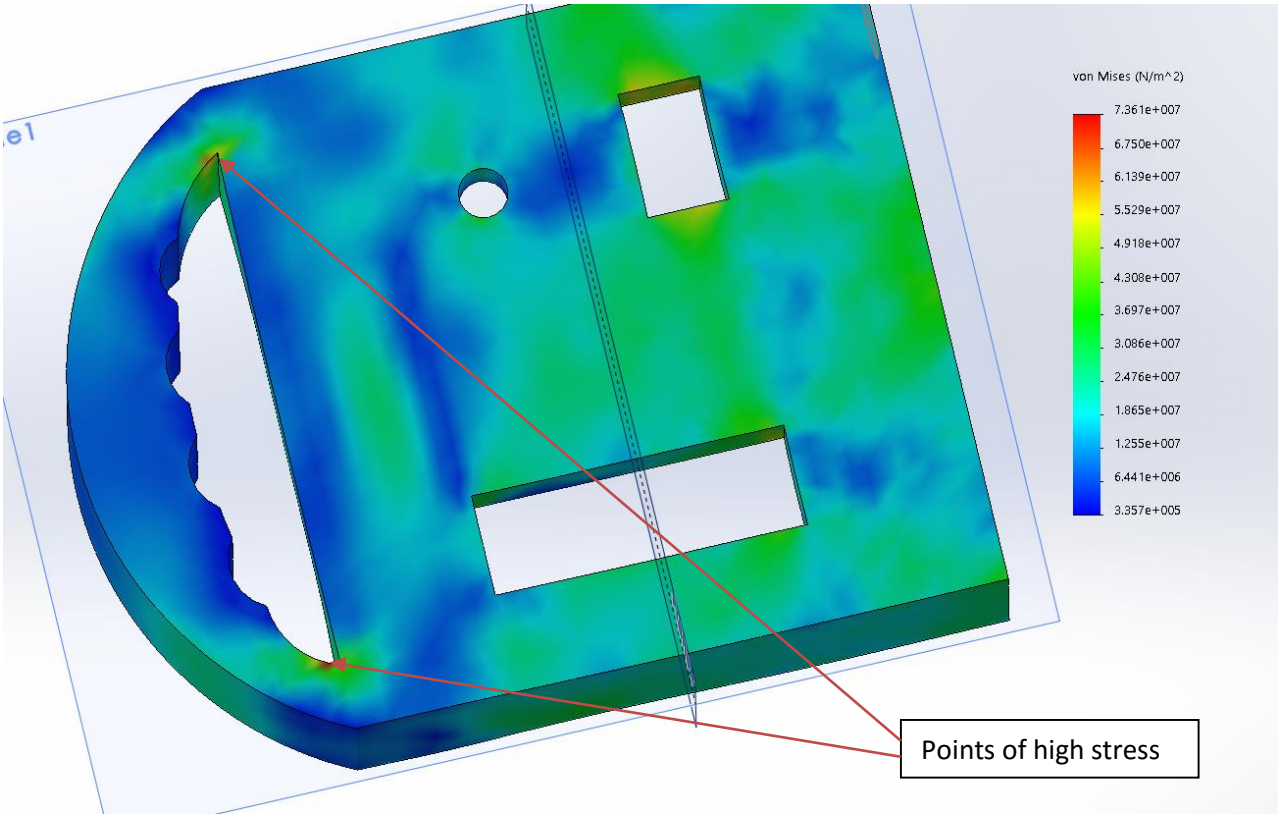


Source: Made using SolidWorks 2016 with SolidWorks Simulation[183]

For the second idea, where the product was dropped onto Plane 1, again from a height of 2 metres, the simulation shows how the stresses are generally much lower, with the vast majority in the order of magnitude of 10^1 von Mises (N/m^2) as opposed to 10^4 as in the first design. It clearly shows the effectiveness of the rounded design, as the stresses are very equally spread out over a large area. This also clearly shows the problems of any squared edges with the insides of the cut-outs for the LCD and Fingerprint Sensor experiencing stresses at these corners. Unfortunately, these shapes are governed by the components themselves so this could not be changed.

[183] (SolidWorks, n.d.)

Figure 96 Stresses encountered on Drop Test of Third Idea



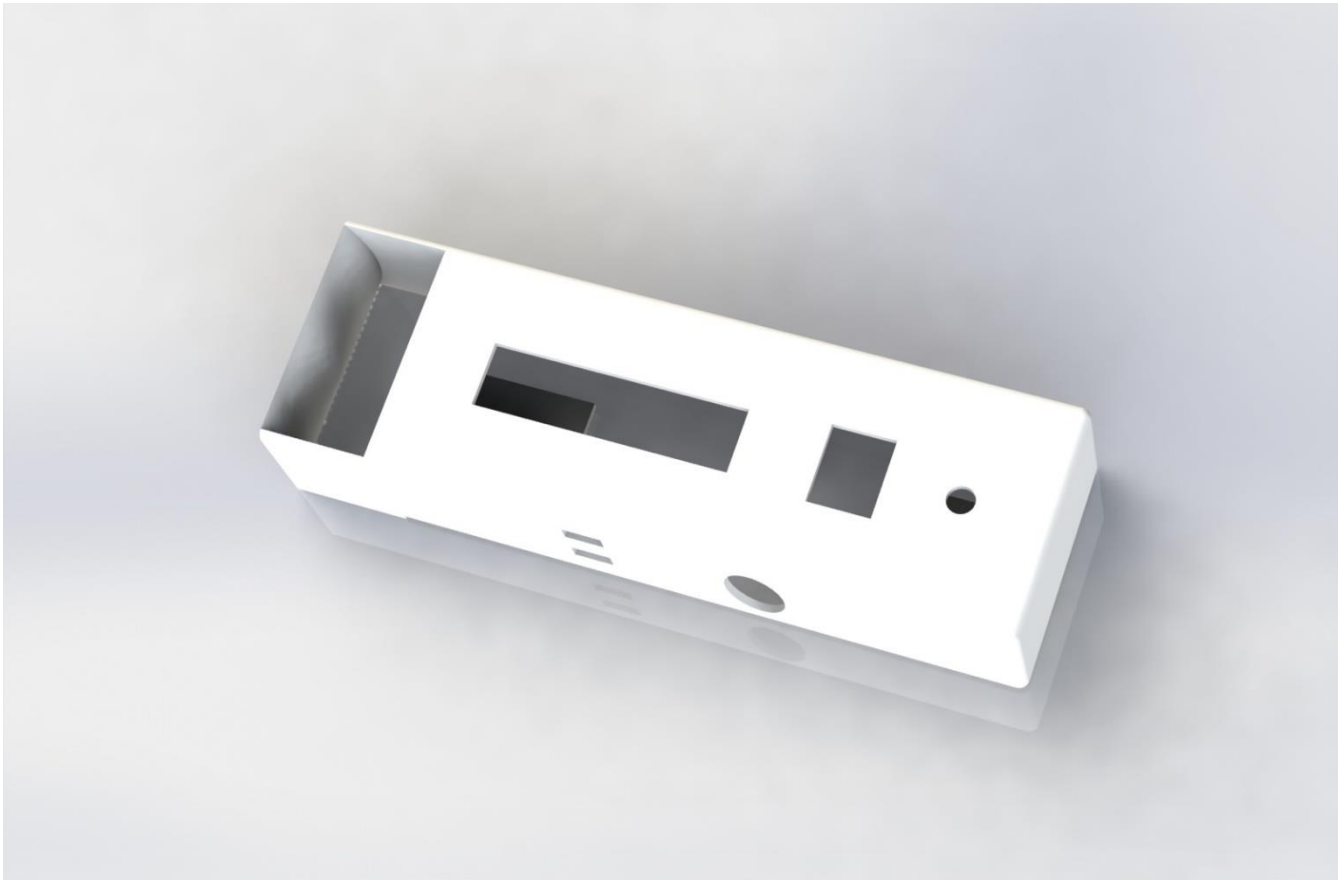
Source: Made using SolidWorks 2016 with SolidWorks Simulation[184]

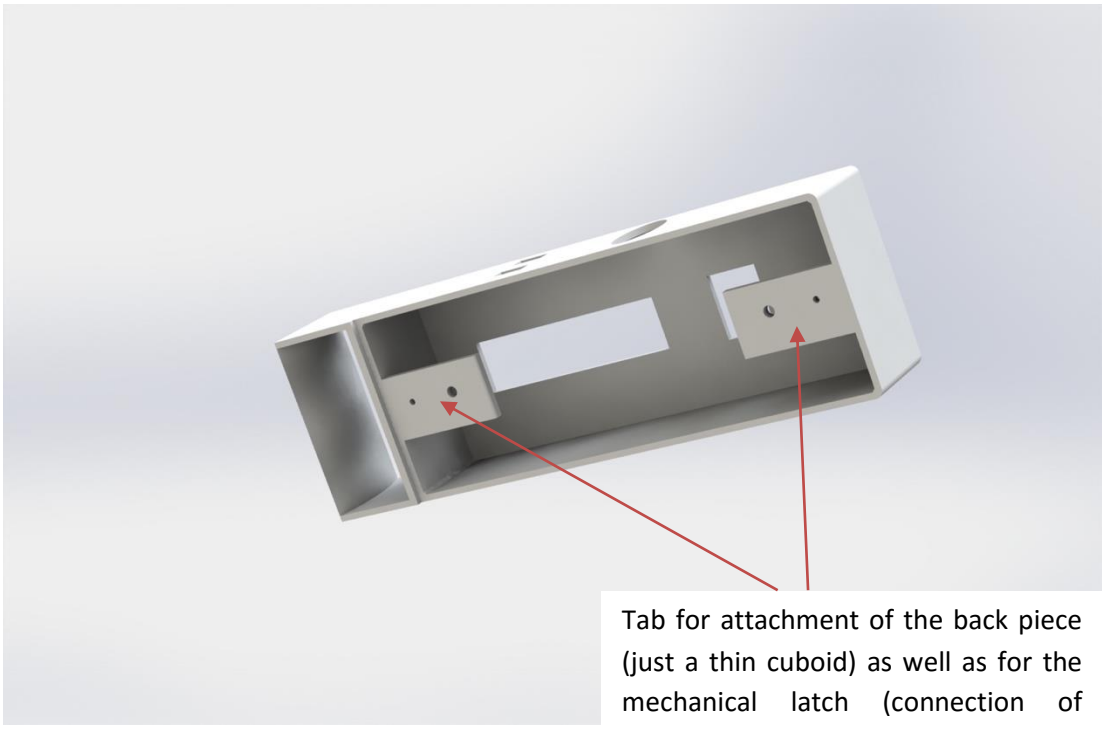
The final idea, being dropped straight down shows stresses very similar to the first design in magnitude, though since it has fallen on a larger surface area, there are more areas of higher stress. However, it is clear that the product is very effective in transferring the stresses inside the design, ensuring that the chances of fracture are minimised, therefore being effective. However, there is one point of improvement, the point at which the handle reaches the main body of the product. As can be seen in the annotation of Figure 96, there is a relatively large stress at these points. Hence, it would be advisable to attempt to further increase this point of connection, in order to reduce the stresses which occur at this point. This could be achieved by simply filleting this edge.

[184] (SolidWorks, n.d.)

6.2.6 Final Design

Figures 97a, 97b and 98 Renders of the Final Design – made using SolidWorks 2014[185]

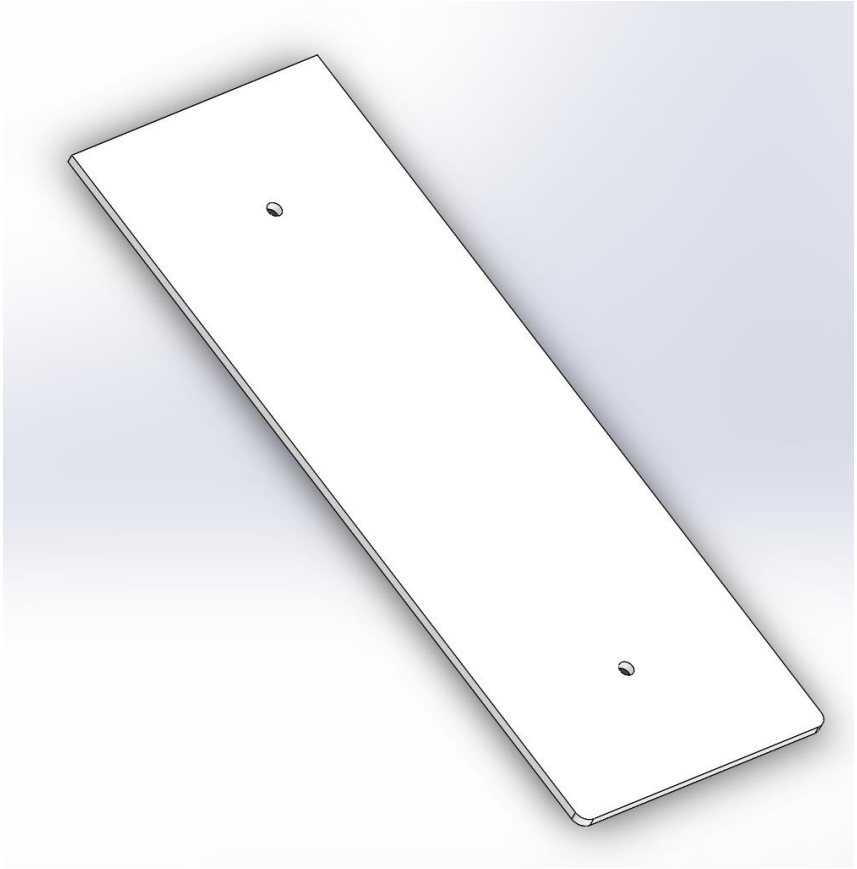




Tab for attachment of the back piece (just a thin cuboid) as well as for the mechanical latch (connection of

[185] (SolidWorks, n.d.)

Figure 99 Render of the Back Piece – made using SolidWorks 2014[186]



The main feature of the final design is the redesigned handle system which is effectively a redesign of the handle in the third design. It adds the additional curve from top to bottom, therefore making the product even easier to grasp.

[186] (SolidWorks, n.d.)

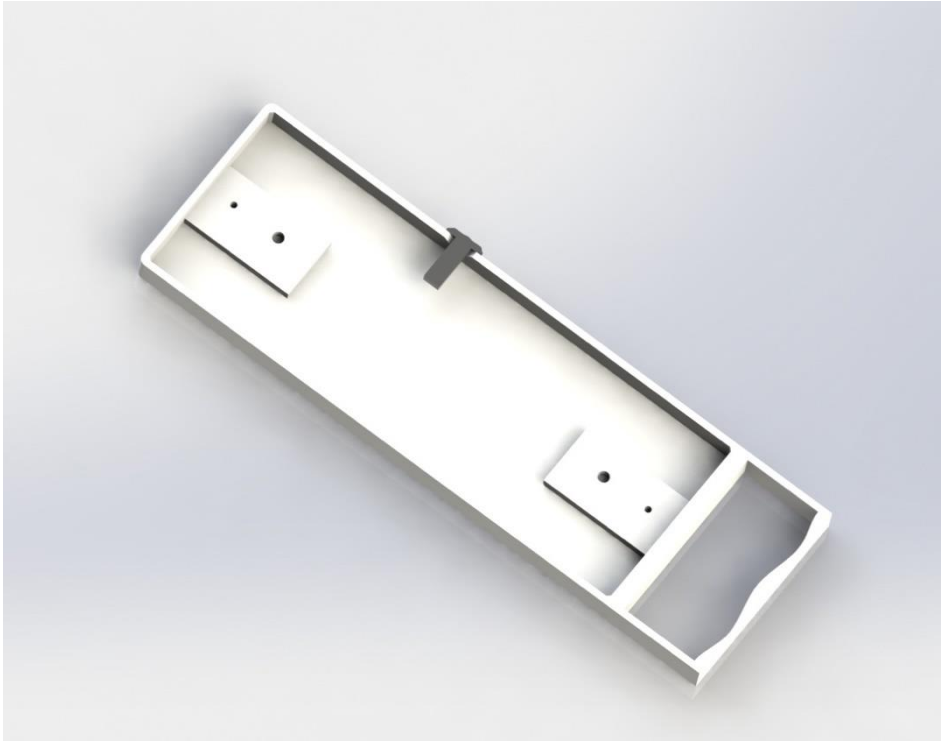
However, instead of providing space for four fingers as the original idea was, the final design only provides space for two fingers therefore helping to reduce the size of the product significantly. This was based on a large amount of research about how people tend to hold relatively small objects[4]. While it is true in products which are generally close to square we tend to use all four fingers to wrap around them, when the products become thinner, it is very normal for us to use two or sometimes three fingers to comfortably grip an object. In fact, once I had realised and designed this I realised how often I personally do this, when carrying everything from backpacks to the break lever of a bicycle. Therefore, it was clear that the handle of the new design would pose little trouble in the possession of less finger holes. Though this might make the product slightly more liable to falling, since fewer fingers were now gripping the product, this was easily overcome by the new, far more ergonomic finger holes, which follow the natural ergonomics of the fingers, as determined in the anthropometric data of elderly people.

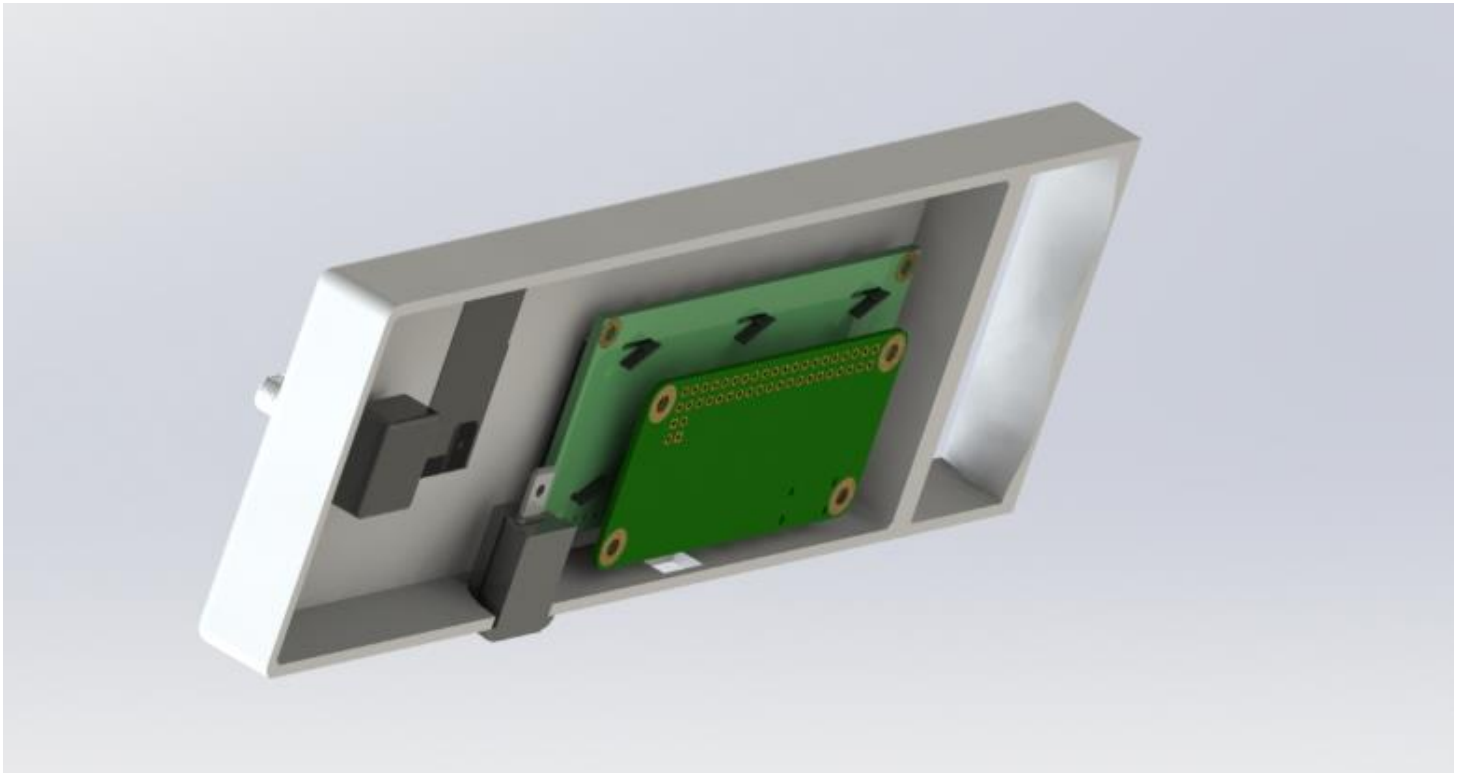
The smaller handle allows the entire product to be far more compact, carrying all the necessary components in a far smaller package. As opposed to the small package of the second idea, the placement of the components has been moved, to allow the user to more easily access the rotary encoder and fingerprint sensor. In addition, the long thin design concentrates the weight in one line, meaning it is even more easy to carry.

Additionally, the back of the device shows the newly designed mechanical latch, which would work using the large connectors which connect to the back piece (Figure 99). This allows the back to be easily removed in case the user want to replace the battery or in case any repairs are necessary. The back connects with two tapped screws going through the tabs, see Figure 101, and then through the connector on the back piece. In addition, this connector will be coated with copper tape, and a wire soldered to it. The second part will also have this copper tape on it, joining the points where the two connectors would be. Therefore, when the back piece is attached, there is a connection between the copper tapes on the two connectors and this can be read by the microcontroller. However, when the back piece is removed, this connection is lost, reporting the opposite to the microcontroller. Hence the product has the ability to ensure all information is secure when the product is opened.

Figures 100, 101 and 102 Renders of Final Design with Components Designed and Mated to the Casing – Made using SolidWorks 2014[187]

[187] (SolidWorks, n.d.)



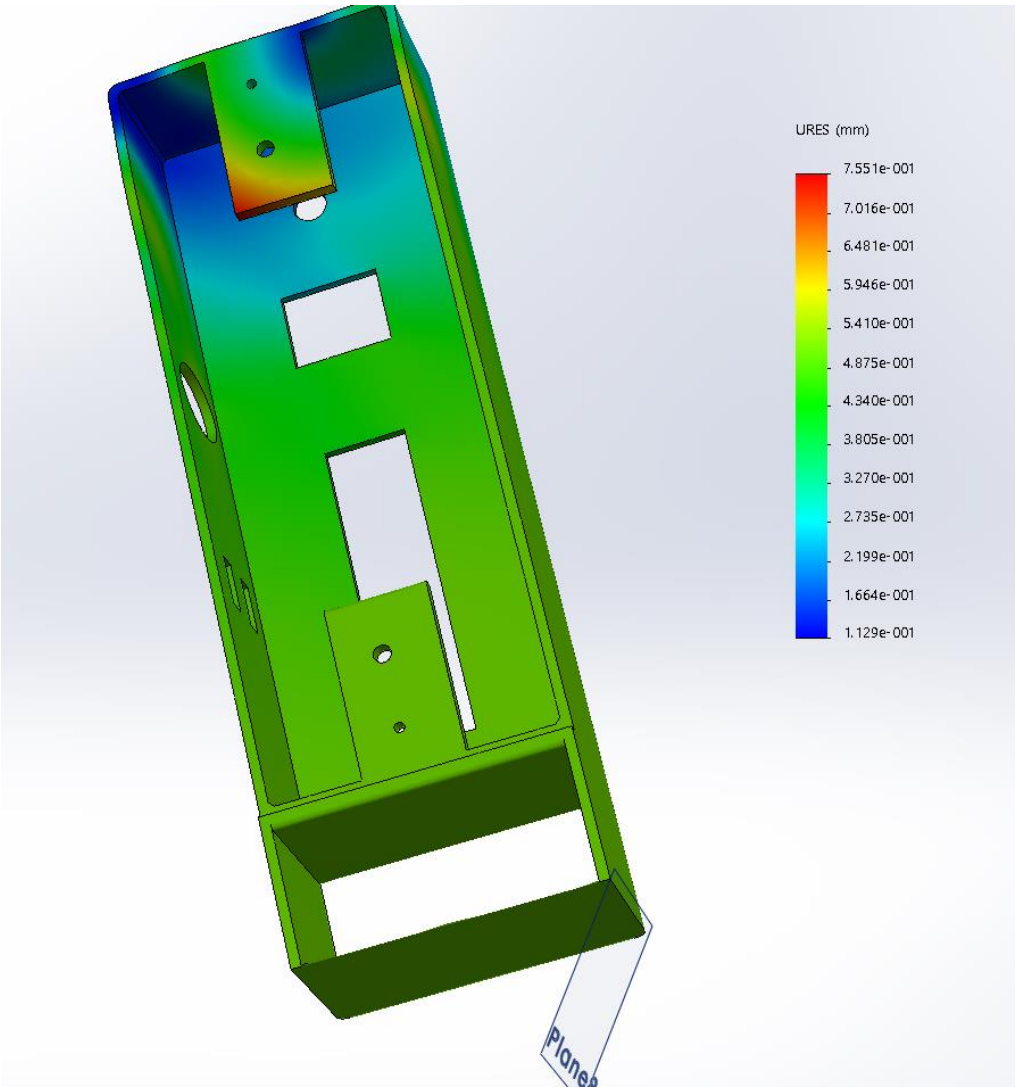


The final step was ensuring that the product passed the drop tests, and at this point, I had to decide a height at which I expected the product to remain intact. I chose the height of 2m, as this was higher than the average person and so if it succeeded at this height then it should be easily able to survive any average drop from one's hands, the main case in which the product will be dropped.

Given I thought the most likely area where the product was to break was the corner of the handle, I dropped it at this point. Finally, as a material, I went for ABS, a material which my initial research had shown to be one of the

possible materials that I could use, while not being the very toughest, so it's results would be indicative of most plastics.

Figure 103 Displacements encountered during a Drop Test of the Final Design



Source: Made using SolidWorks 2016 with SolidWorks Simulation[188]

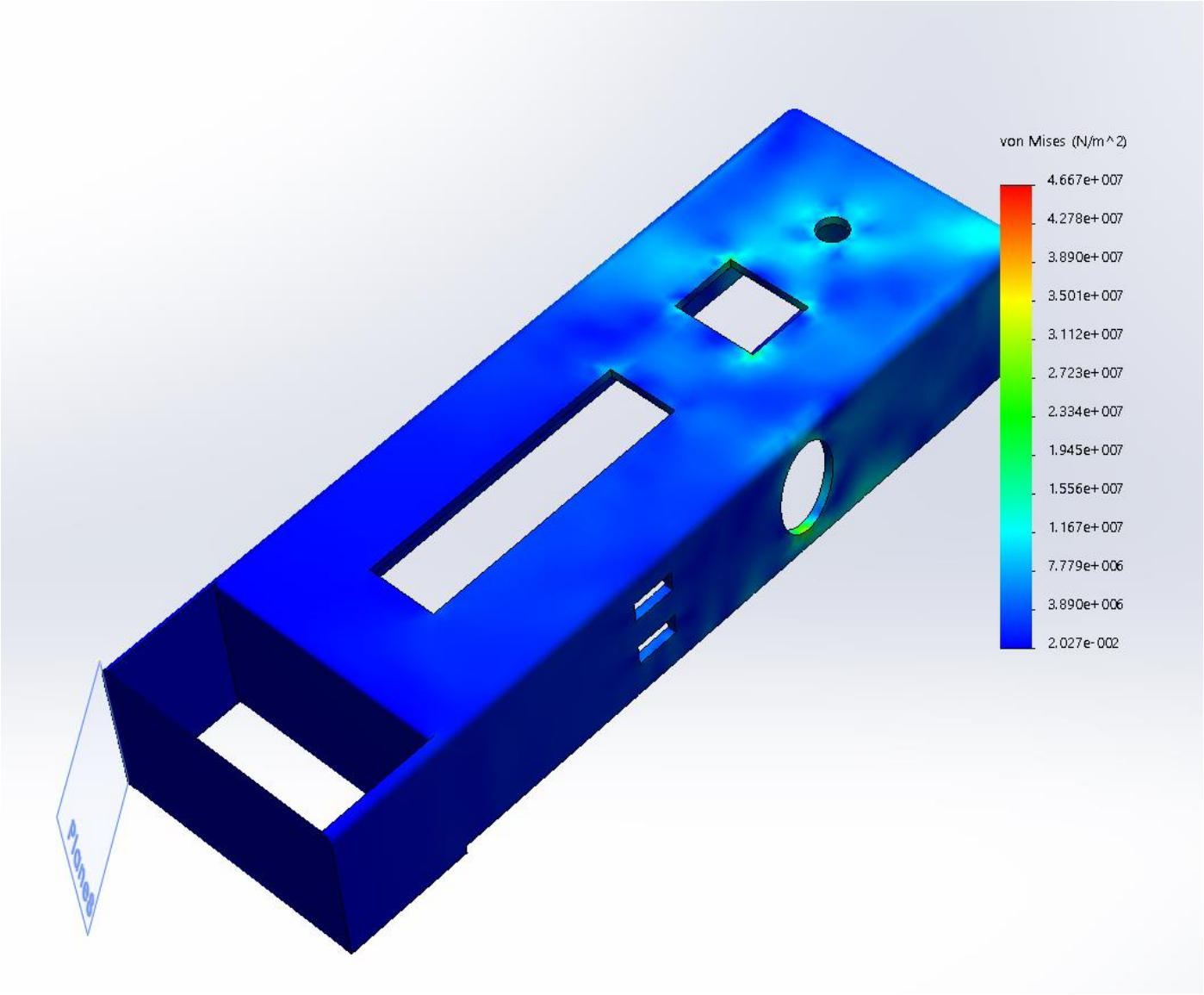
Despite the product being dropped on Plane 8, this part appears to have transferred the stresses almost seamlessly with the handle experiencing no greater displacement than the rest of the product with the exception of the back corner which has experienced very little changes (circa 0.15mm). However, it is clear that even the end of the connector which has experienced the highest stresses of the entire product, has only experienced a displacement of 0.7mm, a rather tiny amount.

In fact, if one looks at a diagram of the von Mises stress incurred by the part, it is clear that the vast majority of the product incurs stresses with magnitude 10^{-1} or 10^{-2}, well within the strength of most materials, as the average. In fact, more importantly the maximum is not much higher, hence not pushing material past the yield point. Basic research into the yield points of the materials show that the yield points for polymers tend to be of the magnitude of 10s of Megapascals therefore 10^{7}, therefore the product would have no parts which would be anywhere near the yield point, and so would not fracture. Additionally, I find it surprising that despite the product being dropped on the

[188] (SolidWorks, n.d.)

edge of the handle, there appear to be very little stresses around the corner, likely due to the price and detailed filleting which ensured that all the stresses are spread over a large surface area at the corners.

Figure 104 Stress encountered during a Drop Test of the Final Design



Source: Made using SolidWorks 2016 with SolidWorks Simulation[189]

6.3 Software Development (Code is available in the Appendices)

Once the Control System design had been decided much of the software development work was done alongside the Control System development, as the system was begun to be constructed in parts, testing individual components and subsystems. Other than the product itself (which posed a relatively large challenge) there was also code for the transfer systems, to allow the user with plenty of options for the creation of the encrypted QR code which the product could read, for example a Web-App and a Mobile Application for iOS and Android.

[189] (SolidWorks, n.d.)

6.3.1 Developing and Testing the new Cipher

In order to test the cipher which would be used for encrypting the data to be transferred from the mobile or desktop device to the product (through a QR code), I had to find a way of finding how hard this would be to hack. This largely used two approaches. One was asking a number of mathematicians and computer scientists to see whether they saw any flaws in the cipher, seeing any way in which the cipher could easily be hacked. This method in fact originally revealed the problem with the first iteration of the cipher (using the basic idea that the key would change in length) and necessitated the move towards the second design of the cipher (where the key itself evolves with encrypting the plaintext). However, in a more extensive search asking more people about the second cipher, I as unable to find any flaws with it, with most agreeing that the design was particularly innovative and offered a large number of advantages over other ciphers that currently existed in the market. The other approach which was taken was the attempt to produce a computer program which could hack the system. I produced a program in C# which tried to hack one of my passwords which was encrypted using this cipher. Though the hacking mechanism was relatively primitive, applying mostly a brute-force technique with a few more caveats which would allow it to work more quickly. However, the one advantage I gave the system was the knowledge of the cipher design which simplifies the task significantly, though it failed in any attempt to find any useful information through frequency analysis.

The results are shown below:

Figure 105 Test of the new cipher

Password	Time taken to hack
hellofirsttest	4 weeks
ashwinspassword123	9 weeks
this!Saf%rm*OreCOMPLEXTEST	Unfinished after 45 weeks

The failure of the system to succeed in managing to break my cipher for a complex password and take a long time for simple passwords clearly appear to show it is very secure. However, it is important to note that hackers would have more sophisticated computers with better applications which could more intelligently attempt and verify key combinations, hence allowing them to break the cipher more effectively.

In comparison, a number of other ciphers were attempted to be broken using the same computer:

Figure 106 Tests conducted to compare the new cipher to other existing ciphers

Shift Cipher (length of twenty characters)	**1 second**
Polyalphabetic Cipher (Length of ten characters)	5 minutes

These clearly show that my cipher is far more secure than any other simple ciphers that exist, largely due to the innovation of the changing of the key as you move through the encryption.

From here, it was important to consider how the cipher could be used in real life, working with the inputted information. While it is easy to apply a shift to a single alphabet it was important to note that the passwords could contain any letters, alphanumeric or special characters. Hence, we had to widen our scope to the Unicode Character Set. This contains the alphabet for every single character that could be typed, including everything from Chinese alphabets to the newest emoji. This provided a system by which I could retrieve a number from each of the inputted

characters, apply a mathematical addition and produce a different character. However, this also required some care, since there were specific periods notably between 120 and 160 where there are no characters (which could create a problem) since it is the gap between the end of the ASCII characters (the old standard) and the start of the Unicode characters.

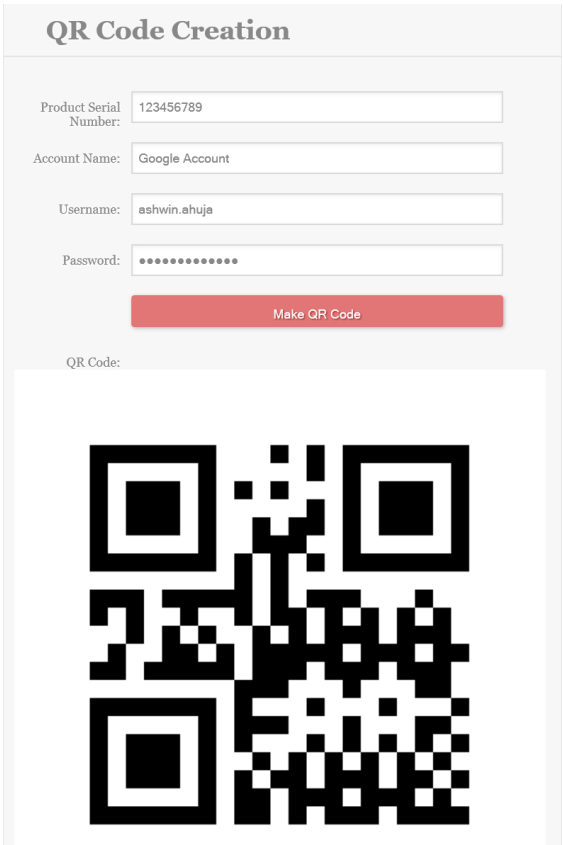
I also briefly considered the possibilities of overflow, however, since the vast majority of the last 1000 Unicode characters (up to around 4000) are foreign or very odd and unlikely to be used, I did not think it a problem, as people would be unlikely to be use one of these, which would then lead to them going over the maximum.

6.3.2 Mobile and Web Apps

As earlier decided the Web app would be produced using JavaScript and HTML and CSS. Hence, I had to find an appropriate API which would produce the QR codes, since this would be a little too complex to do in JavaScript or would have to undertake the rather challenging task of producing a backend (likely in a C based language) to produce a QR code. Luckily, I encountered the Google Charts API which allowed me to send a GET request to the Google servers with the specific information required on the QR code, and the servers would return the QR code image which I could show. While this meant that internet was required, I felt this was not a problem since people would need to be online anyway to use the WebApp.

Once the JavaScript to produce the QR codes had been written, successfully managing to send the requests to the Google servers, I moved onto incorporating the cipher to the system, ensuring that the data that would be a part of the QR code would be encrypted as I had designed. This was quite a large challenge as it had to be written in JavaScript which struggled to cope with even the basic character and numerical manipulation that the cipher required. However, after some struggle, the required functions were created and the attention was moved towards producing an aesthetically pleasing user interface. While an external style sheet could have been made, due to the simplicity of the page, a decision was made to integrate all the CSS in the main HTML, hence simplifying the page.

Figure 107 Screenshot of Web-App



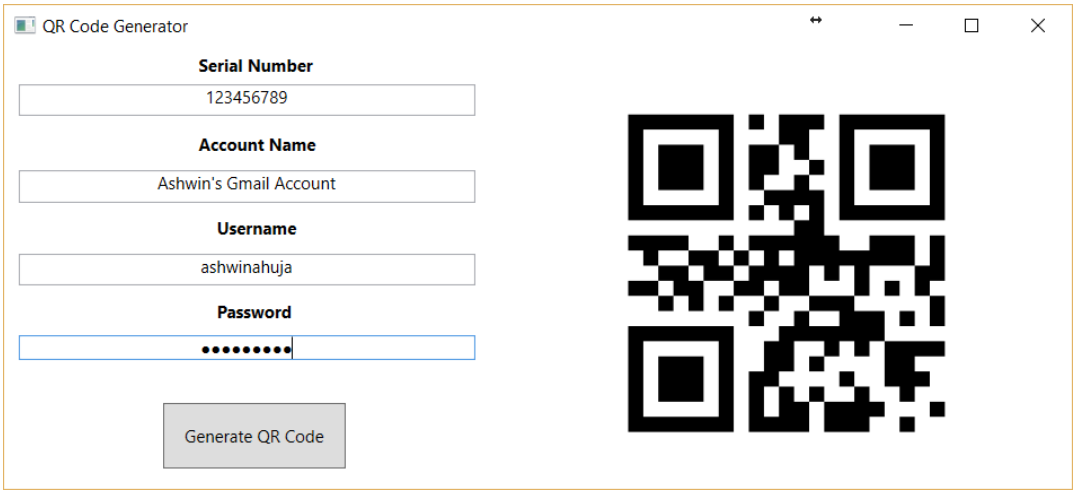
A screenshot of the Webpage shows how it has been designed to be aesthetically pleasing as well as allowing the user to enter all the required information easily.

Meanwhile the same ethos was carried to making the Mobile Apps, attempting to make the user interfaces as simple as possible, while complex enough to allow them to meet their purpose. In this case I attempted to find a different API to use to connect to Xamarin Forms to allow the application to work offline to produce a QR code, as otherwise the user could only add passwords when connected to a network. This was in fact possible through the use of the library ZXing.NET, which had a specific library for use with Xamarin Forms (with a specific part which produced QR codes). This meant that it could be easily implemented allowing me to use a few lines of code to display a QR code in an already initialised UIImageViewer (a space intended for an image). In this case implementing the addition of the cipher proved much simpler since I was much more at home with C# than JavaScript and knew how to use many of the character manipulation functions that I had struggled to find in JavaScript.

Finally, in the process of making a mobile app, I also decided to make a desktop application, using WPF, since I had found through my research that many of the elderly were most at home on the computers as opposed to the mobile apps. However, a number were intimidated by the internet, and the necessity of going on the internet to gain the ability to add passwords to the product might have proved counterproductive for some users.

Hence, a simple PC app could easily solve this. Since this app as well would be written in C#, I was able to recycle large parts of the Universal Apps code including the encryption parts, thereby saving me large portions of time. However, I did have a little struggle learning XAML, the language which would be used for the Storyboarding (choosing where components of the screen will go), however, by reading a comprehensive book on it, I was able to learn how to use XAML relatively quickly. In the end, I chose to use a Background Worker continuously updating the QR code to prevent the person even needing to press the generate QR code as the system would anyway be able to work out that the user had finished entering the information, since there was now no longer any keyboard interrupts being called, and therefore the QR code could be loaded. Though I had originally intended to use ZXing.NET library for this as well, I actually found that this added a lot more complexity, since the commands for WPF were entirely different. Therefore, it would be much easier to use the Google Charts API that had been made for the WebApps. The use of Secure Strings for the passwords (which means that the programmer does not have access to the password at any point) means that even should the computer be hacked; they could not gain access to the raw unencrypted data in the field.

Figure 108 Screenshot of Windows Application



6.3.3 Product Software

6.3.3.1 Encryption

As has been discussed previously, the key for the AES encryption (to be used on the Pi Zero) would be taken from the fingerprint data of the first finger used to enrol on the device (see Section 5.4.3 Encryption for passwords). This means that the key of each person's passwords is specific, and a member of the programming team would not have access to the password, hence adding another layer of security. This is being shortened using the SHA-256 algorithm. Though I had looked into simply using the first 256 bytes of the template, I had quickly found that there was often a lot of overlap between different fingers, so this necessitated the usage of a hashing mechanism, such as the SHA-256 to ensure that all different fingerprints mapped to different encryption keys. In order to use all the encryption systems, the library PyCrypto was used, which contains the ability to use these built-in functions. Upon first decrypting of the device, the library does everything with a set inputted (encrypted file) and a set outputted decrypted file, after the chunk size which has been used has been set. However, encrypting the file is more complex, as it might necessitate the adding of additional information to ensure that the total file size to be encrypted is a multiple of the chunk size that is inherent with block ciphers such as AES. Hence, the file is padded with spaces if necessary.

6.3.3.2 Controlling the Fingerprint Sensor

The fingerprint sensor required a specific library to be produced for it, while the sensor itself handles all the communications with the optical camera (and the processing thenceforth). As defined by the datasheet for the module, the fingerprint sensor accepts a number of set commands, which change depending on the exact data that the person wants the sensor to respond with. These were originally written separately for each command, to ensure that each one worked before they were consolidated into a single library, written in Python, which contains information for all the communications which might occur. In order to complete the final communications, the pySerial library is used, as it allows for easy transmission (and receipt of data) through the Serial Protocol. Much of the designed library works using interrupts, and EventHandlers which are called if there is a receipt of data from the fingerprint sensor. Hence, if any data is received from the fingerprint sensor, it is immediately decoded and depending on where the program is during the run cycle its value can have an impact on the device, for example a confirmed verification would allow the device to unlock.

6.3.3.3 Controlling the LCD screen

I2C character LCDs are few and far between, so I could not find any prior information on getting them working with a Raspberry Pi. In addition, I was unable to obtain the exact screen I wanted, since it was only available in the US and by the time I had decided on the screen it was too late. Therefore, though I produced a basic library based on the commands described in the datasheet of the screen, I had no chance to test it. This made use of the python library i2c tools which would allow me to set an address to write to and a byte to send. I was able to determine the exact values to send by using the database of bytes in the datasheet of the LCD, which included the commands for resetting the cursor at different points as well as clearing the screen, enabling the backlight etc. Instead I was forced to use an Adafruit 128x64 OLED which I already owned, and hence I made use of the Adafruit Python Library. Since I've had personal experience in working with the Adafruit products, I expected the library to be well written and easy to use. However, I had a number of issues and found that the library barely met my need. If I were using the product as the actual screen of the product therefore, I would have written my own custom library to help to simplify the complexities associated with using a Graphical Screen. The library provided a lot of unnecessary options for producing shapes and images, which were unnecessary for me.

6.3.3.4 Controlling the Rotary Encoder

In controlling the Rotary Encoder, I based my code largely on the working Arduino code that I had produced, when the encoder was made to work with the Arduino Uno, therefore I had expected the code to transfer seamlessly. That

wasn't the case. I spent many hours attempting to find any possible issues, since it was not working, however, it transpired that the issue was in fact with the Rotary Encoder. Though there was the possibility of writing a library for the rotary encoder I felt that it was unnecessary, since it could be declared as a function and referred to throughout the main code relatively easily.

In order to ensure that the turning of the rotary encoder surpassed other changes, I used both pins of the rotary encoder as Digital Interrupts with them stopping other things going on and requiring a checking of the values of both pins and determining if the encoder had turned. If it had turned, then based on the current position of the product, locked, which screen it was on, it responded by waking up the screen or moving up or down the screen. In order to control all the interrupts as well as controlling the reading from the pins, I made use of the Raspi.GPIO library. In addition to problems with the Rotary Encoder itself, I had a large amount of problems using the built in Pull-Up resistors that the Raspberry Pi offers, with them sometimes apparently failing to work despite being initialised correctly. Therefore, I chose to implement external resistors which were far more reliable, and solved a number of problems.

6.3.3.5 Reading a QR code using the camera

In order to read the QR code, there were largely two steps. Capturing the required image and then processing it and attempting to find a QR code in the image. With these two steps being repeated over and over. For capturing the image, as the simplest method possible, I attempted to replicate the 'Bash' script which would allow me to call fswebcam and capture the image, with all the required flags, such as setting the location for the image to be saved to, and the resolution of the image to be captured. I managed to do this, by making use of the os.system library which would allow me to specifically call a bash function. After this, I began to look for libraries which would manage to find a QR code and decode it. A number existed, and as the most recommended I attempted to use OpenCV. This created a large number of problems for me, not least due to the fact that the installation was prone to failing, with it only succeeding on the fourth attempt. After this was complete, I managed to get the system finding any QR codes and breaking when the required data was found. However, I had the issue that this was highly time consuming, as OpenCV was not very efficient. Hence, I looked for other possible libraries which might be faster, stumbling upon QR-Tools, a third-party library developed by a Spanish engineer, which appeared to be highly promising. As well as being much simpler to install and use, it was far less time consuming, with the finding and decoding of a QR code generally taking in the vicinity of a 50ms. However, the fswebcam script was less efficient, taking at least 250ms to successfully take an image. Though I attempted to experiment with other imaging libraries such as adapting the far better PiCamera library (which I had used with the Raspberry Pi Camera which was the original choice of camera), I found no other feasible options, and was forced to stick with the fswebcam script, which though time consuming was effective.

6.3.3.6 Backing up passwords to a USB stick

In order to compete with the other options for password management – including word processing and dedicated password management solutions, it was important to ensure that my product met the advantages over paper that those systems offered. One of the major advantages of this was the ability to automatically backup passwords, so that the passwords would never be lost even if there was a corruption of the original. Though given the offline nature of the product, it would be impossible to back-up the data directly to the cloud, it was still possible for data from the product to backed up both locally and to a USB stick.

Theoretically, this offered a dilemma, how to back-up the data. Should it be backed up in the encrypted form or in the decrypted form?

Encrypted Form

If the passwords were backed up in the encrypted form it would be entirely impossible for any hacker or indeed the user to access the password without having a product of mine, as no other product would be able to reverse the encryption, without knowing how the key was generated. While this was acceptable, it offered the problem of whether two products would have the same key, whether a second product of the same person would use the same encryption key as the first, as this would be required were the product to be able to be decrypted. In order to discover whether if a person scanned the same finger twice whether it generated the same exact template – I carried out an experiment – doing this 10 times with the same finger, attempting to place it in as similar a position as possible, before recording the template that the fingerprint sensor generated on enrolment. The results were rather astonishing, with each and every template being generated sharing few bytes. It was clear that even small changes in the position of the finger on the fingerprint sensor made large differences in the template generated. Hence, it was clear that I could not rely upon one product decrypting another products passwords even if the finger used to generate the key to the products was the same.

Decrypted Form

This meant that the passwords would have to be transferred in a decrypted form, as then this could be read by a new product, and then imported and combined with the existing list. Though this also had the advantage of allowing the user to look at the passwords using another tool such as a word processor, it posed quite a challenge to me, since it was clear that this would reduce the security of the passwords, if there was an unencrypted file available. Hence, I also considered using a specific key, which was known by the programmer alone for ensuring that the passwords were always encrypted, even at this point, and therefore the security would not be reduced unless this key was to be leaked. It was clear that this solved a number of the problems with the decrypted backup system, as it meant that nobody could intercept the passwords along the way and easily open them. However, it increased the impetus on ensuring that the code behind the product was unable to be seen by hackers (or anyone) as simply looking at this code would allow them to see the key that was being used for this backup. Therefore, I had to look into mechanisms to ensure this would not happen.

Delete all code when product is opened: Since there is a mechanical latch which when opened would delete all the passwords from the fingerprint sensor as well as ensuring the passwords on the device were encrypted, we could also additionally delete the code that runs the product at this point, as this would prevent anyone from opening the product, getting the SD card and looking at the code. However, this meant that this product was now permanently dead and could never be used again as it had lost all the code that governed how it ran.

Encrypt the SD card's content: By encrypting the content of the SD card, it would ensure that any code and / or information on the device was secure and could not be viewed by anyone opening the product.

It is clear that though the first might be effective, the problems associated with accidental opening of the case, which would prevent you from ever using that product again means that the second is a more preferable choice, with it making the entire system more secure, not just the code!

Practical Application of backing up

Now that the decision had been made to encrypt the SD card[190], it was clear that the only feasible backup location would be an external USB port, which was available on the Raspberry Pi. Therefore, the software had to be able to mount a USB drive and copy an encrypted file across to this USB disk.

[190] Despite originally expecting to find a number of solutions to allow me to encrypt the SD card and decrypt it on login, I was only able to find one possible solution – which involved encrypting the SD card using PyCrypto. Therefore, since all the available encryption techniques were available to me, I decided to go for the Advanced Encryption Standard again since it was by far the

Assuming there was only one USB stick I could assume that the USB stick would be under /dev/sda, while the system could assume the USB only had one partition, hence /dev/sda1, however this later created problems with USB sticks which contained a small first partition which stored information about the USB stick and its contents. This was resolved by instead searching for the largest partition of /dev/sda assuming that is the working partition. The system is then able to create a new directory under /mnt, with the name of the folder following a set pattern, 'USB1', 'USB2', etc. After this the only thing left to do was to practically mount the USB stick which could be done by a set bash command (again carried out in Python through using the os.system library).

Once the USB was mounted in a known location, copying a file was relatively simple, simply requiring another bash command using 'cp filefrom fileto'.

6.3.3.7 Combining the systems

Once each system had been made to work on its own the next step was to combine the systems together, to produce one complete program, which would control everything. Firstly, the screen and rotary encoder were brought together to help to design a complete User Interface, which consisted of a Main Menu, which included an alphabetically ordered list of all the passwords, and a Settings Bar which included all the required settings such as: showing the Serial Number, importing data from a USB stick, importing data from QR code, backing up data to a USB stick and informing the product that the user was about to open the product. After this was completed, the fingerprint sensor was added to the system, which enabled the locking to be added to the main program. After this, the camera system was added, with a switch to emulate a mechanical latch also added. The final step was adding in the power switch, which rather oddly was in this case not connected directly to the battery – as is normal – but to the Raspberry Pi to ensure that shutoffs would lead to the product turning off safely ensuring that the passwords are encrypted.

Once this was all complete, large parts of the final program had to be written, especially involving the way in which certain actions would influence the product. During this phase of prototyping, there were a number of problems, notably with the Adafruit OLED screen which appeared to break, hence meaning the entire system could not be made to function as complete until a new screen was acquired. More than this, the use of Python made this harder as one could not easily tell what was failing, with a large number of rather incomprehensible connection failures which meant that the complete program would not run. Despite this a complete program has been made, as can be seen in the Appendix, which though was working on the prototype unit, a number of additional parts have been added, which have not been able to be tested in completeness.

---

most secure. During the turn on phase, there would a nano script (in the startup configuration script) to decrypt the entire device, and run the main code for the product, while in the turn-off descriptions, the product would encrypt the entire device again.

7.0 Large scale manufacturing

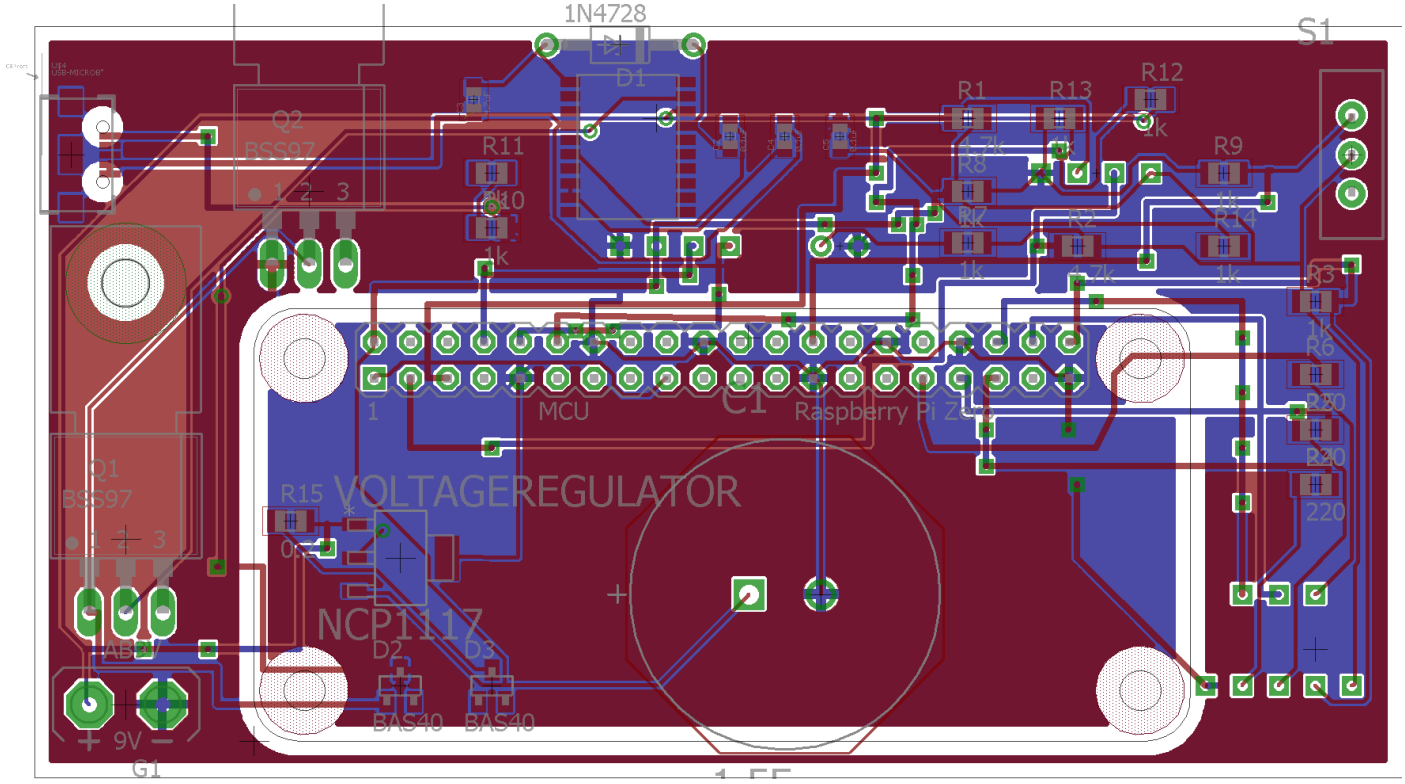
In my proposal form, I discussed the idea that I would carry out some research into how my product could be placed into large scale manufacturing runs, ensuring the idea could move past the point of prototypes and to a real-life use.

7.1 PCB Design

Given the complexity of the electronics, it would be prudent to design a PCB, as this would allow all the passive components to more easily fit onto the board. This is far better than the alternative of using a large number of loose wires, since this would be very space consuming as well as messy. For the PCB there were a number of possibilities, including the number of layers which could be used. The more the layers, the easier it would be to route all the necessary connections, but layers also take more space. In fact, the highest value for money (according to the PCBTrain – a large British PCB manufacturer) in terms of layers / pound for up to 1000 PCBs is the 2-layer PCB, hence I decided on it. This is because this is by far the most common and hence the cost per unit of the PCB reduces significantly.

In order that this be possible, I designed a two-layer PCB using Eagle CAD, hence:

Figure 109: Complete PCB Design



Source: Made using EagleCAD[191]

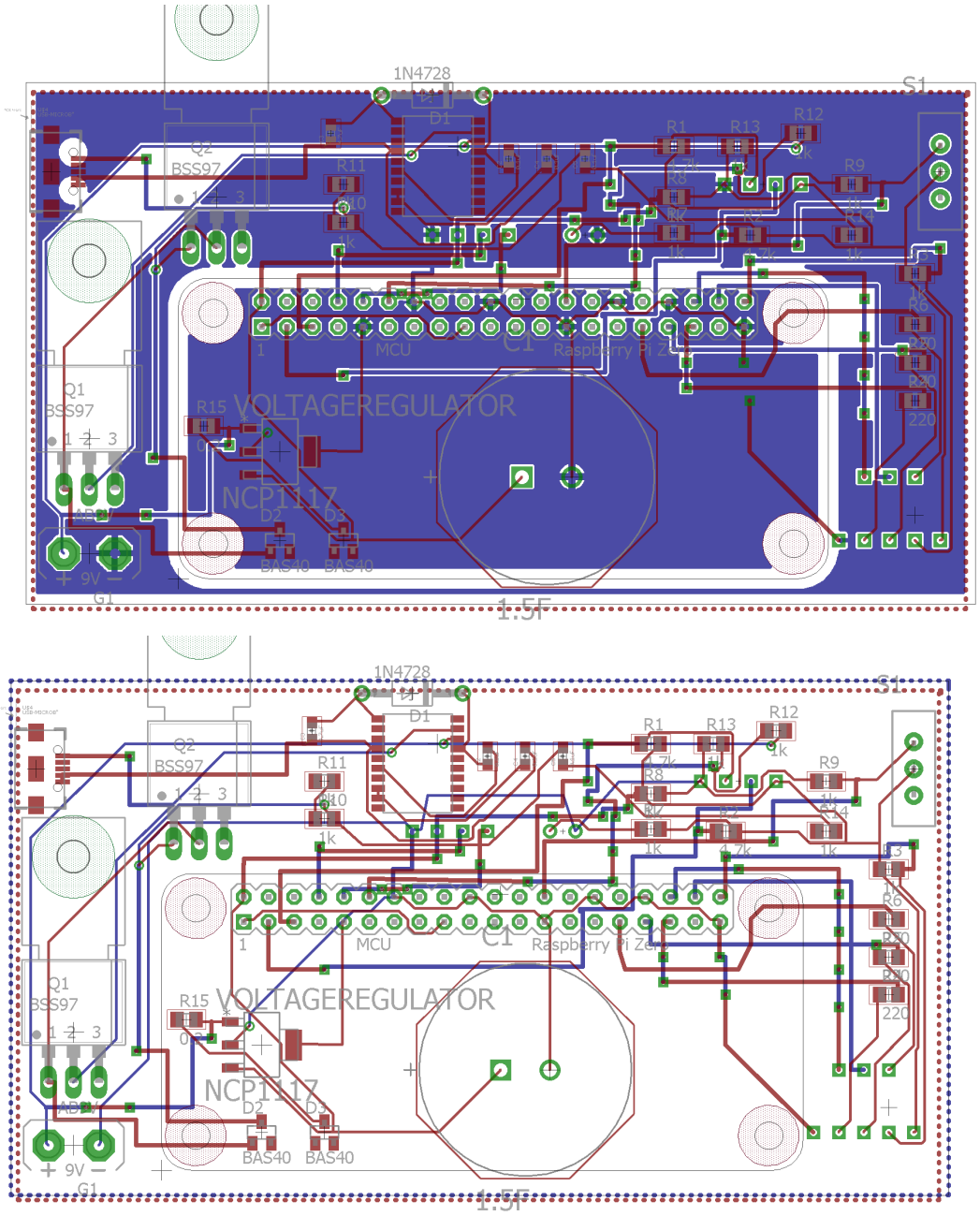
It is designed such that all the required components can either be mounted on the board in the case of the passive components or in the case of many of the other components can have their wires soldered directly to the board (for the power switch, rotary encoder, LCD, fingerprint sensor and the camera). As recommended, the PCB meets the design rules of PCBTrain indicating it could be manufactured without any problems. All the components are mounted on the top, with the Raspberry Pi itself mounted on the bottom with pin headers (the holes for which are easily visible). The rotary encoder wires go to the bottom right, with the design such that the exact look of the rotary

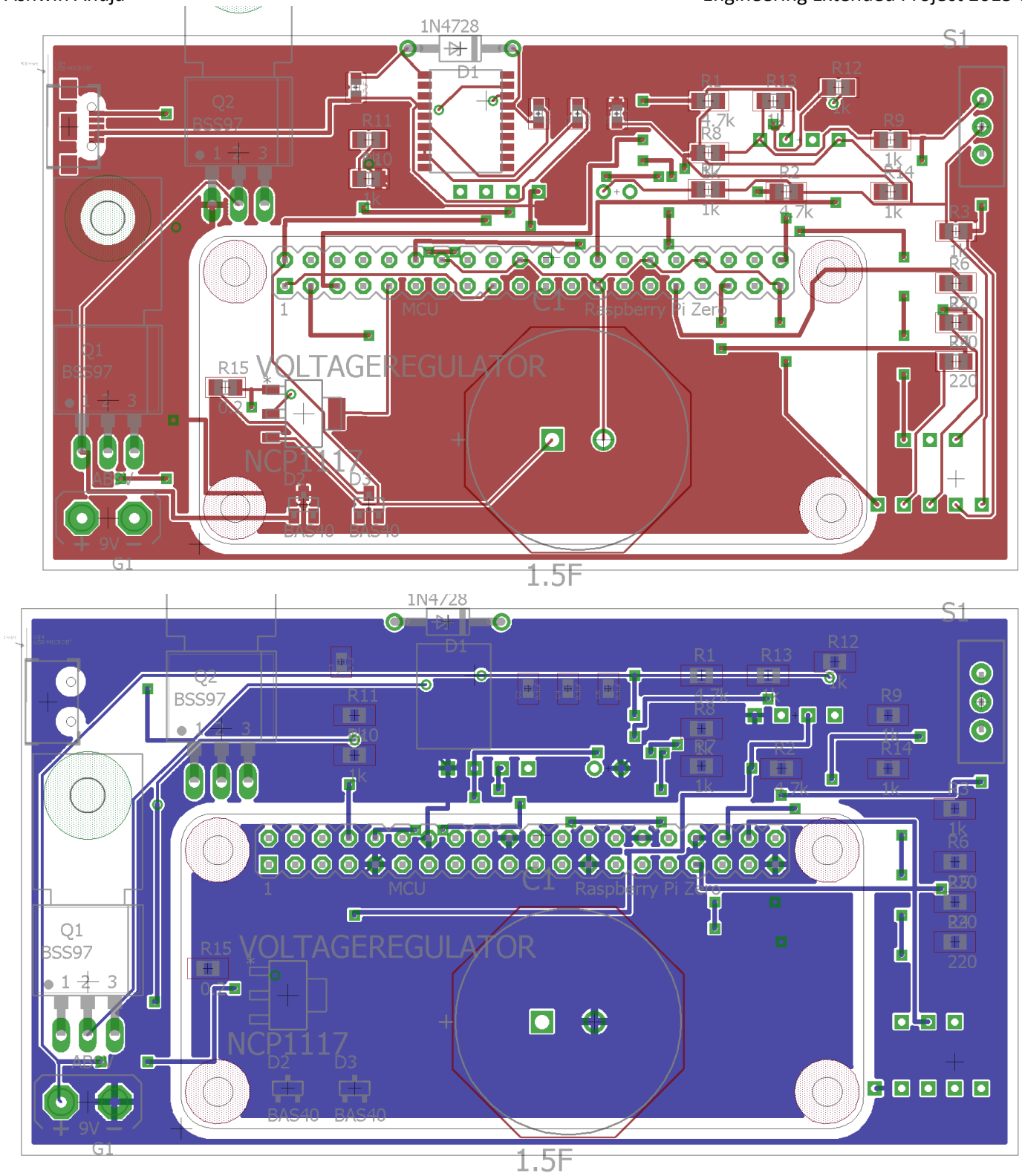
[191] (CadSoft USA, n.d.)

encoder is preserved to reduce the problems with wrong pins that could occur. Meanwhile the four pins of the LCD and the Fingerprint Sensor are mounted in the top part of the PCB, therefore allowing them to easily reach the components themselves which are in these locations. The vast majority of the power system is in the left and top middle, with the MAX745 mounted in the top middle. Meanwhile the Schottky Diodes and Power MOSFETs required for the chip take up much of the left hand side, while the Voltage regulator is in the centre, above the Raspberry Pi, next to the large Supercapacitor.

In addition, to ensure the product is safe, there is a ground plane at the bottom, ensuring that the Raspberry Pi is at no risk of being damaged if there is any contact (though this must be attempted to be avoided) between tracks on the Pi and the PCB. Additionally, though less important since there are no components below it, the top layer also has a ground plane.

Figures 110, 111, 112 and 113 Additional Images of the PCB





Source: Made using EagleCAD[192]

Manufacture of the PCB:

The manufacture of the PCB will follow the manufacture of most PCB today[193], hence using the following stages:

[192] (CadSoft USA, n.d.)
[193] (MadeHow, n.d.)

1. Woven glass fibre will be unwound from a roll and impregnated with epoxy resin, through dipping the fibre itself in the resin. This will then be fed through a roller, to gain a consistent thickness as desired. The fibre plates are then passed through an oven where the glass fibres cure before the plates are cut into large consistently sized panels. These panels are stacked in layers alternating with copper foil, before the entire structure is placed in a press where the temperatures are very high (generally around 170 degrees Celsius) and pressures are similarly high (1500psi). This fully cures the resin and binds the copper foil to the surface of the fibre.
2. The panel is passed through a CNC routing machine, which drills the holes as the design files say, with the holes then being deburred to remove excess material staying inside the hole. Then, some molten copper is poured into each of the holes, to make entire hole conductive.
3. The panel is now passed through a vacuum chamber where a photoresist material is placed onto the top layer of copper foil. On top of this, the PCB mask (with the design desired) is placed.
4. The panel is exposed to intense ultraviolet light. Because the mask is clear in the areas of the PCB pattern, the photoresist in those areas is irradiated and becomes highly soluble. Once the mask is removed, an alkaline developer is placed on the PCB dissolving the irradiated photoresist, only leaving the parts which do not contain tracks of the PCB design.
5. The panels are then electroplated with copper. The foil on the surface of the panel acts as the cathode and the copper is plated in the exposed foil areas. However, parts of the panel which have a layer of photoresist still cannot act as a cathode and are not plated. Tin or a different protective coating would then be placed on the copper (again using electroplating).
6. The photo-resist is stripped from the boards with a solvent, before an acid solution is used to break away the copper foil left under where the photoresist was. Because of the tin protection on the rest of the PCB, it is unaffected by this acid.
7. The panel is then sealed using Epoxy Resin, to ensure that the PCB is stable.

Once the PCB has been made, there are two main options for the adding of components to the PCB. Either it could be done by hand – and much of the PCB has been designed for this, with the components being large enough to solder by hand, with the minimum SMD size of passives being 0805, which is relatively easy to solder by hand. However, were the product to be sold in larger quantity, this process should be automated, as it would save a considerable amount of time and therefore money, since people would otherwise have to be hired to complete the soldering. This would largely be completed using Reflow Soldering, where solder paste would be placed on the pads of the components to be soldered. Then, using a CNC pick-and-place machine, the components would be added to the PCB, on top of the solder paste. Then, the entire PCB would go through a Reflow Oven, which would heat up the solder paste, making it into solder and joining the pad to the component, since there would be an attraction between the hot solder and the conductive component. Another method which could be used, especially at even higher quantities, would be an extension of a pick-and-place machine, where the component would be soldered down by an attachment at the exact time when the component is placed down. This however, requires much more expensive machinery than the Reflow Soldering and hence is only economically viable when the product is successful, hence warranting larger quantities of PCBs being manufactured.

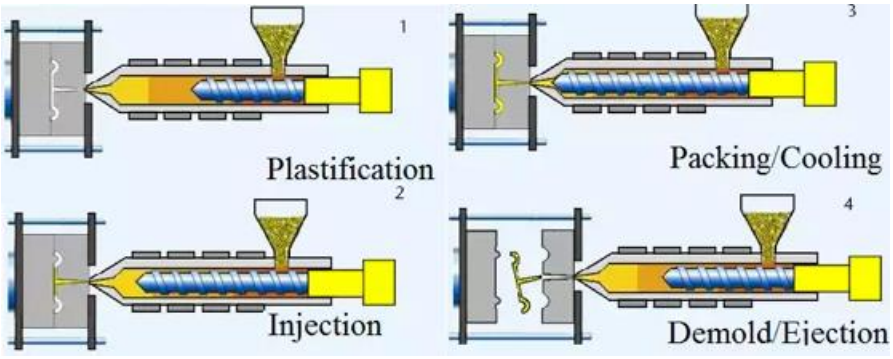
7.2 Mechanical Manufacturing Methods and Materials

Now that the geometry of the product has been defined, it is important to turn to the matter of various manufacturing methods which could be used to manufacture the product. As we have already defined the materials as from the plastics, this vastly reduces the number of manufacturing methods which need to be investigated, and hence we can present this in a table:

7.2.1 Injection Moulding

Granules of a material are preheated until they can be pressured into a pre-made mould. The plastic is allowed to flow around the mould and cool, thereby forming a plastic in the desired shape.

Figure 114 Diagram of the Process of Injection Moulding



Source: Intechopen[194]

Materials: Generally used on thermoplastics, as the plastics can be heated and cooled a number of times, thereby reducing wastage. However, the process can be conducted on other plastics, including thermosetting plastics, and even glass.

Advantages:

- Very fast process, with individual products being produced once every 15-30 seconds once a complete assembly line is made.
- Once the setup is completed, the same parts could be used on different colours and even different materials.
- The running costs of injection moulding are very low, since the vast majority of work is completed by machines.
- There is generally very little waste in the process, thereby being relatively good for the environment.

Disadvantages:

- There is a high initial cost, since one needs expensive machinery and also to produce a number of moulds, which might require CNC equipment to ensure that the mould is exactly as desired
- There are some restrictions to the parts which can be made, given the limitations of the manufacturing process.
- There will necessarily be some imperfections, especially due to there being one point at which the plastic must be added to the mould. Additionally, there can be more imperfections formed during imperfect (not very consistent cooling)

7.2.2 Blow Moulding

Blow moulding is a similar process to injection moulding however, it is a process that allows for the production of hollow shapes. The plastic is applied to the mould, before air is blown into the chamber which forces the liquid plastic to the very outside of the mould. Then the plastic is allowed to cool, and the mould is removed from the outside, the desired shape is present. Blow moulding a process often used for plastic bottles.

[194] (INTECH, n.d.)

Materials: According to the British Plastics Federation, the list of plastics that can be used includes: Polyethylene (PE), Polypropylene (PP), Polyethylene Terephthalate (PET), Polyvinyl Chloride (PVC)

Advantages:

- Produces hollow objects is easier than with injection moulding since a closed mould is not required.

- It has many of the same advantages as injection moulding in the low running costs, speed, ability to change materials and colours, and lack of waste in the process.

Disadvantages:

- It shares the same disadvantages as injection moulding, with expensive machinery, high initial costs, restrictions to parts and imperfections during manufacture.
- In addition, for our project, it is clear that the process is not necessarily applicable, since there are few hollow areas, hence injection moulding as a very similar process might be more applicable.

7.3.3 Vacuum Forming

Vacuum forming is a simplified version of thermoforming whereby a sheet of plastic is heated to a high enough temperature that it is capable of bending. Then, the flexible, hot plastic is forced into the desired shape by use of a vacuum being generated underneath the mould. The mould is generally a solid piece, since more complex parts are likely to suffer from failures in forming.

Then, when the plastic is allowed to cool, the plastics remains in the same shape, in the shape desired around the plastic.

Materials: Materials which are suitable for vacuum forming are conventionally thermoplastics, with the most common material that is used being High Impact Polystyrene (HIPS).

Advantages:

- Fast and generally very cheap.

Disadvantages:

- High start-up costs with the requirement of a mould being produced.
- Additionally, there are a number of limitations in terms of the product which could be formed. It generally has to be very simple, with few complexities. Additionally, it cannot produce any gaps or hollow structures. This means that it would be unable to produce the required geometries for my product and hence must be ruled out.

7.3.4 3D Printing

There are a number of forms of 3D printer, however, they can generally be split into two types, additive and subtractive. Additive 3D printing is generally more common and much cheaper and consists of producing the product by building the structure up in layers, only using the required amount of material.

Meanwhile, subtractive 3D printing generally makes use of Stereolithography where a laser is used to solidify parts of a material which are desired, while the rest of the material is washed away, therefore leaving the desired shape.

This works as the laser is able to alter the bonds in the material, thereby forming a stable structure. This idea Is used for a number of materials from Resin through to metals in Selective Laser Sintering, which is relatively similar.

Materials: There is a large variety of materials which could be used in 3D printing, such as ABS and PLA (which remain the most common for FDM (Fused Deposition Modelling), Nylon, Resins, Polycarbonate and various metals.

Advantages:

- 3D printing offers relatively high quality models, with fewer caveats than other manufacturing methods.
- 3D printers are relatively affordable meaning it could be cheaper than Injection Moulding for small scale production.
- Additionally, there are very low start-up costs for the product itself, as there is no requirement for mould or suchlike.
- Finally, especially with FDM, there is very little material waste.

Disadvantages:

- 3D printing generally is a slow process, requiring hours rather than seconds to produce the average sized product. This means that the potential for producing a number of products quickly is reduced.
- Since materials require pre-preparation, production of products is generally more expensive than other manufacturing processes such as Injection Moulding.
- Models often require post-processing to ensure high finish quality.
- There are often a number of imperfections in the models, meaning the quality of the product is not always very high.

7.3.5 Laser Cutting

Laser Cutting is a technology which makes use of a laser to cut materials, and can generally be controlled by a computer, such that a desired design can be produced.

Materials: Laser Cutting has a very limited list of plastics that can be cut. According to Pololu who offer Laser Cutting services, the list of materials is: ABS (Acrylonitrile Butadiene Styrene), Acrylic, Delrin, High Density Polyethylene, Polyimide, Polyester, Nylon, PETG, PE, PP and Styrene, while many of them such as Nylon come with the caveat that they “melt badly” and so the printing quality would be reduced.

Advantages:

- Generally, relatively fast, with large 2D shapes cut in a few minutes.
- It is highly versatile being able to cut a vast variety of 2D shapes accurately.

Disadvantages:

- There are some imperfections, such as heat colouration that occur.
- Though there are some 3D laser cutting machines (which are very expensive and not very accurate), the machines generally only deal in 2D planes, hence would be unable to manufacture my product.

7.3.6 Decision and Further Discussion

In order to make the decision of which manufacturing process was the best to choose, I made a decision matrix:

Figure 115 Decision Matrix for Manufacturing Method

	Suitability		Speed		Cost		Accuracy		
	Weight	Mark	Weight	Mark	Weight	Mark	Weight	Mark	**Total**
Injection Moulding	10	9	8	10	7	9	8	9	305
Blow Moulding	10	0	8	9	7	9	8	9	207
Vacuum Forming	10	0	8	6	7	9	8	7	167
3D printing	10	9	8	2	7	8	8	6	210
Laser Cutting	10	3	8	5	7	9	8	9	205

The decision matrix clearly shows that Injection Moulding is by far the most effective manufacturing technique, and hence this should be used to manufacture my product.

From here, the next very important decision to make is which material to use, which we know must come from the thermoplastics or thermosetting plastics, since these are the materials which are suitable for Injection Moulding. Also highly important is the toughness of the materials, which is shown by this table, now only showing the thermoplastics or thermosetting plastics.

Figure 116a Table showing usable plastics and their toughness, density and cost

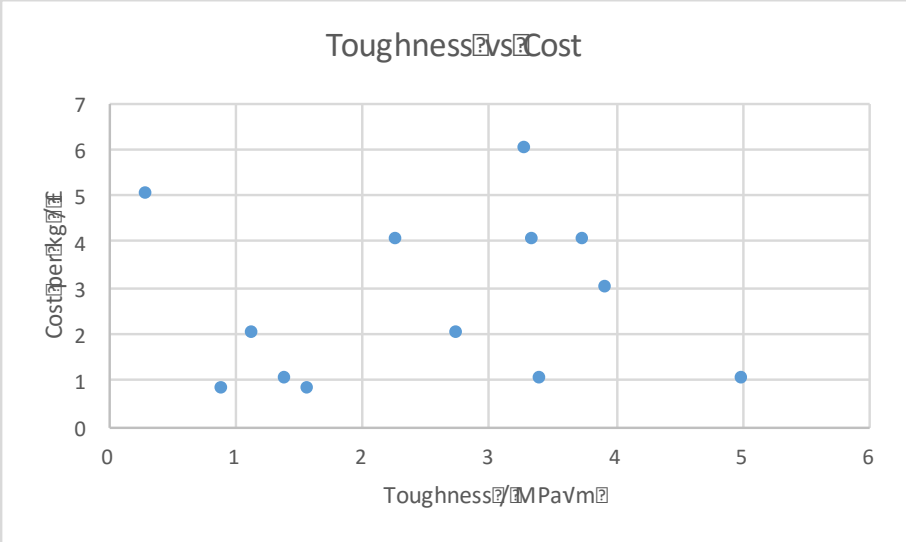
	Average Fracture Toughness / MPa√m	Density Average / gcm^(-3)	Cost per Kilogram / £
eIPU	0.3	1.135	5
ABS	2.745	1.11	2
Ionemers	2.285	0.945	4
Nylons	3.92	1.13	3
PC	3.35	1.175	4
PE	1.58	0.9495	0.8
PET	5	1.345	1
Acrylic	1.15	1.19	2
PP	3.75	0.9	4
PS	0.9	1.045	0.8
tpPU	3.405	1.18	1
PVC	3.29	1.44	6
Polyester	1.395	1.22	1

Source: Statistics from Cambridge University Engineering Department[195]

If we graph toughness vs cost, the most important figure, since the densities all appear relatively similar (with the exception of PVC), we find one standout material which has been mentioned before, PET, which has a very low cost as well as by far the highest toughness.

[195] (Cambridge University Engineering Department, 2003)

Figure 116b Graph showing toughness vs cost of the usable plastics

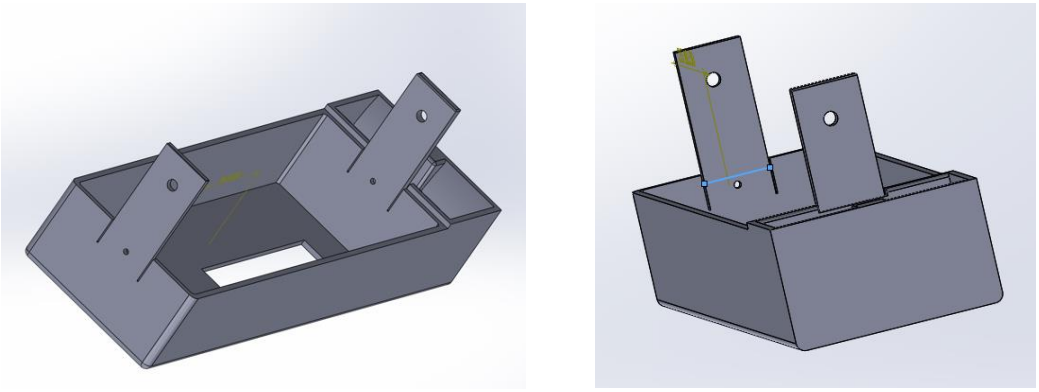


Source: Statistics from Cambridge University Engineering Department[196]

Therefore, the material of the product should clearly be Polyethylene Terephtalate, since it can be easily manufactured using Injection Moulding, while it offers the best combination of Toughness and Cost.

Hence, the exact design process can be defined, including the exact steps which would be required to ensure that the product could easily (without much further work) be mass produced. Hence, the exact requirements during the process of Injection Moulding must be considered. Foremost in this, is the requirement to produce a mould in which the molten plastic would be poured. In order for the plastic to be then removed, it is clear that the part could not be made as is, with the connectors for the back piece, since the mould from the inside of the shell could not easily be removed without damaging the connectors of the piece. Hence, it would be necessary for this piece to not stick into the main shell. There are a couple of methods of ensuring this, the first involving producing these pieces separately and making use of sticking methods such as glues to connect it to the main piece. However, this would clearly not be as strong, as well as increasing the time and cost of manufacture, since a number of more pieces would be required to be made. The other option involves bending the piece into place after the piece has been injection moulded. Since PET is a thermoplastic, the easiest and cheapest manner of doing this is using a CNC line bending tool, which would ensure that the piece would be strong, not losing structural integrity, while remaining relatively cheap.

Figure 117a CAD Design – adapted to show how it could be designed for Injection Moulding – line shows the bending point

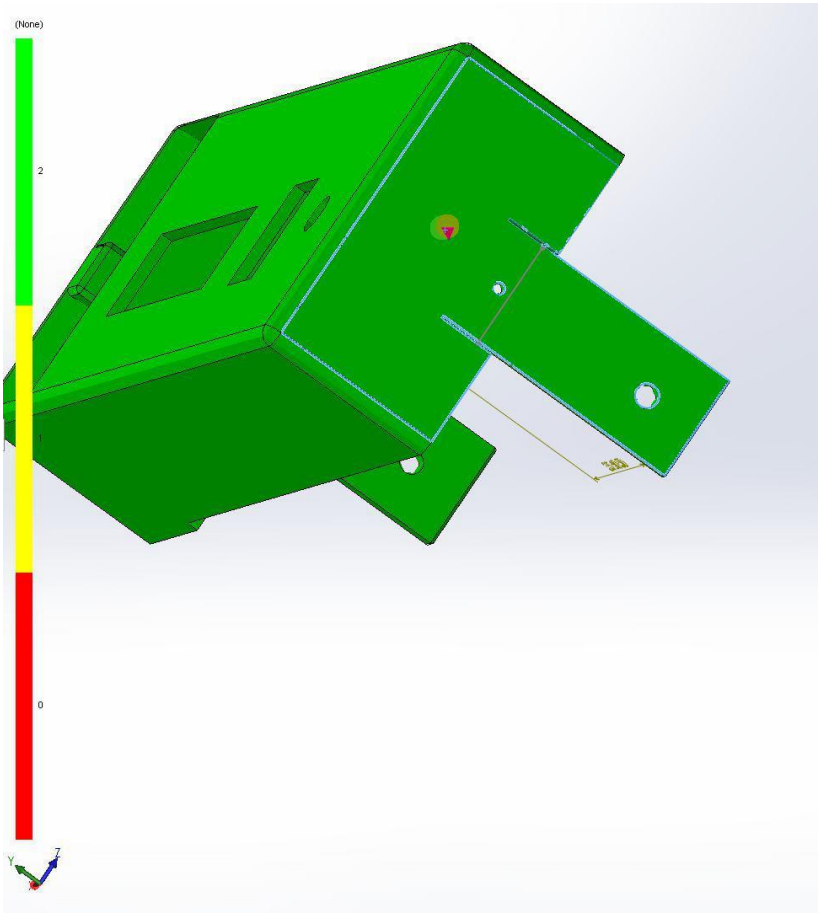


Source: Made using SolidWorks 2016

[196] (Cambridge University Engineering Department, 2003)

Additionally, it is important to show that the entire structure of the product could be easy to reach using Injection Moulding. Using the SolidWorks Plastics Simulation Tool, setting the obvious point in the main structure as the injection point shows that the entire product is easy to fill – ranking as the easiest on SolidWorks ranking scale. This makes use of relatively low temperatures – 260 degrees Celsius, and low pressures – 50MPa, while still producing very fast fill time, with the entire process (according to the simulations) taking around 2 seconds. It is important to be noted that with higher temperatures and pressures, even faster fill times would be highly possible. However, in converting the product to mass manufacture, it would be clear that a couple of problems must be considered in the design of the mould. Firstly, the problem of draft angles must be considered, to ensure the product is easy to remove from the mould. Additionally, Figure 117c shows how when using PET, there is relatively large shrinkage, this must be countered by scaling up the design according to exact shrinkages. Alternatively, the use of a number of injection points would allow the shrinkage to be made more consistent.

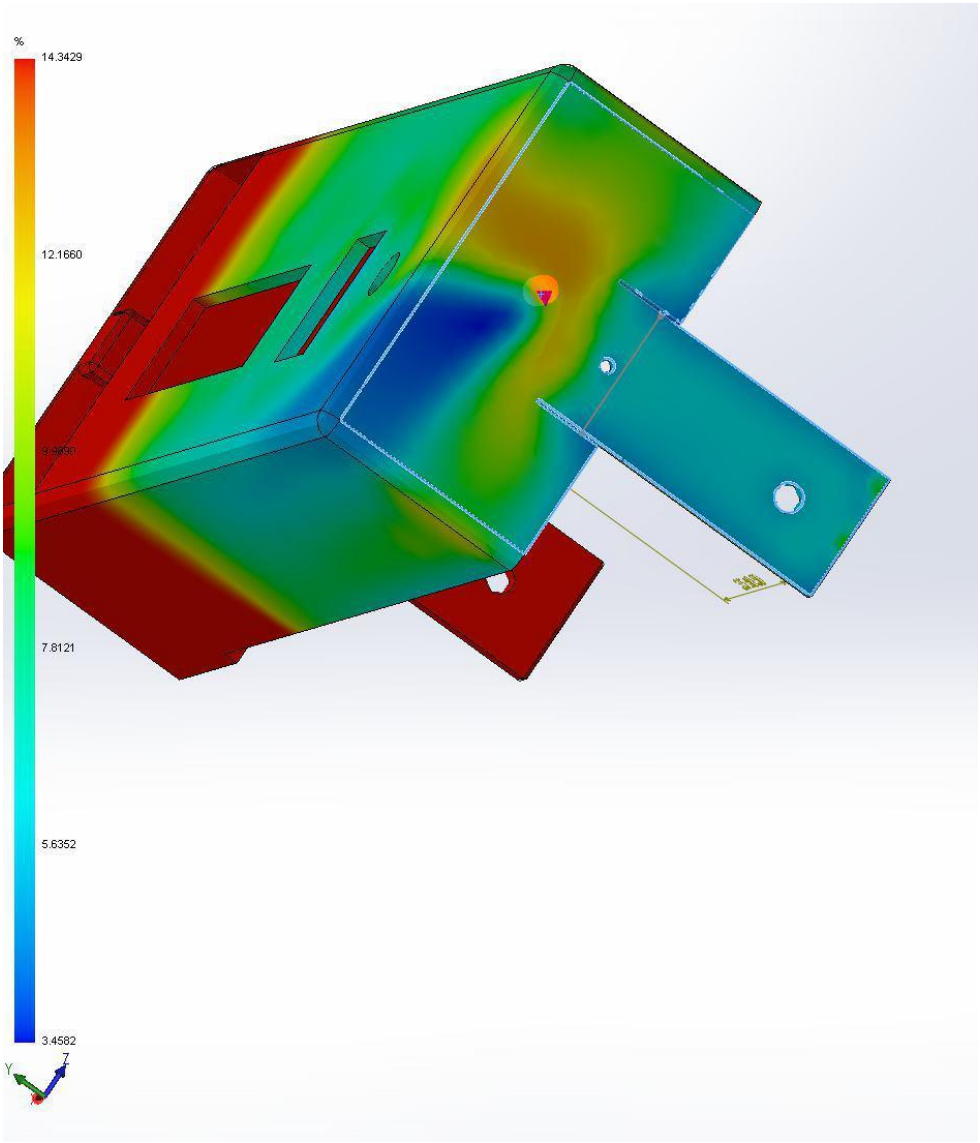
Figure 117b Injection Moulding Simulation showing ease of fill



Source: SolidWorks 2016 with SolidWorks Plastics[197]

[197] (SolidWorks, n.d.)

Figure 117c Injection Moulding Simulation showing Volumetric Shrinkage on end of fill



Source: SolidWorks 2016 with SolidWorks Plastics[198]

[198] (SolidWorks, n.d.)

8.0 Project Evaluation and Conclusion

8.1 Artefact Evaluation

8.1.1 Success in fulfilling the original brief

My original brief was: "To produce a design for the mechanics, electronics and software of a secure handheld method to store passwords, ensuring it is not hackable by current means – and thus more secure. It should make use of biometrics to ensure that the product only responds to the intended user. It should also be especially easy to use, to attract especially the elderly, who are likely to particularly struggle with this problem."

I was highly successful in meeting this brief as detailed below:

- My final artefact contains complete designs for mechanics and electronics and all the code required for the system to function.
- My final artefact is a handheld product with particular focus towards ensuring the passwords are not hackable. As shown by the various tests I made it was clear that the code was not easily hackable by current means.
- My final artefact includes plans to use a fingerprint sensor as a usage of biometric technology to help to ensure the product only responds to the intended user.
- My final artefact includes a complete user interface design with specific thought paid to ensuring the system is as simple for the user as possible. This is accomplished by ensuring the system includes an iPod like interface (a reason why it was chosen), with a list of items as well as layers of screens.

A summary of the artefact as well as complete renders of the mechanics, complete electronics designs and the code can be found in the Appendices.

In addition, it is clear that I was highly successful in my other project criteria.

Figure 118 Success in meeting aims

Aim	Extent of Success	Justification
To research current hacking techniques.	Complete	I conducted a detailed research into a number of hacking techniques, ensuring that I found why they were successful, and how they worked. I then completed research into whether they were relevant to me. From here, I also completed research on how people can design products to be insusceptible to specific hacking techniques.
To research different algorithms for encryption and their various merits and flaws.	Complete	I completed an overview of a number of the most popular algorithms for security looking at how they might be applicable and looked at their flaws, often mathematically looking at why they were successful.
To research the material sciences related to materials which could be used in the product.	Complete	I completed an overview of material sciences before specifically finding the characteristics which were important for me, and then spent more time researching them – making use of both Ashby Plots and raw data for various materials.
To research ergonomics.	Complete	I completed evaluation of products on the market which claim to be highly ergonomic, looking at the features which make them successful and unsuccessful. Then I completed research of

		anthropometrics, finding data that could be used to specifically produce a highly ergonomic design
To research different methods by which the product could be manufactured.	Complete	I researched the major types of manufacturing while I was designing the product, before completing a more thorough investigation once the broad material and geometry had been decided.
To learn how to use 3D simulation software to conduct simulation upon the materials to produce the product.	Complete	Throughout the design stages of the project, I made use of the Drop Test feature of SolidWorks, first gaining an understanding of how to use the SolidProfessor account which I had been allocated. I used this feature to help to evaluate the three ideas, finding any areas of improvements for the final design. I also justified many parts of the final design using these drop tests. In addition to this, I completed simplistic stress analysis on the various points of the device to ensure it could deal with tensile and compressive stresses.
To learn how to design algorithms for encryption.	Complete	In designing the algorithm for the QR code encryption cipher, I conducted research into how ciphers are designed (looking at lots of other ciphers) and also completed research into how I could evaluate my cipher against other encryption techniques.
To define and justify the structure and geometry of the product, as well as the materials used.	Complete	I generated three separate ideas for the structure and geometry, evaluating each one separately, before producing a final design which incorporated a number of their features. This final design is fully justified, with all important decisions mentioned and talked about. I conducted initial research into many materials before making my final decision about materials after choosing a manufacturing method. I justified this material using numerical data. In the appendix there is all the specific mechanical information which allows the reader to replicate the product if they want.
To define and justify the manufacturing processes required to produce the product.	Complete	During the period after defining the geometry of the product, I was able to evaluate the effective manufacturing methods, which could be used, based on the basic materials that I was prepared to use. A number of methods were compared, and a comprehensive weighted decision matrix was used to show why I had made the decision I did.
To define and justify the electronics of the product, to easily allow the user to find their password, and also add them comfortably.	Complete	Initially, I designed a complete control system, justifying all the decisions I had made, referring back to the purpose and target market of the product. During development, I defined and justified any required change, before producing a complete definition of the electronics (in the form of a schematic) with additional designs for a PCB.
To define and justify the main security algorithms used in the product to ensure it is secure and insusceptible to current hacking techniques.	Complete	At the very beginning of my research, I chose an algorithm for the encryption of passwords, ensuring that it met the need of being unhackable, and any additional security algorithms – that of encrypting the SD Card and also for the QR code encryption, I defined all main software algorithms used as well as justifying them, showing why they were successful. Additionally, in the software appendices, most of the code is present (with the rest in the submission folder), where all the other algorithms are used. Where useful, comments have been used to explain why a certain piece of code was used.

To define and justify the biometric system used in the product, to draw the best balance between the product being as secure as possible, as well as easy to use on a frequent basis.	Complete	After researching all possible types of biometrics, I chose and justified (using a decision matrix) one form of biometrics to use, fingerprint identification. In the decision matrices, security and ease of use were important criteria that were used. In the control system sections, I defined and justified how the system would make use of the fingerprint sensor as a manner of unlocking the device, and decrypting the passwords.

8.1.2 Success in meeting the specification

After my introductory research, based on the requirements of the target market and based on the other solutions currently available to the problem, I produced a specification for my product.

Figure 119 Success in meeting specification (primary specification in bold)

Specification Point	Extent of Success	Justification
The product must be able to store passwords.	Complete	The product contains a system by which passwords added to the device are stored on an on-board Micro-SD card connected to the Raspberry Pi.
The product must have a system through which the user could add new passwords	Complete	A complete system has been designed and produced by which using a mobile, web or desktop application, the user can produce an encrypted QR code with the password data. This QR code can then be scanned by the product which has an on-board camera.
The product should have a system to allow the passwords to be backed up – either automatically through the cloud or manually.	Complete	The product has a complete system to backup passwords, through the use of a USB stick plugged into the USB port of the Raspberry Pi.
The product should be easy to use – so that it could be used easily without much explanation by technology-averse elderly.	Complete	The User Interface is very simple to use, with finding passwords being the most important and simple thing to do, simply requiring the user to scroll down a list of passwords until they find the one they are looking for. In fact, when I showed the system to my elderly neighbours, they agreed with my assessment, calling the product's user interface 'perfect'.

The product should have a bright, high contrast screen to ensure it could be used despite any issues that the users might have with sight.	Complete	The product makes use of a backlit LCD display to ensure a high brightness, with the use of white on black text ensuring the greatest contrast.
Any sound used by the product should be loud and avoid making use of particularly high frequencies.	Not Applicable	Though originally considered, the product does not make use of any sounds, as it was felt to be unnecessary.
The product should use of biometric technology to ensure that the product does not require the user to remember a master password.	Complete	The product makes use of a fingerprint sensor (the GT511C3) as the master entry system, meaning that the user does not have to remember a master password.
The product should make use of encryption and other means to ensure that hackers cannot easily gain access to the passwords.	Complete	The product uses a number of encryption mechanisms (including the encryption of both the password and the entire Pi when sleeping) to ensure that a hacker cannot gain access to the passwords.
The product's control system should be designed to ensure that it could be used easily with reduced fine motor skills.	Complete	The product makes use of a rotary encoder as the main input device on the system, allowing it to be used with reduced fine motor skills. The rotary encoder does not require said skills, also providing large amounts of tactile feedback to the user.
The product should be light, and easy to handle.	Complete	The product is very small and light with the size having been minimised as much as possible. When deciding the material, the density was also minimised to ensure that the weight was also minimised – while ensuring that the toughness remained high enough.
The product should be designed with reference to ergonomics and anthropometric data, hence being comfortable in a user's hands.	Complete	The product is based on successful ergonomic products while also making use of anthropometric data to ensure that it fits comfortably into the hands of the users.
The product should be able to withstand normal,	Complete	The product is made of PET, which has a very high toughness and resistance to stresses (tensile, compressive and shear) ensuring that it could easily cope with indoor use.

<table>
<tr><td>indoor use easily.</td><td></td><td></td></tr>
<tr><td>The product should be able to withstand being dropped, as this is something which is likely to happen on occasion.</td><td>Complete</td><td>The product has been simulated being dropped from varying heights, only appearing to break (taken above yield point) from heights above around 5m.</td></tr>
<tr><td>The product should be relatively cheap, definitely under £50.33, the average maximum price that the elderly people I asked said that they would pay for such a product.</td><td>Complete</td><td>
<table>
<tr><td>Raspberry Pi</td><td>£</td><td>5.00</td></tr>
<tr><td>Fingerprint Sensor</td><td>£</td><td>24.91</td></tr>
<tr><td>Screen</td><td>£</td><td>4.70</td></tr>
<tr><td>Rotary Encoder</td><td>£</td><td>1.67</td></tr>
<tr><td>Ultracapacitor</td><td>£</td><td>1.25</td></tr>
<tr><td>Voltage Regulator</td><td>£</td><td>0.89</td></tr>
<tr><td>Charger IC</td><td>£</td><td>5.36</td></tr>
<tr><td>Assorted Passived</td><td>£</td><td>4.00</td></tr>
<tr><td>Rocker Switch</td><td>£</td><td>0.25</td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td>TOTAL</td><td>£</td><td>48.03</td></tr>
</table>
The costing of the parts of the product are as shown, therefore meeting the maximum price by just over £2.
</td></tr>
</table>

8.1.3 Possible improvements

The following aspects of my final artefact could potentially be improved:

- The product currently only caters for those who are right handed and only comes in a single size, so the hand spaces are relatively large (so everyone's hands could fit into the spaces). Therefore, the product could come in a range of sizes to ensure the best fit for people with small hands, and those who are left handed. This would increase the target audience therefore allowing it to reach even more people.
- The product currently only has one size screen, in order to make it compact and due to the complications of the screen requiring to connect over I2C. In future, the I2C to Serial communications could be built into the PCB allowing the increase in size of screen, since this was a major concern when showing the prototype to a few old people in my area. In fact, the product could come in a range of screen sizes to allow the individual to choose the balance between portability and screen size.
- I did not define anything for the specific colour of the product. It would be worth considering whether the product could be made in a range of colours to allow the user to choose their favourite colour, using the product as an extension of their personality. This is generally very common in other electronics, with most phones coming in a variety of models. Even the iPhone, despite Apple championing the art of simplicity and few models, comes in a variety of colours, from white and black (Space Grey) to pink (Rose Gold) and gold.
- In future, I would more closely consider the usage of sound in the product as this was not very well investigated during the production of this device. Talking to an elderly neighbour at the end of the process, he said that sound would be highly useful, providing even more feedback when using the product, in addition to the tactile and visual feedback already provided by the product. This would be very simple, simply requiring a single Piezo, but could make the product even easier to use, especially for those who suffer from vision problems.
- In future it would be worth considering producing the fingerprint sensor and indeed Linux Computer (both of which were beyond the scope of the Extended Project) in house, as this would help to reduce the cost and allow the entire system to be further compressed.

8.2 Project Management Evaluation

8.2.1 Time Management

Given that I have been highly successful at fulfilling my brief, I think my time management was relatively successful. A lot of the credit for that goes to proper planning and management tools, without which I would have been unable to do this. However, there were a number of failures in the time management which I could allocate largely to a naivety about some of the complexities of the project, and hence taking longer than I had expected especially in the development phase of the project.

8.2.2 Reflection on Process

Prior to the production of a plan, during the preparation, I suffered, as I had a lack of efficiency, working on a number of tasks at the same time, including choosing a problem while attempting to research various basic elements of Engineering, such as SolidWorks and theoretical Engineering in classes. If I had chosen my problem earlier this would have been better, as it would have allowed me to focus my work more on the relevant parts. In fact, I thought my EP would have little requirement of Mechanics and hence did not work hard on SolidWorks – therefore meaning I had to learn complex features when I came to use them in the final CAD, including Lofts.

Once I had developed a plan, I was largely able to follow it successfully, ensuring that I remained on course to complete the project on time. However, as I started completing the Control System and Mechanical research, I struggled to meet the deadlines I had set putting the work back. In particular, research of encryption was found to be far more complex than I had expected, as I dedicated time to the fundamental mathematics behind the cipher. Much of this was not entirely relevant to the project, and I would not use it, and so I could have prioritised the other pieces of basic research, including the more vital Control System research. Additionally, time had to be taken out during the middle of this research to complete the Preliminary Research Report. In retrospect, this time could have been better organised, ensuring that I focused my energy on finishing the research before my first deadline (in the middle of December). Additionally, I missed out on some important pieces of research in the beginning, notably anthropometric and ergonomics research, requiring this to be done later, during the design phase of the project. This could have been fixed by some more careful thinking in the planning stages of the product, as I envisioned the possible handheld product. With regards to materials research, I had a number of problems sufficiently understanding the problems which would affect my product, requiring a longer period of time than I had envisioned to do this. Even once the fact that the toughness was the large factor was determined, there was a challenge in finding appropriate data which could be used to make a properly considered decision. However, with additional research, I was eventually able to find a useful document made by the University of Cambridge, which included lots of materials data (including toughness). Finally, I struggled on a number of decisions, notably the decision of the Raspberry Pi over an Arduino, which pushed all the other work and initial design work back. This was because largely I knew that the Pi would be more effective (as it is much more powerful) but I had far less prior knowledge using Pis compared to Arduinos. It would have been useful to be slightly more decisive in these cases, simply making a decision, and working with it, given that I had found all the required information. In fact, I had already produced a decision matrix in order to make the decision, which showed that the Pi was better, yet I was still undecided due to the risk it represented. Additionally, in this phase, I was inefficient in the manner in which I completed a number of tasks at the same time. This in many ways meant that time was spent attempting to recap research which I had already done.

The initial design phase of the project was among my best; I designed the initial electronics relatively quickly - meaning the development phase of this could begin ahead of schedule. At this point, the highly comprehensive research (taking longer than I had expected) became very useful, as most decisions had been made, thus dictating the electronics design. The mechanics ideas as well progressed very quickly allowing me to more comprehensively evaluate each idea than I had expected, including completing a comprehensive simulation of each idea for drops,

making use of the inbuilt abilities that SolidWorks had to do this. I did however, have a little trouble with the CAD designing element for each idea, as I had failed to do requisite preparations, and was forced to return to SolidProfessor specially to understand how to make use of Revolves and Sweeps. This was also true for my final idea, with the complex structure of the handle and the requirements of using Lofts. However, the access to tutorials such as those on SolidProfessor made this much simpler.

During the next phase of the development of my artefact, I was short of time, struggling to ensure that I met my deadlines. This was at least partially due to the amount of time taken by other Engineering projects. However, a large part of the issues was simply the unexpected complexity of a number of the actions. A number of aspects of the project were complex than anticipated, especially electronics and software of the prototype, programming in a programming language with which I had modest experience, Python, meaning more than preferred work was required into how to complete certain programming tasks. With regards to electronics, I was often forced to complete more research into exactly how the components worked, sometimes testing them with an Arduino before moving to a Raspberry Pi, due to relative simplicity of the system as well as the existence of sample code for Arduino, in the case of the Rotary Encoder. Moreover, this time was cut short as the parts arrived late, and in fact one did not even arrive at all (the screen), since it would not arrive before March, by which point I would have finished the project. This was largely due to the delays in Control System research and could have been resolved by prioritising this research, ensuring it was done earlier. It would have been important to understand the ramifications this research would have on the prototype. Though I was aware of the dependency in the plan, I had allocated no time for the arrival of the components, hence this put me behind schedule.

At this point, I was couple of weeks behind schedule, and hence was quite time constrained in the production of the final design and completing the development of the code, due largely to excess prioritisation of small issues in the prototype. Hence, less evaluation of the final design than I expected, meaning a few of the decisions could be even further developed, such as the point at which the device would fracture.

The write-up took much longer than I had expected, since I underestimated the amount of writing I would have to complete. Due to the breadth of the project, there were a rather large amount of decisions which I had to make and hence justify, especially in the Control System Research section, which requires the discussion of most components, a number of which were not very well discussed at the time of decision. Hence, a greater appreciation for this at the time of making the decisions would have prevented problems such as these.

Finally, the evaluation was completed rapidly and quickly, though it was pushed later due to the delays during the CAD Design stages. The initial specification and criteria established in the Project Proposal Form proved useful in showing that I had met the brief that I had intended to. Some time was required to think about changes that would be made to the artefact if I had to do the process again.

8.2.4 Project Management

Throughout the project, I found the planning that done very useful, as it offered structure to ensure that I completed work on time, and motivated me to ensure that I was not forced to do large amounts of work towards the end of the project. Particularly useful was the complete task list (Appendix A), which proved as a checklist of sort to ensure that at the end of a section, I would ensure that I had met the expectations of the task. However, this also proved as an issue of a sort, as Anthropometric Research was delayed as I had forgotten to include it when first making the task list. Also useful was the Gantt Chart, which allowed me at a glance to see the dependencies and the overview of the success of the project, ensuring that I was not falling behind my plan. This was also particularly helpful when I was delayed completing a task, such as the research into encryption, as I could easily plan my time by altering the Gantt Chart, while ensuring that important tasks (which was a prerequisite for other tasks) would be completed as a priority. The Gantt Charts are available in Appendix B as well as the various major changes that occurred between

the various iterations. However, I encountered a problem with the scale of the Gantt Chart, since I largely planned tasks for each week, meaning there were sometimes a number of tasks that were being completed at the same time. This meant that I chose which one I wanted to do, creating issues where less desirable tasks – such as the final CAD design were delayed in order that I could focus on the prototype which I found more interesting. Hence, in future, it would be useful to allocate time by the day, which would ensure that the required task be completed. Additionally, among the extremely effective mechanisms for ensuring that I did the work were the two Project Milestones, for the completion of the research and for the definition of the artefact. Hence, in another such project, I would make use of more regular goals and milestones which would encourage me to work towards and beat them.

8.2.3 Things I would do differently

- I would have benefited from choosing the best possible problem to complete earlier as this would have allowed me to tune my preparation towards ensuring I had the requisite skills to complete the task, reducing the amount of work required later.
- During the initial phase of the research, a greater appreciation for the important features of the project would have improved it, including ensuring the control system research was completed early to ensure time for the arrival of the parts for the prototype.
- In the same way, an appreciation for the necessities of the project notably the lack of requirement of the mathematical details about the ciphers, which though interesting, proved to add little to the decisions made, and in many parts do not feature in the report, despite the large period of time dedicated to them.
- Additionally, it would have been useful to ensure that the decisions were better documented (outside of my rough notes) as this would have relieved the stress that occurred during the period of the Preliminary Research Report and the Final Report. This was particularly important with regards to specific component decision, as there were often small but important reasons where a component was chosen, but this decision was not properly written down. This required the completion of the same research at the point when the final report was being written.
- During the development phase of the project, it would have been useful to give more priority to the CAD as opposed to finishing the prototype as this led to time being short for adequately finishing and evaluating the final design. This in many ways was due to my own personal preference to work on Electronics and Software and could be fixed by allocating tasks for specific days instead of a number of tasks for a period of time as I did.
- Additionally, I was rather shabby in the ordering of components, in fact completing the initial ideas phase of the mechanics before ordering the components. This in many ways was due to my forgetting to add this to the task list.
- A greater amount of attention could be paid during the preparation stages to ensuring that I had the required abilities of CAD. I had thought that this would likely pay a quite small part in the project, and so my skills that I acquired were rather minimal and hence I was required to (re)learn topics such as Lofts and Revolves when designing both the ideas and the final design.

9.0 Bibliography

AARP RealPad, n.d. *AARP RealPad.* [Online]
Available at: http://www.aarprealpad.org/
[Accessed 14 02 2016].

Adafruit, n.d. *Monochrome 1.3" 128x64 OLED graphic display.* [Online]
Available at: https://www.adafruit.com/product/938
[Accessed 06 03 2016].

Adafruit, n.d. *PiTFT - Assembled 320x240 2.8" TFT+Touchscreen for Raspberry Pi.* [Online]
Available at: https://www.adafruit.com/products/1601
[Accessed 06 03 2016].

Ahmed Sabbir Arif, W. S., 2013. *Pseudo-Pressure Detection and Its Use in Predictive Text Entry on Touchscreens,* Toronto: s.n.

Ahuja, A., n.d. [Online]
Available at: http://twitter.com/ashwin_ahuja

AimBrain, 2015. *AimBrain, the Mobile Biometrics Platform.* [Online]
Available at: https://aimbrain.com/
[Accessed 10 03 2016].

AlphaRubicon, n.d. *Battery Sizes.* [Online]
Available at:
https://www.google.co.uk/url?sa=i&rct=j&q=&esrc=s&source=images&cd=&cad=rja&uact=8&ved=0ahUKEwjXraTS8tXLAhVFng4KHXmBDxUQjRwIBw&url=http%3A%2F%2Fwww.alpharubicon.com%2Faltenergy%2Fbatteryfred.htm&bvm=bv.117218890,d.ZWU&psig=AFQjCNFQHDemSmYLZHNmn0kx_OV8FGTLSQ&ust=1458791133243559
[Accessed 06 03 2016].

Alzheimer's Disease International, n.d. *Early Symptoms.* [Online]
Available at: http://www.alz.co.uk/info/early-symptoms
[Accessed 30 12 2015].

Alzheimers Organisation, 2015. *Dementia 2014 report statistics.* [Online]
Available at: https://www.alzheimers.org.uk/statistics
[Accessed 30 12 2015].

Amazon, n.d. *BIGTrack Trackball with Drag Technology.* [Online]
Available at: http://www.amazon.co.uk/BIGTrack-Trackball-Technology-Non-Switch-Adapted/dp/B004KS2E6G
[Accessed 14 02 2016].

Amazon, n.d. *Power Grip Utility Knife - Model 553279.* [Online]
Available at: http://www.amazon.com/Power-Grip-Utility-Knife-553279/dp/B002BUEB3Y
[Accessed 14 02 2016].

Anon., 2012. *Hexadecimal Conversion Table.* [Online]
Available at: http://freepages.genealogy.rootsweb.ancestry.com/~dav4is/Misc/images/X2D2X.gif
[Accessed 05 12 2015].

aPlaceForMom, 2015. *Top 7 Physical Alzheimer's Symptoms.* [Online]
Available at: http://www.aplaceformom.com/blog/alzheimers-physical-changes-7-9-13/
[Accessed 30 12 2015].

Apple, 2016. *LastPass – Free Password Manager & Secure Vault with Private Notes & Passcode Generator.* [Online]
Available at: https://itunes.apple.com/gb/app/lastpass-free-password-manager/id324613447?mt=8
[Accessed 19 03 2016].

Apple, n.d. *Use Touch ID on iPhone and iPad.* [Online]
Available at: https://support.apple.com/en-gb/HT201371
[Accessed 20 03 2016].

Arduino CC, n.d. *Intel Edison.* [Online]
Available at: https://www.arduino.cc/en/uploads/ArduinoCertified/Intel_Edison_Back_450px.jpg
[Accessed 06 03 2016].

BBC, 2015. *TalkTalk hack 'affected 157,000 customers'.* [Online]
Available at: http://www.bbc.co.uk/news/business-34743185
[Accessed 1 12 2015].

BBC, 2015. *The generation that tech forgot.* [Online]
Available at: http://www.bbc.co.uk/news/technology-32511489
[Accessed 10 12 2015].

Berent, A., n.d. *AES Simplified.* [Online]
Available at: https://www.ime.usp.br/~rt/cranalysis/AESSimplified.pdf
[Accessed 04 12 2015].

Berkley Fishing, n.d. *Knife & Scissor Sharpener.* [Online]
Available at: http://www.berkley-fishing.co.nz/product/knide-scissor-sharpener/
[Accessed 14 02 2016].

Brodsky, M. C., 2010. *Pediatric Neuro-Opthalmology.* Second Edition ed. Rochester: Springer.

Business Insider, 2014. *Nearly 7 Million Dropbox Passwords Have Been Hacked.* [Online]
Available at: http://www.businessinsider.com/dropbox-hacked-2014-10
[Accessed 6 12 2015].

CadSoft USA, n.d. *Eagle CAD.* [Online]
Available at: http://www.cadsoftusa.com/
[Accessed 20 03 2016].

Cambridge University Engineering Department, 2003. *Materials Data Book.* [Online]
Available at: http://www-mdp.eng.cam.ac.uk/web/library/enginfo/cueddatabooks/materials.pdf
[Accessed 10 06 2016].

CERN Computer Security, 2013. *Password Recommendations.* [Online]
Available at: https://security.web.cern.ch/security/recommendations/en/passwords.shtml
[Accessed 14 12 2015].

CircuitLab, n.d. *Circuit Simulation.* [Online]
Available at: https://www.circuitlab.com/
[Accessed 15 02 2016].

Clarke University, 2015. *Euclid's Elements.* [Online]
Available at: http://aleph0.clarku.edu/~djoyce/java/elements/toc.html
[Accessed 04 12 2015].

Duke Department of Neurology, n.d. *Movement Disorders Definitions.* [Online]
Available at: http://neurology.duke.edu/specialty-programs/movement-disorders/definitions
[Accessed 31 12 2015].

Eli Carmeli, H. P. a. R. C., 2013. The Aging Hand. *The Geronotological Society of America,* pp. 146-151.

Entrepreuneur, 2015. *Password Statistics: The Bad, the Worse and the Ugly (Infographic).* [Online]
Available at: http://www.entrepreneur.com/article/246902
[Accessed 10 12 2015].

Experian, n.d. *One in six adults has fallen victim to cyber-crime.* [Online]
Available at: http://www.experian.co.uk/blogs/latest-thinking/one-six-adults-fallen-victim-cyber-crime/
[Accessed 05 12 2015].

Financial Times, 2016. *Obama seeks $19bn to boost cyber defences.* [Online]
Available at: http://www.ft.com/cms/s/0/35249c8c-cf53-11e5-92a1-c5e23ef99c77.html
[Accessed 26 02 2016].

GanttProject, 2016. *Free project scheduling and management app for Windows, OSX and Linux..* [Online]
Available at: https://www.ganttproject.biz/
[Accessed 18 3 2016].

Google, 2015. *Stronger Security for your Google Account.* [Online]
Available at: https://www.google.com/landing/2step/
[Accessed 13 12 2015].

Google, n.d. *Google Drive.* [Online]
Available at: https://drive.google.com/drive/my-drive

Henry, A., 2013. *Five best food and nutrition tracking tools.* [Online]
Available at: http://lifehacker.com/five-best-food-and-nutrition-tracking-tools-1084103754
[Accessed 10 12 2015].

HyperPhysics, n.d. *Capacitors.* [Online]
Available at: http://hyperphysics.phy-astr.gsu.edu/hbase/electric/capchg.html
[Accessed 06 01 2016].

Images, W. C., n.d. *Letter Frequency.* [Online]
Available at:
https://upload.wikimedia.org/wikipedia/commons/thumb/d/d5/English_letter_frequency_(alphabetic).svg/2000px-English_letter_frequency_(alphabetic).svg.png
[Accessed 20 03 2016].

Indiegogo, 2016. *Indiegogo.* [Online]
Available at: https://www.indiegogo.com/#/picks_for_you
[Accessed 10 12 2015].

Information is Beautiful, 2016. *Buggest Data Breaches.* [Online]
Available at: http://www.informationisbeautiful.net/visualizations/worlds-biggest-data-breaches-hacks/
[Accessed 10 12 2015].

Internet Live Stats, 2016. *Internet Users.* [Online]
Available at: http://www.internetlivestats.com/internet-users/
[Accessed 10 12 2015].

Jenkins, B. B., 2015. *10 Life-Changing Reasons to Drink More Water.* [Online]
Available at: http://breakingmuscle.com/health-medicine/10-life-changing-reasons-to-drink-more-water
[Accessed 10 12 2015].

Khan Academy, 2014. *Journey into Cryptography.* [Online]
Available at: https://www.khanacademy.org/computing/computer-science/cryptography
[Accessed 04 12 2015].

Kickstarter, 2016. *Kickstarter.* [Online]
Available at: https://www.kickstarter.com/
[Accessed 10 12 2015].

LastPass, 2016. *LastPass, Simplify Your Life.* [Online]
Available at: https://lastpass.com/
[Accessed 06 12 2015].

LastPass, 2016. *LastPass, the features.* [Online]
[Accessed 05 12 2015].

LifeHacker, 2015. *LastPass Hacked, Change Your Master Password Now.* [Online]
Available at: http://lifehacker.com/lastpass-hacked-time-to-change-your-master-password-1711463571
[Accessed 05 12 2015].

Lifehackers, 2011. *The Only Secure Password is the One You Can't Remember.* [Online]
Available at: http://lifehacker.com/5785420/the-only-secure-password-is-the-one-you-cant-remember
[Accessed 12 12 2015].

Medical Daily, 2016. *75% of Americans May Suffer From Chronic Dehydration, According to Doctors.* [Online]
Available at: http://www.medicaldaily.com/75-americans-may-suffer-chronic-dehydration-according-doctors-247393
[Accessed 10 12 2015].

MedlinePlus, n.d. *Aging changes in the senses.* [Online]
Available at: https://www.nlm.nih.gov/medlineplus/spanish/ency/article/004013.htm
[Accessed 30 12 2015].

Microsoft, n.d. [Online]
Available at: https://onedrive.live.com/

Miller, J., n.d. *Elasticity. Hook's Law. Tensile, compressive and shear stresses. Strain. Elastic, shear and bulk modulus..* [Online]
Available at: http://www.solitaryroad.com/c1020.html
[Accessed 10 12 2015].

MyFitnessPal, 2015. *How do I use the barcode scanner to log foods?.* [Online]
Available at: https://myfitnesspal.desk.com/customer/portal/articles/13640-how-do-i-use-the-barcode-scanner-to-log-foods-
[Accessed 10 12 2016].

National Eye Institute, 2011. *Cataracts.* [Online]
Available at: https://nei.nih.gov/eyedata/cataract
[Accessed 30 12 2015].

NDT Resource Center, n.d. *Fracture Toughness.* [Online]
Available at: https://www.nde-ed.org/EducationResources/CommunityCollege/Materials/Mechanical/FractureToughness.htm
[Accessed 15 03 2016].

O'Brien, J. T., 1999. *Age-associated memory impairment and related disorders.* [Online]
Available at: http://apt.rcpsych.org/content/aptrcpsych/5/4/279.full.pdf
[Accessed 15 02 2016].

PC World, 2008. *The 7 Worst Tech Predictions of All Time.* [Online]
Available at: http://www.pcworld.com/article/155984/worst_tech_predictions.html
[Accessed 10 12 2015].

Poh Kiat Ng, A. S., 2014. Hand Anthropometry: A Descriptive Analysis on Elderly Malaysians. In: *Anthropometric Research in Malaysia.* s.l.:s.n.

Posturite, n.d. *Soft Pencil Grips Pack of 3.* [Online]
Available at: http://www.posturite.co.uk/soft-pencil-grips.html
[Accessed 14 01 2016].

QRStuff, n.d. *Sample QR Code.* [Online]
Available at: http://www.qrstuff.com/images/sample.png
[Accessed 06 03 2016].

Quentons, n.d. *8" Ceramic Chef Knife (Titanium Coated).* [Online]
Available at: http://quentons.com/products/SKU1234-8''-Ceramic-Chef-Knife-Titanium-Coated
[Accessed 14 02 2016].

Random-ize, n.d. *How Long to Hack My Password.* [Online]
Available at: http://random-ize.com/how-long-to-hack-pass/
[Accessed 20 03 2016].

Rapid Electronics, n.d. *Winstar WH1602B3-SLL-JWV 16x2 LCD VATN White on Black I2C Interface Report an error.* [Online]
Available at: http://www.rapidonline.com/electronic-components/winstar-wh1602b3-sll-jwv-16x2-lcd-vatn-white-

on-black-i2c-interface-55-3098
[Accessed 06 03 2016].

Reuters, 2015. *5.6 million fingerprints stolen in U.S. personnel data hack: government.* [Online]
Available at: http://www.reuters.com/article/us-usa-cybersecurity-fingerprints-idUSKCN0RN1V820150923
[Accessed 06 12 2015].

Roboform, 2015. *Password Security Survey Results- Part 1.* [Online]
Available at: http://www.roboform.com/blog/password-security-survey-results
[Accessed 05 12 2015].

Rush University Medical Center, n.d. *5 Facts About Parkinson's Disease.* [Online]
Available at: https://www.rush.edu/health-wellness/discover-health/5-parkinsons-disease-facts
[Accessed 30 12 2015].

Schneier, B., 2009. *Hacking Two-Factor Authentication.* [Online]
Available at: https://www.schneier.com/blog/archives/2009/09/hacking_two-fac.html
[Accessed 13 12 2015].

Scientific American, 2015. *Record 232-digit number from Cryptography Challenge factored.* [Online]
Available at: http://blogs.scientificamerican.com/observations/record-232-digit-number-from-cryptography-challenge-factored/
[Accessed 04 12 2015].

Seeed Studio, n.d. *Grove Fingerprint Sensor.* [Online]
Available at: http://www.seeedstudio.com/wiki/Grove_-_Fingerprint_Sensor
[Accessed 06 03 2016].

Singh, S., n.d. *Arab Code Breakers.* [Online]
Available at: http://simonsingh.net/media/articles/maths-and-science/arab-code-breakers/
[Accessed 05 12 2015].

SolidWorks, n.d. *3D CAD Design.* [Online]
Available at: http://www.solidworks.co.uk/
[Accessed 20 03 2016].

Sophos, 2014. *Average person has 19 passwords – but 1 in 3 don't make them strong enough.* [Online]
Available at: https://nakedsecurity.sophos.com/2014/10/17/average-person-has-19-passwords-but-1-in-3-dont-make-them-strong-enough/
[Accessed 13 12 2015].

Sparkfun Electronics, n.d. *Fingerprint Scanner - TTL (GT511C3).* [Online]
Available at: https://www.sparkfun.com/products/11792
[Accessed 06 03 2016].

Sparkfun Electronics, n.d. *Rocker Switch - SPST (Round).* [Online]
Available at: https://www.sparkfun.com/products/11138
[Accessed 06 03 2016].

Sparkfun Electronics, n.d. *Rotary Encoder.* [Online]
Available at: https://www.sparkfun.com/products/10982
[Accessed 06 03 2016].

Sparkfun Electronics, n.d. *Rotary Potentiometer.* [Online]
Available at: https://www.sparkfun.com/products/9939
[Accessed 6 03 2016].

Sparkfun Electronics, n.d. *Sliding Potentiometer.* [Online]
Available at: https://www.sparkfun.com/products/9119
[Accessed 06 03 2016].

Sparkfun Electronics, n.d. *SPDT Mini Power Switch.* [Online]
Available at: https://www.sparkfun.com/products/102
[Accessed 06 03 2016].

Sparkfun Electronics, n.d. *Toggle Switch.* [Online]
Available at: https://www.sparkfun.com/products/9276
[Accessed 06 03 2016].

Sparkfun, n.d. *Thumb Joystick.* [Online]
Available at: https://www.sparkfun.com/products/9032
[Accessed 06 03 2016].

Statista, 2015. *Global PC shipments from 1st quarter 2009 to 4th quarter 2015, by vendor (in million units)*.* [Online]
Available at: http://www.statista.com/statistics/263393/global-pc-shipments-since-1st-quarter-2009-by-vendor/
[Accessed 10 12 2015].

SurveyMonkey, 2016. *Make Better Decisions with the UK's Leading Survey Platform.* [Online]
Available at: https://www.surveymonkey.co.uk/
[Accessed 20 03 2016].

TechTarget, n.d. *Anna Kournikova Virus.* [Online]
Available at: http://searchenterprisedesktop.techtarget.com/definition/Anna-Kournikova-virus
[Accessed 20 03 2016].

The Guardian, 2015. *Security v usability: cracking the workplace password problem.* [Online]
Available at: http://www.theguardian.com/media-network/2015/oct/27/password-security-usability-workplace-problem
[Accessed 10 12 2015].

TheVerge, 2015. *Bluetooth Hacked.* [Online]
Available at: http://www.theverge.com/2013/11/14/5103820/bluetooth-hacked
[Accessed 5 11 2015].

Thorzt, 2015. *8 Things Every Employer Should Know About Worker Dehydration.* [Online]
Available at: http://thorzt.com/8-things-every-employer-should-know-about-worker-dehydration/
[Accessed 10 12 2015].

TIME, 2016. *Netflix Accounts Hacked.* [Online]
Available at: http://time.com/4230367/netflix-account-hacked/
[Accessed 26 02 2016].

United States Department of Justics, 2015. *Online Identify Theft.* [Online]
Available at: http://www.justice.gov/criminal-fraud/identity-theft/identity-theft-and-identity-fraud
[Accessed 04 12 2015].

University of California San Diego, 2008. *Lecture Notes on Cryptography.* [Online]
Available at: https://cseweb.ucsd.edu/~mihir/papers/gb.pdf
[Accessed 05 12 2015].

University of Cambridge, 2009. *Material and Process Selection Charts.* [Online]
Available at: http://www.mie.uth.gr/ekp_yliko/2_materials-charts-2009.pdf
[Accessed 10 12 2015].

VeloSpace Forums, n.d. *Custom Framebuilders are they worth it.* [Online]
Available at: http://velospace.org/forums/discussion/5522/custom-framebuilders-are-they-worth-it/
[Accessed 10 12 2015].

WebMD, n.d. *Healthy Aging - Normal Aging.* [Online]
Available at: http://www.webmd.com/healthy-aging/tc/healthy-aging-normal-aging
[Accessed 30 12 2015].

Wikimedia Common Images, n.d. *Caesar Cipher.* [Online]
Available at: https://upload.wikimedia.org/wikipedia/commons/thumb/2/2b/Caesar3.svg/2000px-Caesar3.svg.png
[Accessed 05 12 2015].

WIRED, 2013. *How Apple and Amazon Security Flaws Led to My Epic Hacking.* [Online]
Available at: http://www.wired.com/2012/08/apple-amazon-mat-honan-hacking/
[Accessed 04 12 2015].

Xamarin, n.d. *Xamarin.Forms.* [Online]
Available at: https://www.xamarin.com/forms
[Accessed 22 03 2016].

YouGov, 2014. *Elderly and Technology.* [Online]
Available at: https://yougov.co.uk/news/2015/12/14/prejudice-data/
[Accessed 20 03 2016].

10.0 Appendices

A Tasks List

Preparation:

- Install SolidWorks on computer
- Create comprehensive list of tasks
- Create Gantt and PERT Chart
- Install other required software such as Xamarin Studio and Eagle CAD

Background Research

- Identify the problem that is to be solved – complete general research on passwords and password management
- Identify current solutions and the successes and flaws of them
- Identify the optimal target market – and complete basic research on them and their needs.
- Produce initial specification for the product based on the identified needs of the target market.

Further Research

- Security Research
  - Complete research on current hacking techniques and therefore how the product can be designed to ensure that it is not susceptible to these hacking techniques.
  - Complete research on encryption as a major method for preventing the sensitive information reaching the hands of the hackers. In this look at the history of encryption as well as modern encryption, evaluating the benefits and downsides of each of the encryption methods
  - Complete biometrics research, looking into the various forms of biometrics and their advantages and disadvantages.
- Control System Research
  - Complete research on how the user will enter passwords into the product.
  - Conduct research on the general user interface and how the product will be used.
  - Conduct research to choose all the required components presenting a number of options in each case, and therefore making a justified decision.
- Mechanical Research
  - Conduct research on the stresses that the product will incur and therefore research into the most effective materials that could be used.
  - Complete research into possible geometries of the product – looking particularly at ergonomics of handheld products.
- Software Design Research
  - Conduct initial research on how the various programming elements could be completed for example looking at the possible programming languages that could be used.

Development

- Control System Testing and Development
  - Conduct research on the components chosen – testing them if possible, to ensure that their usage would be possible.

- Correct any problems found as well as making improvements if possible, especially to further enhance the security of the product.

- Mechanical Ideas and Development
  - Produce a number of possible ideas for the geometry of the product.
  - Evaluate each of the ideas, looking at the various pros and cons of them.
  - Produce a final design for the structure and geometry of the product.
- Software Design Development
  - Produce the code required for all the elements of the project.

Manufacturing

- PCB Production
  - Conduct research into the necessities of a PCB for the electronics as well as how the PCB would be manufactured if required.
  - Design a PCB, which would be able to contain all the required electronics.
- Casing Manufacture
  - Research and evaluate multiple methods for producing the casing of the product, concluding what manufacturing technique would be the best.
  - After this has been decided, return to materials research and ensuring that it could be manufactured in real life, define the material(s) used for the casing.

Evaluation

- Evaluate the artefact against the original brief and specification designed.
- Evaluate the process of the Extended Project, looking at how particular sections of work went – whether they were successful or more painful than originally anticipated.
- Produce a number of changes, which would be useful to make the product more successful.

Write-Up

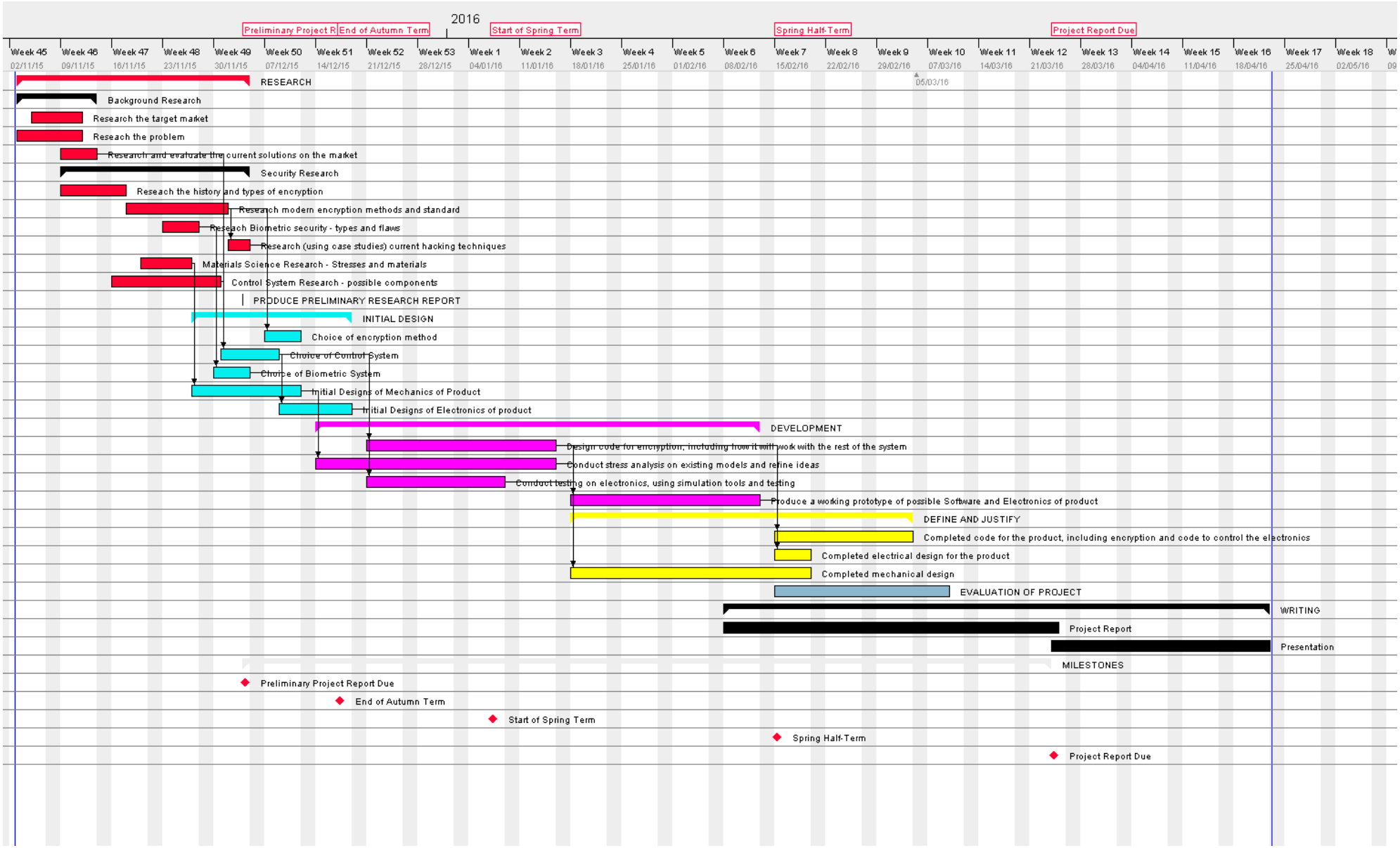
- Consolidate all previous writing into a complete report, filling in any gaps, which are found.
  Read report, ensuring that the report makes sense as well as fulfils the mark scheme.

B Gantt Charts

11/11/2015

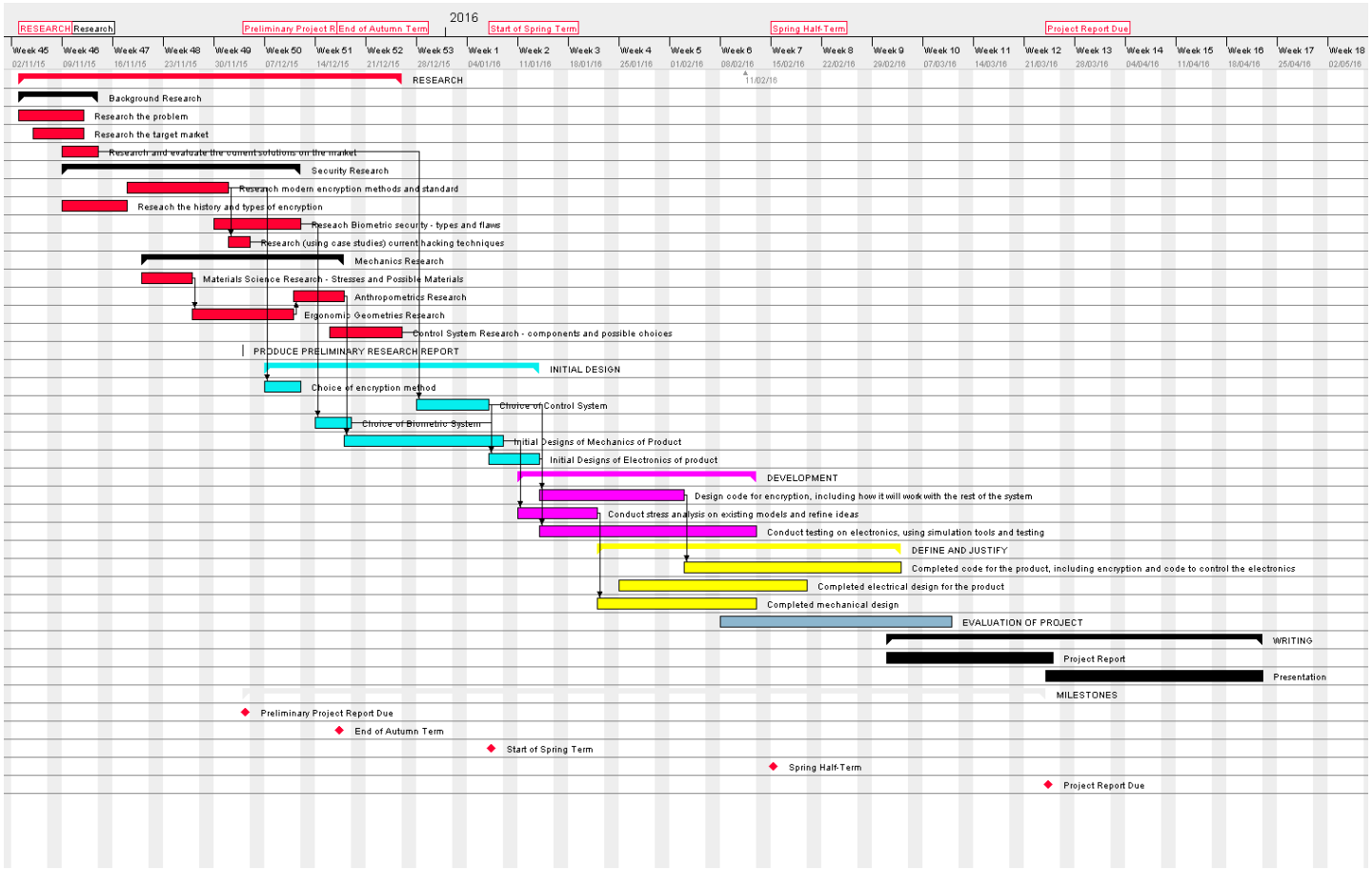
In the first plan, I took the task list from my Proposal Form into a new more comprehensive list, adding tasks such as breaking down the Security Research further into investigating the various options for encryption as well as for biometrics. In addition, for a number of sections where the proposal form contained very broad estimations of time required, I did a little bit of research to ensure that the time required would be largely accurate. Additionally, I made a list of the dependencies (on a piece of paper) to ensure that I understood the flow of the project as well as the list of resources which would be required to complete the task, ensuring that if I required anything I would ensure that this arrived.

Meanwhile, I attempted to find the best way of representing this plan, looking at mechanisms such as online calendars as well as Gantt Charts. In the end, the Gantt Chart proved the most effective, allowing a complete overview into the project, as well as an in depth view if necessary, with a plan for every single week. However, I also continue to maintain a comprehensive list of tasks, which proved a checklist. This tasklist in its final form is available in Appendix A.


2016
Preliminary Project R
End of Autumn Term
Start of Spring Term
Spring Half-Term
Project Report Due
Week 45 02/11/15
Week 46 09/11/15
Week 47 16/11/15
Week 48 23/11/15
Week 49 30/11/15
Week 50 07/12/15
Week 51 14/12/15
Week 52 21/12/15
Week 53 28/12/15
Week 1 04/01/16
Week 2 11/01/16
Week 3 18/01/16
Week 4 25/01/16
Week 5 01/02/16
Week 6 08/02/16
Week 7 15/02/16
Week 8 22/02/16
Week 9 29/02/16
Week 10 07/03/16
05/03/16
Week 11 14/03/16
Week 12 21/03/16
Week 13 28/03/16
Week 14 04/04/16
Week 15 11/04/16
Week 16 18/04/16
Week 17 25/04/16
Week 18 02/05/16
RESEARCH
Background Research
Research the target market
Reseach the problem
Research and evaluate the current solutions on the market
Security Research
Reseach the history and types of encryption
Research modern encryption methods and standard
Reseach Biometric security - types and flaws
Research (using case studies) current hacking techniques
Materials Science Research - Stresses and materials
Control System Research - possible components
PRODUCE PRELIMINARY RESEARCH REPORT
INITIAL DESIGN
Choice of encryption method
Choice of Control System
Choice of Biometric System
Initial Designs of Mechanics of Product
Initial Designs of Electronics of product
DEVELOPMENT
Design code for encryption, including how it will work with the rest of the system
Conduct stress analysis on existing models and refine ideas
Conduct testing on electronics, using simulation tools and testing
Produce a working prototype of possible Software and Electronics of product
DEFINE AND JUSTIFY
Completed code for the product, including encryption and code to control the electronics
Completed electrical design for the product
Completed mechanical design
EVALUATION OF PROJECT
WRITING
Project Report
Presentation
MILESTONES
Preliminary Project Report Due
End of Autumn Term
Start of Spring Term
Spring Half-Term
Project Report Due

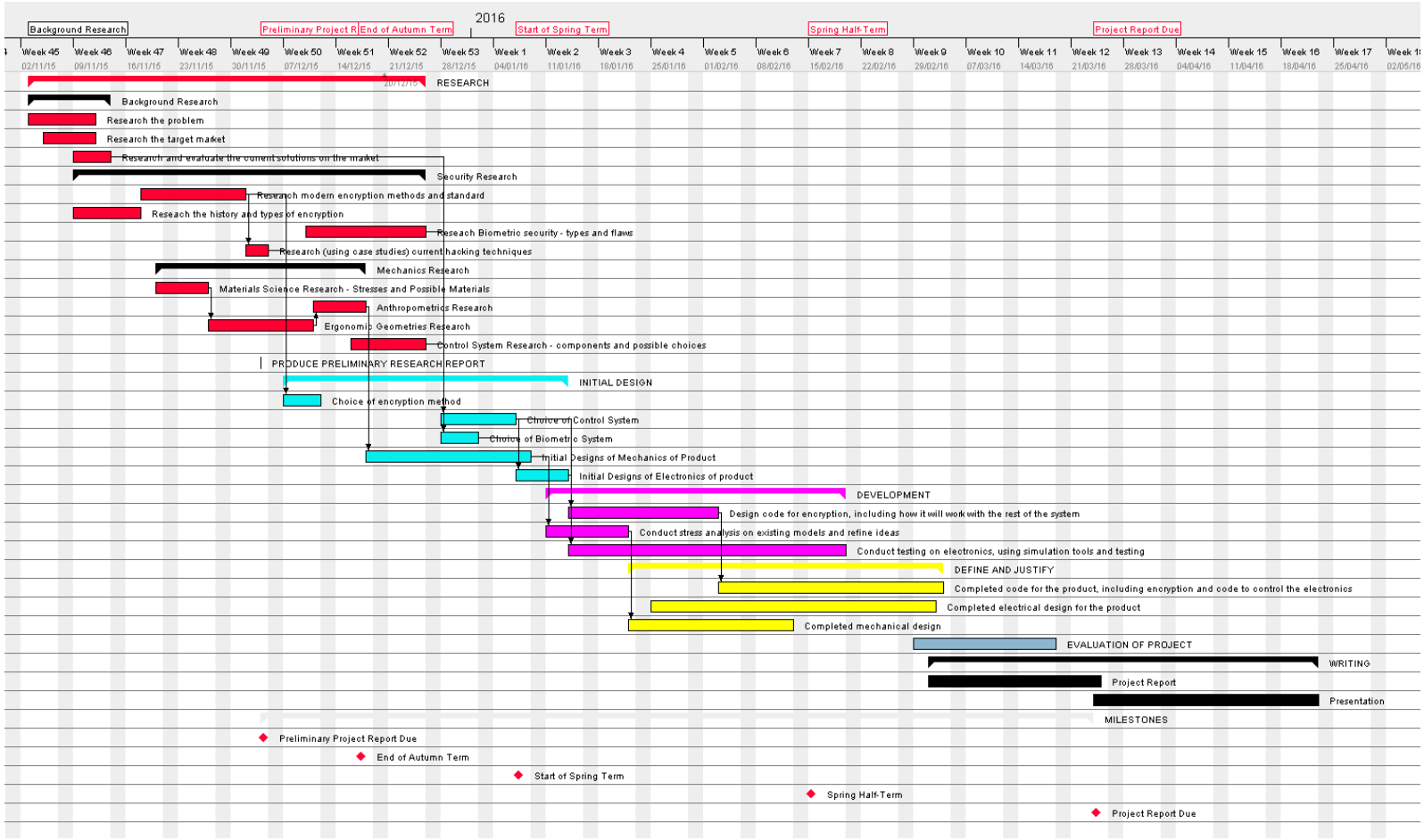
21/12/2015

By the second iteration of the plan, I had found a number of the complexities of the project, and the Gantt Chart included an increased focus on the parts which required focus, such as the realisation that more time would be required for the various elements of the Electronics and Software, given its relative complexity. Additionally, there was the delay of a few research task, including the control system, as I had to forgo these parts to produce the Preliminary Research Report before the deadline for this. Finally, when completing the beginning to consider the mechanical design of the product, I realised that I had forgotten an important section of the research, including research into the ergonomics of the design and indeed into anthropometric data.



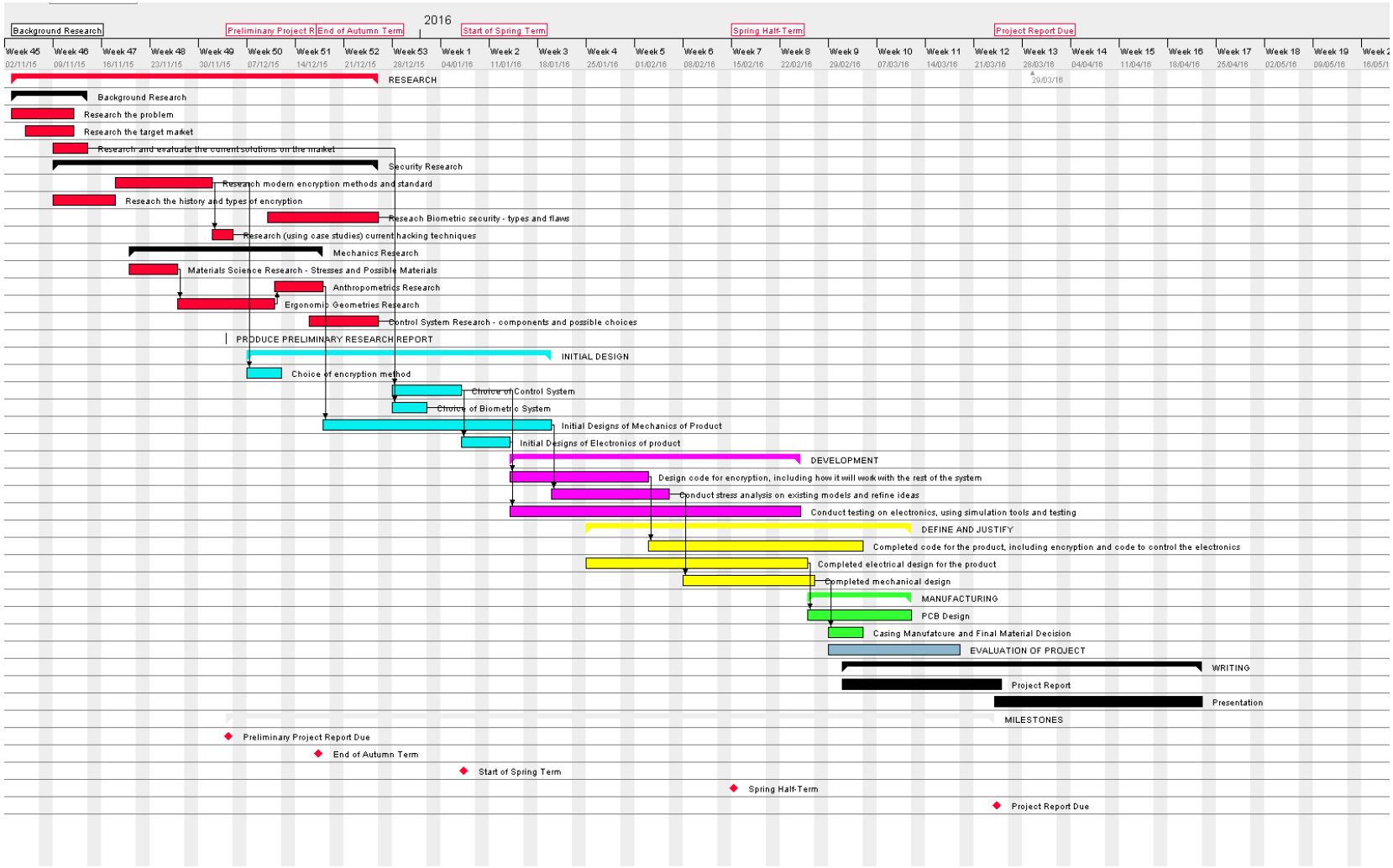
10/01/2016

Over Christmas, the initial designs had been largely completed and so there was much optimism that the mechanical design could be completed ahead of schedule, ensuring that I would meet the deadline at the very least. However, in many ways, this was at the cost of the Control System development, which had suffered, leaving plenty of work to do. This meant that more time was dedicated to various elements of this, also ensuring that at the time when there were multiple tasks to be completed at the same time, the electronics and software would take priority. It also planned to ensure that my new milestone of completing the definition by the beginning of the Spring Half-Term would be achieved.

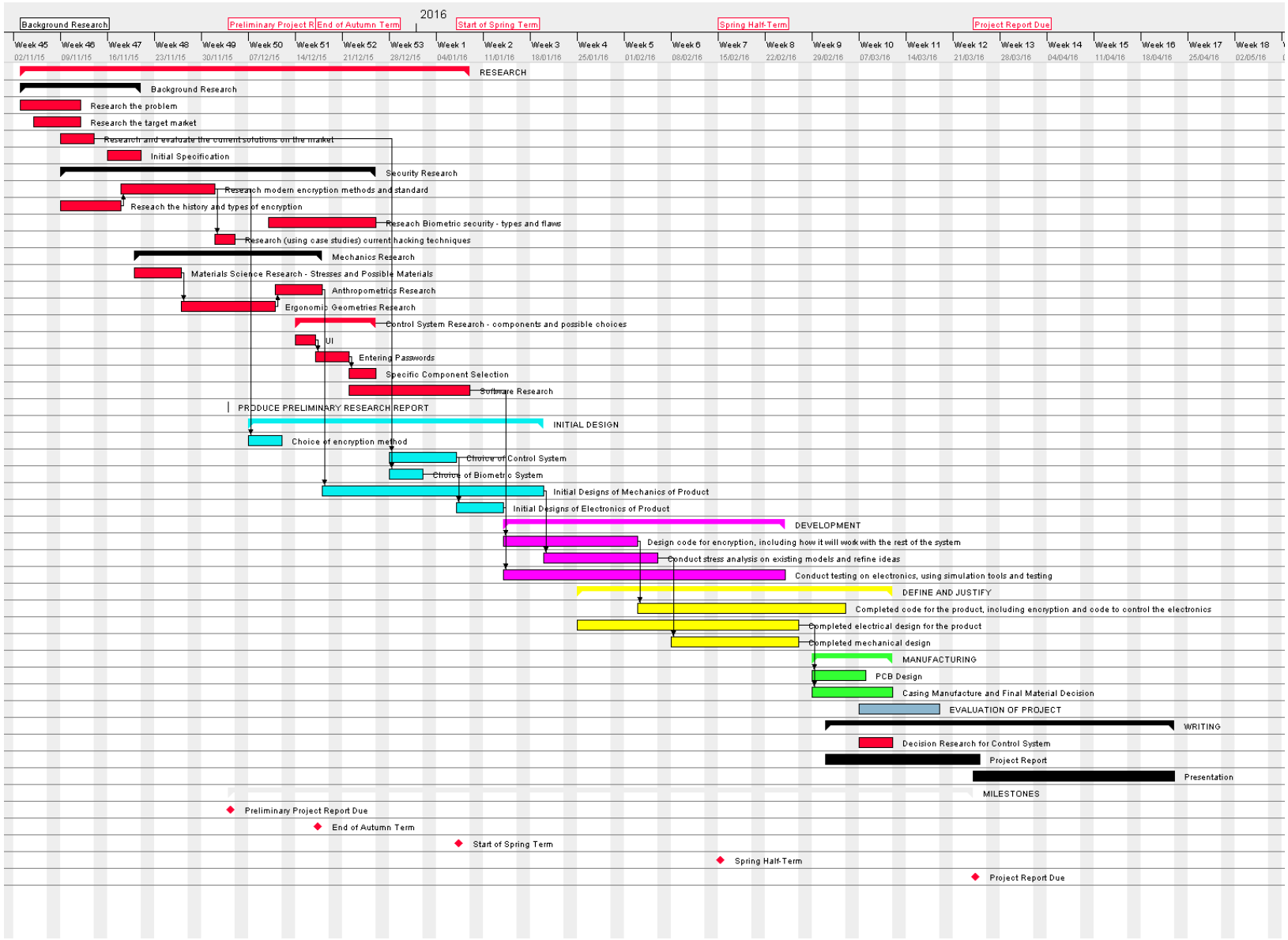


21/02/2016

I had at this point in time just about met my milestone of defining all the various elements, though this had come at the cost of working on coding the product (which meant that time had to be allocated for this in the future). I also found that I had failed to include a few mechanical parts in my plan, including the final decision on materials (I had delayed this decision after the stresses research as this yielded the idea that multiple materials would work) and a discussion of the viability of mass-manufacturing, including the possible design of a PCB from the electronics side and the discussion of manufacturing methods of the casing. Hence, time for this was added to the plan.



20/03/2016 – Final Gantt Chart



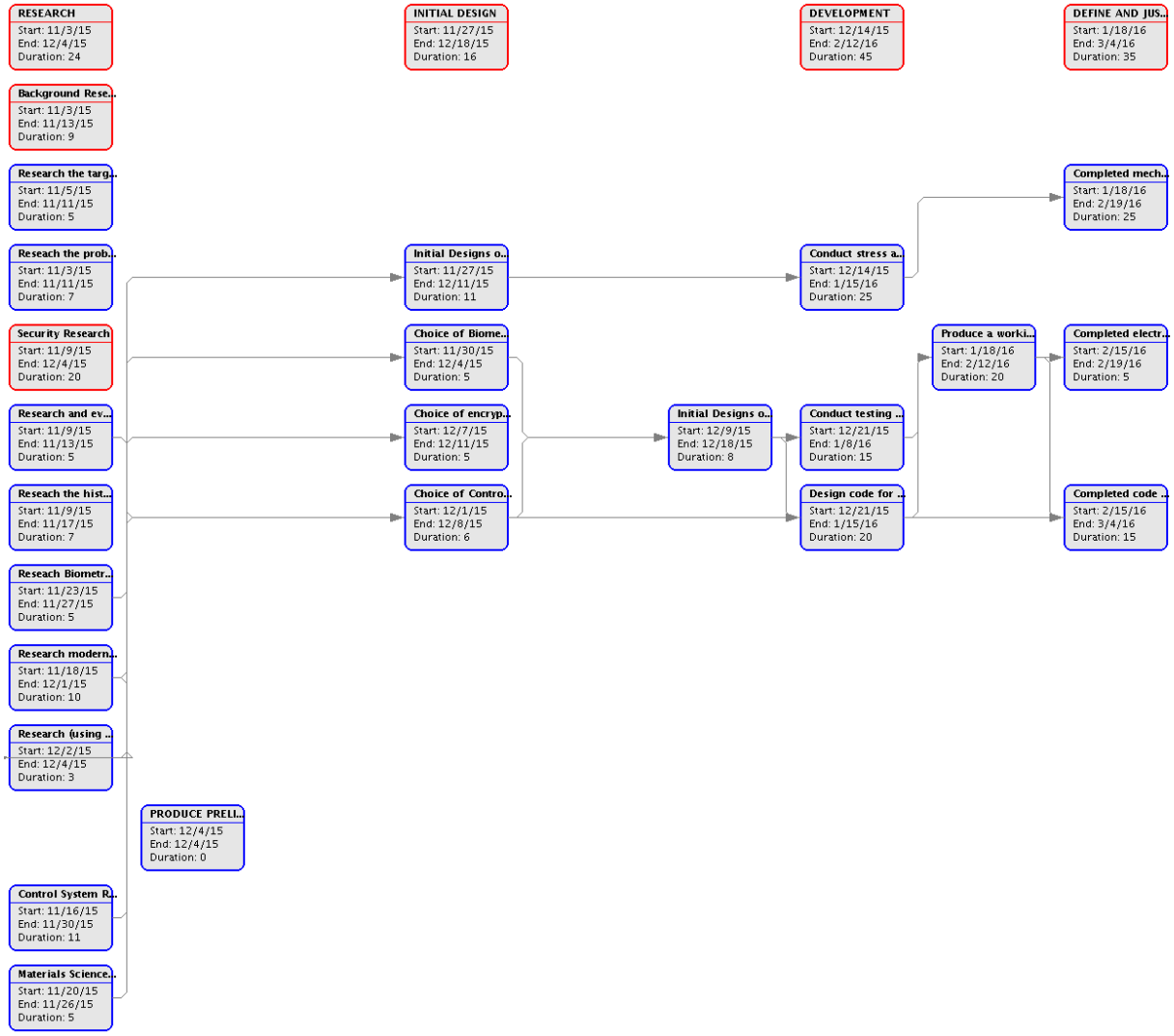
During this period of time, the production of the code passed quicker than expected, therefore helping to satisfy the extra requirements in completing additional Control System to help clarify the decisions that had been made. Additionally, PCB design proved easier, being able to be completed in a week, but Manufacturing of the product and the final material took longer, requiring additional CAD work to show how the design would be adapted for Injection Moulding. In the report writing stage, I also went through the Gantt Chart and made sure that all tasks, however short were covered, using the Activity Log to show where they had been completed.

The final task list from the Gantt Chart was hence:

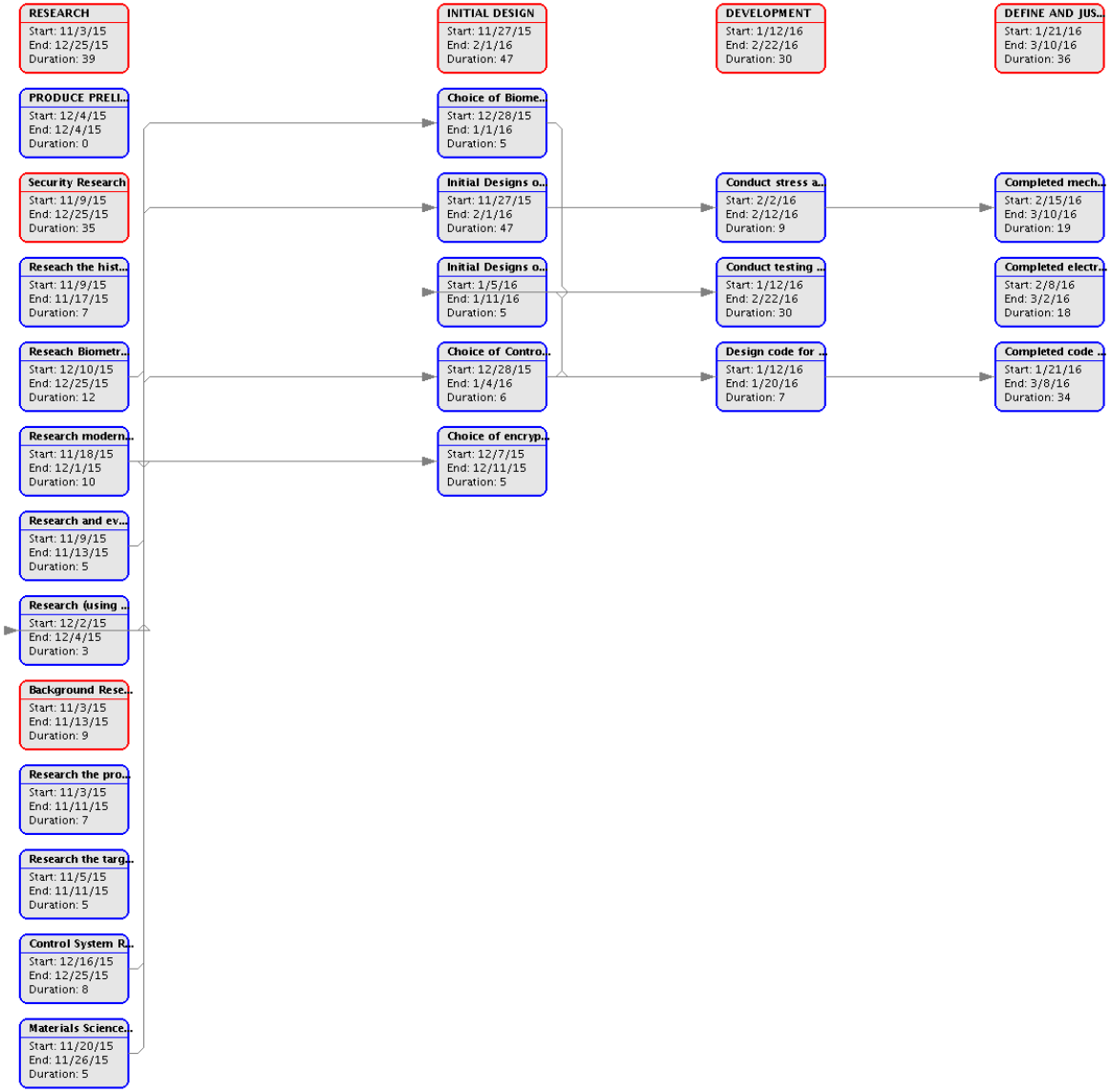
Name	Begin date	End date
RESEARCH	03/11/15	08/01/16
Background Research	03/11/15	20/11/15
Research the problem	03/11/15	11/11/15
Research the target market	05/11/15	11/11/15
Research and evaluate the current solutions on the market	09/11/15	13/11/15
Initial Specification	16/11/15	20/11/15
Security Research	09/11/15	25/12/15
Research modern encryption methods and standard	18/11/15	01/12/15
Reseach the history and types of encryption	09/11/15	17/11/15
Reseach Biometric security - types and flaws	10/12/15	25/12/15
Research (using case studies) current hacking techniques	02/12/15	04/12/15
Mechanics Research	20/11/15	17/12/15
Materials Science Research - Stresses and Possible Materials	20/11/15	26/11/15
Anthropometrics Research	11/12/15	17/12/15
Ergonomic Geometries Research	27/11/15	10/12/15
Control System Research - components and possible choices	14/12/15	25/12/15
UI	14/12/15	16/12/15
Entering Passwords	17/12/15	21/12/15
Specific Component Selection	22/12/15	25/12/15
Software Research	22/12/15	08/01/16
PRODUCE PRELIMINARY RESEARCH REPORT	04/12/15	04/12/15
INITIAL DESIGN	07/12/15	19/01/16
Choice of encryption method	07/12/15	11/12/15
Choice of Control System	28/12/15	06/01/16
Choice of Biometric System	28/12/15	01/01/16
Initial Designs of Mechanics of Product	18/12/15	19/01/16
Initial Designs of Electronics of Product	07/01/16	13/01/16
DEVELOPMENT	14/01/16	24/02/16
Design code for encryption, including how it will work with the rest of the system	14/01/16	02/02/16
Conduct stress analysis on existing models and refine ideas	20/01/16	05/02/16
Conduct testing on electronics, using simulation tools and testing	14/01/16	24/02/16
DEFINE AND JUSTIFY	25/01/16	11/03/16
Completed code for the product, including encryption and code to control the electronics	03/02/16	04/03/16
Completed electrical design for the product	25/01/16	26/02/16
Completed mechanical design	08/02/16	26/02/16
MANUFACTURING	29/02/16	11/03/16
PCB Design	29/02/16	07/03/16
Casing Manufacture and Final Material Decision	29/02/16	11/03/16
EVALUATION OF PROJECT	07/03/16	18/03/16
WRITING	02/03/16	22/04/16
Decision Research for Control System	07/03/16	11/03/16
Project Report	02/03/16	24/03/16
Presentation	24/03/16	22/04/16
MILESTONES	04/12/15	23/03/16
Preliminary Project Report Due	04/12/15	04/12/15
End of Autumn Term	17/12/15	17/12/15
Start of Spring Term	07/01/16	07/01/16
Spring Half-Term	15/02/16	15/02/16
Project Report Due	24/03/16	24/03/16

C PERT Charts

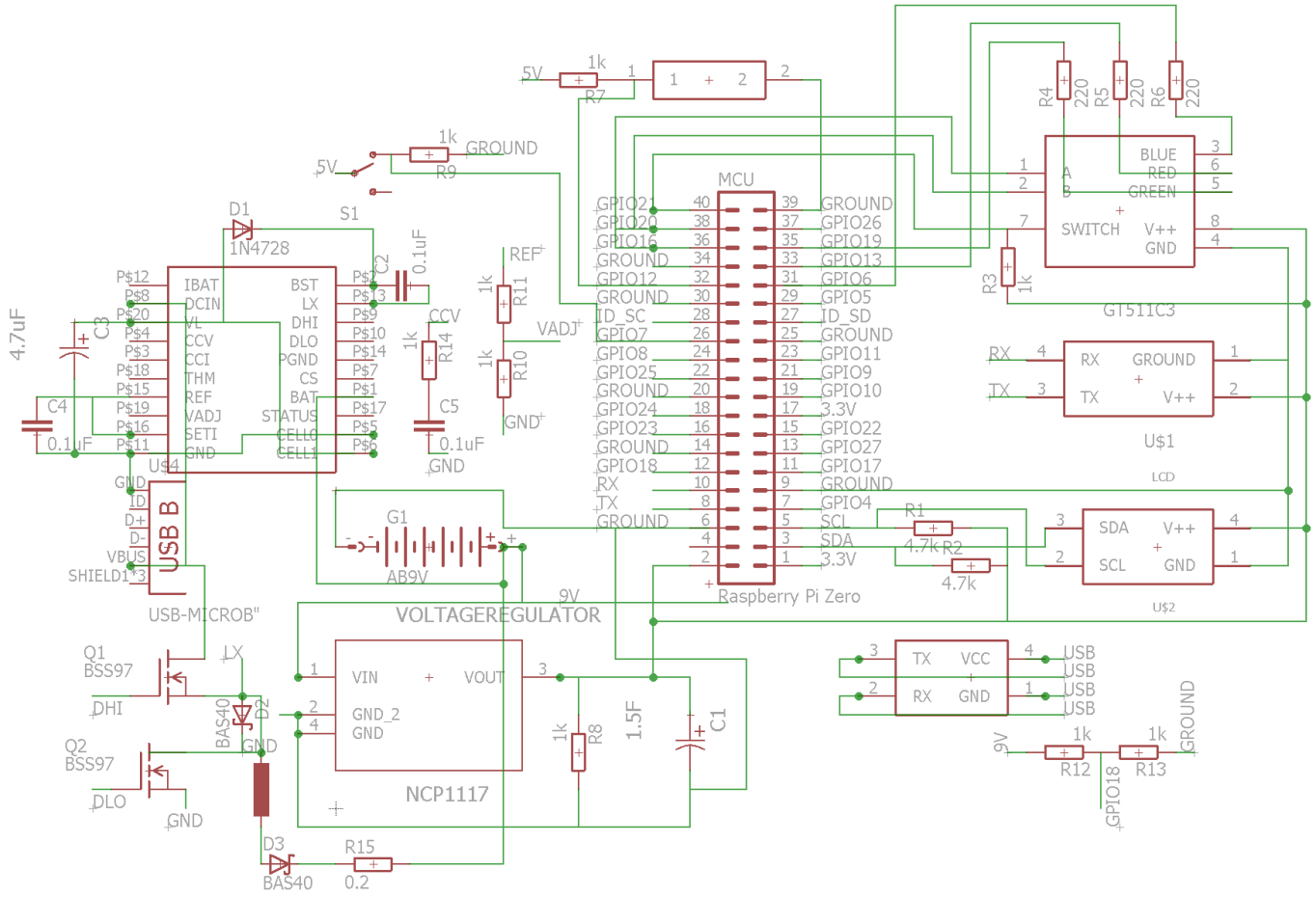
Initial PERT Chart

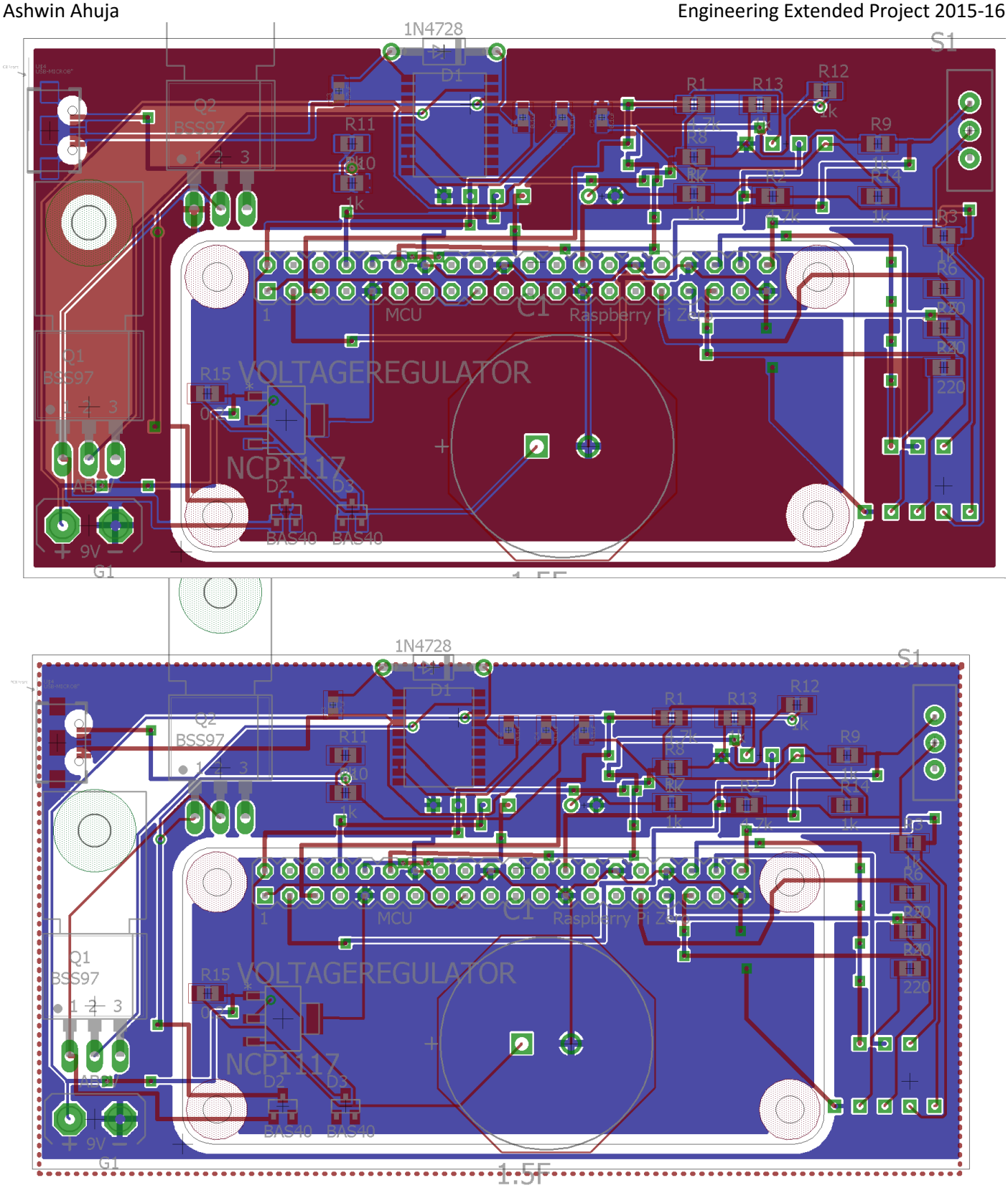


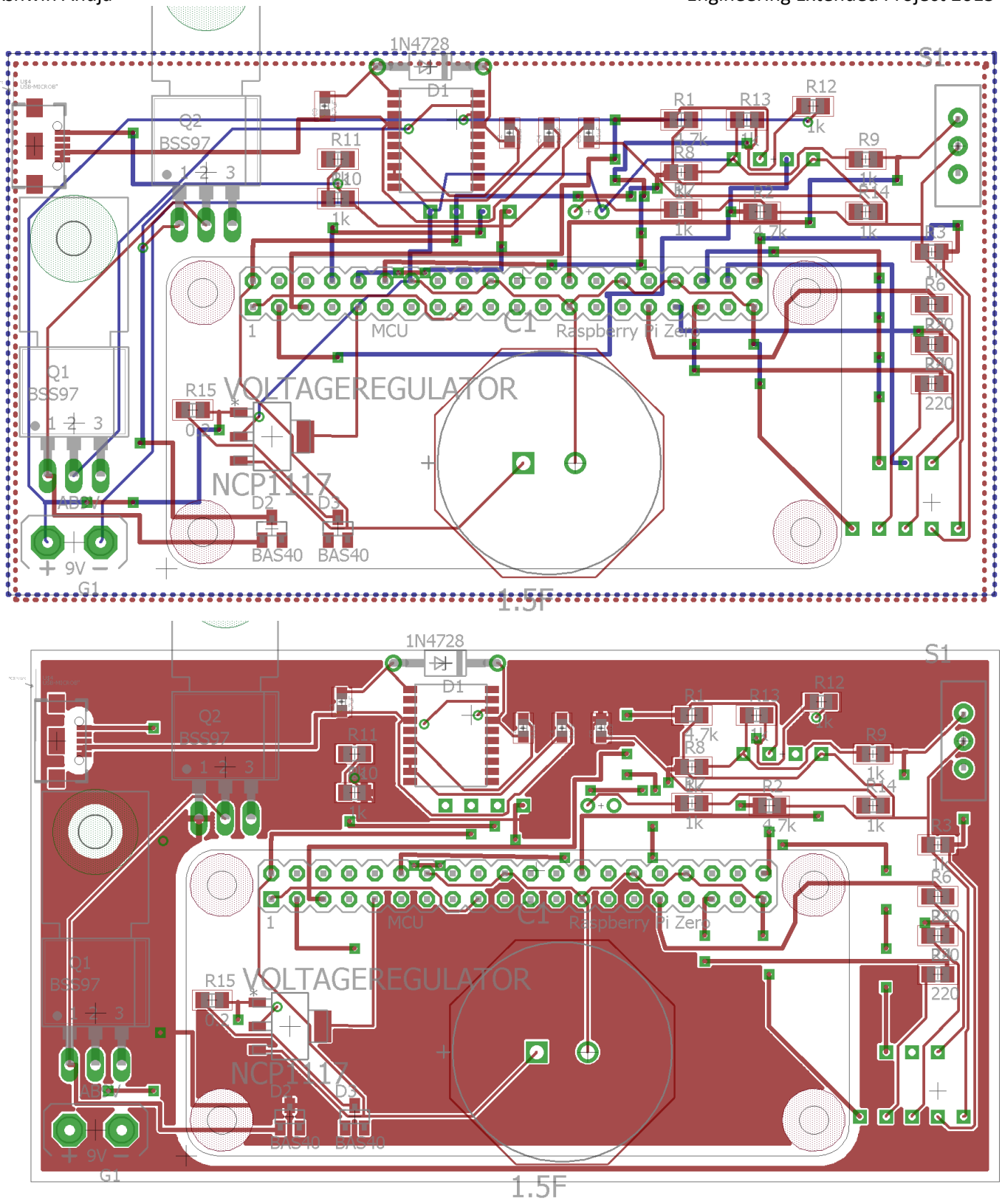
Final PERT Chart

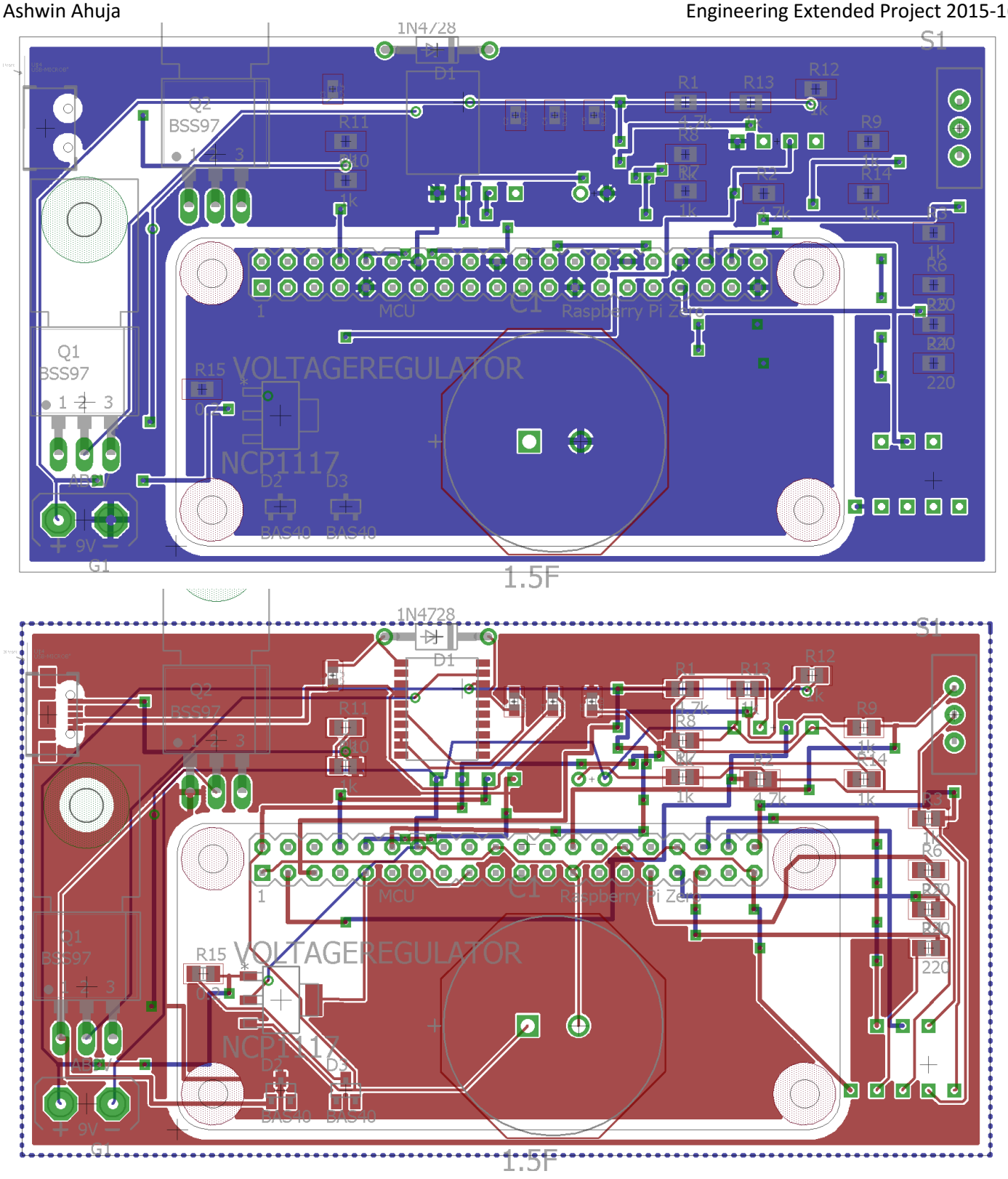


D Electronics Designs of Artefact




1N4728
D1
S1
Q2
BSS97
R11
R1
R13
R12
R9
R14
R3
R6
MCU
C1
Raspberry Pi Zero
Q1
BSS97
R15
VOLTAGEREGULATOR
NCP1117
D2
D3
BAS40
BAS40
9V
G1
220
1.5F


1N4728
D1
S1
Q2
BSS97
R11
1k
R1
R13
R12
4.7k
1k
R8
R9
1k
R14
1k
R2
4.7k
R3
R6
220
MCU
C1
Raspberry Pi Zero
Q1
BSS97
R15
VOLTAGEREGULATOR
NCP1117
D2
D3
BAS40
BAS40
9V
G1
1.5F



E Software Designs of Artefact

Since there are a number of small sections of code in the project, in every part, these are not included in the report, but are included in the Software Folder with the rest of the artefact. In fact, the Mobile Apps are not at all included in this report, due to the code being fragmented into so many different files, each with some code which comes pre-included and hence is not interesting as well as the code for the app itself.

```
<html>
   <style>
      .form {
   margin-left:auto;
   margin-right:auto;
   max-width: 500px;
   background: #F7F7F7;
   padding: 25px 15px 25px 10px;
   font: 12px Georgia, "Times New Roman", Times, serif;
   color: #888;
   text-shadow: 1px 1px 1px #FFF;
   border:1px solid #E4E4E4;
}
.form h1 {
   font-size: 25px;
   padding: 0px 0px 10px 40px;
   display: block;
   border-bottom:1px solid #E4E4E4;
   margin: -10px -15px 30px -10px;;
   color: #888;
}
.form h1>span {
   display: block;
   font-size: 11px;
}
.form label {
   display: block;
```

```
    margin: 0px;
}
.form label>span {
    float: left;
    width: 20%;
    text-align: right;
    padding-right: 10px;
    margin-top: 10px;
    color: #888;
}
.form input, .basic-grey input[type="email"], .basic-grey textarea, .basic-grey select {
    border: 1px solid #DADADA;
    color: #888;
    height: 30px;
    margin-bottom: 16px;
    margin-right: 6px;
    margin-top: 2px;
    outline: 0 none;
    padding: 3px 3px 3px 5px;
    width: 70%;
    font-size: 12px;
    line-height:15px;
    box-shadow: inset 0px 1px 4px #ECECEC;
    -moz-box-shadow: inset 0px 1px 4px #ECECEC;
    -webkit-box-shadow: inset 0px 1px 4px #ECECEC;
}
.form textarea{
    padding: 5px 3px 3px 5px;
```

```
}
.form select {
    background: #FFF url('down-arrow.png') no-repeat right;
    background: #FFF url('down-arrow.png') no-repeat right);
    appearance:none;
    -webkit-appearance:none;
    -moz-appearance: none;
    text-indent: 0.01px;
    text-overflow: '';
    width: 70%;
    height: 35px;
    line-height: 25px;
}
.form textarea{
    height:100px;
}
.form .button {
    background: #E27575;
    border: none;
    padding: 10px 25px 10px 25px;
    color: #FFF;
    box-shadow: 1px 1px 5px #B6B6B6;
    border-radius: 3px;
    text-shadow: 1px 1px 1px #9E3F3F;
    cursor: pointer;
}
.form .button:hover {
    background: #CF7A7A
```

```
}

</style>


<script>

function generateQRCode(){

var serialNumber = sn.value.trim();

var accountName = an.value.trim();

var username = un.value.trim();

var password = pword.value.trim();

var valid = true;


if (valid == true)

{

   var key = serialNumber;

   var totalToBeEncrypted = accountName + "|"+ username + "|" + password;

   var encrypted = "";

   var keyCounter = 0;

   var endKey = key.length;

   for (i = 0; i < totalToBeEncrypted.length; i++)

   {

      if (totalToBeEncrypted[i] == '|')

      {

         encrypted += "|";

      }

      else

      {

     var characterToDealWith = totalToBeEncrypted[i];

     var intOfCharacter = characterToDealWith.charCodeAt(0);
```

```
    if(i>0)
    {
    var intOfLastCharacter = totalToBeEncrypted[i-1].charCodeAt(0);


    }
    else
    {
       var intOfLastCharacter = 0;
    }
    keyBit = parseInt(serialNumber[keyCounter]);
    intOfCharacter = intOfCharacter + keyBit + intOfLastCharacter;
    var toAdd = String.fromCharCode(intOfCharacter);
    encrypted = encrypted + toAdd;
    keyCounter++;
    if (keyCounter == endKey)
    {
       endKey = endKey - 1;
       keyCounter = 0;
       if (endKey == 0)
       {
          endKey = key.length;
       }
    }
     }
}
alert(encrypted);
this.qrImage.style.display ='none';
this.qrImage.src="https://chart.googleapis.com/chart?cht=qr&choe=UTF-8&chs=500x500&chl=" + encrypted;
```

```
   this.qrImage.style.display ='inline';

}

}

</script>

<body>

   <form action="" method="post" class="form">

   <h1>QR Code Creation

   </h1>

   <label>

      <span>Product Serial Number:</span>

      <input id="sn" type="text" name="name" placeholder="To find out look at settings of the product" />

   </label>


   <label>

      <span>Account Name:</span>

      <input id="an" type="text" name="name" placeholder="" />

   </label>


<label>

      <span>Username:</span>

      <input id="un" type="text" name="name" placeholder="" />

   </label>

    <label>

         <label>

      <span>Password:</span>

      <input type="password" id="pword"  name="name" placeholder="" />

   </label>

   </label>
```

```
    <label>
      <span> </span>
      <input type="button" class="button" value="Make QR Code" onclick="javascript:generateQRCode();" />
    </label>
    <label>
      <span>QR Code:</span>
      <img id='qrImage' style='display:inline;'/>
    </label>
</form>

</body>
</html>
```

Fingerprint Sensor Library

```
'''
Ashwin Ahuja - ashhwin.ahuja@gmail.com
December 2015
This code is published under the MIT License, please look at the main folder for more information
This code makes use of the module obtained from SparkFun - the GT511C3 - and makes lots of use of the
library for arduino provided by them - hence, please look at this.
This code has been tested on a Raspberry Pi 2 B, however, will in finality be used with a Raspberry Pi
Zero, so code has been designed to work with both of these.

This inherits majorly from Jean Machuca's attempts to do the same - please look at his library as well

'''

'''
Connections are as follows:
```

```
FP (Fingerprint Sensor) TX - RXD of Raspberry Pi GPIO

FP (Fingerprint Sensor) RX - TXD of Raspberry Pi GPIO

VCC of Fingerprint Sensor  - 3V3 of Raspberry Pi GPIO

GND of Fingerprint Sensor  - GND of Raspberry Pi GPIO


'''


import os

import serial

import time

import binascii


def delay(milliseconds):

   #allows me to maximise the code that can be lifted directly from the sparkfun library

   time.sleep(milliseconds)


deviceName = '/dev/cu.usbserial-A601EQ14' #this is the Pi Serial Connection


class packet:

   byte1 = 0x55;

   byte2 = 0xAA;

   bytedevid1 = 0x01;

   bytedevid2 = 0x00;


   def returnHighByte(self, word):

      return (w>>8)&0x00FF

   def returnLowByte (self, word):

      return w&0x00FF
```

```
  def CalculateCheckSum(self,bytearr):

    return sum(map(ord,bytes(bytearr)))


  def serializeToSend(self,bytearr):

    return ' '.join(binascii.hexlify(ch) for ch in bytes(bytearr))

class commandpacket(packet):

  cmd = ''

  command = bytearray(2)

  commands = {

        'NotSet'          : 0x00,    # Default value for enum. Scanner will return error if sent this.

        'Open'            : 0x01,    # Open Initialization

        'Close'           : 0x02,    # Close Termination

        'UsbInternalCheck'    : 0x03,    # UsbInternalCheck Check if the connected USB device is valid

        'ChangeBaudrate'      : 0x04,    # ChangeBaudrate Change UART baud rate

        'SetIAPMode'          : 0x05,    # SetIAPMode Enter IAP Mode In this mode, FW Upgrade is available

        'CmosLed'         : 0x12,    # CmosLed Control CMOS LED

        'GetEnrollCount'      : 0x20,    # Get enrolled fingerprint count

        'CheckEnrolled'       : 0x21,    # Check whether the specified ID is already enrolled

        'EnrollStart'         : 0x22,    # Start an enrollment

        'Enroll1'         : 0x23,    # Make 1st template for an enrollment

        'Enroll2'         : 0x24,    # Make 2nd template for an enrollment

        'Enroll3'         : 0x25,    # Make 3rd template for an enrollment, merge three templates into one
template, save merged template to the database

        'IsPressFinger'       : 0x26,    # Check if a finger is placed on the sensor

        'DeleteID'            : 0x40,    # Delete the fingerprint with the specified ID

        'DeleteAll'           : 0x41,    # Delete all fingerprints from the database

        'Verify1_1'           : 0x50,    # Verification of the capture fingerprint image with the specified ID

        'Identify1_N'         : 0x51,    # Identification of the capture fingerprint image with the database

        'VerifyTemplate1_1'   : 0x52,    # Verification of a fingerprint template with the specified ID
```

```
            'IdentifyTemplate1_N'    : 0x53,      # Identification of a fingerprint template with the database
            'CaptureFinger'          : 0x60,      # Capture a fingerprint image(256x256) from the sensor
            'MakeTemplate'           : 0x61,      # Make template for transmission
            'GetImage'               : 0x62,      # Download the captured fingerprint image(256x256)
            'GetRawImage'            : 0x63,      # Capture & Download raw fingerprint image(320x240)
            'GetTemplate'            : 0x70,      # Download the template of the specified ID
            'SetTemplate'            : 0x71,      # Upload the template of the specified ID
            'GetDatabaseStart'       : 0x72,      # Start database download, obsolete
            'GetDatabaseEnd'         : 0x73,      # End database download, obsolete
            'UpgradeFirmware'        : 0x80,      # Not supported
            'UpgradeISOCDImage'      : 0x81,      # Not supported
            'Ack'                    : 0x30,      # Acknowledge.
            'Nack'                   : 0x31       # Non-acknowledge
        }

    def __init__(self,*args,**kwargs):
        commandName=args[0]
        kwargs.setdefault('UseSerialDebug', True)
        self.UseSerialDebug= kwargs['UseSerialDebug']
        if self.UseSerialDebug:
            print 'Command: %s' % commandName
        self.cmd = self.commands[commandName]

    UseSerialDebug = True
    Parameter = bytearray(4)

    def GetPacketBytes(self):
        self.command[0] = self.GetLowByte(self.cmd)
```

```
        self.command[1] = self.GetHighByte(self.cmd)


        packetbytes= bytearray(12)
        packetbytes[0] = self.COMMAND_START_CODE_1
        packetbytes[1] = self.COMMAND_START_CODE_2
        packetbytes[2] = self.COMMAND_DEVICE_ID_1
        packetbytes[3] = self.COMMAND_DEVICE_ID_2
        packetbytes[4] = self.Parameter[0]
        packetbytes[5] = self.Parameter[1]
        packetbytes[6] = self.Parameter[2]
        packetbytes[7] = self.Parameter[3]
        packetbytes[8] = self.command[0]
        packetbytes[9] = self.command[1]
        chksum = self.CalculateCheckSum(packetbytes[0:9])
        packetbytes[10] = self.GetLowByte(chksum)
        packetbytes[11] = self.GetHighByte(chksum)


        return packetbytes;


    def ParameterFromInt(self, i):
        self.Parameter[0] = (i & 0x000000ff);
        self.Parameter[1] = (i & 0x0000ff00) >> 8;
        self.Parameter[2] = (i & 0x00ff0000) >> 16;
        self.Parameter[3] = (i & 0xff000000) >> 24;


class Response_Packet(Packet):
    '''
        Response Packet Class
```

```
    '''

    errors = {
            'NO_ERROR'              : 0x0000,    # Default value. no error
            'NACK_TIMEOUT'          : 0x1001,    # Obsolete, capture timeout
            'NACK_INVALID_BAUDRATE' : 0x1002,    # Obsolete, Invalid serial baud rate
            'NACK_INVALID_POS'      : 0x1003,    # The specified ID is not between 0~199
            'NACK_IS_NOT_USED'      : 0x1004,    # The specified ID is not used
            'NACK_IS_ALREADY_USED'  : 0x1005,    # The specified ID is already used
            'NACK_COMM_ERR'         : 0x1006,    # Communication Error
            'NACK_VERIFY_FAILED'    : 0x1007,    # 1:1 Verification Failure
            'NACK_IDENTIFY_FAILED'  : 0x1008,    # 1:N Identification Failure
            'NACK_DB_IS_FULL'       : 0x1009,    # The database is full
            'NACK_DB_IS_EMPTY'      : 0x100A,    # The database is empty
            'NACK_TURN_ERR'         : 0x100B,    # Obsolete, Invalid order of the enrollment (The order was not as:
EnrollStart -> Enroll1 -> Enroll2 -> Enroll3)
            'NACK_BAD_FINGER'       : 0x100C,    # Too bad fingerprint
            'NACK_ENROLL_FAILED'    : 0x100D,    # Enrollment Failure
            'NACK_IS_NOT_SUPPORTED' : 0x100E,    # The specified command is not supported
            'NACK_DEV_ERR'          : 0x100F,    # Device Error, especially if Crypto-Chip is trouble
            'NACK_CAPTURE_CANCELED' : 0x1010,    # Obsolete, The capturing is canceled
            'NACK_INVALID_PARAM'    : 0x1011,    # Invalid parameter
            'NACK_FINGER_IS_NOT_PRESSED' : 0x1012,    # Finger is not pressed
            'INVALID'               : 0XFFFF     # Used when parsing fails
        }

    def __init__(self,_buffer=None,UseSerialDebug=False):
        '''
        creates and parses a response packet from the finger print scanner
```

```
    '''
    self.UseSerialDebug= UseSerialDebug

    if not (_buffer is None ):
        self.RawBytes = _buffer
        self._lastBuffer = bytes(_buffer)
        if self.UseSerialDebug:
            print 'readed: %s'% self.serializeToSend(_buffer)
        if _buffer.__len__()>=12:
            self.ACK = True if _buffer[8] == 0x30 else False
            self.ParameterBytes[0] = _buffer[4]
            self.ParameterBytes[1] = _buffer[5]
            self.ParameterBytes[2] = _buffer[6]
            self.ParameterBytes[3] = _buffer[7]
            self.ResponseBytes[0]  = _buffer[8]
            self.ResponseBytes[1]  = _buffer[9]
            self.Error = self.ParseFromBytes(self.GetHighByte(_buffer[5]),self.GetLowByte(_buffer[4]))

_lastBuffer = bytes()
RawBytes = bytearray(12)
ParameterBytes=bytearray(4)
ResponseBytes=bytearray(2)
ACK = False
Error = None
UseSerialDebug = True

def ParseFromBytes(self,high,low):
```

```
        '''
        parses bytes into one of the possible errors from the finger print scanner
        '''
        e  = 'INVALID'
        if high == 0x01:
            if low in self.errors.values():
                errorIndex = self.errors.values().index(low)
                e = self.errors.keys()[errorIndex]
        return e



    def IntFromParameter(self):
        retval = 0;
        retval = (retval << 8) + self.ParameterBytes[3];
        retval = (retval << 8) + self.ParameterBytes[2];
        retval = (retval << 8) + self.ParameterBytes[1];
        retval = (retval << 8) + self.ParameterBytes[0];
        return retval;


class SerialCommander:
    def __serialize_args_hex__(self,*arg,**kwargs):
        return bytes(bytearray([v for v in kwargs.values()]))
    def serializeToSend(self,bytearr):
        return ' '.join(binascii.hexlify(ch) for ch in bytes(bytearr))
    def unserializeFromRead(self,char_readed,bytearr):
        bytearr.append(char_readed)
        return bytearr
```

```
def connect(device_name=None,baud=None,timeout=None,is_com=True):
    _ser = None
        is_com = false
    baud = 9600
    device_name = '/dev/ttyAMA0'
    timeout = 2000
    return _ser




BAUD = 9600


class fingerprintseonsor(SerialCommander):
    _serial = None
    _lastResponse = None
    _device_name = None
    _baud = None
    _timeout= None

    UseSerialDebug = True

    def __init__(self,device_name=None,baud=None,timeout=None,is_com=True):
        self._device_name = device_name
        self._baud=baud
        self._timeout = timeout
        self._serial = connect(device_name,baud,timeout,is_com=is_com)
        if not self._serial is None:
            delay(0.1)
```

```
            self.Open()

        elif self.UseSerialDebug:

            print 'No connection with this device:- %s' % self._device_name



    def Open(self):


        self.ChangeBaudRate(BAUD)

        delay(0.1)

        cp = Command_Packet('Open',UseSerialDebug=self.UseSerialDebug)

        cp.ParameterFromInt(1)

        packetbytes = cp.GetPacketBytes()

        self.SendCommand(packetbytes, 12)

        rp = self.GetResponse()

        del packetbytes

        return rp.ACK



    def Close(self):

        cp = Command_Packet('Close',UseSerialDebug=self.UseSerialDebug)

        cp.Parameter[0] = 0x00;

        cp.Parameter[1] = 0x00;

        cp.Parameter[2] = 0x00;

        cp.Parameter[3] = 0x00;

        packetbytes = cp.GetPacketBytes()

        self.SendCommand(packetbytes, 12)

        rp = self.GetResponse()

        if not self._serial is None:
```

```
            self._serial.close()
        del packetbytes
        return rp.ACK

    def SetLED(self,on=True):
        cp = Command_Packet('CmosLed',UseSerialDebug=self.UseSerialDebug)
        cp.Parameter[0] = 0x01 if on else 0x00;
        cp.Parameter[1] = 0x00;
        cp.Parameter[2] = 0x00;
        cp.Parameter[3] = 0x00;
        packetbytes = cp.GetPacketBytes()
        self.SendCommand(packetbytes, 12)
        rp = self.GetResponse()
        retval = rp.ACK
        del rp
        del packetbytes
        return retval

    def CheckEnrolled(self,ID):
        cp = Command_Packet('CheckEnrolled',UseSerialDebug=self.UseSerialDebug)
        cp.ParameterFromInt(ID)
        packetbytes = cp.GetPacketBytes()
        del cp
        self.SendCommand(packetbytes, 12)
        del packetbytes
        rp = self.GetResponse()
        retval = rp.ACK
        del rp
```

```
        return retval

    def EnrollStart(self,ID):
        cp = Command_Packet('EnrollStart',UseSerialDebug=self.UseSerialDebug)
        cp.ParameterFromInt(ID)
        packetbytes = cp.GetPacketBytes()
        del cp
        self.SendCommand(packetbytes, 12)
        del packetbytes
        rp = self.GetResponse()
        retval = 0
        if not rp.ACK:
            if rp.Error == rp.errors['NACK_DB_IS_FULL']:
                retval = 1
            elif rp.Error == rp.errors['NACK_INVALID_POS']:
                retval = 2
            elif rp.Error == rp.errors['NACK_IS_ALREADY_USED']:
                retval = 3
        del rp
        return retval
    def Enroll1(self):
        cp = Command_Packet('Enroll1',UseSerialDebug=self.UseSerialDebug)
        packetbytes = cp.GetPacketBytes()
        del cp
        self.SendCommand(packetbytes, 12)
        del packetbytes
        rp = self.GetResponse()
        retval = rp.IntFromParameter()
        retval = 3 if retval < 200 else 0
```

```
        if not rp.ACK:
            if rp.Error == rp.errors['NACK_ENROLL_FAILED']:
                retval = 1
            elif rp.Error == rp.errors['NACK_BAD_FINGER']:
                retval = 2
        return 0 if rp.ACK else retval
    def Enroll2(self):
        cp = Command_Packet('Enroll2',UseSerialDebug=self.UseSerialDebug)
        packetbytes = cp.GetPacketBytes()
        del cp
        self.SendCommand(packetbytes, 12)
        del packetbytes
        rp = self.GetResponse()
        retval = rp.IntFromParameter()
        retval = 3 if retval < 200 else 0
        if not rp.ACK:
            if rp.Error == rp.errors['NACK_ENROLL_FAILED']:
                retval = 1
            elif rp.Error == rp.errors['NACK_BAD_FINGER']:
                retval = 2
        return 0 if rp.ACK else retval
    def Enroll3(self):
        cp = Command_Packet('Enroll3',UseSerialDebug=self.UseSerialDebug)
        packetbytes = cp.GetPacketBytes()
        del cp
        self.SendCommand(packetbytes, 12)
        del packetbytes
        rp = self.GetResponse()
```

```
        retval = rp.IntFromParameter()
        retval = 3 if retval < 200 else 0
        if not rp.ACK:
            if rp.Error == rp.errors['NACK_ENROLL_FAILED']:
                retval = 1
            elif rp.Error == rp.errors['NACK_BAD_FINGER']:
                retval = 2
        return 0 if rp.ACK else retval
    def IsPressFinger(self):
        cp = Command_Packet('IsPressFinger',UseSerialDebug=self.UseSerialDebug)
        packetbytes = cp.GetPacketBytes()
        self.SendCommand(packetbytes, 12)
        rp = self.GetResponse()
        pval = rp.ParameterBytes[0]
        pval += rp.ParameterBytes[1]
        pval += rp.ParameterBytes[2]
        pval += rp.ParameterBytes[3]
        retval = True if pval == 0 else False
        del rp
        del packetbytes
        del cp
        return retval

    def DeleteID(self,ID):
        cp = Command_Packet('DeleteID',UseSerialDebug=self.UseSerialDebug)
        cp.ParameterFromInt(ID)
        packetbytes = cp.GetPacketBytes()
        self.SendCommand(packetbytes, 12)
```

```
        rp = self.GetResponse()
        retval = rp.ACK
        del rp
        del packetbytes
        del cp
        return retval

    def Identify1_N(self):
        cp = Command_Packet('Identify1_N',UseSerialDebug=self.UseSerialDebug)
        packetbytes = cp.GetPacketBytes()
        self.SendCommand(packetbytes, 12)
        rp = self.GetResponse()
        retval = rp.IntFromParameter()
        if retval > 200:
            retval = 200
        del rp
        del packetbytes
        del cp
        return retval


    def CaptureFinger(self,highquality=True):
        cp = Command_Packet('CaptureFinger',UseSerialDebug=self.UseSerialDebug)
        cp.ParameterFromInt(1 if highquality else 0)
        packetbytes = cp.GetPacketBytes()
        self.SendCommand(packetbytes, 12)
        rp = self.GetResponse()
        retval = rp.ACK
```

```
        del rp
        del packetbytes
        del cp
        return retval


    def GetTemplate(self, ID):
        '''
            Gets a template from the fps (498 bytes) in 4 Data_Packets
            Use StartDataDownload, and then GetNextDataPacket until done
            Parameter: 0-199 ID number
            Returns:
              0 - ACK Download starting
              1 - Invalid position
              2 - ID not used (no template to download
        '''
        cp = Command_Packet('GetTemplate',UseSerialDebug=self.UseSerialDebug)
        cp.ParameterFromInt(ID)
        packetbytes = cp.GetPacketBytes()
        self.SendCommand(packetbytes, 12)
        rp = self.GetResponse()
        retval = 0
        if not rp.ACK:
            if rp.Error == rp.errors['NACK_INVALID_POS']:
                retval = 1
            elif rp.Error == rp.errors['NACK_IS_NOT_USED']:
                retval = 2
        return retval
```

Main Product Code

```
import RPi.GPIO as GPIO
```

```
import time

import os

import FPS, sys

import Adafruit_GPIO.SPI as SPI

import Adafruit_SSD1306

import Image

import ImageDraw

import ImageFont

from Crypto.Cipher import AES

import hashlib

import random

import struct

RST = 24

asleep = False

disp = Adafruit_SSD1306.SSD1306_128_32(rst=RST)

disp.begin()

disp.clear()

disp.display()


width = disp.width

height= disp.height

padding = 1

x=padding

top = padding

bottom = height - padding

image = Image.new('1', (width, height))

draw = ImageDraw.Draw(image)

font = ImageFont.load_default()
```

```
draw.text((x, top), 'Hello', font = font, fill = 255)

draw.text((x, top+20), 'Product Loading', font = font, fill = 255)

disp.image(image)

disp.display()

powerSwitchPin = 20

latchPin = 21


GPIO.setmode(GPIO.BCM)

encoderPinA = 23

encoderPinB = 25

GPIO.setup(encoderPinA, GPIO.IN)

GPIO.setup(encoderPinB, GPIO.IN)

GPIO.setup(powerSwitchPin, GPIO.IN)

GPIO.setup(latchPin, GPIO.IN)

GPIO.add_event_detect(powerSwitchPin, GPIO.BOTH, callback=turnOff)

GPIO.add_event_detect(latchPin, GPIO.BOTH, callback=turnOff)

switchPin = 16

GPIO.setup(switchPin, GPIO.IN, pull_up_down=GPIO.PUD_UP)

fps = FPS.FPS_GT511C3(device_name='/dev/ttyAMA0',baud=9600, timeout=2, is_com=False)

fps.Open()

int serialNumber  = 123456789

tempValueForEncoder = 0

encoderPosition = 0

encoderLastPin = 0


from threading import Timer


locked = True
```

```
importKey = "a"

redPin = 26

greenPin = 19

bluePin = 13

GPIO.setup(redPin, GPIO.OUT)

GPIO.setup(greenPin, GPIO.OUT)

GPIO.setup(bluePin, GPIO.OUT)


previousSettingsShown = False;

def sleepProperly():
   asleep = True
   time.Sleep(10)



def turnOff():
   if asleep:
      asleep = False
      main()
   else:
      key = fps.GetTemplate()
      key2 = hashlib.sha256(key).digest()
      encrypt_file(key2, "EP.txt")
      os.system("rm EP.txt")
      sleepProperly()



def remainInSleep(fps):
   while GPIO.input(switchPin):
```

```
        print GPIO.input(switchPin)
        time.sleep(1)



def timeout():
    password = fps.GetTemplate()
    key = hashlib.sha256(password).digest()
    encrypt_file(key, "EP.txt")
    os.system("rm EP.txt")
    locked = True



def importDataFromQR():
    success = False
    result = ""
    while not GPIO.input(switchPin) and not success:
        os.System("fswebcam -r 1280x720 --no-banner image.jpg")
        myCode = QR(filename = "image.jpg")
        if(myCode.decode()):
            result = myCode.data_to_string()
            success = True
    if success:
        disp.clear()
        draw.text((x, top), "Data Imported", font = font, fill = 255)
        draw.text((x, top+20), 'from QR', font = font, fill = 255)
        disp.image(image)
```

```
    disp.display()

    key = str(serialNumber)

    decrypted = ""

    counter = 0

    keyCounter = 0

    keyEnd = serialNumber.Length

    for i in result:

      counter = counter + 1

      if (i == '|'):

        decrypted = decrypted + "|";

      else:

        c = i;

        a = ord(c);

        last = 0;

        if (counter >  1):

          last = ord(i-1);

        int keyBit = int(SerialNumber[keyCounter]);

        a = a - last - keyBit;

        strl = str(unichr(a))

        decrypted = decrypted + strl

        keyCounter = keyCounter + 1;

        if (keyCounter == keyEnd):

          keyCounter = 0;

          keyEnd = keyEnd - 1;

          if (keyEnd == 0):

            keyEnd = SerialNumber.Length;

    with open("EP.txt", "a") as f:

      f.write(decrypted)
```

```
        main()
    else:
        disp.clear()
        draw.text((x, top), "Data Import", font = font, fill = 255)
        draw.text((x, top+20), 'FAILED', font = font, fill = 255)
        disp.image(image)
        disp.display()
        main()



def importDataFromUSB():
    os.System("sudo mkdir /media/usb")
    os.System("poonam123")
    os.System("sudo chown -R pi:pi /media/usb")
    os.System("sudo mount /dev/sda1 /media/usb -o uid=pi,gid=pi")
    os.System("sudo cp /media/USB/out.txt.enc /")
    key = hashlib.sha256(importKey).decode()
    decrypt_file(key, "out.txt.enc")
    with open("/EP.txt") as output:
        with open("/out.txt") as inputFile:
            for line in inputFile:
                output.write(line)



def backupDataToUSB():
    os.System("sudo mkdir /media/usb")
```

```
    os.System("poonam123")

    os.System("sudo chown -R pi:pi /media/usb")

    os.System("sudo mount /dev/sda1 /media/usb -o uid=pi,gid=pi")

    key = hashlib.sha256(importKey).decode()

    encrypt_file(key, "EP.txt")

    os.System("sudo cp EP.txt.enc out.txt.enc")

    os.System("sudo cp out.txt.enc /media/USB/out.txt.enc")



def displayMenu (counter, lines):

    if (counter == 0 and not previousSettingsShown):

        disp.clear()

        draw.text((x, top), "Settings", font = font, fill = 255)

        disp.image(image)

        disp.display()

        counter = -1

        previousSettingsShown = True;


    str4 = lines[counter]

    str2 = str4.split('|', 3)

    disp.clear()

    draw.text((x, top), str2[0], font = font, fill = 255)

    disp.image(image)

    disp.display()

    previousSettingsShown = False;


def decrypt_file(key, in_filename, out_filename=None, chunksize=24*1024):

    """ Decrypts a file using AES (CBC mode) with the
```

```
    given key. Parameters are similar to encrypt_file,
    with one difference: out_filename, if not supplied
    will be in_filename without its last extension
    (i.e. if in_filename is 'aaa.zip.enc' then
    out_filename will be 'aaa.zip')
  """
  if not out_filename:
    out_filename = os.path.splitext(in_filename)[0]

  with open(in_filename, 'rb') as infile:
    origsize = struct.unpack('<Q', infile.read(struct.calcsize('Q')))[0]
    iv = infile.read(16)
    decryptor = AES.new(key, AES.MODE_CBC, iv)

    with open(out_filename, 'wb') as outfile:
      while True:
        chunk = infile.read(chunksize)
        if len(chunk) == 0:
          break
        outfile.write(decryptor.decrypt(chunk))

      outfile.truncate(origsize)

def encrypt_file(key, in_filename, out_filename=None, chunksize=64*1024):
  """ Encrypts a file using AES (CBC mode) with the
    given key.

    key:
```

```
        The encryption key - a string that must be
        either 16, 24 or 32 bytes long. Longer keys
        are more secure.

    in_filename:
        Name of the input file

    out_filename:
        If None, '<in_filename>.enc' will be used.

    chunksize:
        Sets the size of the chunk which the function
        uses to read and encrypt the file. Larger chunk
        sizes can be faster for some files and machines.
        chunksize must be divisible by 16.
"""
if not out_filename:
    out_filename = in_filename + '.enc'

iv = ''.join(chr(random.randint(0, 0xFF)) for i in range(16))
encryptor = AES.new(key, AES.MODE_CBC, iv)
filesize = os.path.getsize(in_filename)

with open(in_filename, 'rb') as infile:
    with open(out_filename, 'wb') as outfile:
        outfile.write(struct.pack('<Q', filesize))
        outfile.write(iv)
```

```
        while True:
            chunk = infile.read(chunksize)
            if len(chunk) == 0:
                break
            elif len(chunk) % 16 != 0:
                chunk += ' ' * (16 - len(chunk) % 16)

            outfile.write(encryptor.encrypt(chunk))

def displaySettingsMenu():
    disp.clear()
    draw.text((x, top), serialNumber, font = font, fill = 255)
    draw.text((x, top+20), 'Serial Number', font = font, fill = 255)
    disp.image(image)
    disp.display()
    tenpCounter = 0
    while not GPIO.input(switchPin):
        t = twistedClockwise()
        if not t=="no":
            if t == "down":
                tempCounter = tempCounter + 1
            if t == "up":
                tempCounter = tempCounter - 1
        if(tempCounter == 0):
            disp.clear()
            draw.text((x, top), serialNumber, font = font, fill = 255)
            draw.text((x, top+20), 'Serial Number', font = font, fill = 255)
            disp.image(image)
```

```
        disp.display()
    elif(tempCounter==1):
        disp.clear()
        draw.text((x, top), "Import Data", font = font, fill = 255)
        draw.text((x, top+20), 'from QR', font = font, fill = 255)
        disp.image(image)
        disp.display()
    elif(tempCounter == 2):
        disp.clear()
        draw.text((x, top), "Import Data", font = font, fill = 255)
        draw.text((x, top+20), 'from USB', font = font, fill = 255)
        disp.image(image)
        disp.display()
    elif(tempCounter == 3):
        disp.clear()
        draw.text((x, top), "Backup Data", font = font, fill = 255)
        draw.text((x, top+20), 'to USB', font = font, fill = 255)
        disp.image(image)
        disp.display()
    elif(tempCounter == 4):
        disp.clear()
        draw.text((x, top), "Enroll a Finger", font = font, fill = 255)
        disp.image(image)
        disp.display()
    elif(tempCounter < 0):
        tempCounter = 4
    elif(tempCounter > 4):
        tempCounter = 0
```

```
    Time.sleep(3)
    if(GPIO.Input(switchPin)):
        locked = False
        main()
    else:
        if(tempCounter = 1):
            importDataFromQR()
        elif(tempCounter == 2):
            importDataFromUSB()
        elif(tempCounter == 3):
            backupDataToUSB()
        elif(tempCounter == 4):
            enroll(fps)
```

```
def displayMoreInfo(counter, lines):
    if(counter == -1):
        displaySettingsMenu()
    str4 = lines[counter]
    str2 = str4.split('|', 3)
    disp.clear()
    draw.text((x, top), str2[1], font = font, fill = 255)
    draw.text((x, bottom), str2[2], font = font, fill = 255)
    disp.image(image)
    disp.display()
```

```
def enroll(fps):
    enrollid=0
    okid=False
    #search for a free enrollid, you have max 200
    while not okid and enrollid < 200:
        okid = fps.CheckEnrolled(enrollid)
        if not okid:
            enrollid+=1
    if enrollid <200:
        #press finger to Enroll enrollid
        print 'Press finger to Enroll %s' % str(enrollid)
        disp.clear()
        draw.text((x, top), 'PLACE YOUR FINGER', font = font, fill = 255)
        draw.text((x, bottom), 'ON SCANNER')
        disp.image(image)
        disp.display()
        fps.EnrollStart(enrollid)
        while not fps.IsPressFinger():
            FPS.delay(1)
        iret = 0
        if fps.CaptureFinger(True):
            #remove finger
            print 'remove finger'
            disp.clear()
```

```
        draw.text((x, top), 'REMOVE FINGER', font = font, fill = 255)
        draw.text((x, bottom), 'FROM SCANNER')
        disp.image(image)
        disp.display()
        fps.Enroll1()
        while not fps.IsPressFinger():
            FPS.delay(1)
        #Press same finger again
        print 'Press same finger again'
        disp.clear()
        draw.text((x, top), 'PLACE SAME FINGER', font = font, fill = 255)
        draw.text((x, bottom), 'ON SCANNER')
        disp.image(image)
        disp.display()
        while not fps.IsPressFinger():
            FPS.delay(1)
        if fps.CaptureFinger(True):
            #remove finger
            print 'remove finger'
            disp.clear()
            draw.text((x, top), 'REMOVE FINGER', font = font, fill = 255)
            draw.text((x, bottom), 'FROM SCANNER')
            disp.image(image)
            disp.display()
            fps.Enroll2()
            while not fps.IsPressFinger():
                FPS.delay(1)
            #Press same finger again
```

```
        print 'press same finger yet again'
        disp.clear()
        draw.text((x, top), 'PLACE SAME FINGER', font = font, fill = 255)
        draw.text((x, bottom), 'ON SCANNER')
        disp.image(image)
        disp.display()
        while not fps.IsPressFinger():
            FPS.delay(1)
        if fps.CaptureFinger(True):
            #remove finger
            iret = fps.Enroll3()
            if iret == 0:
                print 'Enrolling Successfull'
                disp.clear()
                draw.text((x, top), 'ENROLL SUCCESSFUL', font = font, fill = 255)
                disp.image(image)
                disp.display()
            else:
                print 'Enrolling Failed with error code: %s' % str(iret)
                disp.clear()
                draw.text((x, top), 'ENROLL FAILED', font = font, fill = 255)
                disp.image(image)
                disp.display()
        else:
            print 'Failed to capture third finger'
            disp.clear()
            draw.text((x, top), 'ENROLL FAILED', font = font, fill = 255)
            disp.image(image)
```

```
            disp.display()
        else:
            print 'Failed to capture second finger'
            disp.clear()
            draw.text((x, top), 'ENROLL FAILED', font = font, fill = 255)
            disp.image(image)
            disp.display()
    else:
        print 'Failed to capture first finger'
        disp.clear()
        draw.text((x, top), 'ENROLL FAILED', font = font, fill = 255)
        disp.image(image)
        disp.display()
else:
    print 'Failed: enroll storage is full'
    disp.clear()
    draw.text((x, top), 'ENROLL FAILED', font = font, fill = 255)
    disp.image(image)
    disp.display()

#ENROLLMENT DETAILS




def verifyAUser(fps):
    disp.clear()
    draw.text((x, top), 'PLACE YOUR FINGER', font = font, fill = 255)
    draw.text((x, bottom), 'ON SCANNER')
```

```
    disp.image(image)
    disp.display()
    while not fps.IsPressFinger():
        time.sleep(1)
    while not fps.CaptureFinger(False):
        disp.clear()
        draw.text((x, top), 'Failed to capture', font = font, fill = 255)
        disp.image(image)
        disp.display()
        print "FAILED to CAPTURE FINGER"
    returnedValue = fps.Identify1_N()
    if returnedValue > 199:
        disp.clear()
        draw.text((x, top), 'No Entry', font = font, fill = 255)
        disp.image(image)
        disp.display()
        return false
    else:
        disp.clear()
        draw.text((x, top), 'Verified', font = font, fill = 255)
        disp.image(image)
        disp.display()
        return True


def twistedClockwise():
    n = GPIO.input(encoderPinA)
    if((encoderLastPin == False) and (n == True)):
        if not GPIO.input(encoderPinB):
```

```
            encoderPosition = encoderPosition + 1
            encoderLastPin = n
            return "down"
        else:
            encoderPosition = encoderPosition - 1
            encoderLastPin = n
            return "up"
    encoderLastPin = n
    return "no"


def main():
    locked = True
    while True:
        while(locked):
            GPIO.output(redPin, GPIO.LOW)
            GPIO.output(greenPin, GPIO.HIGH)
            GPIO.output(bluePin, GPIO.HIGH)
            disp.clear()
            fps.SetLED(True)
            remainInSleep(fps)
            for i in range(0,9):
                if(verifyAUser(fps)):
                    locked = false
                    i = 10
                    break
        t = Timer(60000, timeout)
        t.Start()
```

```
    GPIO.output(redPin, GPIO.HIGH)

    GPIO.output(greenPin, GPIO.LOW)

    menuCounter = 0

    key = hashlib.sha256(fps.GetTemplate())

    decrypt_file(key, "EP.txt.enc")

    with open('EP.txt') as f:

      lines = f.readlines()

    while not locked:


      while not GPIO.input(switchPin):

        #decodeFile

        #display menu and allow the user to move up and down the list

        #displayMenu

        if locked:

          break

        t = twistedClockwise()

        if not t=="no":

          if t == "down":

            if (menuCounter + 1 >= len(lines)):

              menuCounter = 0

            else:

              menuCounter = menuCounter + 1

            displayMenu(menuCounter, lines)

          if t == "up":

            if (menuCounter - 1 <= 0):

              menuCounter = len(lines)-1

            else:

              menuCounter = menuCounter - 1
```

```
            displayMenu(menuCounter, lines)


        while not GPIO.input(switchPin):
            if locked:
                break
            displayMoreInfo()


                #moveMenuRight



try:
    main()
except KeyboardInterrupt:
    GPIO.cleanup()
```

Windows Application

MainWindow XAML

```
<Window x:Class="QR_Code_Generator.MainWindow"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
        xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
        xmlns:local="clr-namespace:QR_Code_Generator"
        mc:Ignorable="d"
        Title="QR Code Generator" Height="350" Width="772.079">
    <Grid>


        <TextBox x:Name="textBox" HorizontalAlignment="Left" Height="23" TextWrapping="Wrap" Text="The Serial Number is available in the Product's Settings" VerticalAlignment="Top" Width="324" Margin="10,26,0,0" ToolTip="Information is available in the product's settings" TextAlignment="Center"/>
```

```
    <Label x:Name="label2" Content="Account Name" HorizontalAlignment="Left" Margin="133,56,0,0"
VerticalAlignment="Top" FontWeight="Bold"/>

    <TextBox x:Name="textBox_Copy" HorizontalAlignment="Left" Height="23" TextWrapping="Wrap" Text="e.g.
Yahoo Mail" VerticalAlignment="Top" Width="324" Margin="10,87,0,0" ToolTip="Information is available in the
product's settings" TextAlignment="Center"/>

    <Label x:Name="label2_Copy" Content="Username" HorizontalAlignment="Left" Margin="146,115,0,0"
VerticalAlignment="Top" FontWeight="Bold"/>

    <TextBox x:Name="textBox_Copy1" HorizontalAlignment="Left" Height="23" TextWrapping="Wrap" Text="e.g.
joebloggs" VerticalAlignment="Top" Width="324" Margin="10,146,0,0" ToolTip="Information is available in the
product's settings" TextAlignment="Center"/>

    <Label x:Name="label2_Copy1" Content="Password" HorizontalAlignment="Left" Margin="146,174,0,0"
VerticalAlignment="Top" FontWeight="Bold"/>

    <PasswordBox x:Name="passwordBox" HorizontalAlignment="Left" Margin="10,204,0,0"
VerticalAlignment="Top" Width="324" HorizontalContentAlignment="Center"/>

    <Label x:Name="label2_Copy2" Content="Serial Number" HorizontalAlignment="Left" Margin="133,0,0,0"
VerticalAlignment="Top" FontWeight="Bold"/>

    <Button x:Name="button" Content="Generate QR Code" HorizontalAlignment="Left" Margin="113,252,0,0"
VerticalAlignment="Top" Width="129" Height="46" Click="button_Click"/>

    <Image x:Name="imageBox" HorizontalAlignment="Left" Height="321" VerticalAlignment="Top" Width="353"
Margin="395,0,0,-0.2" IsEnabled="True" Loaded="Image_Loaded"/>


  </Grid>

</Window>
```

MainWindow CS

```
using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;
```

```
using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;


namespace QR_Code_Generator

{

   /// <summary>

   /// Interaction logic for MainWindow.xaml

   /// </summary>

   public partial class MainWindow : Window

   {

      public MainWindow()

      {

         InitializeComponent();

      }


      private void Image_Loaded(object sender, RoutedEventArgs e)

      {

         string SerialNumber = textBox.Text;

         string username = textBox_Copy1.Text;

         string accountName = textBox_Copy.Text;

         string password = passwordBox.Password;

         string complete = accountName + "|" + username + "|" + password;

         string encrypted = "";

         int keyCounter = 0;

         int keyEnd = SerialNumber.Length;
```

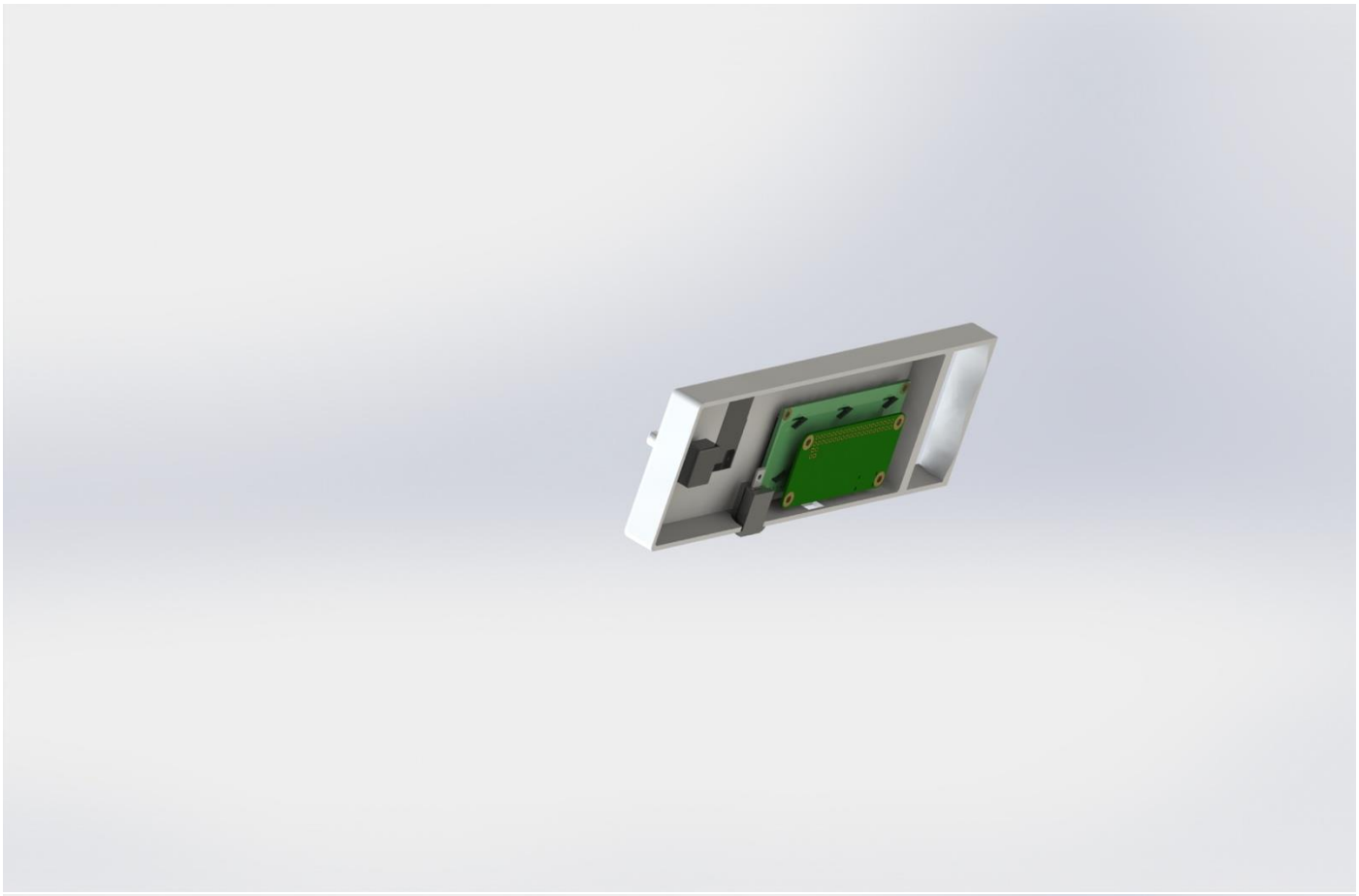
```
        for (int i = 0; i < complete.Length; i++)
        {
            if (complete[i] == '|')
            {
                encrypted = encrypted + "|";
            }
            else
            {
                char c = complete[i];
                int a = Convert.ToInt32(Char.GetNumericValue(c));
                int last = 0;
                if (i > 0)
                {
                    last = Convert.ToInt32(Char.GetNumericValue(complete[i - 1]));
                }
                int keyBit = Convert.ToInt16(SerialNumber[keyCounter]);
                a = a + last + keyBit;
                char l = (char)a;
                string strl = Convert.ToString(l);
                encrypted = encrypted + strl;
                keyCounter++;
                if (keyCounter == keyEnd)
                {
                    keyCounter = 0;
                    keyEnd--;
                    if (keyEnd == 0)
                    {
                        keyEnd = SerialNumber.Length;
```

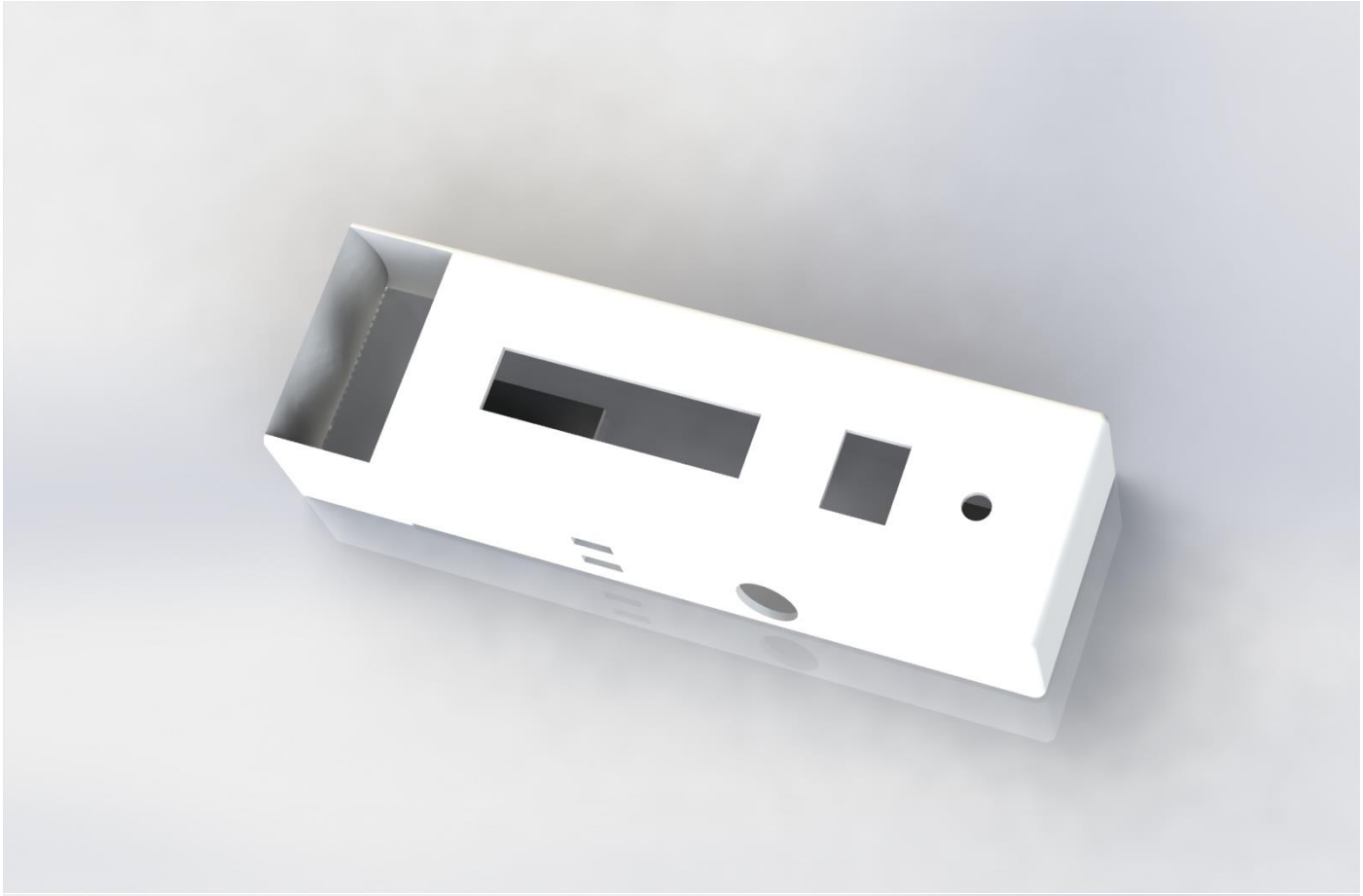
```
                }

            }

        }

    }

    BitmapImage b = new BitmapImage();

    b.BeginInit();

    b.UriSource = new Uri("https://chart.googleapis.com/chart?chs=450x450&cht=qr&chl=" + encrypted +
"&choe=UTF-8");

    b.EndInit();

    var image = sender as Image;

    image.Source = b;

}


private void button_Click(object sender, RoutedEventArgs e)

{

    string SerialNumber = textBox.Text;

    string username = textBox_Copy1.Text;

    string accountName = textBox_Copy.Text;

    string password = passwordBox.Password;

    string complete = accountName + "|" + username + "|" + password;

    string encrypted = "";

    int keyCounter = 0;

    int keyEnd = SerialNumber.Length;

    for (int i = 0; i < complete.Length; i++)

    {

        if (complete[i] == '|')

        {

            encrypted = encrypted + "|";

        }
```

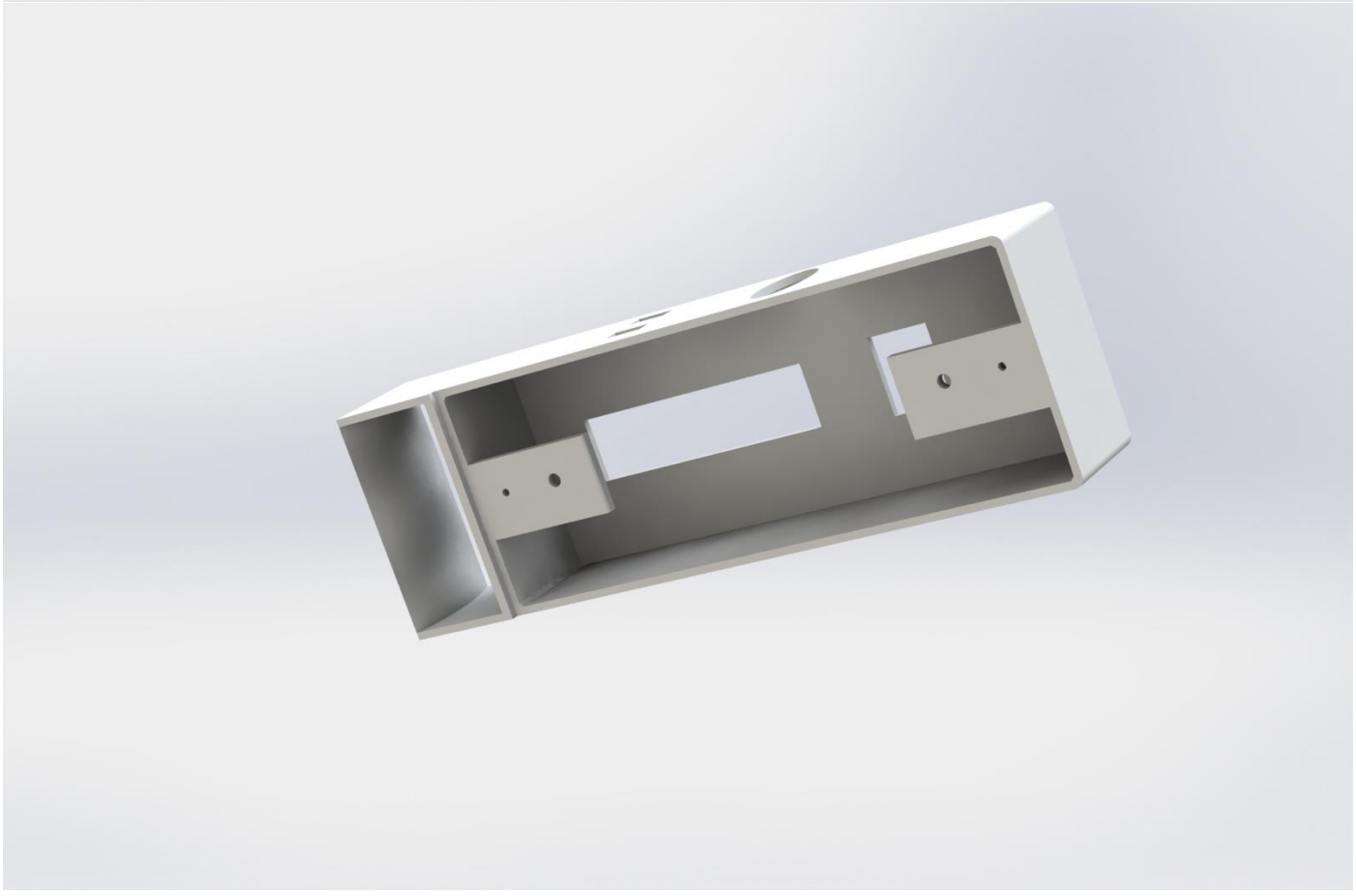
```
        else
        {
            char c = complete[i];
            int a = Convert.ToInt32(Char.GetNumericValue(c));
            int last = 0;
            if (i > 0)
            {
                last = Convert.ToInt32(Char.GetNumericValue(complete[i - 1]));
            }
            int keyBit = Convert.ToInt16(SerialNumber[keyCounter]);
            a = a + last + keyBit;
            char l = (char)a;
            string strl = Convert.ToString(l);
            encrypted = encrypted + strl;
            keyCounter++;
            if (keyCounter == keyEnd)
            {
                keyCounter = 0;
                keyEnd--;
                if (keyEnd == 0)
                {
                    keyEnd = SerialNumber.Length;
                }
            }
        }
    }
    BitmapImage b = new BitmapImage();
    b.BeginInit();
```

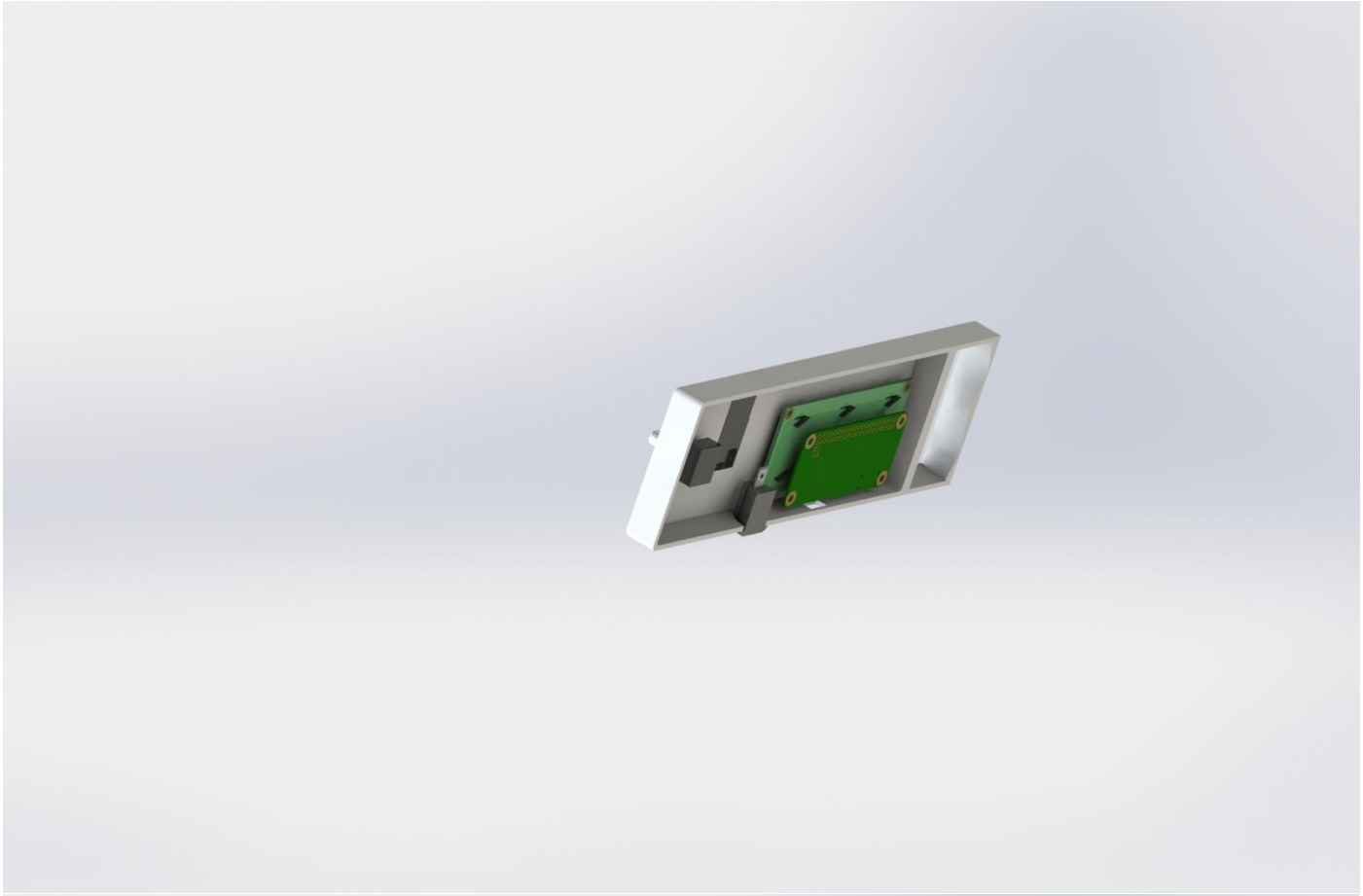
```
        b.UriSource = new Uri("https://chart.googleapis.com/chart?chs=450x450&cht=qr&chl=" + encrypted +
"&choe=UTF-8");

        b.EndInit();

        var image = imageBox as Image;

        image.Source = b;

      }

   }

}
```

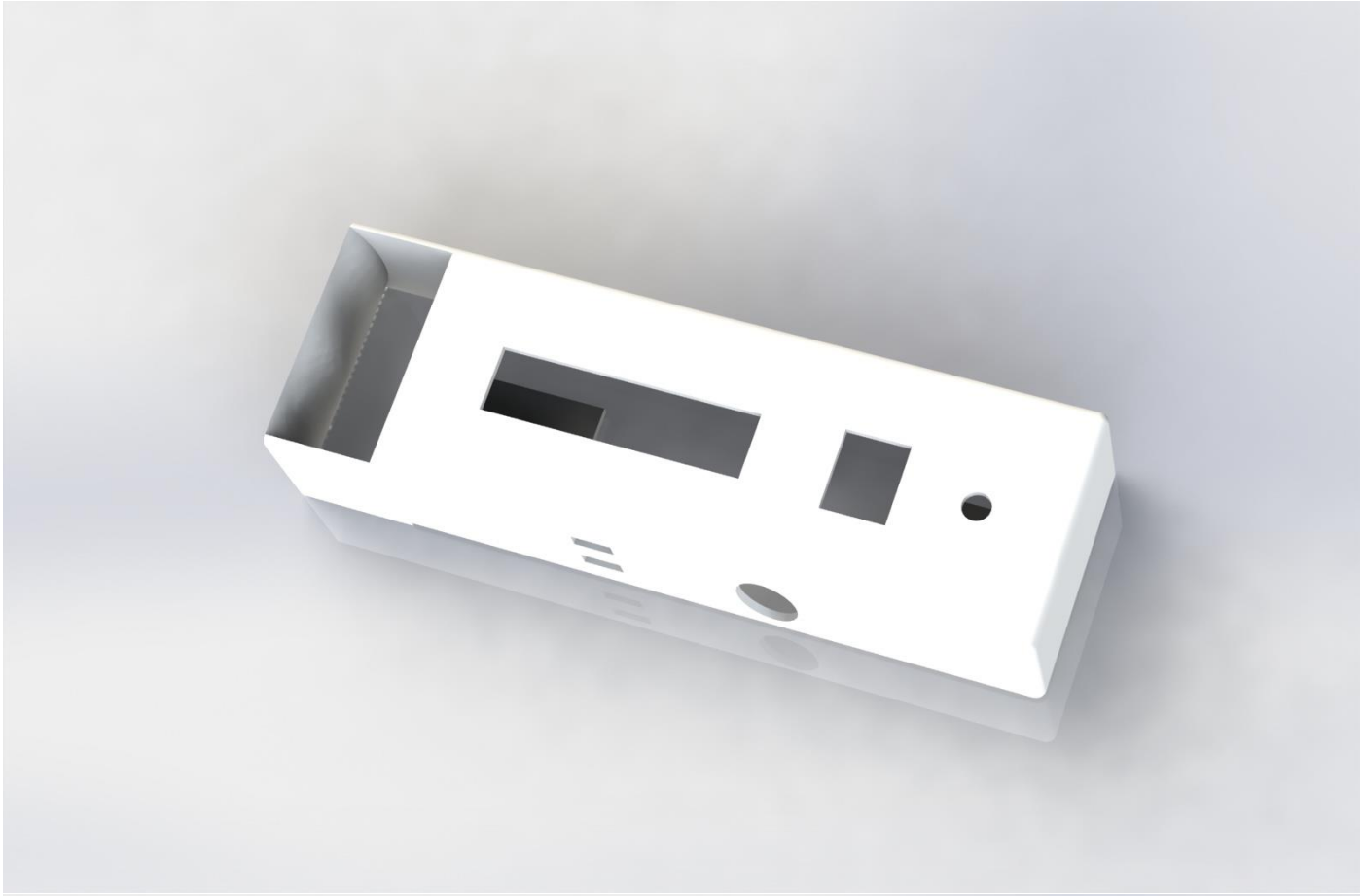
F Renders of Mechanical Designs of Artefact

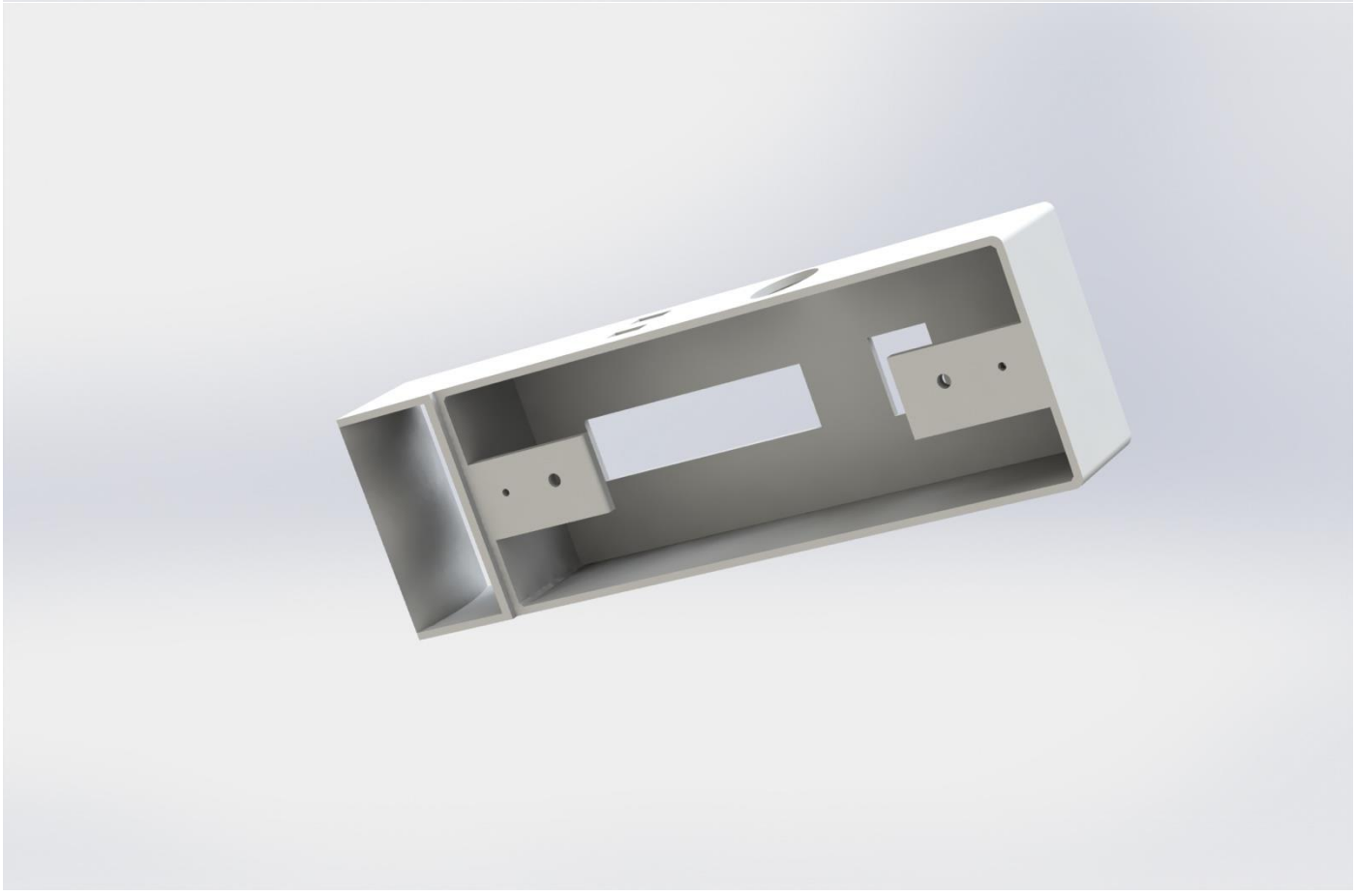


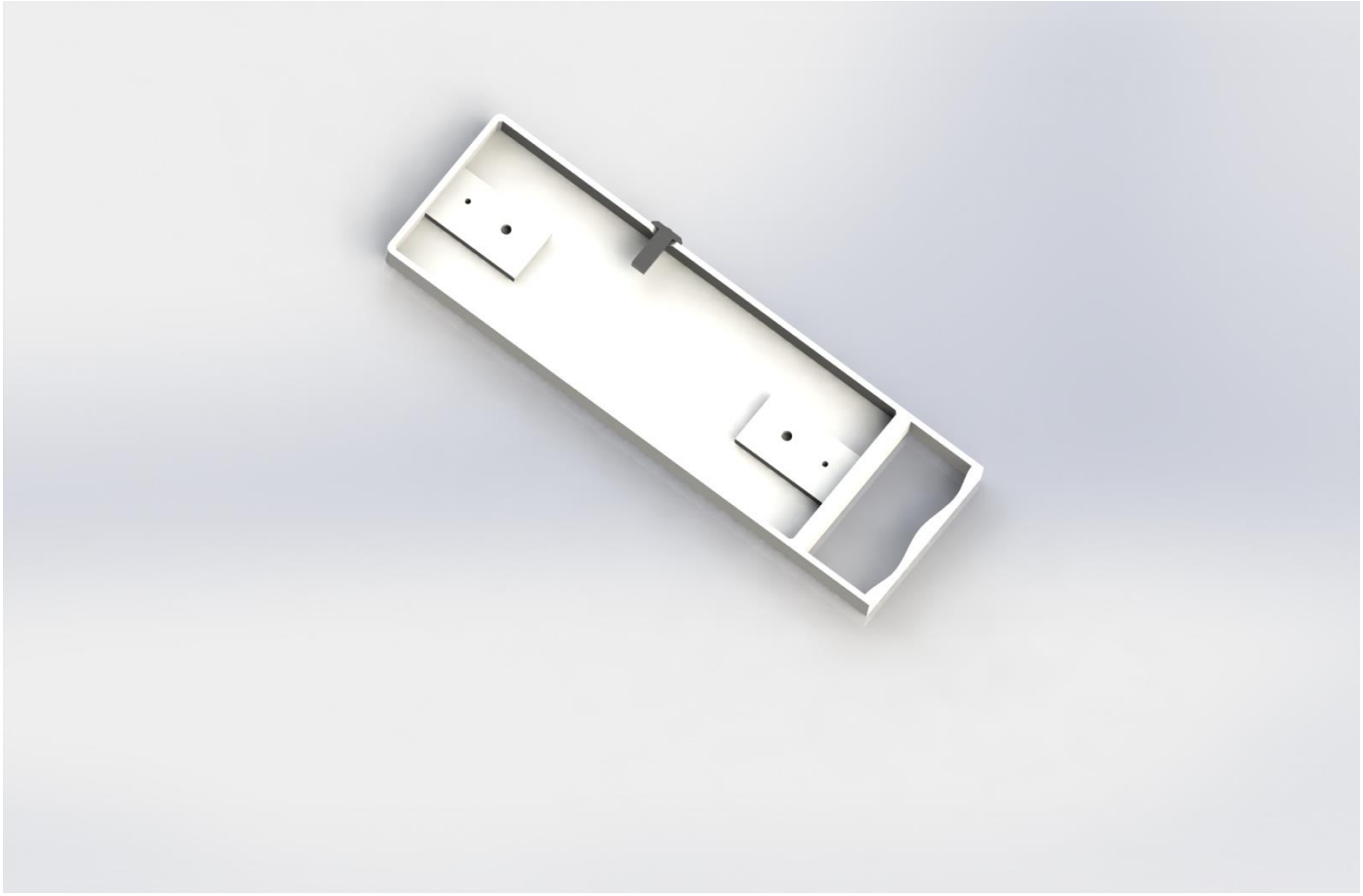


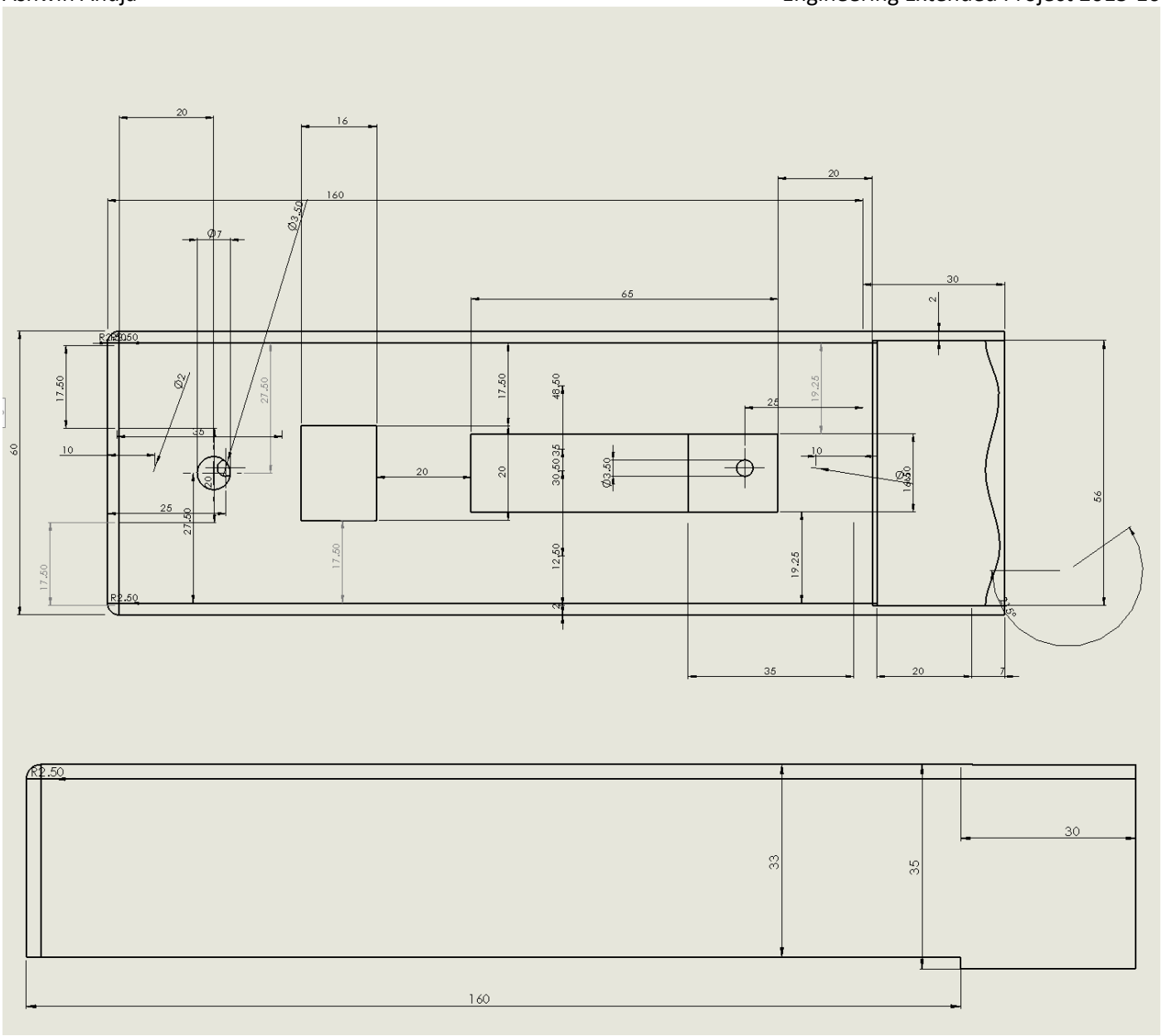


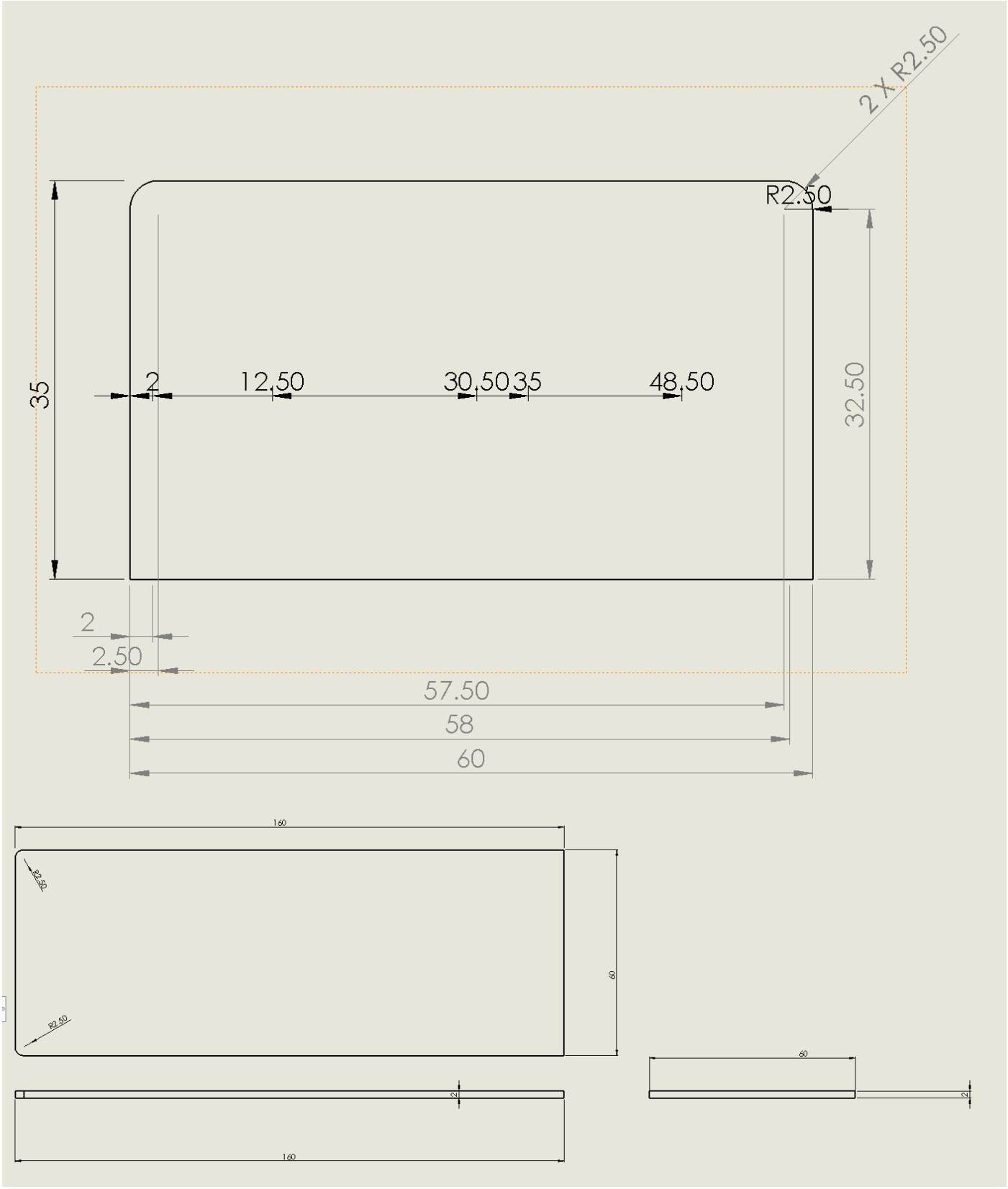









20
16
160
20
Ø7
Ø3.50
65
30
2
R2.50
17.50
Ø2
27.50
48.50
25
19.25
60
10
20
20
30.50
35
Ø3.50
10
Ø3.50
56
25
27.50
17.50
12.50
19.25
R2.50
2
3.50
35
20
7
R2.50
33
35
30
160



G Proof that Euler's Totient Function is multiplicative

To prove this, we must establish a one to one correspondence (each element maps directly to

$one\ of\ the\ other\ between\ \Phi(mn)\ and\ \Phi(\mathrm{m})(\mathrm{n}))$

$Thus\ we\ establish\ two\ sets$:

$$S1 = \{a: 1 \leq a \leq mn\ and\ \gcd(a, mn) = 1\} - this\ clearly\ has\ \phi(mn)\ number\ of\ elements$$

$$S2 = \{(b, c): 1 \leq b \leq m\ and\ \gcd(b, m) = 1\}$$

$$and\ 1 \leq c \leq n\ and\ \gcd(c, n) = 1\}$$

$We\ can\ show\ that\ this\ has\ length\ of\ \Phi(m)\Phi(n)\ by\ thinking\ how\ we\ could\ vary\ each\ of\ the$

$the\ first\ of\ the\ ordered\ pair\ by\ \Phi(m)\ and\ the\ second\ by\ \Phi(n), thus\ the\ two\ coordinates,$

$number\ of\ combanations\ possible\ = \Phi(m)\Phi(n)$

$By\ considering\ any\ value\ for\ m\ an\ n, we\ can\ show\ there\ is\ an\ association\ between\ them,$

$, for\ example\ m = 4\ and\ n = 5$

$\mathbf{1} \rightarrow (\mathbf{1}, \mathbf{1})$

$\mathbf{2} \rightarrow (\mathbf{3}, \mathbf{3})$

$\mathbf{7} \rightarrow (\mathbf{3}, \mathbf{2})$

$\mathbf{9} \rightarrow (\mathbf{1}, \mathbf{4})$

$\mathbf{11} \rightarrow (\mathbf{3}, \mathbf{1})$

$\mathbf{13} \rightarrow (\mathbf{1}, \mathbf{3})$

$\mathbf{17} \rightarrow (\mathbf{1}, \mathbf{2})$

$\mathbf{19} \rightarrow (\mathbf{3}, \mathbf{4})$

$In\ order\ to\ show\ that\ different\ numbers\ in\ S1\ get\ sent\ to\ different\ numbers\ in\ S2$:

$If\ this\ is\ true - Let\ a1\ and\ a2\ be\ distinct\ elements\ of\ S1\ and\ are\ mapped\ to\ the\ same\ pair\ in\ S2$

$$a1 \equiv a2\ mod(m)\ and\ a1 \equiv a2\ mod(n)$$

$$This\ implies: mn|(a1 - a2)\ since\ m\ and\ n\ are\ relatively\ prime$$

$$\rightarrow a1 \equiv a2\ mod\ (mn)\ which\ contradicts\ the\ original\ assumption$$

$$that\ a1\ and\ a2\ were\ different$$

$In\ order\ to\ show\ that\ for\ ever\ pair\ in\ S2, there\ is\ an\ association\ with\ an\ alement\ in\ S1$:

$For\ any\ example: a \equiv b\ mod(m)\ AND\ a \equiv c\ mod(n)$

$According\ to\ the\ Euclidean\ Algorithm, this\ is\ only\ always\ true\ when\ the\ GCD(m,n) = 1 \rightarrow$ $they\ are\ coprime.$

H Summary Marking Criteria

Key

§1.1 = Section 1.1 of Main Report
PPF = Project Proposal Form
A.B = Appendix B
AL = Activity Log

Mark Scheme	Extent of Success	LOCATION OF EVIDENCE	DISCUSSION OF EVIDENCE
AO1: Manage			
The proposed outcome is identified and developed with limited guidance, support and assistance from the tutor-assessor but then finalised and refined independently by the learner individually or within groups.	High	§2.0, §3, PPF	• During the Project Selection Phase, the outcome of the project is discussed, including what I aimed to get out of the project. • A number of possible ideas were considered and evaluated with minimal assistance from my tutor-assessor, who had not been selected at the time. • A decision matrix was used to select the most suitable project. • In the Project Proposal Form, a complete reasoning is given for my selection. • In Section 3, there are a list of objectives which I wanted to meet, which would maximize the usefulness of the Extended Project, as well as allowing me to meet my brief.
The proposed outcome is well defined and clearly focused.	High	§2, §3, PPF	• In Section 3, the aim of the project is stated outright, making use of the brief proposed in the Project Proposal Form. • The aim is broken down into a number of other

			objectives to be met.
The project plan is clear and concise, with clear and detailed objectives and rationale.	High	§3, §8.2, A.A, A.B, A.C	• Section 3 contains considerable discussion at to how the project plan was made, as well as looking at the actual project plan. • The Appendices contain Gantt Charts as well as PERT charts, which were used to visually show the plan, while there is also a comprehensive list of tasks.
All of the main tasks to be completed are provided in an appropriate order and described in detail, with an appropriate time span allocated for each task.	High	PPF, §3.2, A.A, A.B, A.C	• The Project Proposal Form contains a preliminary list of tasks to be completed with approximate time spans. • The Appendices contain all my Gantt Charts which specifically show the plans for individual tasks and how long they would take – with Appendix A breaking the tasks into a number of sections.
The learner shows a high level of organisational ability and time management skills when managing the project.	High	§3, §8.2, A.A, A.B, AL	• A complete journal was maintained throughout the project which showed all work completed – being written in every time any EP work was done. • The activity log is comprehensive, assessing the activities completed in that week as well as the problems encountered, and solutions found. • Gantt Charts were often updated, with 5 different versions, showing how I often reassessed my progress through the project and therefore replanned the future to ensure that I would succeed.
The learner maintains clear and detailed records of activities undertaken during the project, including problems encountered and steps taken to overcome them.	High	§3, §8.2, A.A, A.B, A.C, AL	• The weekly activity log has a number of set discussion points every week, including 'Activities Undertaken', 'Problems Encountered', 'Steps taken to overcome problems' and the effect on the plan. • I also maintained a complete journal which I wrote in any time I completed any EP work.
Progress is monitored against the original plan and adjustments made to the plan where necessary.	High	§8.2, A.B, AL	• On a weekly basis, the activity log comments on any issues as well as the effect on the plan. • A number of revisions were made to my Gantt Chart and plan during the year.
	High		
AO2: Use Resources			
A wide range of different types of possible materials and techniques have been thoroughly investigated.	High	§4.2, §5 (§5.2.2, §5.2.3, §5.3.2, §5.3.3), §6.1.4, §7.2	• Throughout the report, whenever there was a major decision, a number of alternatives were provided, with them being assessed for their pros and cons, before a justified decision was made. • This was especially true for Control System parts where at least two options were offered for all parts before a decision matrix (with a number of weighted criteria) were used to make a decision.

			• With particular reference to 'materials and techniques', a huge variety of materials were assessed in Section 5.3, before a group (the polymers were chosen). Then in section 7.2, manufacturing techniques were assessed in conjunction to the chosen geometry, before a decision on materials was also made (after looking at specific data of polymers)
Research sources are referenced appropriately and consistently and a bibliography is included, listing the sources in an appropriate and consistent format.	High	§9, throughout the report (footnotes)	• Throughout the report, the Harvard Referencing Style was used, with citations in the footnotes for any pieces of information which I had got from any other sources. • A full bibliography is included in Section 9, which includes all information which would be required to find the original source. • The figures are all labelled with the source underneath, with it also being referenced fully in the footnotes.
From the research carried out, appropriate information and resources have been selected for use in the project.	High	§4, §5, §9 (shows the sources used)	• Throughout the project, a number of useful sources were found and used, where they were appropriate. The effectiveness of the sources is shown by how often during the Development phase of the project, I referred back to these sources.
Information has been analysed and synthesised in reference to the project.	High	§4.2, §4.3, §5	• On a number of occasions, sources are distilled into quotations, which contain the salient information, which the report requires, while the reference allows the reader to go and look at the full source. • At the end of a discussion about a piece of research, the research is connected directly to the project and how it effects the artefacts design.
Clear, concise and detailed links have been established between the research carried out and the project.	High	§5 (especially §5.2.3), §6, §8	• Throughout the development phase of the project and later I often referred to the decisions or research that had been completed, talking about how the ideas had been influenced by this research.
A thorough understanding of the complexities of the resources and research required for the development and production of the artefact has been shown.	High	§3, §8, A.A, A.B, A.C, AL, PPF	• A wide variety of types of sources were used in an attempt to reduce the biases involved in the sources. • The activity log on a number of occasions includes reference to the process I followed to ensure that bias was found and eliminated from sources, by ensuring that the statements were well backed up with data.
AO3: Develop & Realise			

The supporting information that relates to the development process is structured and presented clearly.	High	§5, §6, §7	• A largely chronological structure was used, ensuring that the development follows on from information already stated. • I used multi-level section headings as well to ensure that it would be easy to find particular sections of research and development, especially through the use of the Table of Contents.
The information contained within it is consistently clear and relevant.	High	§6	• Wherever possible I used bullet points, or small sections to help to separate the information into digestable chunks. • Additionally, I made use of tables in order to more graphically represent the options, and the data, wherever possible. • Finally, whenever information was deemed to be unnecessary – though it may have been investigated simply for interest, this was not included in the final project.
Learners demonstrate a thorough understanding of the developmental process.	High	§6, AL, A.A	• The Activity Log list my activities week by week, showing how I made use of my time. • Additionally, there is a comprehensive Development section, split into Control System, Mechanical and Software Development, which largely track the development process chronologically.
There is clear evidence of development of ideas	High	§5, §6, §7	• During the Control System section, I talk in depth about the process of testing and improving the system, as well as the various decisions made on the way. Additionally, wherever more concerns are found, full research is conducted into solutions, with a number of possible solutions considered and one justified and chosen. • During the Mechanical Design Development, a number of ideas a proposed, with the justification of why they were designed the way they were, as well as the final design that was made being shown and the changes being discussed.
and that alternative ideas and approaches have been considered carefully and evaluated,	High	§5 (§5.2.2, §5.2.3, §5.3.3), §6.1.3, §6.1.4, §6.3, §7.2	• Throughout the process, wherever there were decisions where there were multiple possible solutions, a number were considered and the final solution was found through the evaluation of advantages and disadvantages of these solutions.
with a well-thought out and well argued explanation of the decisions taken, eg relating to choosing the most appropriate materials, processes, techniques, design.	High	§5 (§5.2.2, §5.2.3, §5.3.3), §6.1.3, §6.1.4, §6.3, §7.2	• For most major decisions, a Decision Matrix was used as the method of ensuring the decision was well considered, with a number of relevant weighted criteria. • In a number of places, this is not used however, because it is clear that one option is by far the best, or a large number of options are completely flawed.

There is clear evidence that the artefact has been carefully and perceptively refined during the developmental process, showing innovation	High	§5.2, §6.1 (§6.1.9 shows how the development refined the design), §6.2.6 (shows how idea have been taken to generate a final (better) design.	• A number of changes are completed along the Developmental Process to further refine the artefact, or fix any problems that arised during the testing. • Innovation was shown in a number of sections, including the design of an entirely new cipher, with a novel and effective idea of the changes in the key as you proceed through encryption.
Resources and skills are applied consistently successfully in creating the artefact	High	§5, §6, §7, AL, §8	• A vast variety of skills are used in order to fulfill the broad project, including electronics skills including the use of Eagle Cad. • Additionally, a number of different programming languages are used to show the breadth of skills that I have. • Finally, the mechanical section shows the use of complex CAD skills with SolidWorks including Simulation. • The processes required to ensure that I had the requisite skills in each section are discussed in the Activity Log.
The artefact is highly successful at fulfilling the original brief.	High	§8.1	• The success of the project is discussed in an evaluation discussion, where the artefact produced is compared against the brief, objectives that I had set as well as the Initial Specification that I created. • This clearly shows that the project was highly successful at meeting it's brief.
AO4: Review			
Overall the learner shows a high level of insight and self-awareness in evaluating the project	High	§8.1	• In Section 8, I conducted a thorough evaluation of the artefact, comparing it to the brief, objectives and the initial specification that I had created.
and the extent to which they have achieved their aims and met the original brief.	High	§8.1	• The artefact is assessed in tabular form against each of the aims and initial brief, with justification showing how it met (or failed to meet) the objective that I had set.
The learner is highly adept at assessing how well they managed at different stages.	High	§8.2	• In section 8, I assessed how the process had gone, individually looking at the phases of the project.
The learner explains and justifies ideas for what they could do	High	§8.1.3, §8.2.3	• During Section 8, I assess how I could have improved my artefact, both through changes in the Project Management (found through the evaluation) and to

differently next time.			improve the artefact itself.
They have drawn clear and perceptive conclusions about process of producing an artefact that could help them in future.	High	§8.2	• During Section 8 there is a complete discussion of a number of changes that could have been made to further enhance the process and allow me to produce an even better artefact, that would even more effectively solve the problem, that I had set out to.
The presentation is clearly and logically structured so that it is completely clear to the audience how the different parts link together	High	Presentation	• The presentation offers a brief discussion of the entire project, showing how the various parts of electronics, mechanics and electronics fit together to create a complete product. • The presentation imitates the structure of the report, with the exception of the lack of separated research and development sections, instead keeping these as a single part.
and the learner shows a high level of ability to convey the main ideas.	High	Presentation	• The most innovative parts of the project are particularly emphasized, whilst allowing the audience to find more information in the report.