WSC501 Project; Interim Report

ID Number:	B929062
Degree Programme:	Product Design Engineering
Project Title:	Release.

How to complete the report

(a) Select your module code in the header and title.

(b) Add your text in the places indicated with "your text here"; leave the headings and descriptions in place to remind your assessors what you have been told to do. Do not change text formatting.

(c) There is limit of 4 pages of text, including question text. The Gantt chart (or equivalent) is not included in the page count. The list of references and any images should be placed at the end of the document and are also not included in the page count.

(d) The report should be submitted electronically in a single file in Portable Document Format (.pdf) form via the **22WS50META - Individual Projects** page on Learn. The filename should be your ID number and interim (hyphenated) with the .pdf extension, for example **A123456-interim.pdf**. Your name will be associated with the file by Learn.

Notes

(a) The interim report contributes 70% towards the final Gateway 2 assessment mark, with the other 30% derived from the technical understanding and project ownership demonstrated at the Gateway 2 meeting. Therefore, the percentages allocated to each section in the interim report add to 70%.

(b) Check the project handbook on Learn for the Gateway 2 assessment rubric.

Project Description (5%): *A definitive description of your project written for a technically competent audience. This can be an expansion of the original description or entirely new, but it should describe the project that you are doing in one or two paragraphs, including expected outcomes.*

Design and prototype a robust, waterproof, smart trolley lock system to reduce theft and incorporate IoT into supermarket chain ecosystems. The consensus is that the world is moving towards a cashless society, so why do we still need to carry a £1 coin to unlock a £127 trolley? [1] This project aims to remove the need for a coin to unlock a trolley.
The prototype produced in part B was simply a proof of concept. Significant development work is needed to produce a robust functional prototype to meet a potential supermarket chain's needs. The problems identified from the prototype breakdown were:

- embedded hardware used is "hobbyist level"
- the current draw is too high for an embedded device,
- no attempt was made to ensure the enclosure was water resistant
- the final prototype did not fit in one enclosure a
- high cost per unit for electronics damage product feasibility.

The product must level up in terms of reliability and robustness to meet client needs. Therefore the expected outcomes for this project are: six months of battery life, water-resistant to ISO20653 with a rating of IP34, following the U.S. Open-Source Software Security Initiative Workshop request to program in a memory-safe language [2], produce an enclosure that fits all the components.

Project Context (5%): *Using language applicable to a wider non-technical audience, describe: the context of the project; why this work is relevant; and how it could be used.*

Supermarket customers increasingly don't carry cash, specially coins but if they don't carry coins, how will they unlock the trolley? The Release Project aims to solve this problem by unlocking the trolley using their phone or loyalty card. The project also has the potential to lower the barrier to entry for experimental technologies like Location Tracking using mesh BLE (low energy Bluetooth). The barrier to entry is lowered because it is easier to implement electronics when all the required hardware is already in place like a water-resistant enclosure and battery to draw power from.

Project Aim (2%): *A definitive project aim; it should state in one sentence what are you trying to do.*

The Release project aims to remove the need for a coin to unlock a supermarket trolley; instead, customers unlock the trolley wireless using contactless technology with their phone or loyalty card.

Project Objectives (6%): *What do you need to do to satisfy the aim? A list of objectives should be SMART (simple, measurable, actionable, relevant, and timely). Ask yourself the question can I decide if I have achieved this or not? Note that personal objectives are not relevant; "learning to use a soldering iron" is not a project objective. Hint: when writing they should ideally begin with a verb.*

1. Research to determine the security concerns with IoT products
2. Design and build the PCB using professional MCU's like ESP32 or STM32
3. Design and test mechanical enclosure to meet ISO/DIS 20653 waterproofing standards.
4. Develop the battery life of the product to achieve a minimum of 6 months of theoretical battery life
5. Program the MCU in (MSL), Memory Safe Languages, like Rust or C# due to cyber security risks around personal information leaks in loyally card database.
6. Prototype a proof of concept

Project Deliverables (5%): *What do you need to produce to meet the objectives; this can be processes, software, hardware, products, artefacts or even the exposition of a theory. This should not include the written deliverables for the project, those are understood.*

1. The product must have design changes that reduce security concerns
2. A full PCB must to be manufactured
3. A mechanical enclosure must meet IP rating specified
4. While testing the prototype, the device must achieve a minimum of 6 months of battery life
5. The final program running on the mcu must be compiled from a memory safe language like Rust.
6. The final prototype needs to be tangible and enclosed, unlike part B prototype where the enclosure had an open lid.

Background Research (11%): *Pick a minimum of four of your research sources that are most relevant to your project. Briefly describe their contents and assess their relevance and importance to your project. Make sure that you cite and/or quote your source material correctly. Sources relevance will be specific to discipline, but the expectation is to see peer reviewed content (i.e. journal papers, conference papers, books, etc.). An in-depth critical analysis is not required here (but will be in the final report).*

1 Cyber security

For the release project to be successfully implemented in a supermarket store, it would require access to the online loyalty card database to allow for customer verification. Turning the product into an IoT system, this elevates the risk of cyberattacks by intruders using the product as a gateway into the confidential database, where credit card numbers, zip codes and legal names are stored. Attackers could use this information to perform identity theft or simply sell the data for millions. Therefore research is needed to reduce the risk.
The **open-source Software Security Initiative workshop** recommends new IoT products to be developed in memory safe languages. [2] This virtual workshop took place on the 24-25th of August organised by three bodies: Nation Science Foundation, National Insitute for Standards and Technology, Office of Management and Budget (which directly serves the President of the United States). With the goal to join stakeholders from the open source community, private sector and academia with the US Government. A large focus of this workshop was Memory-Safe Programming Languages including how to incentives adoption.
The recommendations relevant to the Release project where:

- System level IoT devices should be programmed in memory safe languages specially critical infrastructure
- Universities need to play a part in refocusing programming curriculum to use memory safe languages
- Software developers who are not security specialists need to be convinced about the importance of memory safe language adoption
- "Rust programming language, despite its initial learning curve, is particularly well-suited for safe systems-level development." [2, p. 6]

The workshop's recommendations are well matched for this project, the author is not a security specialist but plays the role of a software developer, the product being created is a IoT device which requires embedded IoT programming and lastly the project is being created in a University environment.
Are there other companies or papers recommending MSL? In 2019 Microsoft concluding from their internal tests that 70% of their vulnerabilities are due to memory safety issues.
Due to these reasons, the Release Project is a perfect candidate to follow the guidelines and program in Rust for embedded systems.
KEY WORDS: MSL means memory safe languages like Rust, IoT is Internet of Things

2 Enclosure Design

Supermarket trolleys are often left outside in the unfavorable conditions like heavy rain or intense sun with zero regard to keeping it sheltered. As the Release product needs to survive in its environment, how do we ensure the product will function even in a storm? The problem can be broken down into 2 questions.

- How waterproof do we need the design?
- How to design and manufacture water resistant electronic enclosures?

How waterproof do we need the design? Waterproof is an outdated term. Manufactures starting steered away from the word "waterproof", in favour of water resistant when ISO 2281 was introduced in 1990, prohibiting the term waterproof to be used in watch making. The consumer electronic market followed suit, not long after, creating a standard called ISO 20653.
ISO 20653:2013 [3], is the current standard used to classify how protected a product is against foreign objects, water and access. If the reader has ever heard advertising for a consumer device's stating a rating of IP68, this rating is derived from this ISO standard.
What does IP68 mean? Using [Fig. 4], the first number is in protection against dust and the second number is protection against water. IP68 means the product is dust-tight and water resistant against continuous submersion. Note: the product is also rated for all e.g. IP57.
Now that the reader understands the industry standard for water resistance, the author can go back to the initial question.
What IP rating does the release product require? If the product will sit outside in heavy rainfall, looking at second element, [Tab. 1], the minimum requirement is 'splash water' number 4, which includes inferior ratings like water driping at 15° inclination and water spray. For the first element it is less certain, due to the locking and unlocking mechanism, it will be unfeasible to get a 'dust-tight' enclosure without interfering with the servo. Thus a conservative dust IP rating of 3 is set, which requires an opening of less than 2.5 mm. To conclude, the release product shall pass a ISO20653 water resistance test of IP34.
How to design and manufacture water resistant electronic enclosures? Looking at industry solution's for electronic enclosures that are IP certified. Polycase is a good sample for industry solutions, the company has been designing electronic enclosures for 30+ years. Looking at their existing products two design choices keep appearing for the waterproofing problem, gasket and O rings. These solutions are found irrespective of enclosure material selection, both metal CNC or injection moulded plastic enclosures find these design choices. Polycase's family of plastic enclosures are the most relevant for the Release Project. The majority of these designs are secured by four screws threaded into brass inserts instead of directly into the plastic enclosure, as seen on [4].

Current Project State (20%): *In the context of the timeline given in the last section, explain clearly where you are in your project and what you have done. This is a brief progress report and should be written in a less formal and more chronological fashion compared to the final report. You should include preliminary designs and results.*

On the 25th of October I attended Hardware Pioneers, UK's largest convention dedicated to cutting-edge technologies for smart and connected devices. Here I created contacts with industry suppliers, attended workshops and asked experts about the feasibility of the Release Project. [Fig. 9]
On the 10th of November I booked the electronics workshop, T101, to analyse and disassemble part B prototype. [Fig. 8] Reviewing the product I created two years ago, with my current knowledge, gave me a different perspective. The main findings from this product tear-down were:

- The battery life calculated of the prototype using a fresh 9V battery was 7h 42mins . Multiple IoT devices have months if not years of battery life, 7h is simply not good enough. The source for the poor battery life was the servo stays on when the mechanism is open, causing excessive current draw (175mA). Average current draw on standby was also too high at at 65mA. This was due to the Arduino Uno not being designed for low power consumption.
- No attempt was made to ensure enclosure was water resistant, like gaskets or o-rings
- By using an Arduino Uno and a shield it caused an extremely high cost per unit making it infeasible in a competitive industry.

To produce the desired outcome in the project. I must become proficient at both embedded programming and the rust language. I have done this by following two "books" for beginner embedded rust, one for STM32 Discovery board and one for esp32c3 rust board. [5, 6] The reason for following two books simultaneously was to mitigate the risk 'stm32 vs esp32', as illustrated in [Fig. 5].
On the 14th of November, after following these books for 3 months, it become clear that STM32 was easier for beginners due to many example programs and vast documentation found online. I had managed to print hello world and a blink sketch from the STM32 all written in rust, more importantly I understood what I was writing while the esp32 was running into bugs constantly. Using this new knowledge, I created a documentation guide to facilitate the workflow of future projects like writing a servo or RFID sensor program.
On the 23rd of November, I successfully programmed the 9g servo to actuate 180° with the STM32 and uploaded the program to Github to ensure it is not lost.
On the 26th of November, I successfully programmed the RC522 sensor to read the RFID card values shown in [Fig. 6 , 7]

On the 30th of November I picked up components for the hardware development phase which is due to start on the 15th of December.
Where to go now? Once this report is complete, I can focus on merging the RFID and servo code. Then perform tests and experiment with sleep functions to reduce current draw. Finally I will start PCB hardware development like the gantt chart recommends. [Fig. 1]

Risk Assessment of Project Delivery (6%): *List the risks associated with the successful completion of your project (note these are separate from a health and safety risk assessment for specific activities). For each, numerically assess their severity and likely effects on project delivery. Give suggestions for their mitigation.*

Look at [Fig .5] for the breakdown of the Risk Assessment

Ethical and Environmental Issues (5%): *Discuss the ethical and environmental issues associated with the project. This could be those directly associated with the project, or those linked with a future application of the work. Also consider ethical behaviour in conducting project work. You are encouraged to use the 'Engineering Council Statement of Ethical Principles' as a guide (https://www.engc.org.uk/ethics)*

Starting with **ethical issues**, the Release product works in conjunction with customer loyally cards, which by themselves have caused a range of controversy. The problem is not the card, it is the decisions and information that can be derived by the retailers about the consumer using their purchasing data. An example of an ethical issue retailers could pursue in search of maximising profits; is excluding less profitable customers from schemes by postcodes. An illegal example could be identifying a change in purchase behavior, concluding the customer is pregnant, discriminate against the customer's protected characteristic in search for higher profits. [7] On the contrary, some argue that the customer loyally cards, are in the customers best interest as they allow retailers to target offers about how best to satisfy the consumer. The ethical risks of customer loyally cards are definitely apparent but investigating these issues is outside the scope of the Release project. In any case, the ethical issues should definitely be understood and declared following the Honesty and integrity section in the **Engineering Council** Statement of Ethical Principles.
Lastly for **environmental issues**, "why fix it, if it ain't broken". The biggest worry in the Release product, is replace a product that works satisfactory for little net gain.
When releasing a new product, there is always a worry that it is contributing to the rising Consumer Society. Therefore when releasing a new product, the team must ensure that the product brings more positives than the negatives caused by the life cycle of the product.
An LCA could be done to better understand the damages caused by the launch of the Release product to a major supermarket retailer in the UK. Ensuring that the Release project can simply slip on to existing supermarket trolleys instead of forcing retailers to buy trolleys would lower the overall impact. Another suggestion for future application is focusing on the end of life of the product. What happens when the batteries capacity decreases to the point of requiring replacement?

Project Timeline (5%): *In a style of your choice (i.e. a Gantt chart; V-diagram; etc.) highlight the activities (referring to the objectives and deliverables) that you have identified as necessary for the successful completion of your project and how these interact. This should be on a page by itself and does not count in the document page count. This page can be included in landscape format if it helps.*

All the project management documents and techniques used on the Release project are inside an application called Notion. Here I customised a page to include a log book, a database with tasks and deadlines that the project must respect to succeed. Using this application instead of a excel document allows me to access the timeline and make quick alterations from any device, phone or laptop.
The page has three viewing options for the database: timeline view, task view and status view.
Timeline view is where the Gantt chart is located. It uses a database property called Timeline, when you create a new task, add the timeline property with the start and end date of the section. Then the program automatically populates the timeline view with the database to produce the Gantt chart. [Fig. 1]
Task view is used to as a to-do list and to quickly jot tasks/ideas from a phone. Each task can be as simple or complex as you like. [Fig. 2]
Status view is a kanban style project management technique. The tasks are separated by "Not started", "In progress" and "Complete". Used infrequently but useful when needing to track the task progression and completion. [Fig. 3]

References

[1] "Supermarket trolley 240l," 2022. [Online]. Available: https://www.shopequip.co.uk/trolleys+shopping+baskets/supermarket+trolley+240+litre+traditional-C55-I1202.html

[2] A. D. Keromytis, "Recommendations from the workshop on open-source software security initiative," 09 2022. [Online]. Available: https://cpb-us-w2.wpmucdn.com/sites.gatech.edu/dist/a/2878/files/2022/10/OSSI-Final-Report.pdfpage=5

[3] *Road vehicles — Degrees of protection (IP code) — Protection of electrical equipment against foreign objects, water and access, ISO 20653:2013*, 2013.

[4] Polycase2022. Wc-20 outdoor enclosure with clear cover. [Online]. Available: https://www.polycase.com/wc-20

[5] E. Systems, "Embedded rust on espressif," https://esp-rs.github.io/book/, 2021.

[6] R. on Embedded Devices Working Group, "The embedded rust book," https://docs.rust-embedded.org/book/, 2020.

[7] A. Smith and L. Sparks, *"Making Tracks: Loyalty Cards As Consumer Surveillance", in E - European Advances in Consumer Research Volume 6*. eds. Darach Turley and Stephen Brown, Provo, UT : Association for Consumer Research, 2003. [Online]. Available: https://www.acrwebsite.org/volumes/11294/volumes/e06/E-06

Appendices

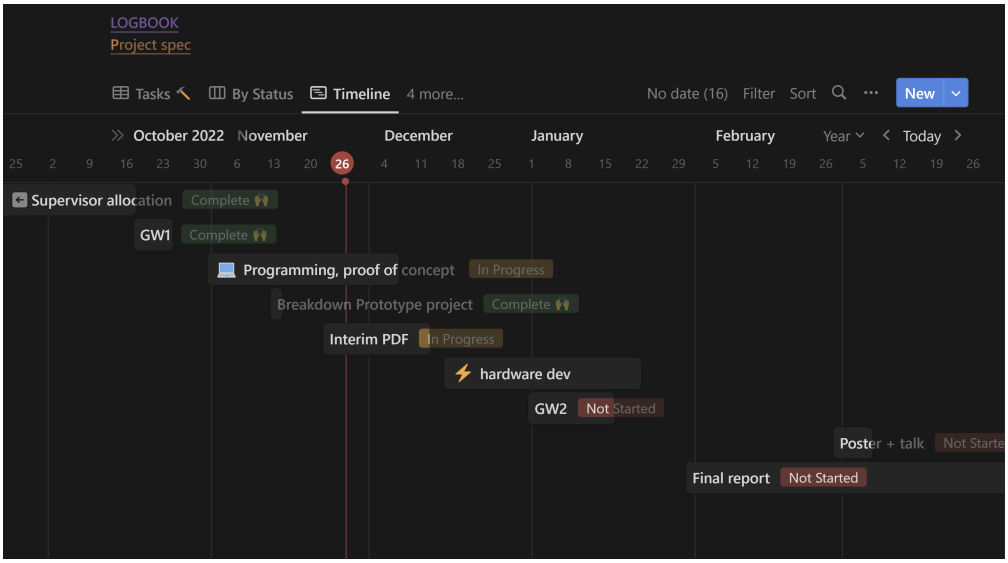

Figure 1: Notion status view

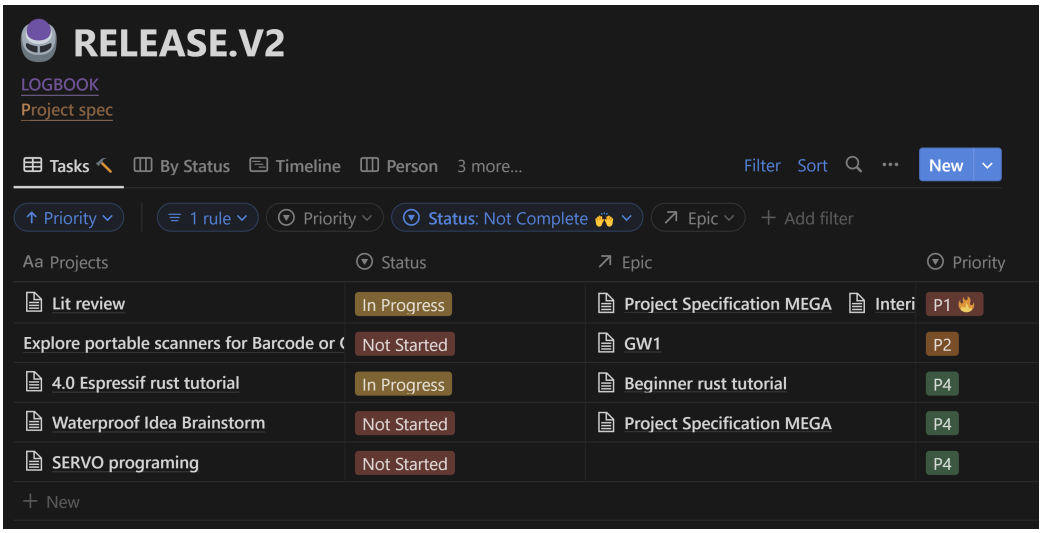

Figure 2: Notion Task View

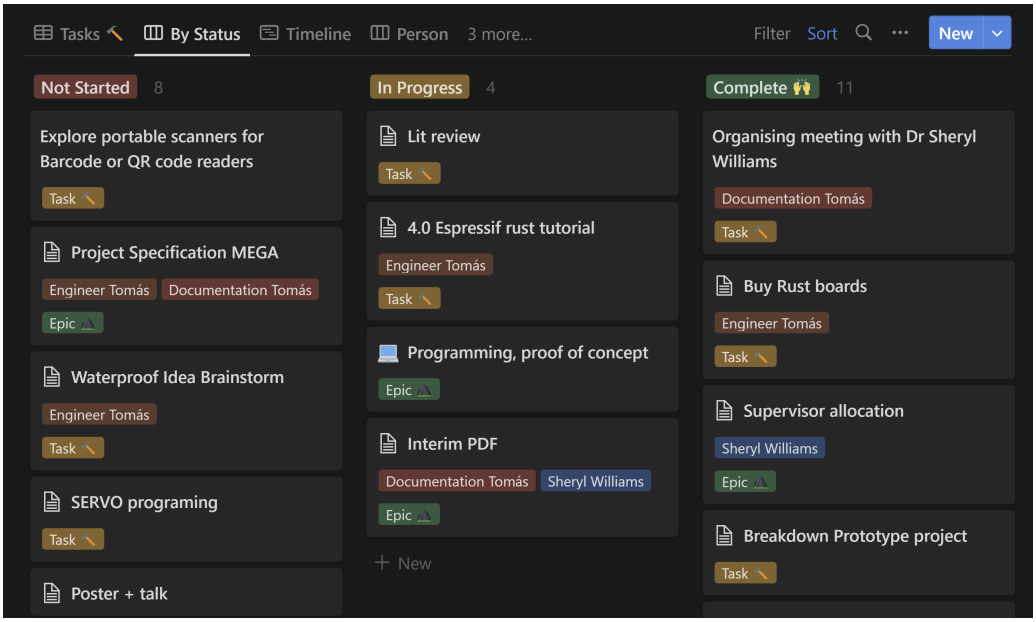

Figure 3: Notion kanban view

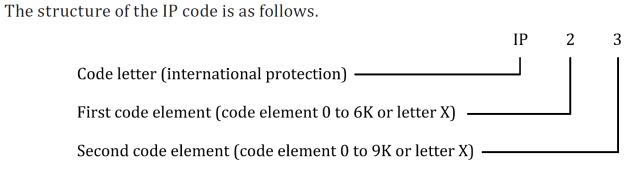

Figure 4: IP structure

	Risk Pre mitigation		Risk Post mitigation	
Risk Number	**Description of risk**	**Overall risk**	**Mitigation Strategy**	**Overall risk**
Name+3 digit code	*What is the risk? what negative effect will it have on the*	1-9	*What can will be done to reasonably reduce the risk?*	1-9
CW_Deadlines-001	Course work deadlines from other modules mean we have little time to work on the project during week 9-10	6	Include other module deadlines on gantt chart and rearrange subtask delivery according to these new constraints	2
OrderingDelays-002	Parts ordered for prototyping might take longer than expected	6	Specify and order parts from external partners early. Specify order by dates on gantt chart. Use reputable suppliers which provide delivery	2
MCU_comparison-003	Embedded development platform STM32, might be too difficult causing the product to not function	9	Have another embedded platform ready, incase STM32 environent provides to programatic. Embedded platform chosen incase STM32	3
PCB_deisgn-004	PCB manufactured might not work first time, causing huge delays, prototype wont function until the PCB is	9	Follow a reliable guide on designing a PCB for STM32, order PCB early in the project to allow for V2 of the PCB.	6
IP_certification_005	To aquire IP rating certifications, a company would be required to test the enclosure. This price would be outside the project's budget and the time taken to perform these tests in a lab would be past the deadlines	4	Instead I will performing these tests using university facilties while following BS ISO 20653:2013 recommendations. This will reduce the cost and development cycle	2

Figure 5: Risks

```
ERROR error Timeout
└─ stm32f3_mfrc522::print_err @ src/main.rs:144
ERROR WUPA error
└─ stm32f3_mfrc522::__cortex_m_rt_main @ src/main.rs:83
ERROR error Timeout
└─ stm32f3_mfrc522::print_err @ src/main.rs:144
INFO  new card detected
└─ stm32f3_mfrc522::__cortex_m_rt_main @ src/main.rs:65
INFO  card uid [74, 152, 138, 25]
└─ stm32f3_mfrc522::__cortex_m_rt_main @ src/main.rs:68
INFO  read [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
└─ stm32f3_mfrc522::handle_card @ src/main.rs:114
ERROR WUPA error
└─ stm32f3_mfrc522::__cortex_m_rt_main @ src/main.rs:83
ERROR error Timeout
```

Figure 6: RFID proof screenshot

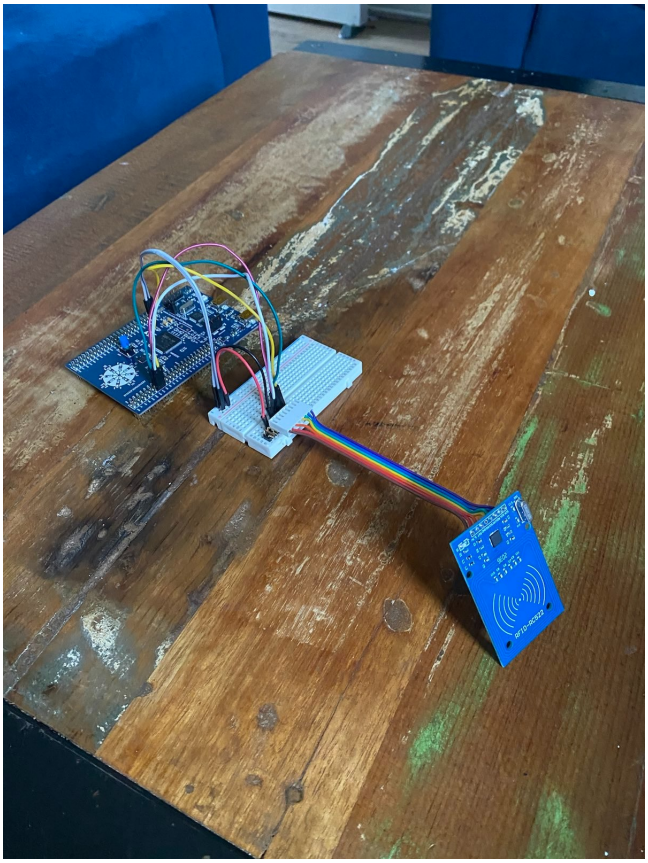

Figure 7: RFID with Discovery Board

Figure 8: Product tear-down

Figure 9: Hardware Pioneers Badge

Element	**IP**	**Meaning for protection of electronic equipment**
First element	0	no protection
	1	with diameter $\geq$ 50 mm
	2	with diameter $\geq$ 12,5 mm
	3	with diameter $\geq$ 2,5 mm
	4	with diameter $\geq$ 1,0 mm
	5	dust protected
Second element	0	no protection
	1	vertical water drips
	2	water drips (15° inclination)
	3	water spray
	4	splash water
	5	high-velocity water
	6	strong high-velocity water

Table 1: 20653:2013 details