LAM-CHARITY RUN

Feil Charel

JANUARY 28, 2023

BTS-IOT
Lycée des Arts et Métiers
Project CREPO3 2022/2023
1, rue de l'école,
Bissen L-7768
CREPO3
Supervisors:
THIERRY BERNARD
JEAN DAUBENFELD
ROBERT FISCH
GILLES GARDULA
GAETAN HOLDERBEKE
GUY WEILER

Table of contents

1 Description of the task

The task given for this project was to produce a working prototype of an RFID scanning system for a charity run which counts the laps of individual runners. The device that scans the RFID tags was our goal to produce. The device needs to be connected to the internet or a network and send the information read from the RFID tag to the server. The server counts the laps and holds all the information of the individual runners. We did not have to make an interface where people might see who scanned and the information of the individual.

The RFID tags are provided and look like this: BILD

RFID stands for **R**adio-**F**requency **ID**entification and uses electromagnetic fields to read the data from tags. The tag is a small object which reacts to electromagnetic interrogation pulses, it then transmits its data to the interrogating receiver. Our tags are so called passive tags. These only get powered by the electromagnetic field our scanner produces and doesn't transmit data when being far away from a power source.

The scanner is provided and looks like this: BILD

The scanner operates with 12V and sends the scanned data via the Wiegand protocol. The Wiegand interface is a wiring standard for card readers, which has been emancipated in the 1980s. This standard has become so popular because of the Wiegand effect, which was discovered by John R. Wiegand. The Wiegand data is being transmitted via 2 wires, which are called Data0 and Data1, both are standard high and are low when data is being sent. Data0 is the line for the 0s and Data1 is the line for the 1s.

The standard protocol has one parity bit, 8 bits of facility code, 16 bits of ID code and a trailing parity bit. This comes to a total of 26 bits. We are however operating our scanners on the W34 standard.

The microcontroller we used is also POE compatible. POE stands for Power over Ethernet and is very simple, it sends power over the ethernet cable. This can range from 44V-57V. This means that only one cable is needed to provide the device with power and network connection.

2 Introduction

2.1 Step by step description

How my scanner works:

My scanner firstly checks if it is connected via Ethernet or if a specific network had already been saved in the filesystem. If this is not the case an Access Point (AP) is being opened. (AP = Wifi network coming from the scanner) On this network, you can navigate to the URL http://192.168.4.1/ where you can configure the network, this means entering the SSID (name of the network) and its password. BILD.

The user then gets asked to connect with his end device to the correct network where he can access another webpage which shows the last scanned person, the battery percentage and in which mode the scanner is in. BILD

Then the scanner is connected and is waiting for an input in form of RFID tags. The information contained on these tags is being sent to the server via an API, those handle the data processing, so the lap count, the money earned etc. The scanner can operate in 2 modes: battery powered mode and POE (Power over Ethernet) powered mode.

When connected to the school WiFi, my scanner has gotten the IP 192.168.129.75, this means that the consulting page can be accessed by navigating to this address on the browser: http://192.168.129.75/ . When connected to the Ethernet, the scanner has gotten the IP 192.168.129.223. The consulting website is therefore available on this link: http://192.168.129.223/ .

Note: these links only work when the scanner and your end device are connected to the BTS-HUB network and when the scanner receives the same IP address.

I tried to change this to simply use a hostname like charityRun to get a link like http://charityRun.local/ but I could not implement this anymore.

AP:192.168.4.1

Webserver: http://192.168.129.75/ , http://192.168.129.223/

3 Main Part

3.1 Project progress

3.1.1 Initial thoughts

The first thing I did was to research how professional solutions for marathons or other charity runs would work. I thought this would be a good starting point because those applications are very similar to our use case in school. Most of them used a large long range RFID scanner and tags which are attached to the runner, on the shoe or the dossard for example. These scanners were either stretched on the ground to receive every runners id while passing or a very high power long range unit which would sit on multiple points of the run and know the positions of the individual runners at practically all times. Afterwards we were told that we needed to use a specific RFID module and tags, which were given to us by our teachers.

3.1.2 First tests

One of the first things I tested was the functionality of the RFID scanner, because it operates on the Wiegand protocol, and I haven't worked with that before. That's why I built a simple reading program, which already sent the data via MQTT to my PC. This allowed me to easily see that the Scanner was working as well as the network connectivity. I found out that the scanner was working good when holding the tag very close and a little longer to the scanner. However, when the tag is swiped a little faster and/or further away, the scanner showed some reliability issues. This meant that the tag wasn't scanned properly and was useless to send to the server.
This had been tested by some of our class including myself by hanging the RFID scanner on the wall of the hallway and running past it. This was done to test if the scanner would recognize people/kids swiping swiftly past the scanner. Furthermore this was also done to find out the optimal position of the RFID tag to give a recommendation for the users. Which you can see here:

VID_20220923_110032.mp4

VID_20220923_110118.mp4

3.1.2.1 Test results

Through this testing we found that the scanner was sometimes very reliable and other times very unreliable while running past it, some of it was due to how the scanner was mounted to the wall and to how the RFID tags were fixed to the testing bodies. However, when the person would slow down a little while running past it and making sure the tag would hit the scanner, then it would scan very reliably. This is an obvious inconvenience for more experienced runners, but there is no obvious possible fix in sight.

3.1.3 Further testing

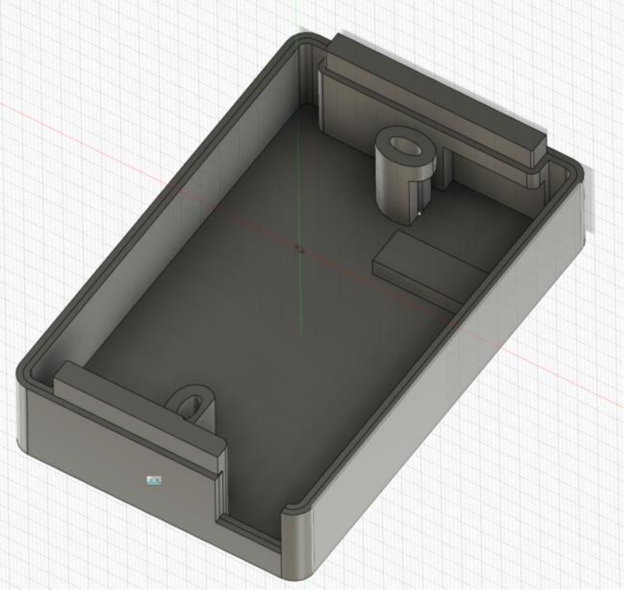
The next thing which was tested was to check if the scanner module would work with lower voltages than advertised. On the back of the module was written that it should be operated with 12V, but through testing we found out that it would work with very similar/ the same results with 5V instead. This meant that we didn't need to boost the voltage when using POE. However, to prevent the ESP32 to take harm from the high voltages, all of us needed to implement a level converter which changes the voltage of the signal, in this case from 5V to 3.3V and back.

3.1.4 *Managing the project*

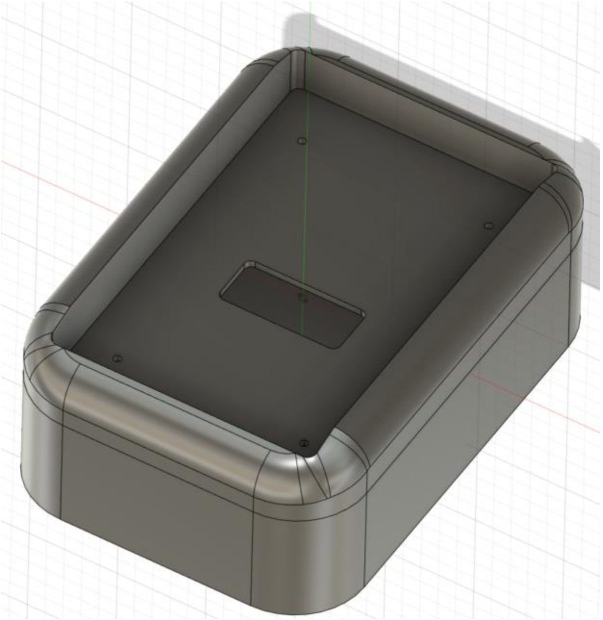
This was a turning point for me, because after testing the basic functionality of the scanner, I concentrated more and more on the design of the case. I have made several iterations of the case because some ideas turned out to be not very functional and/or aesthetically pleasing. Furthermore, during the time of our work, the requirements changed on how the scanner and case would be functioning. This meant that our scanner now needed to include a battery and be relatively water resistant due to the possibility of the device being outside and exposed to the elements. To make the case water resistant we were offered rubber gaskets to seal our boxes. Another added requirement was that the device should be able to be operated in a handheld mode, that meant that a battery needed to be added, which would increase the thickness of the overall casing, at least in my case. I dedicated a lot of time to the design of the case, perhaps too much, but I was confident with my previous software tests that the firmware and the electronic circuit would not take up as much time.

3.1.5 *Design process*

These added requirements threw me off somewhat at that moment, but I searched for solutions and found one with which I am pretty happy. My first idea was to utilize the indentations of the scanning module to hold it somewhat in place and later screw it in place to secure it in further. This meant that I had the microcontroller on one side and the rest of the circuit on the other, which gave me lots of space to work with. When the requirement came to add a battery, that wasn’t much of a problem, this only meant that I had a little less space to work with for the circuit. This and the rubber gasket made me abandon my first idea.

First idea: 

After that I tried to design multi-piece design with a similar way to attach the scanner module as Yves Fischer, but I soon realized that I couldn’t bring what I had in mind to reality and therefore abandoned that idea too.

Second idea: 

3.1.6 Final design idea

Lastly, I sat down and redesigned the whole case with inspiration of my first idea but starting with the battery, gasket and space for the circuit already in mind. This then took 16 versions until I landed on the final one, which I am fairly satisfied with. In this one I took in consideration the indentations of the scanner to hold it in place without screws, I planned in space for switches, the opening for the Ethernet port is angled on the inside to allow more easily to insert or remove the microcontroller, I also added space to allow for the circuit to be mounted on a strip board, I planned a way to put the battery in place without screws and an opening for the micro-USB port. The last addition was to allow for easy testing and charging of the battery without a POE injector. I later realized that this is a huge security risk as anyone could flash other software on the device and that water and dust could enter very easily. This is why I added a plug which can be squeezed in the opening. It is very hard to remove when applied, so that I couldn't remove it without pliers, this is obviously not an optimal solution, to make it perfect I would add a locking slider which could only be unlocked by authorized personnel. You can see the design here.

3.1.7 Code and electronical circuit

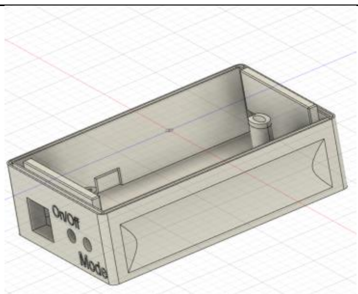
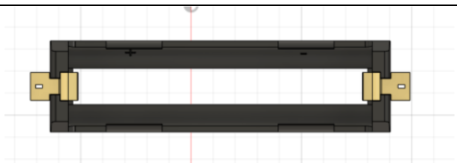
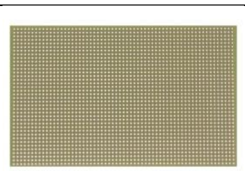
When the case was 90% done, I started to concentrate more and more on the software and electronical circuit. Most of the software I wrote for the initial testing with MQTT could be reused for the final product, this meant that I could concentrate on making it work good and reliably and I could concentrate on the electronic circuit.

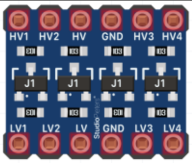
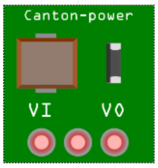
The last part didn't seem very complicated to me because it didn't consist of many components. The main components are the microcontroller, the scanner and the battery I thought. To be sure that the microcontroller wouldn't be harmed with high voltage, I wanted to add a level converter to the circuit which would change the voltage from the signal. However, while testing I noticed that the unit, I had for testing was probably faulty because no matter what direction, it always amplified the input signal instead of dropping it in one direction. This concerned me as for the longevity of the product, but I took a new unit and this worked fine.

3.1.8 Result

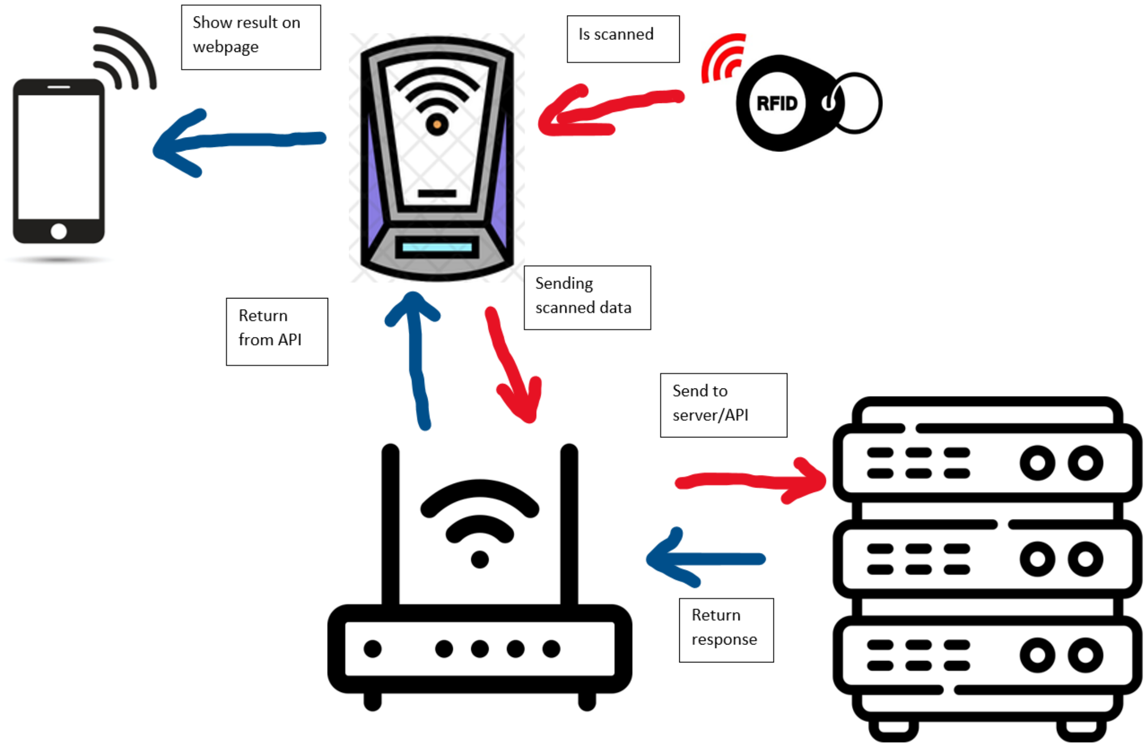
Then I found out that the microcontroller wouldn't output 5V when powered by the battery, so I needed to add a step-up converter which boosts the voltage from 3.3V to 5V for the RFID scanner. The circuit is made up of connectors in order to be able to disconnect and replace every component if needed. However, I soldered 2 cables without a connector, which resulted in it being not as replaceable as I wanted.

3.2 List of material/ electronics

Material	
	Case (3d Printed)
	Battery Holder
	Rubber seal
	Strip board
Electronics	
	Olimex Poe-ISO
	RFID Module

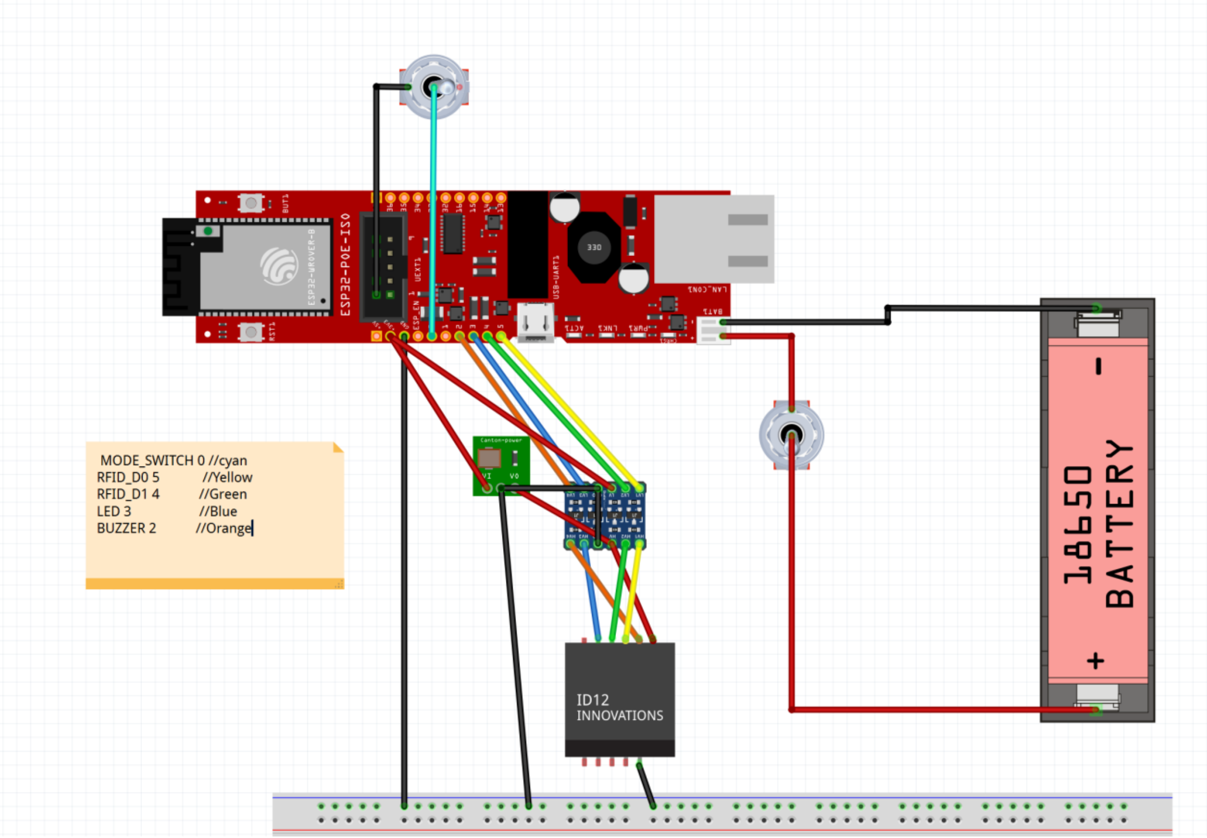
	Level converter
	Step-up converter
	18650 Battery
	RFID-Tags

3.3 Logical schema of my infrastructure

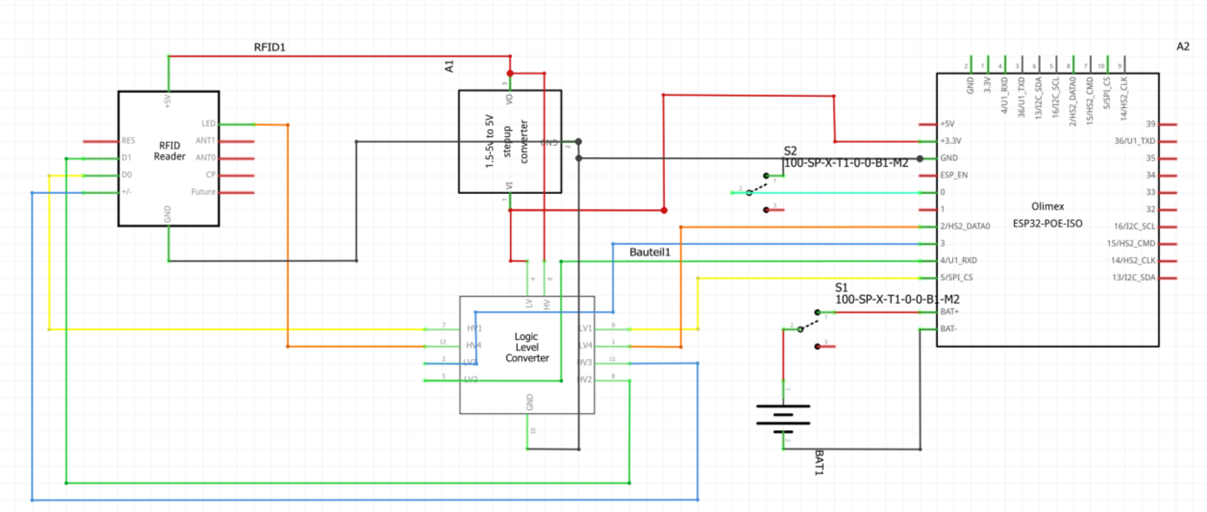


3.4 Electronic circuit schema

Electronics connected on a breadboard for visualisation:

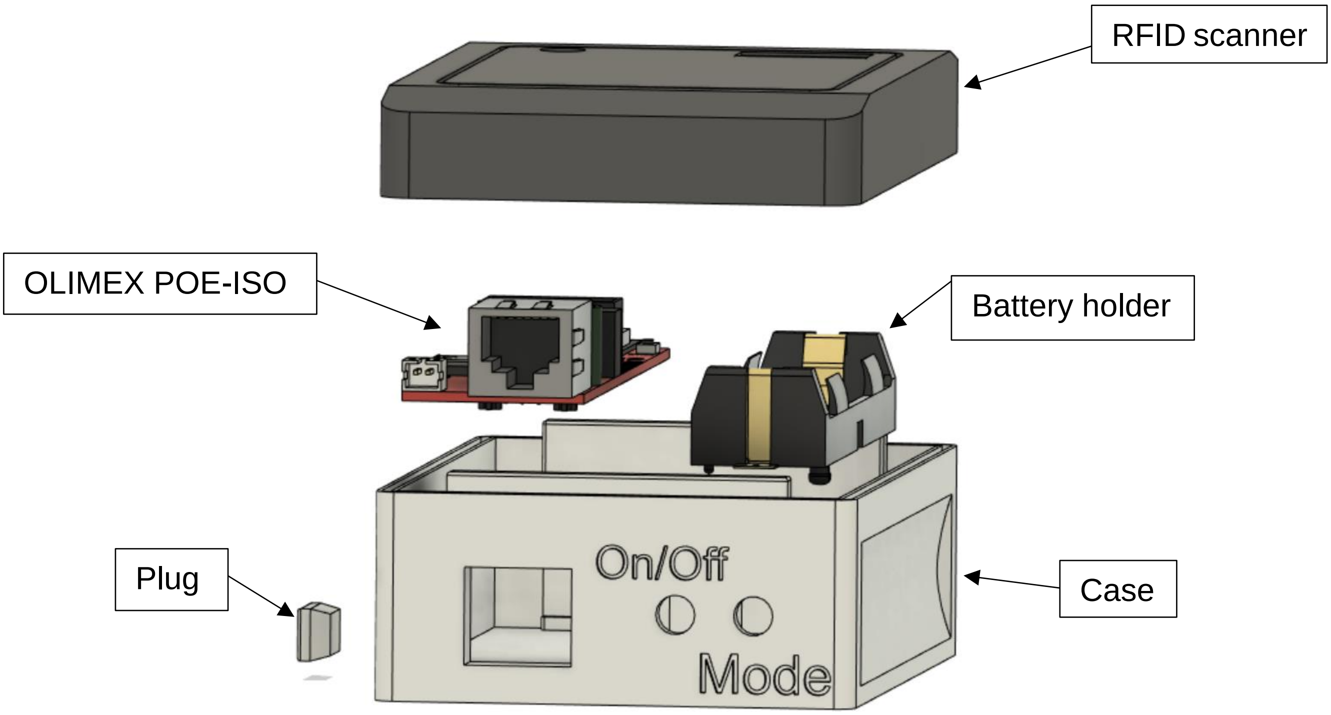


Correct schematic of all connected parts:



Pictures of the circuit can be found [here]

3.5 Pictures of the final product



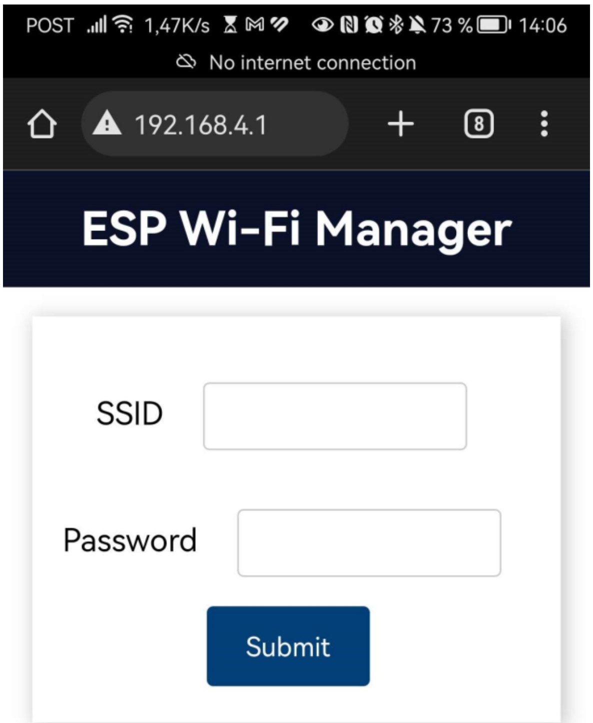
OFF

ON

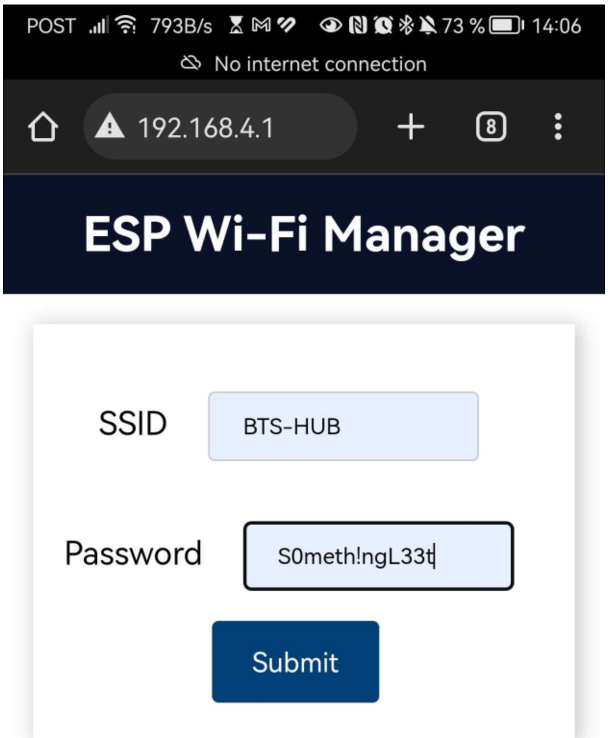
Bottom View

Left View

Back View



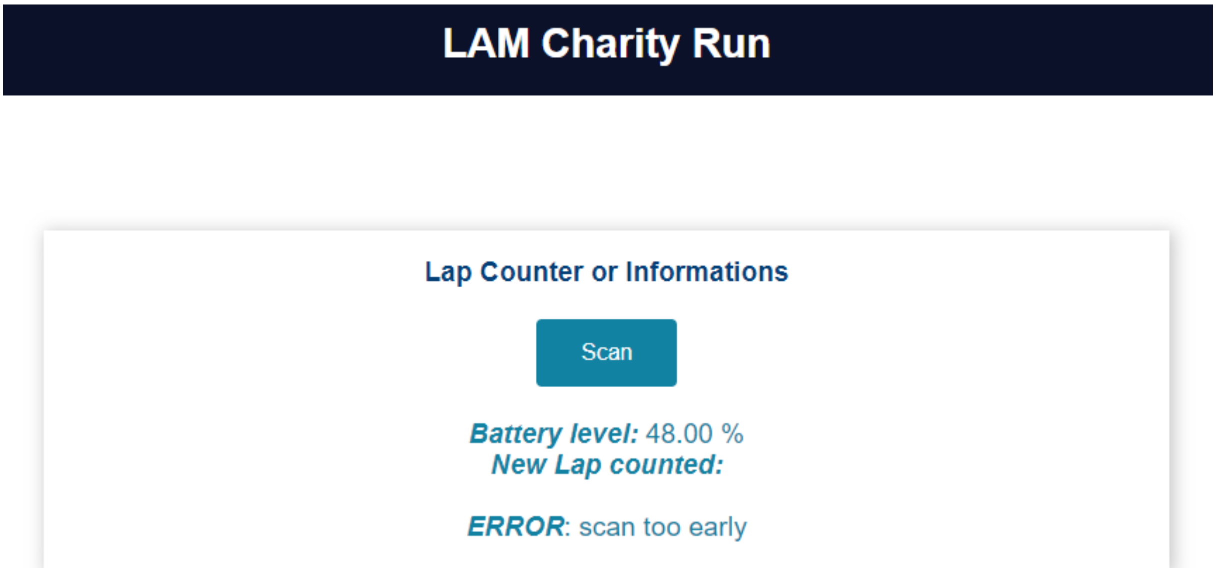
Wifi Configurator Page on the Access Point



Filled in with BTS Wifi

Webpage with the information about the scanned runner:

When an error message occurs:

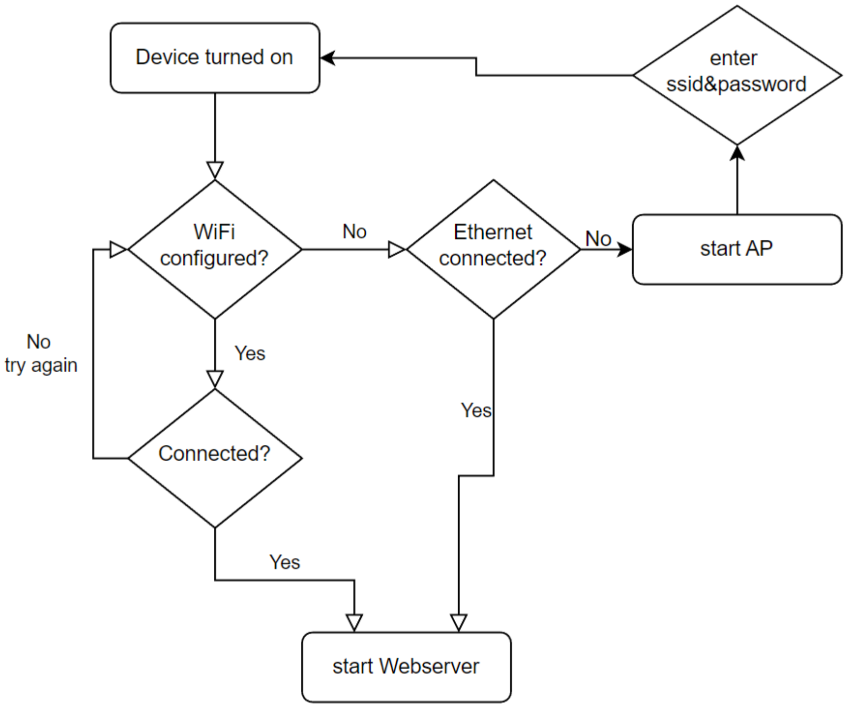


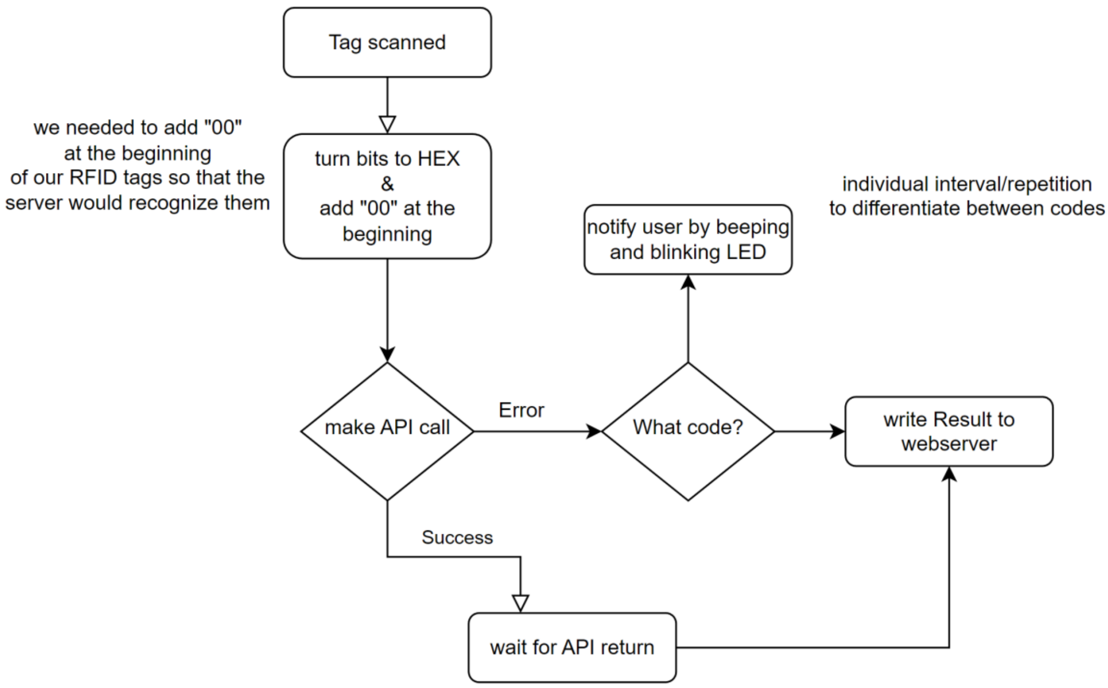
3.6 Testing table

What has been tested?	Comparison	Comments, solutions
Running past the scanner	Works less reliably than walking past it	Slowing down while scanning/passing the scanner
Sending Wiegand read data via MQTT	Works very good	Simple solution for testing and good base for final code
Testing Level converter with function generator and oscilloscope	Did not work as intended	Boosted the signal either way, probably faulty unit, second one works fine
Nearly every iteration of the case, the fit, what could be improved…	Turned out great	Would probably have been better to print less iterations and design a few steps ahead

3.7 Description of firmware

Every major part of the code is commented and to support the comments these flow charts should help to understand it.





3.8 Retrospective analysis

For this project we weren't given much creative freedom, unlike the last 2 projects, which I found a little disappointing. I personally spent the most time on the case, where we had some wiggle room, but after having done the testing with MQTT, I realized that the firmware could be put together fast because most of it was already running at this point.

The project was also interesting because the whole class had to design the same system, but still be different in some way. This was a little challenging regarding the firmware, because everyone needed the same functionality and most used the same library. To make the code differ or have something extra was for me a personal requirement. That's why I added the access point to configure the WiFi and the webserver where you can see the information of the runner, battery percentage and what mode it is in, running or information.

The project seemed easy at the beginning because most of the pieces that needed to be done had already been done by a lot of other people and there is no sense in reinventing the wheel, so it's better to simply understand these pieces.

3.9 Market analysis

There are many RFID scanning devices on the market, even for these use cases like marathons or charity runs. Although these can be very expensive due to their high accuracy and/or range.

This could make our solution viable as it is not very expensive regarding parts and production. However, the scanner is not as reliable and doesn't have a high enough range to be a reliable and viable option to always scan the runners.

4 Executive summary

The device can scan an RFID tag and send the received data to the provided API and receive the error or success message back.

The device can be configured via an Access Point to connect to any WIFI network. It can also be connected via Ethernet. Then no configuration via the AP is needed.

The device opens a small web server where the received information of the sent RFID is displayed for people to review.

5 Bibliography and list of sources

WIFI config example:
https://randomnerdtutorials.com/esp32-wi-fi-manager-asyncwebserver/
Wiegand Library used:
https://github.com/paulo-raca/YetAnotherArduinoWiegandLibrary
Fusion designs :
https://lukat.lu/files/fusion360/olimex_esp32-poe-iso.f3d
https://lukat.lu/files/fusion360/ailova-rfid-scanner.f3d

6 Acknowledgements

Special thanks to Yves Fischer for helping me with the design of the case.
Special thanks to Luana Do Vale Daniel for helping me all around the project.
Special thanks to Luka Theisen for helping me understand the API calls.

7 Declaration of autonomy

I certify herby that I have written the thesis and report independently without the help of other aids than those explicitly stated.

Name: Charel Feil

Place and Date: Bissen, Luxembourg 29.01.2023

Declaration: I certify herby that I have written the thesis and report independently without the help of other aids than those explicitly stated.

Signature: