Proceedings of the project

OPEN RUCHE

Edouard GENTY

Alain DAVASSOU

Anthonio MEZEÏ

Alaaeddine ABOULMAJDI

Tutors :

Yann DOUZE

Sylvain VIATEUR

January 25th 2021

Table of Contents

I - Birth of the project

The request comes from a competition organized by the Syndicat National d'Apiculture (SNA), the Fédération Nationale des Groupements Techniques Apicoles (FNGTA) and the Demain la Terre association. The kit to be produced, if it is convincing, shall equip partner beekeepers in an indefinite proportion.

II - Initial plans

The request from associations wasn't explicit about the necessary functions. Our tutors have advised us of the primary needs:

- Weigh the beehive.
- Measure the interior and exterior temperature and humidity.
- Be autonomous in energy.
- Communicate wirelessly.

(See the bill of specifications for further details)

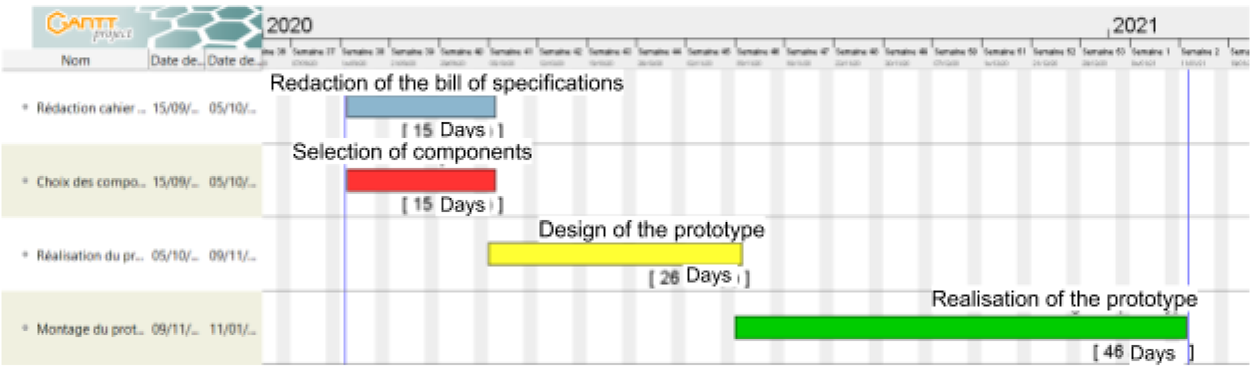
Free for us to imagine additional features. This is what we did. So we have initially planned to add these sensors:

- A microphone to hear the sound of bees, linked to their health.
- A magnetometer to measure the electromagnetic field.
- A GPS module to place the beehive on a map and track if the hive is being stolen.
- A rain gauge to measure the pluviometry.
- A SMS or email alert system in case of emergency.

Unfortunately, the circumstances have complicated the task as we will describe it furtherly.

III - Preparation

We were a group of four students and we had many sessions of four hours from the middle of September to the middle of January to achieve the project.



So after setting up objectives and specifications, we have divided the different tasks depending on our respectives skills. By tasks, we talked about the programmation of each sensor. However, we soon stop working like this and focus all of us on the same thing because the advice of everyone is useful to others.

IV - Programmation of sensors

The components given were chosen by us but validate and ordered by mister Viateur. It results in a certain homogeneity between each group. For the code itself, we use MBED, a platform and operating system for internet-connected devices based on 32-bit ARM Cortex-M microcontrollers.

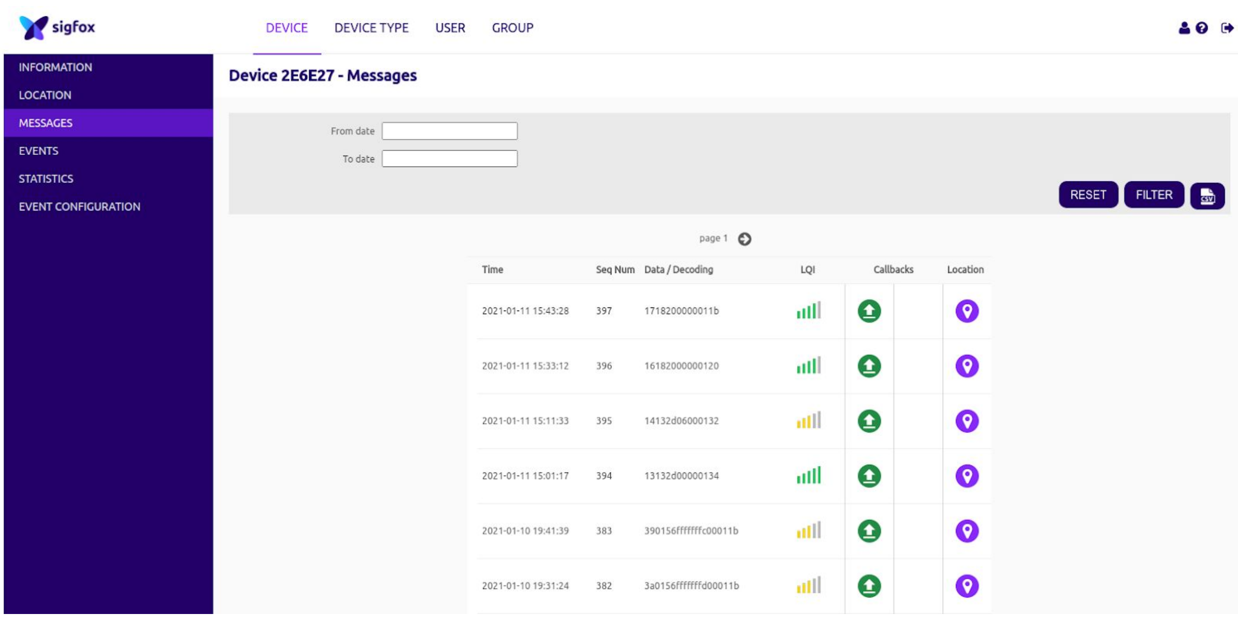
1) Thermometers and hygrometers

Most thermometers sold on public markets include a hygrometer. For the exterior device, we needed a wide measuring range and a precision not extremely accurate. The cheap DHT11 is perfect for this task. It is not waterproof but, placed under the beehive, it is not disturbing. Otherwise, for the interior device, we need a huge accuracy on a small range (see the bill of specification for further details). It also needs to be resistant because a lot of dirtiness will accumulate on the sensor. The DS18B20 is perfect for it.

The programmation was quite simple with the importation of useful libraries with MBED Online. They are included in the Codes.zip folder.

2) The wireless communications

In order to communicate data, we have used a Sigfox module. Sigfox is a company but also the name of a low-power wide-area network (LPWAN). It allows, depending on the plan subscribed, to send around 140 messages of 12 bytes per day. We can see raw data on the Sigfox backend.



Overview of the Sigfox's backend

This technology is perfect for us because it requires very little energy and the electromagnetic field generated is too low to affect the health of the hive.

3) The energy management

Setting up a solar panel and a battery is quite simple using a Lipo-Rider. We saw that even under artificial light, the solar panel is able to give sufficient electricity to make the whole machine work. We work a few hours on battery but we haven't tested if the system can work a whole day of winter outside without shutting down.

4) Weight sensor

The HX711 and its transducer were given to us with a tray on it designed to hold a beehive. The programming was quite difficult because we haven't reached the desired accuracy and repeatability. That's why we have chosen to only print the result in integers. The first step, the calibration, is the most important of all.

5) Anemometer and weathercock

As before, with the right libraries, the programming was quite fast and the tests realized were conclusives. However, the weathercock doesn't return data in degree or in direction (North, South, etc…) but in a short area of number which needs to be indexed into a direction. But each time we start the weathercock up, those values have changed.

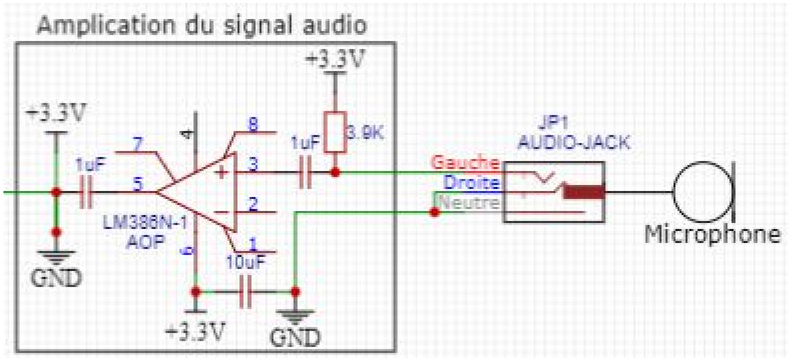
6) Canceled sensors

Because of a lack of technical issues and a deep disruption in our initial planning due to the Covid-19 pandemy, a lot of sensors haven't been deployed in our prototype.

First, the GPS TEL0094, because despite all our efforts, we haven't managed to make it work and because it wasn't a primary need, we cancelled it.

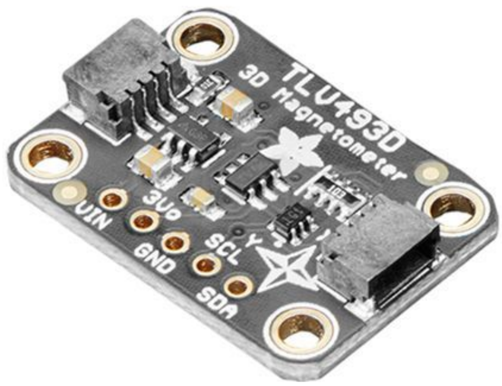
TEL0094 GPS

Then was the microphone. From a lapel microphone, we have designed an audio amplifier circuit for the microcontroller. We have reached to get a decent signal. We have started a program on Matlab to analyse the signal but it requires too much work for the limited remaining time.



Audio amplifier circuit

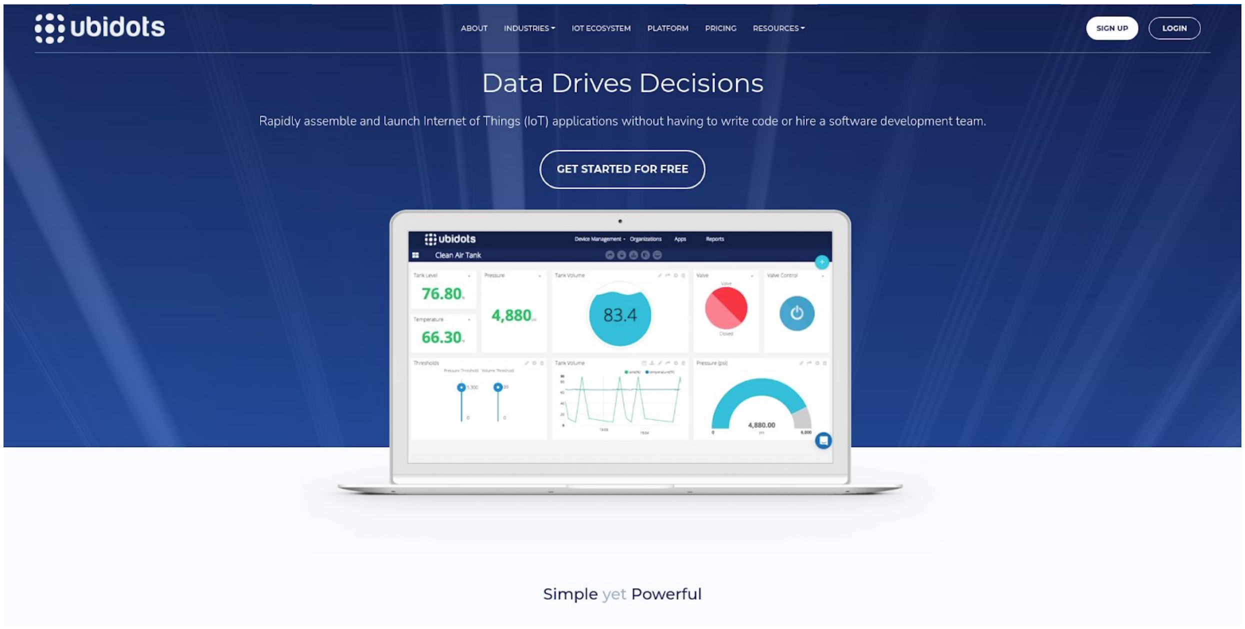
Last was the magnetometer. We haven't succeeded in achieving a working program and the time wasn't sufficient.



TLV493D Magnetometer

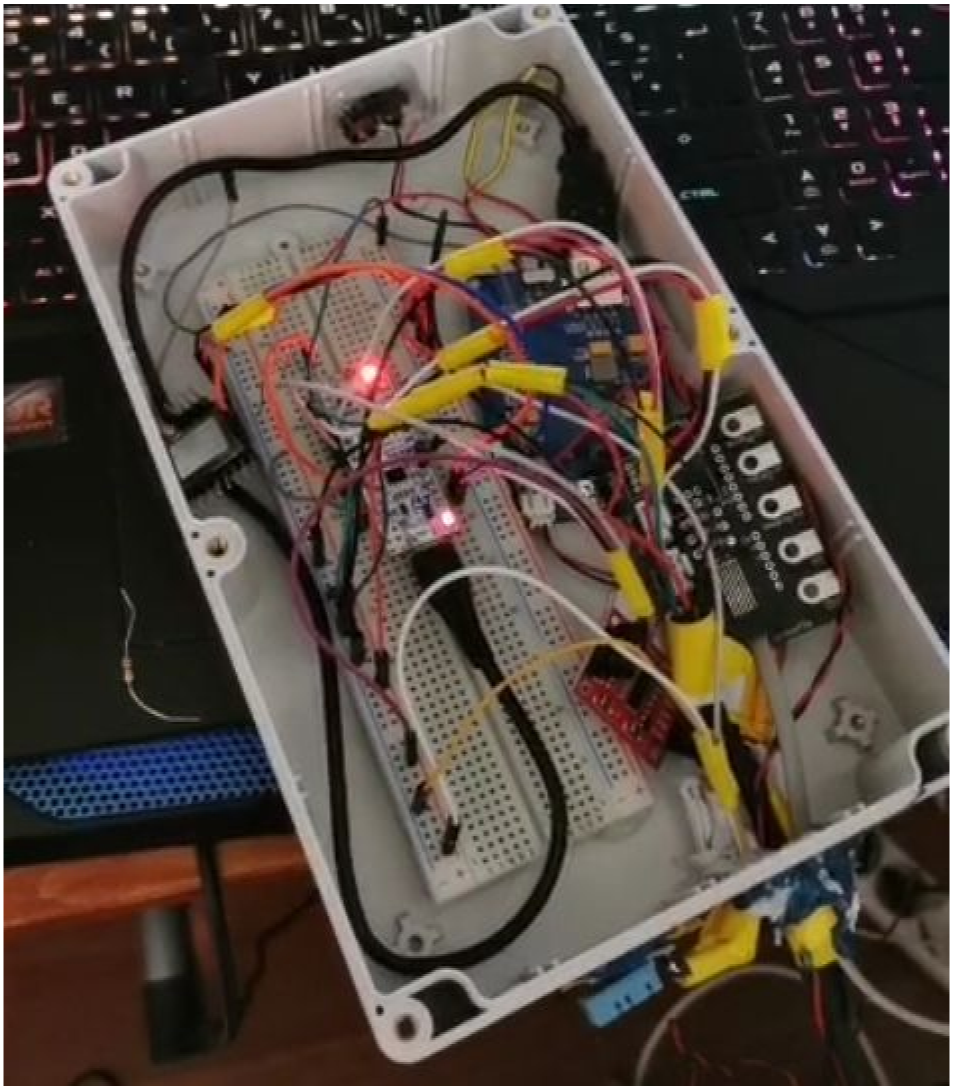
V - Web interface

The backend of Sigfox is not very user-friendly. Fortunately, there is Ubidots. Ubidots is an IoT Platform used to send data to the cloud from any Internet-enabled device. We can then configure and unlock the value of our data through visual tools.



VI - Assembly Operations

The most difficult part of the project was the assembly operation. We have tested each sensor one by one and when we've tried to assemble it all together, we've faced a lot of contact failures. This problem is frequent with breadboards and if the initial plan was to realize a PCB, it was cancelled because of the pandemy. So every time we move the box, contact failures append.



Assembled prototype

VII - Tests

Due to the accumulation of delays because of the difficulty with assembly operations, we haven’t been able to test our prototype on a true beehive but only in our garden and in our house. For all remaining sensors, the test was conclusive and Ubidots show us relevant charts.

Screenshots from the Ubidot’s terminal

VIII - Conclusion

Even if the result is far below the expectations, we have enjoyed releasing this project because we've learnt a lot about IoT which is a fundamental knowledge for every electronic engineer in the near future.

We would like to express all our gratitude to mister Viateur and mister Douze for their advice and their patience.