Wildfire Warning System

Project Engineering

Year 4

Simon GABORIT

Bachelor of Engineering (Honours) in Software and

Electronic Engineering

Galway-Mayo Institute of Technology

2021/2022

Declaration

This project is presented in partial fulfilment of the requirements for the degree of Bachelor of Engineering (Honours) in Software and Electronic Engineering at Galway-Mayo Institute of Technology.

This project is my own work, except where otherwise accredited. Where the work of others has been used or incorporated during this project, this is acknowledged and referenced.

___ Simon GABORIT _____

Acknowledgements

I would like to acknowledge my classmates, who help me for technical question and the support, my housemates, friends and family for their support (especially my sister that provided me useful graphical/design advice) and my supervisor Brian O'SHEA.

Table of Contents

1 Summary 5

2 Poster 6

3 Introduction 7

4 Current Equivalent 8

5 Project Architecture 9

6 Project Plan 10

7 Hardware 11

8 Software 11

8.1 Frontend 11

8.1.1 Next.Js 11

8.1.2 Introduction page 12

8.1.3 Maps page 12

8.1.4 Weather page 13

8.2 Backend 13

8.2.1 Mango DB 13

8.2.2 Node.js 15

9 Problems Encountered 15

9.1 Technical issues 15

9.2 Project Management issues 16

10 Conclusion 16

11 References 17

1 Summary

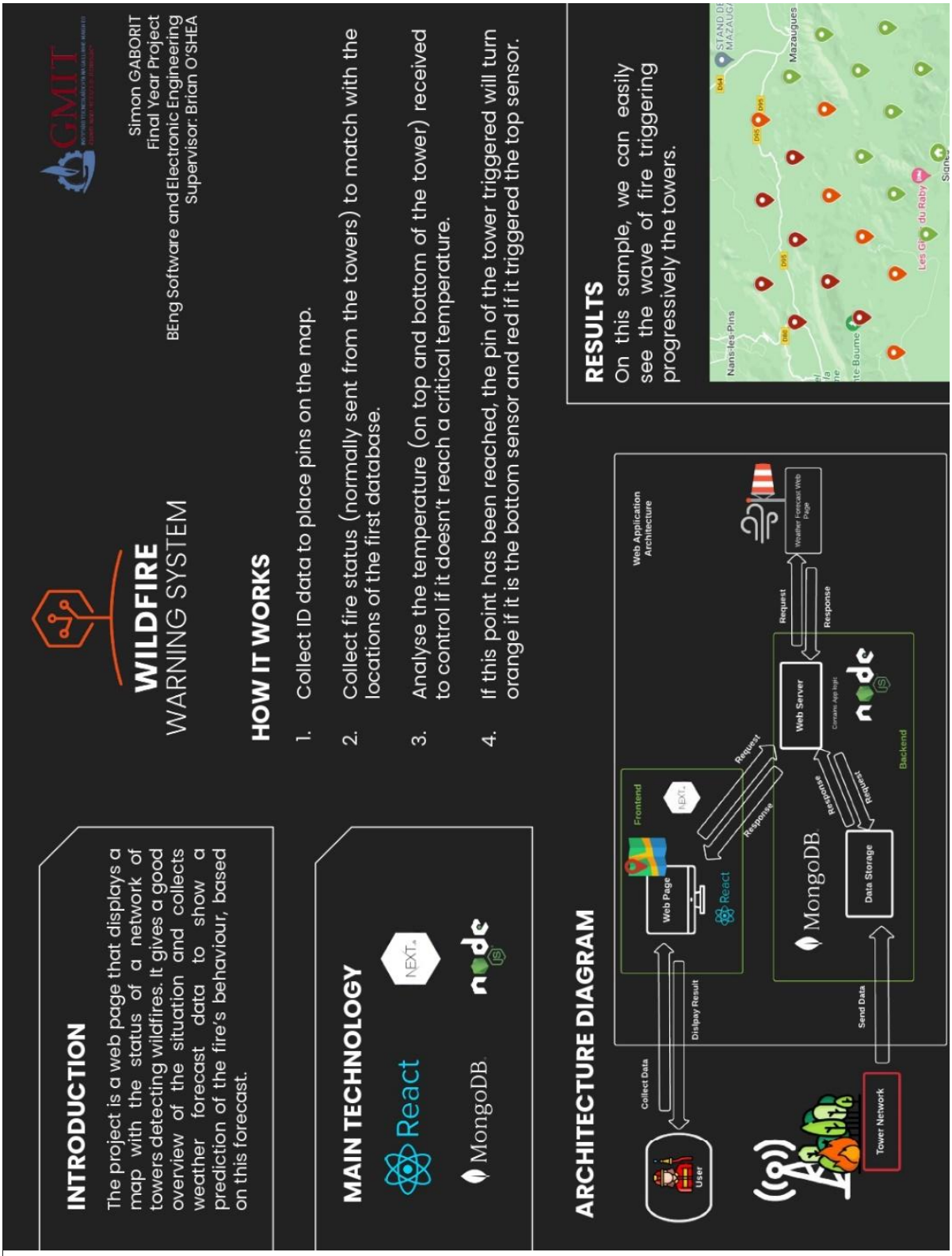
The project is a system of tower network that monitors the situation in the risky areas or the zone difficult to access and report the status of all the towers on a web page. Moreover, the web page collects the weather forecast (Humidity, wind direction and strength, …) at the locations of all the towers to predict the "behaviour" of the fire and display it on a map. Globally, the project gives an overview to the firefighters before, at the beginning and during the wildfire. To demonstrate a realistic project, I decided to not develop the Hardware part and focus on the Software part. Instead, I will create a test file containing the data that are sent by the tower. The web page contains three pages: an introduction page, a map page that displays the map and a forecast page that display the forecast data collected on the weather forecast web page.

The web page is coded in JavaScript, Html and CSS under Next.Js, using the software Visual Studio Code.

MongoDB is hosting a local database where two files are stored. Then, the frontend fetches the data previously called by the backend developed with Node.Js (Express, Mongoose).

Finally, the weather forecast, at the location of the towers, are collected on a web page that provides free online data to display them on the "weather forecast" page and use them in an algorithm to predict the behaviour of the wildfire.

2 Poster



1 Poster

3 Introduction

This project has been thought around the problem of the wildfire that has been illuminated during these last years in different countries all around the world. The consequences on the biodiversity have been terrible in despite the immense effort from all the firefighting task forces of different countries. The lack of resources (material and human) has been one of the main issues of these environmental crisis even if we have seen a close collaboration between the countries to share these resources. Hence, the objective of the project is to prevent these wildfires using accurately these resources, especially during the main crisis.

Wildfire [6]

4 Current Equivalent

Nowadays, this technology would replace and improve two systems currently used for:

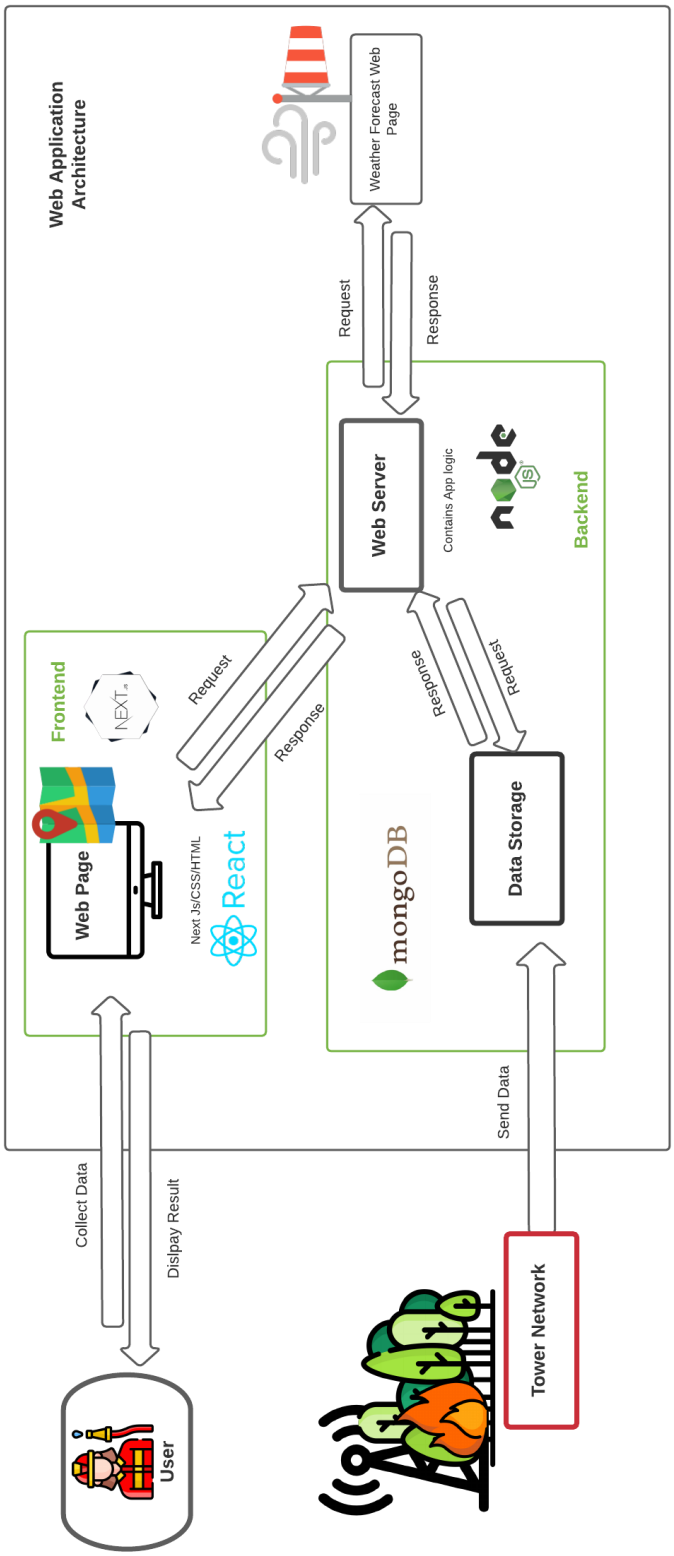
- The patrols: nowadays, one of the only ways to detect the start of a wildfire is the patrols. These teams have to roam thousands of kilometres without any guarantee to be at the good place at the good moment. Moreover, their different means of transport limit them to cover some areas difficult to access. The network watches a large area in real time, even in the area not easily accessible.

Firemen patrol in bush fire [5]

- The aerial point of view: during the intervention to control the wildfire, the only possibility to get an overview on the situation, take the good decision and guide the teams that are fighting the fire is to have, at least, one officer in charge of the operations, in a helicopter or a plane. He is the only one able to take the decision because he is also the only one to know about it. And the fire can go 16 to 20 km/h (9-12.5 mph), hence it is impossible to take decisions based on the visual situation on the ground, and it can be even difficult to manage from the sky. With the new technologies, they also equipped planes with thermal vision camera that communicate real-time images from the sky to the Headquarter. Another expensive solution that could be replaced (in part) by the network.

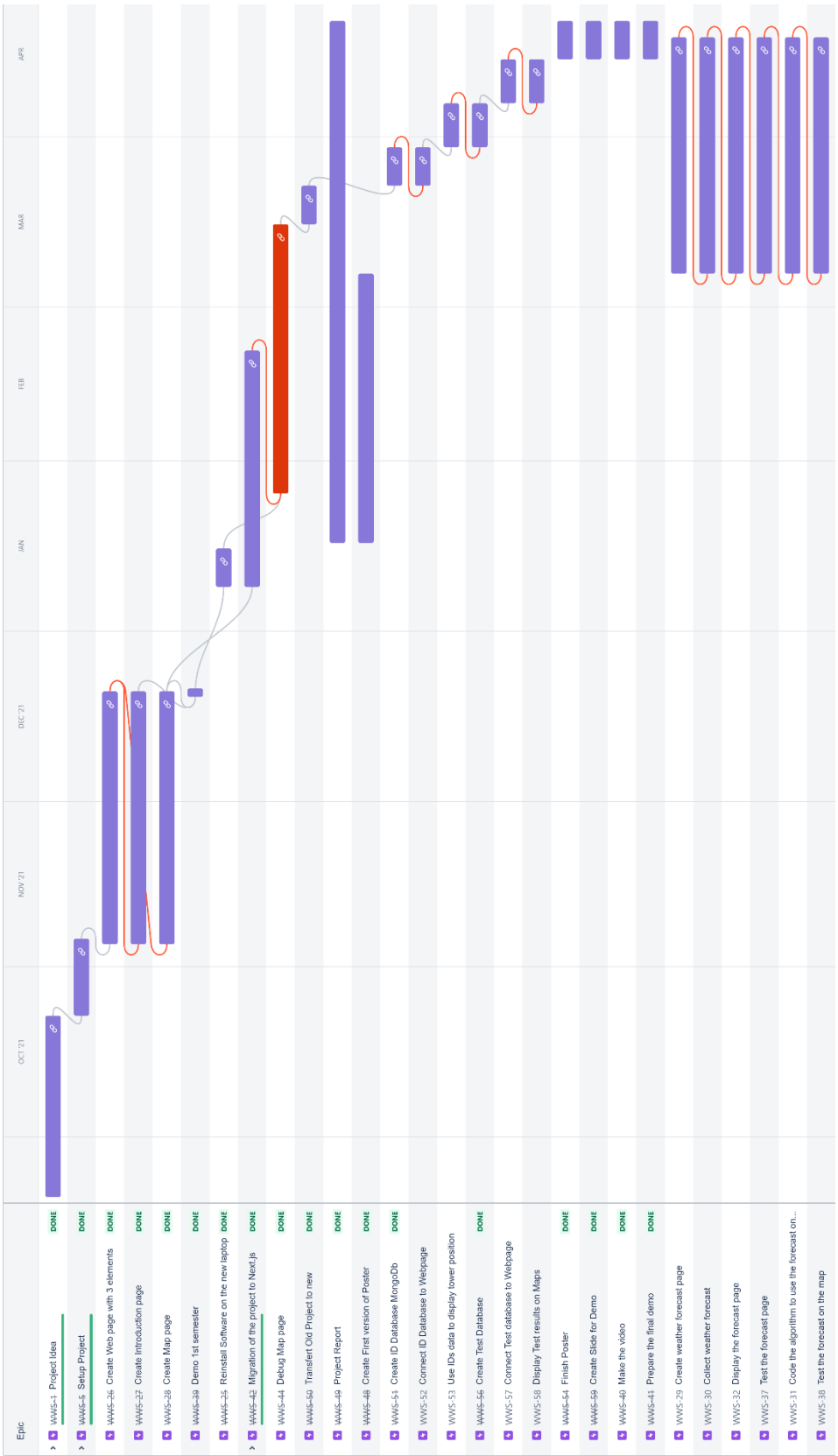
- FIRMS by NASA: the NASA are currently using satellites to detect the wildfire. But these satellites detect sources of heat. Hence, it could be any other sources that gives a low level of accuracy. You also must include the latency of ≈3h between the detection and the upload on the webpage. I don't even mention the price of using a resource like a satellite to detect wildfires.

5 Project Architecture



Architecture Diagram

6 Project Plan



7 Hardware

This part has not been developed in the project for reasons mentioned earlier, but there are some points that are important to mention. Some points are even directly linked to the software.

It is obvious that the selection of the component is essential; they will have to resist to temperatures higher than 1200°C in the worst case. Sometimes, the towers will be isolated in areas difficult to reach. Consequently, the hardware has to be reliable and allow some sleeping/low energy consumption modes. Additionally, the network is built with a IoT communication, such as LoRa or Sigfox. It has a consequence on the structure of the data sent with one of these systems. Hence, to avoid sending a long string message containing the ID, the latitude, the longitude, the high temperature and the low temperature; the latitude and longitude are stored in a database also containing the ID, that the program can use to get the location of the tower (matching the ID).

8 Software

8.1 Frontend

8.1.1 Next.Js

Next.js is an open-source web development framework built on top of Node.js enabling React based web applications functionalities such as server-side rendering and generating static websites. [1]

I started to code my web page with React during the first semester, but then we started to code with Next.Js in Cloud Computing module. It was easier for me to code; it simplifies the structure of it and because it is built on Node.js I could program the backend with the node tools to collect the data from the database. The migration from React to Node was not significant but brought some troubles. Nevertheless, this migration has been a good modification in the project to continue to build on strong basements for the second semester.

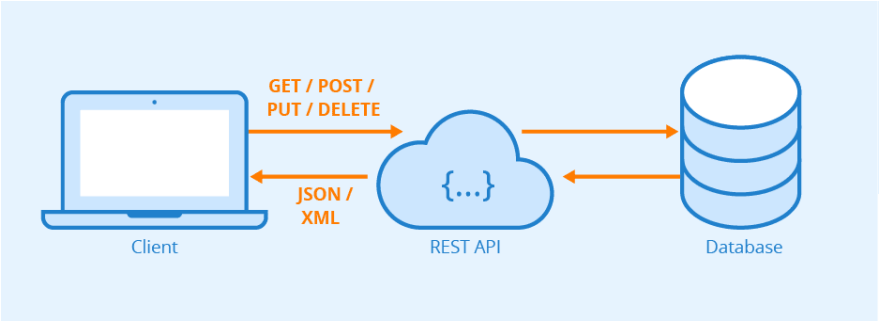
8.1.2 Introduction page

The Introduction page explains the origins of the project and the reasons why I decided to start this project. It also shows some skills in React/Next.js learned previously during the Cloud Computing module or other features learned by myself that brings a fancy and professional look to the web page.

8.1.3 Maps page

This maps page is the heart of the project; it displays the map where you can find all the information about the situation of the wildfire. On this map, there are, permanently the position of the towers.

The communication to collect the data from the Server part is made with a REST API. The **A**pplication **P**rogramming **I**nterface is a Software interface that allows the communication between the application and the server part in this project for example. The data is fetched to the database from the GET. But the code, keeps all the features to use a potential future use of the other functions (POST, PUT, DELETE); this is part of the reliability of the code.



REST API structure [2]

First, this Map page is using the coordinates contained in an IDs database to place the points of the towers on the map with the latitude and longitude.

Secondly, after a short analysis of the temperatures, the program will update the colour of the pins if it reaches a certain temperature: Orange if the lower sensor exceeds the limit and red if the higher sensor exceeds it as well. Then, the weather forecast (explained in the following section) is displayed on this map as well to have an efficient reading of the situation for the people who use the webpage.

8.1.4 Weather page

This page is an intermediate step in the project. The final goal of the project is to collect weather forecast data from a web page at the position of the towers. Then, after an analyse of this information, an algorithm would propose a potential behaviour of the wildfire to anticipate it, according to humidity, temperature, wind, …

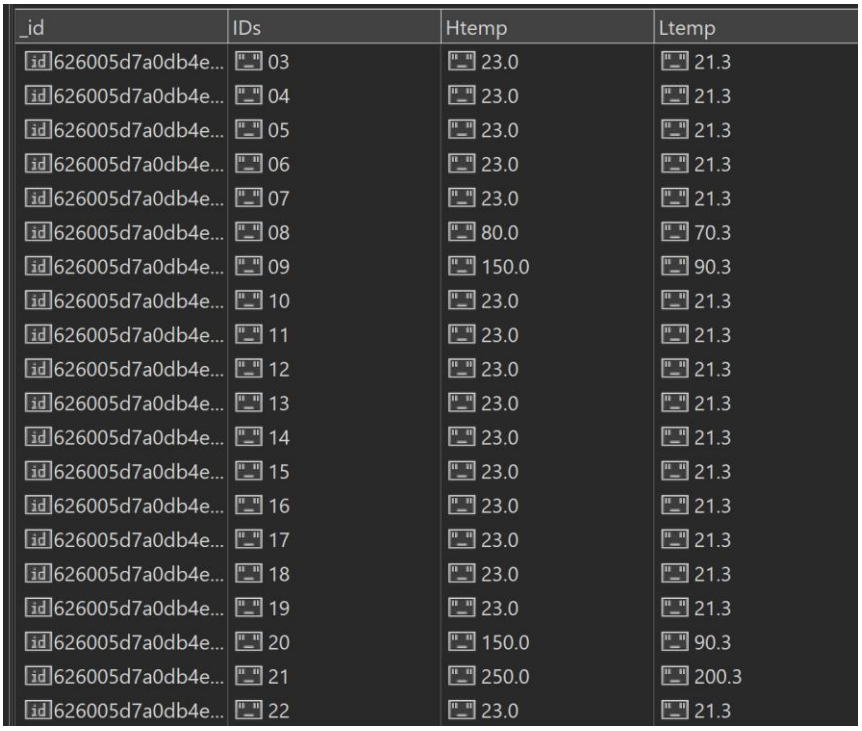
So, this page displays the data collected from the web page before to analyse them and add them to the map. It is only an intermediate step for the final demonstration in case I don't have time to implement the algorithm and update the map, I will have a visual proof of the implementation of this last step.

8.2 Backend

8.2.1 Mango DB

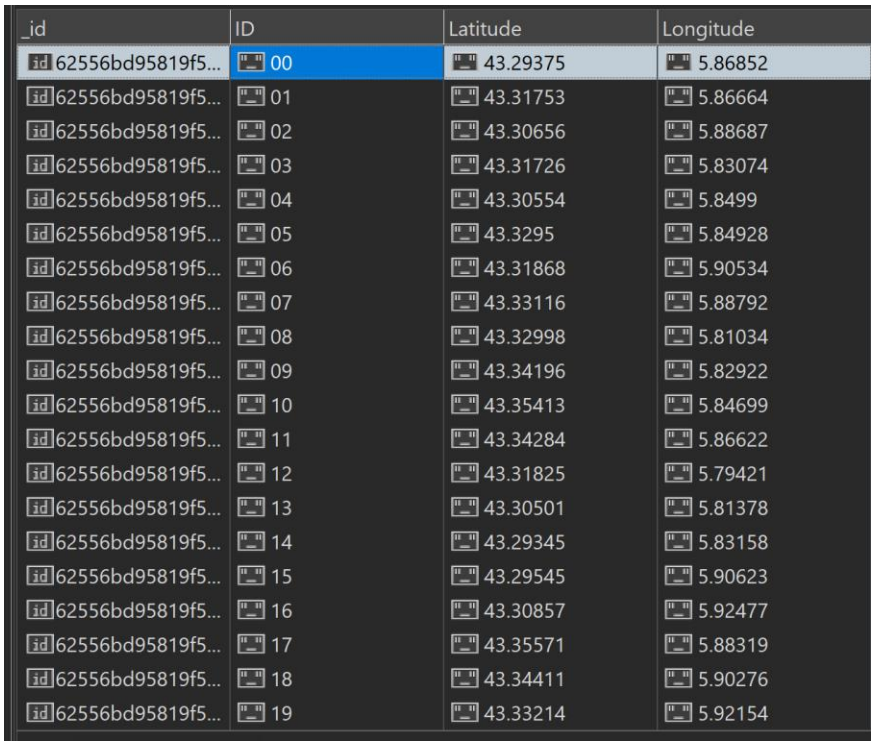
MongoDB is a source-available cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas. MongoDB is developed by MongoDB Inc. and licensed under the Server Side Public License (SSPL). [3]

I used MongoDB as a database containing two files that that I needed in my project. I used Studio3T (Robot3T in the past) as a software to manage more easily my database, but MongoDB includes a shell window where you can use all the management tools with the command lines. To create the database, I imported CSV files, previously written with Excel. Following a certain configuration, the software automatically creates the database.



_id	IDs	Htemp	Ltemp
626005d7a0db4e...	03	23.0	21.3
626005d7a0db4e...	04	23.0	21.3
626005d7a0db4e...	05	23.0	21.3
626005d7a0db4e...	06	23.0	21.3
626005d7a0db4e...	07	23.0	21.3
626005d7a0db4e...	08	80.0	70.3
626005d7a0db4e...	09	150.0	90.3
626005d7a0db4e...	10	23.0	21.3
626005d7a0db4e...	11	23.0	21.3
626005d7a0db4e...	12	23.0	21.3
626005d7a0db4e...	13	23.0	21.3
626005d7a0db4e...	14	23.0	21.3
626005d7a0db4e...	15	23.0	21.3
626005d7a0db4e...	16	23.0	21.3
626005d7a0db4e...	17	23.0	21.3
626005d7a0db4e...	18	23.0	21.3
626005d7a0db4e...	19	23.0	21.3
626005d7a0db4e...	20	150.0	90.3
626005d7a0db4e...	21	250.0	200.3
626005d7a0db4e...	22	23.0	21.3

The adjacent picture is the first database that I call in my code. I created this independent database to avoid sending the geolocation at each update of the status of the towers, because it is static and permanent data. As mentioned previously, with the IoT network technology, such as LoRa or Sixfox, and the low consumption policy, due to the geographical condition of the project, the message sent must remain as short as possible.



_id	ID	Latitude	Longitude
62556bd95819f5...	00	43.29375	5.86852
62556bd95819f5...	01	43.31753	5.86664
62556bd95819f5...	02	43.30656	5.88687
62556bd95819f5...	03	43.31726	5.83074
62556bd95819f5...	04	43.30554	5.8499
62556bd95819f5...	05	43.3295	5.84928
62556bd95819f5...	06	43.31868	5.90534
62556bd95819f5...	07	43.33116	5.88792
62556bd95819f5...	08	43.32998	5.81034
62556bd95819f5...	09	43.34196	5.82922
62556bd95819f5...	10	43.35413	5.84699
62556bd95819f5...	11	43.34284	5.86622
62556bd95819f5...	12	43.31825	5.79421
62556bd95819f5...	13	43.30501	5.81378
62556bd95819f5...	14	43.29345	5.83158
62556bd95819f5...	15	43.29545	5.90623
62556bd95819f5...	16	43.30857	5.92477
62556bd95819f5...	17	43.35571	5.88319
62556bd95819f5...	18	43.34411	5.90276
62556bd95819f5...	19	43.33214	5.92154

This adjoining picture on the left show the second database that contains the sample data that mocks the information that are supposed to be sent by the towers, such as the ID (to match with the location of the first database), the temperatures on top and bottom of the tower that will be analysed and used to display the situation on the map. The database and the communication with it will be developed in the Backend section (Node).

8.2.2 Node.js

Node.js is an open-source, cross-platform, back-end JavaScript runtime environment that runs on the V8 engine and executes JavaScript code outside a web browser. Node.js lets developers use JavaScript to write command line tools and for server-side scripting—running scripts server-side to produce dynamic web page content before the page is sent to the user's web browser. [4]

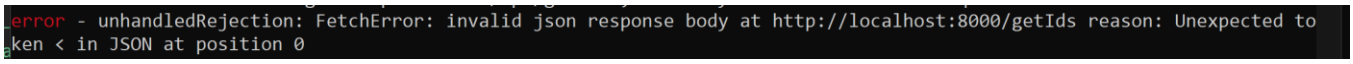
I use Node in my backend to control and manage my database. Node.Js works faster to collect the data that I need. It is important for this project to update quickly the information, especially crisis when the situation can change rapidly. The Hardware is already potential long to detect the change of state, so the Hardware is not allowed to have any lack of speed.

9 Problems Encountered

9.1 Technical issues

The entire project has been a challenge for me; to implement different technologies and knowledge in one project, but two problems have been really time consuming:

- Display Google maps on my Web page: I spent a lot of time to find a solution that came when my supervisor ran the original code on his machine to confirm that was a safe code, and then I injected piece by piece the original code to mine. The error was a simple point in the CSS file that was blocking the display while the code was running well.
- Collect the data from my database: this is the error that blocked me utile the end of the project. An error appeared in the shell:



I couldn’t figure out what is at the origin of this problem. I received help from one of my classmates, hence I will try a last solution before the demonstration.

9.2 Project Management issues

The management of the project has been one of the most difficult things since the beginning of the project. I forgot to include all the time spent to study the other modules. Also, the project plan didn't include a large enough time for technical issues that are incredibly time consuming.

10 Conclusion

This project has been a real challenge for me but unless these difficulties I learned a lot and developed many skills and knowledge that will help me in my future career. I am, obviously I little bit sad to not achieve my goals, but I learned from these "fails" and I am planning to finish a part of this project after the presentation.

This project also confirmed my idea about the master that I want to do to obtain the skill needed to be a systems analyst rather than a Hardware/Software engineer. I preferred working on the project management issues than the technical issues.

Canadair/ Fire bomber plane [7]

11 References

[1] “Wikipedia,” [Online]. Available: https://en.wikipedia.org/wiki/Next.js.

[2] “ASTERA,” [Online]. Available: https://www.astera.com/type/blog/rest-api-definition/.

[3] Wikipedia, “Wikipedia,” [Online]. Available: https://en.wikipedia.org/wiki/MongoDB.

[4] Wikipedia, “Wikipedia Node.Js,” [Online]. Available: https://en.wikipedia.org/wiki/Node.js.

[5] stevepb, “pixabay,” [Online]. Available: https://pixabay.com/photos/firefighter-wildfire-grass-fire-807379/.

[6] Unknown, “Pixabay,” [Online]. Available: https://www.pexels.com/photo/forest-on-fire-51951/.

[7] fish96, “Pixabay,” [Online]. Available: https://pixabay.com/photos/canadair-aircraft-firefighting-6768428/.