SUP'COM
Higher School of Communication of Tunis

Scope Statement of WOT project

Project name :

Help controlling and reporting the situation of animals

We **C**are

Submitted by :

Mariem Mnasser

Rihab Ghrab

Under the guidance of:

Mr.Mohamed Becha Kaaniche

Academic year : 2020 - 2021

Table of contents :

Table of figures :

Introduction :

Animals have been considered human’s closest friends for a very long time. We are dependent on them for food and security (Habitat). Every animal in the world needs to be taken care of by providing them food, shelter and health care. The behavior of most of the domestic animals is mainly observed by how we treat them. If we treat them with good care, love and compassion, they will definitely show the same love and respect towards us.

Animals are dependent on us whether in our presence or absence, so we should provide them with the same safety level in both cases .

The impact of our absence , lack of care and abundance of animals whether in home or even in the protectorates and zoo , is highly contributing to the illness and weakness of those , specially rare species and those which need highly health care (pregnant, ill...).

Objective of the project :

The project objective is to create a system capable of measuring an animal's body temperature and heart rate (ECG: electrocardiogram) then send the measures taken to an application that will process this data and extract the health status of the animal and send a status report to the keeper. In case of danger, the application will trigger an alert that will be received by the owner who will intervene to save them before reaching any high level complications .

Technologies used :

Throughout this project we will be using multiple tools and technologies that are essential to realize the overall solution that we proposed .

1. Tools :

1.1 ESP8266 NodeMCu:

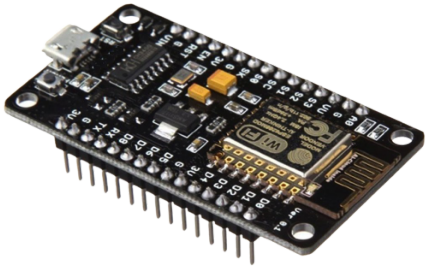


figure 1: ESP8266 NodeMCu

NodeMCU is an open-source Lua based firmware and development board specially targeted for IoT based Applications. It includes firmware that runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which is based on the ESP-12 module.

Specifications :

- Microcontroller: Tensilica 32-bit RISC CPU Xtensa LX106:
- Operating Voltage: 3.3V
- Input Voltage: 7-12V
- Digital I/O Pins (DIO): 16
- Analog Input Pins (ADC): 1
- UARTs: 1

- SPIs: 1
- I2Cs: 1
- Flash Memory: 4 MB
- SRAM: 64 KB
- Clock Speed: 80 MHz
- USB-TTL based on CP2102 is included onboard, Enabling Plug n Play
- PCB Antenna
- Small Sized module to fit smartly inside your IoT projects

1.2 Heart rate sensor :

The sensor we chose was Pulsesensor Heart Rate Pulse Pulse Sensor Amplifier Module for Arduino .

figure 2: Heart rate sensor

• The pulse sensor is used to test the heart rate sensor.

• It has an integrated optical heart rate sensor amplifier and noise-cancelling circuit.

Specifications:

- Power supply: 3-5V
- Diameter: 16 mm
- Magnification: 330b
- LED wavelength: 609 nm

The characteristics of the sensor respond well to the needs of the project: the size is

small and practical, it is able to be connected through the internet.

1.3 Temperature sensor LM-317:

The LM35 sensor is a type of sensor whose output is an analog DC voltage signal and can measure a temperature range of -40℃ to 125℃ , which is suitable enough to measure the targeted animal body temperature.

This type of sensor is preferable in this project because it does not cause any damage. In fact, it requires only a low output voltage. With an accuracy of 0.5°C, it can be converted from voltage to degree Celsius depending on the function.

Temp(C) =Output Voltage *V_{input}/1024

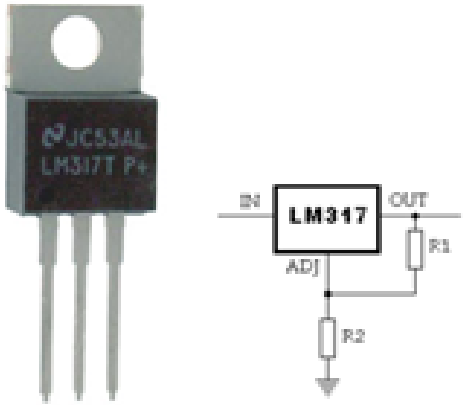


figure 3: Temperature sensor LM-317

2. Technologies :

2.1 Frontend :

2.1.1 Ionic :

figure 4: ionic logo

Ionic definition :

Ionic is a complete open-source SDK for hybrid mobile app development. Known as an open source mobile toolkit for building high quality, cross-platform native and web app experiences. Ionic provides tools and services for developing hybrid mobile, desktop, and progressive web apps based on modern web development technologies and practices, using Web technologies like CSS, HTML5, and Sass. In particular, mobile apps can be built with these Web technologies and then distributed through native app stores to be installed on devices by utilizing Cordova or Capacitor.

Reasons :

As stated earlier, the Ionic Framework allows a mix of languages such as JavaScript, HTML, and CSS to be used, and because it is an open-source platform, it has proven to be a favored framework for experts in mobile application development.

- Across Platforms
- OpenSource
- Angular Base
- Large Community

2.1.1 React.js :

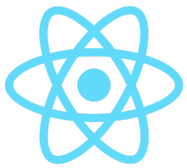


figure 5: React.js logo

React.js definition :

Is a free and open-source front-end JavaScript library for building user interfaces or UI components. It is maintained by Facebook and a community of individual developers and companies. React can be used as a base in the development of single-page or mobile applications. However, React is only concerned with state management and rendering that state to the DOM, so creating React applications usually requires the use of additional libraries for routing, as well as certain client-side functionality.

Reasons :

- React lets you build rich user-interfaces easily. Quality of user-interfaces is important because a poorly designed user-interface is generally less user-friendly and the users will not like a poorly designed UI.
- It is fast in development
- Trusted by great companies

2.1.3 Capacitor :

figure 6: Capacitor logo

Capacitor definition :

Capacitor is an open source project that runs modern Web Apps natively on iOS, Android, Electron, and Web (using Progressive Web App technology) while providing a powerful and easy-to-use interface for accessing native SDKs and native APIs on each platform. As an alternative to Cordova, Capacitor delivers the same cross-platform benefits, but with a more modern approach to app development, taking advantage of the latest Web APIs and native platform capabilities.

Using Capacitor, developers can build one app and target one set of APIs regardless of the platform the app is running on, as opposed to managing multiple APIs for each target platform.

Reasons :

- First-class Progressive Web App (PWA) Support
- Simple yet powerful CLI tooling that is version managed per app
- Capacitor manages plugins in a different and efficient way
- Capacitor allow adding hybrid and web to an existing native app
- Capacitor is well maintained

2.2 Middleware :

2.2.1 Node.js :

figure 7: Node.js logo

NodeJs definition :

As an asynchronous event-driven JavaScript runtime, Node.js is designed to build scalable network applications. It is easily employed as a server-side proxy where it can handle a large amount of simultaneous connections in a non-blocking manner. It's especially useful for proxying different services with different response times, or collecting data from multiple source points.

Reasons:

Node.js is an excellent tool for implementing IoT development projects.

First, Node js is fast and powerful, and therefore capable of handling large data flows. Second, you can easily integrate Node.js with IoT protocols (the integration with MQTT and WebSockets can be a good example).

And finally, the Node Package Manager, briefly described above, has a significant number of helpful IoT modules that help implement the most ambitious IoT projects.

2.3 Backend :

2.3.1 MongoDB :

figure 8 : MongoDB logo

MongoDB definition :

MongoDB is an open-source document-oriented database. It is used to store a larger amount of data and also allows you to work with that data. MongoDB is not based on the table-like relational database structure but provides an altogether different mechanism for storage and retrieval of data, that’s why it is known as NoSQL database. Here, the

term ‘NoSQL’ means ‘non-relational’. The format of storage is called BSON (similar to JSON format).

Reasons :

MongoDB can help you rapidly capture the most value from the IoT. Its flexible schema makes it easy to evolve and store data in a way that is easy for programmers to work with. MongoDB is also built to scale up quickly and supports all the main features of modern databases such as transactions.

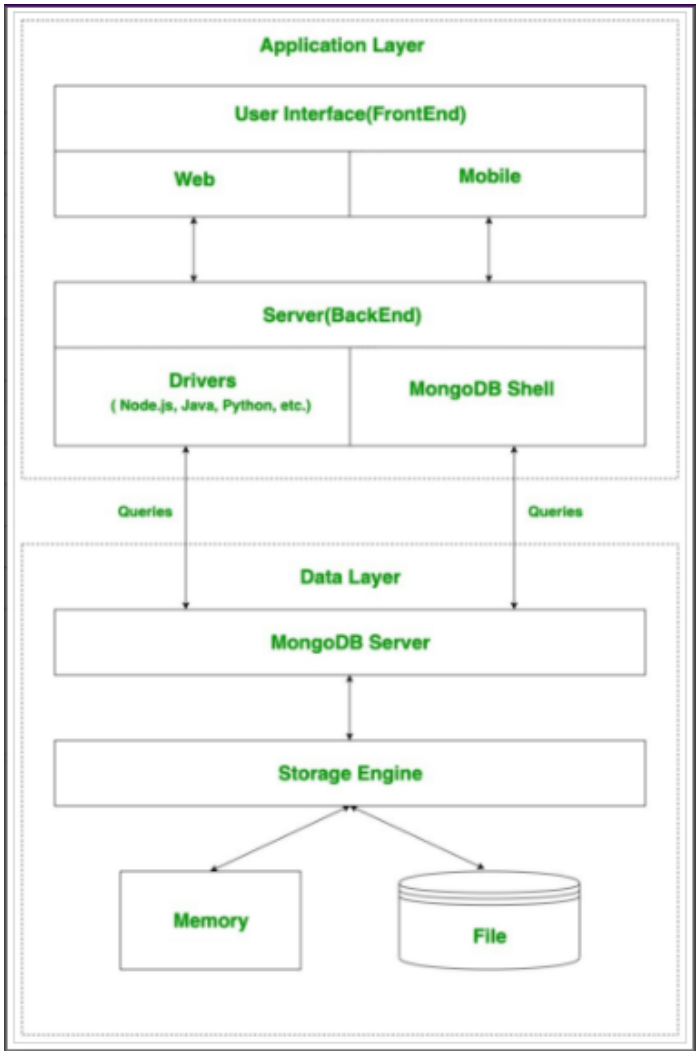


figure 9: MongoDB architecture

- React lets you build rich user-interfaces easily. Quality of user-interfaces is important because a poorly designed user-interface is generally less user-friendly and the users will not like a poorly designed UI.
- It is fast in development
- Trusted by great companies

2.3.2 Mosquitto :

figure 10: Mosquitto logo

Mosquitto definition :

Eclipse Mosquitto is an open source (EPL/EDL licensed) message broker that implements the MQTT protocol versions 5.0, 3.1.1 and 3.1. Mosquitto is lightweight and is suitable for use on all devices from low power single board computers to full servers.

The Mosquitto project also provides a C library for implementing MQTT clients, and the very popular mosquitto_pub and mosquitto_sub command line MQTT clients.

Reasons :

Because it is a lightweight method of carrying out messaging using a publish/subscribe model. This makes it suitable for Internet of Things messaging such as with low power sensors or mobile devices such as phones, embedded computers or microcontrollers.

2.3.3 Node-RED :

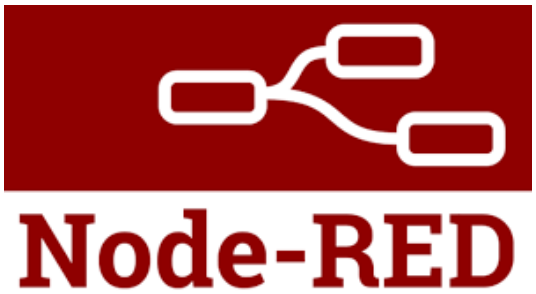


figure 11: Node-RED logo

Node-RED definition :

Node-RED is a flow-based development tool for visual programming developed originally by IBM for wiring together hardware devices, APIs and online services as part of the Internet of Things.

Node-RED provides a web browser-based flow editor, which can be used to create JavaScript functions. Elements of applications can be saved or shared for re-use. The runtime is built on Node.js. The flows created in Node-RED are stored using JSON. Since version 0.14, MQTT nodes can make properly configured TLS connections.

Reasons :

- As the name no-code/low-code indicates, coding is eliminated and programming is intuitively completed with a minimal number of operations needing to be used.
- Efficiency : Node-RED flow editor takes care of building the application execution environment, library synchronization, the integrated development environment (IDE), and editor preparation so that you can concentrate on development.
- High quality is the true value of flow-based and visual programming. Each node provided as a component is a complete module that has been unit tested.

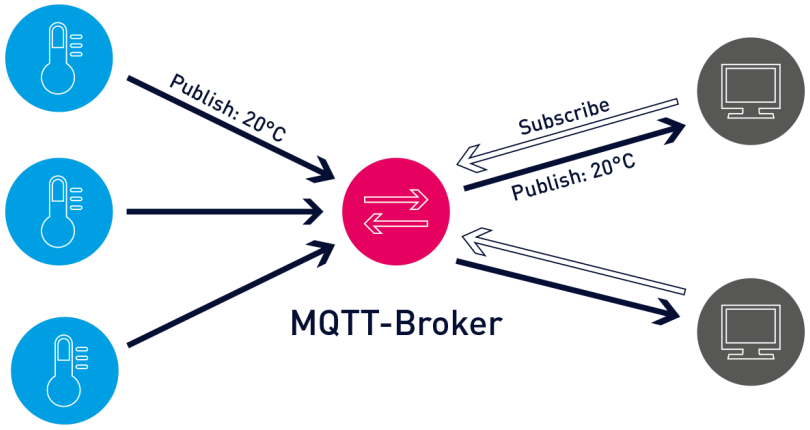


figure 12: MQTT-Broker

General representation :

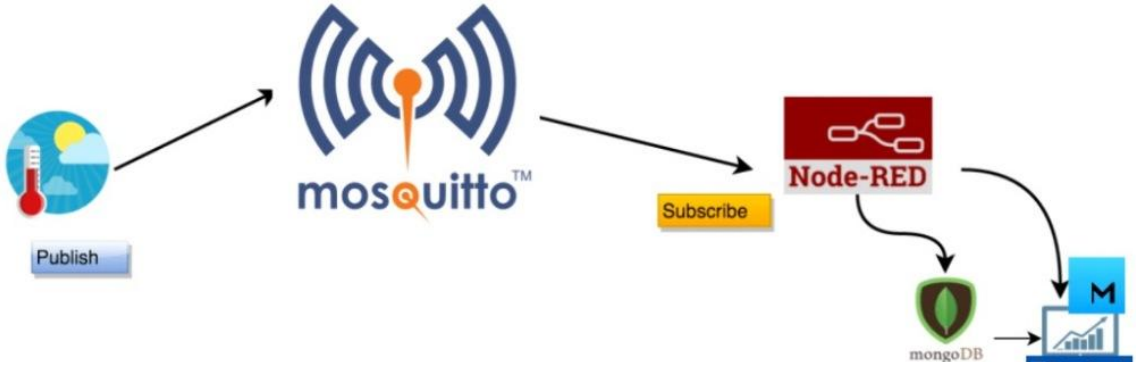


figure 13: General representation

Conception :

1. Use case diagram :

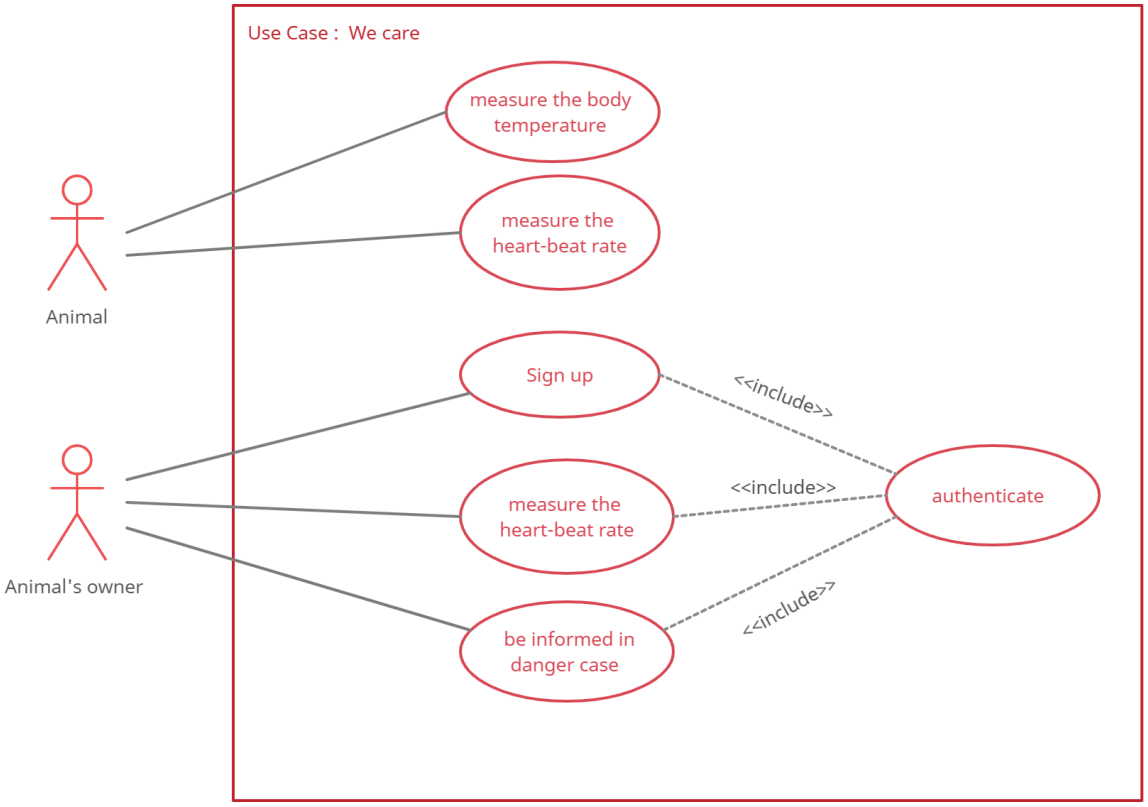


figure 14: Use case diagram

2. Component diagram :

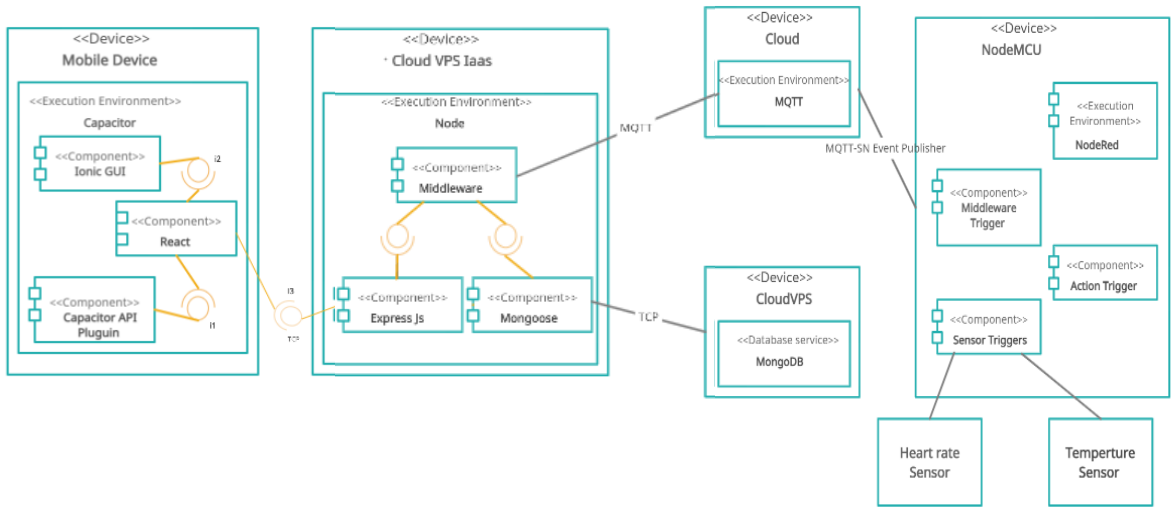
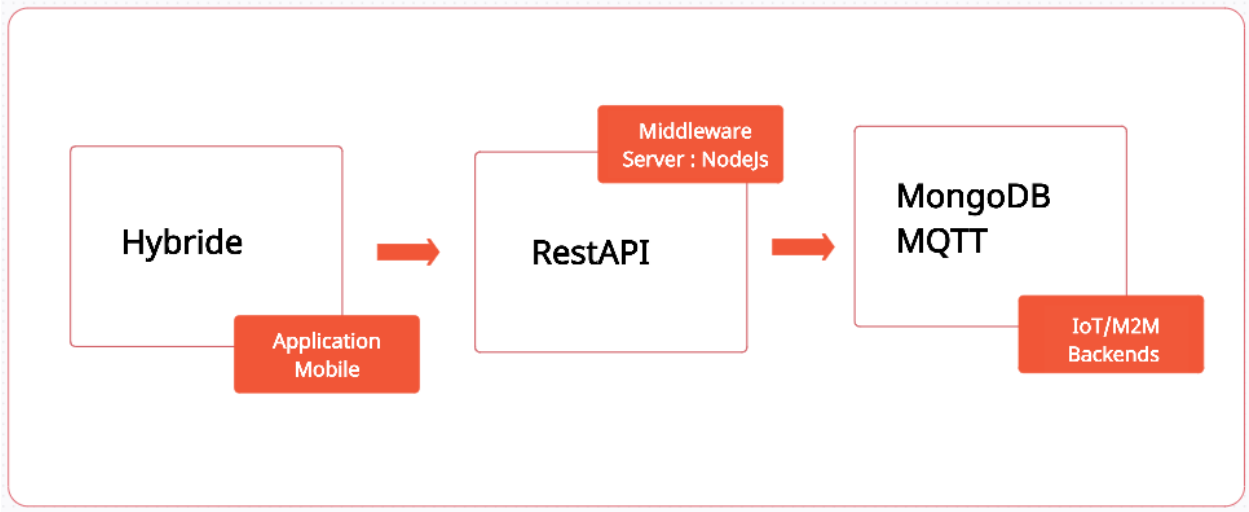
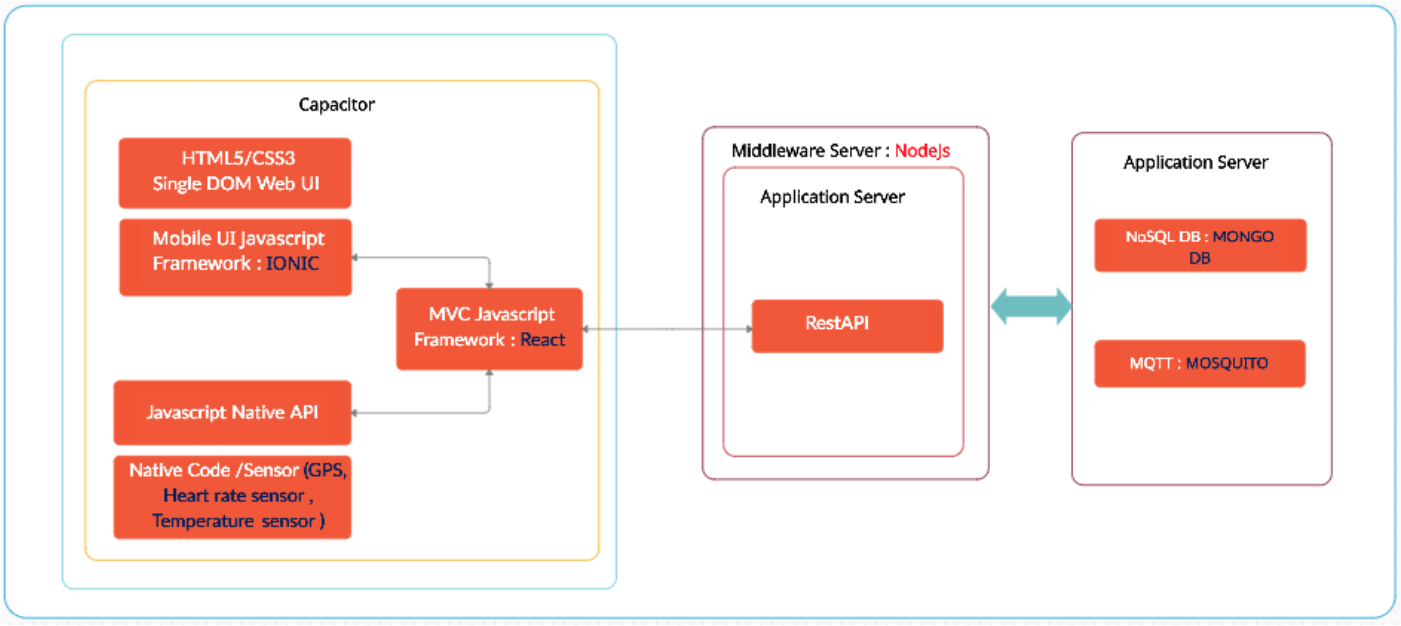
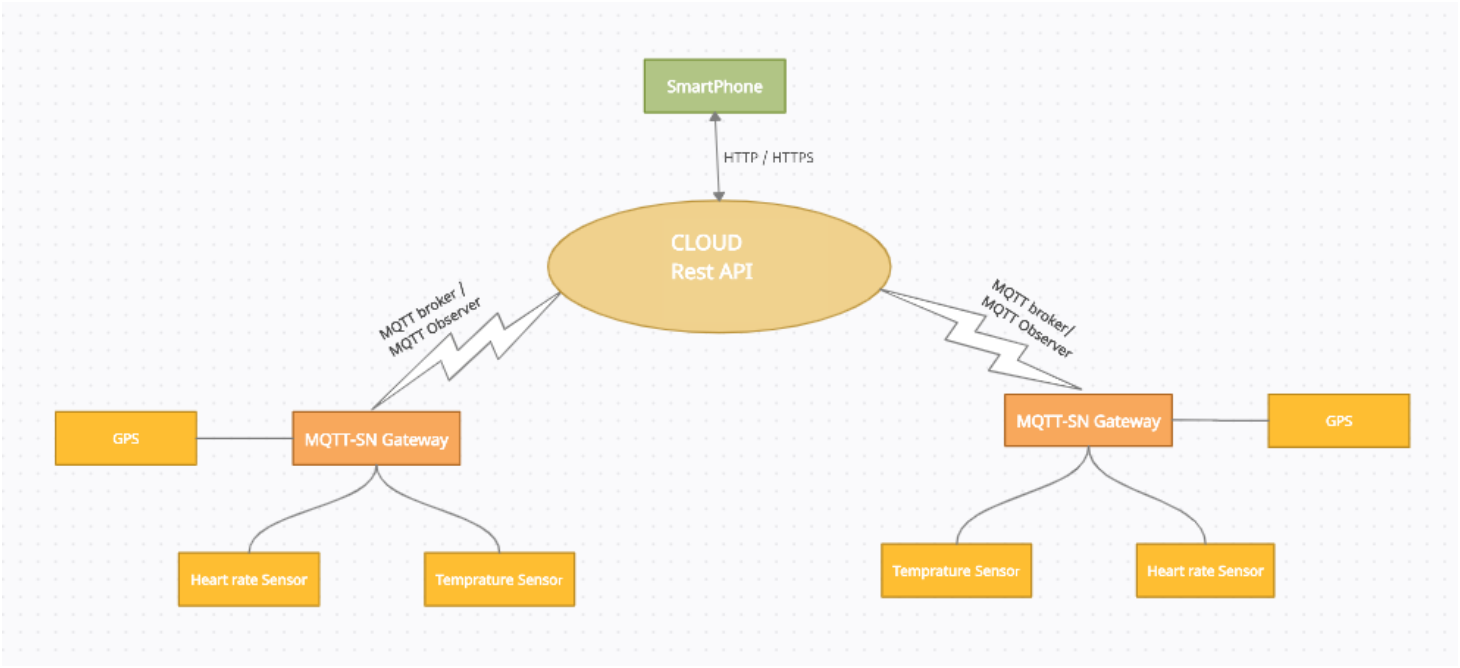


figure 15: Component diagram

General Overview :




Capacitor
HTML5/CSS3
Single DOM Web UI
Mobile UI Javascript
Framework : IONIC
MVC Javascript
Framework : React
Javascript Native API
Native Code /Sensor (GPS,
Heart rate sensor ,
Temperature sensor)
Middleware Server : NodeJs
Application Server
RestAPI
Application Server
NoSQL DB : MONGO
DB
MQTT : MOSQUITO


SmartPhone
HTTP / HTTPS
CLOUD
Rest API
MQTT broker /
MQTT Observer
MQTT broker /
MQTT Observer
GPS
MQTT-SN Gateway
MQTT-SN Gateway
GPS
Heart rate Sensor
Temprature Sensor
Temprature Sensor
Heart rate Sensor

Conclusion :

This project will be highly beneficial for animals as well as animal's keepers . We aim to create a more trustful and certain environment where animals are kept in high health standards . Thus we will be essentially contributing in healing animals wherever and whatever : rare animals , those in the zoo , in shelters and in homes . We will be assuring the health of all of them thanks to our project concept.

In this project we considered the idea of a web of things application where we set up sensors to control the animal then in real-time we send and visualize results in the mobile app designed to be the communication tool with the animal's keeper . Thus we keep an eye on animals to prevent any dangerous situation.