# 

Design and Analysis of Iot

(SOFE 4610U)

Project Design and Architecture

Dr. Ramiro Liscano

Fall 2021

GitHub Link:

<https://github.com/janeCoralde/IoT_Assignment2>

Zainab Rehan - 100653955

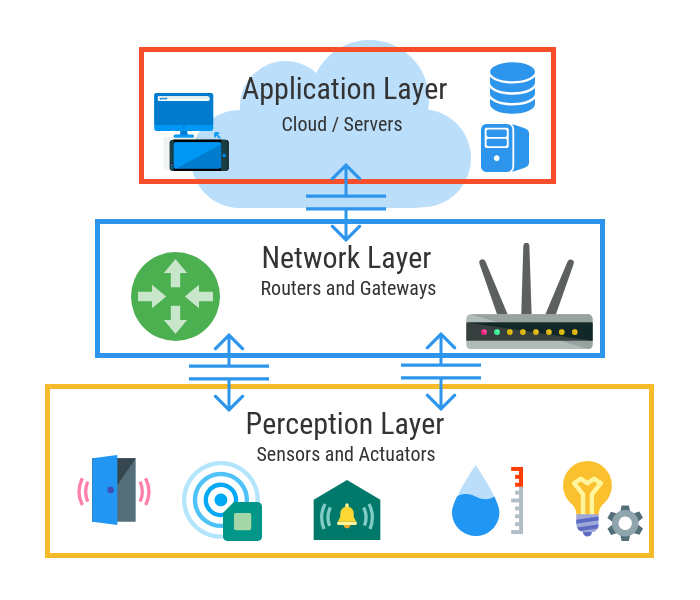
Ramin Sami - 100655257

Jane Coralde - 100660214

Amin Khakpour - 100669547

# Project Architecture

Image 1: Simple and generic IoT architecture was referenced to explain project architecture. No single standard reference for an IoT architecture due to variety of technologies.

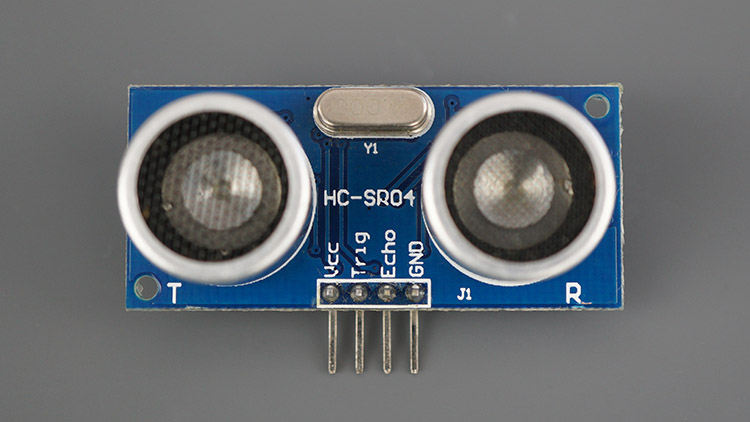


## Perception Layer - Sensors and Actuators

## HC-SR04 Ultrasonic Sensor

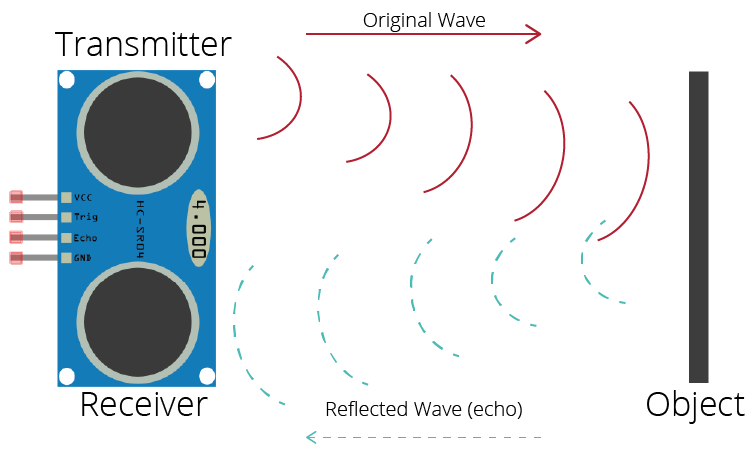
Helpful link: https://randomnerdtutorials.com/esp8266-nodemcu-hc-sr04-ultrasonic-arduino/

See image below



It works by using sonar to determine the distance to an object. As follows:

1. The ultrasound transmitter (trig pin) emits a high-frequency sound (40 kHz).
2. The sound travels through the air. If it finds an object, it bounces back to the module.
3. The ultrasound receiver (echo pin) receives the reflected sound (echo).



For calculation, formula used to get value of distance (in centimeters) is distance to an object = ((speed of sound in the air)\*time)/2

One factor to add is that the sensor does not work properly without supplying the proper voltage when connected to the microcontroller - see sensor’s technical data. In order to overcome this, power is set up in parallel (and not using a VIN pin that supplies 3.3V) in order to get 5V.



### 1.1.2 ESP8266

Developed code was loaded to microcontroller to do the following actions:

1. Connect to WiFi using given credentials
2. Connect MQTT Broker
3. Establish connection
4. Use sensor to get data
5. Publish using MQTT

## Perception Layer - Sensors and Actuators

Cloud instance was created using DigitalOcean platform. Basic software installed in a virtual environment consists of Apache (to serve virtual hosts), MySQL, PHP.

We procured a domain from GoDaddy and configured the nameservers in order to connect our digital droplet in DigitalOcean.

Image1: Created domain for droplet in DigitalOcean

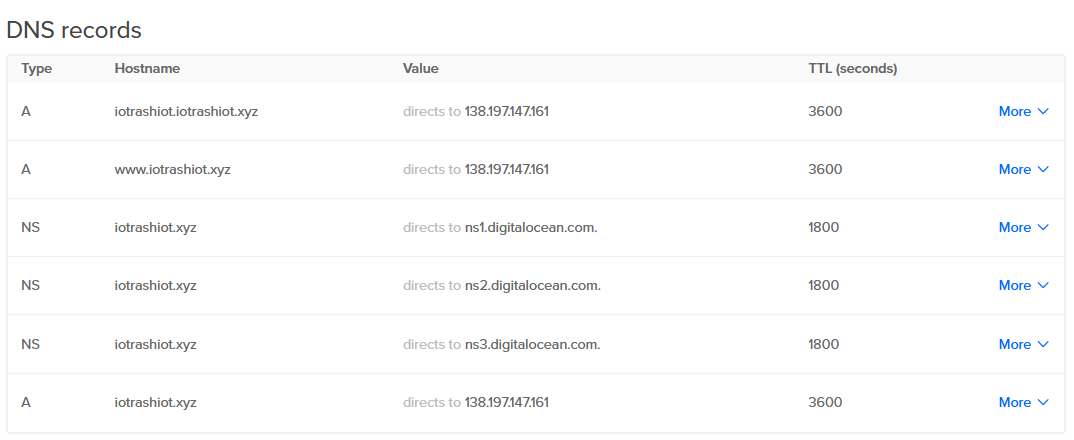


Image2: Updated our nameservers with a domain registrar

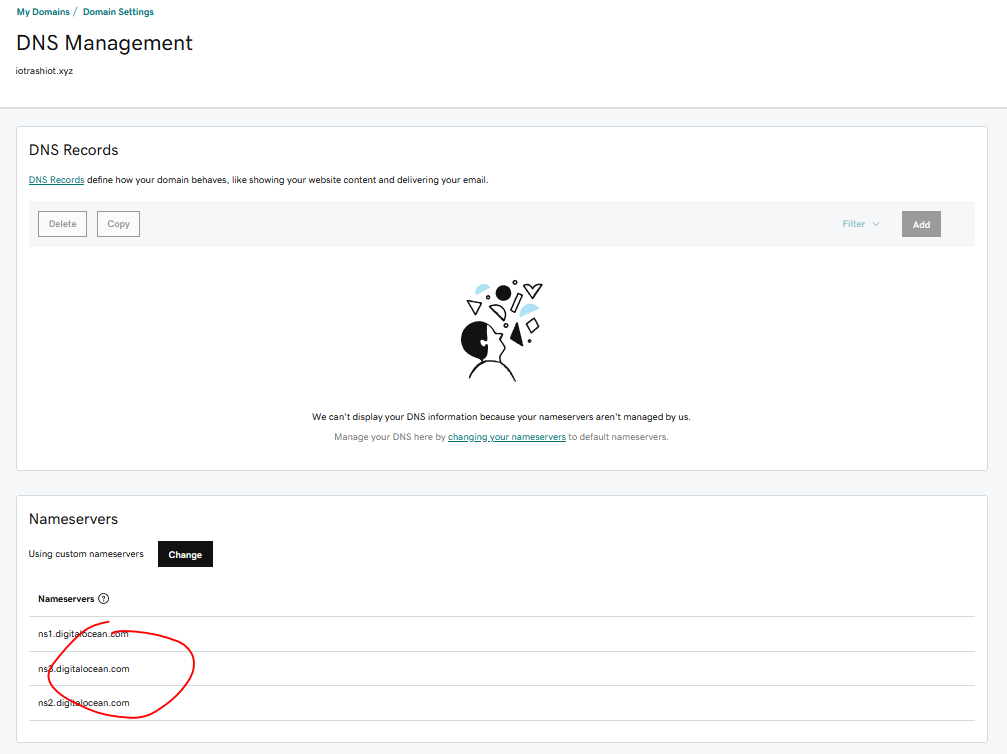
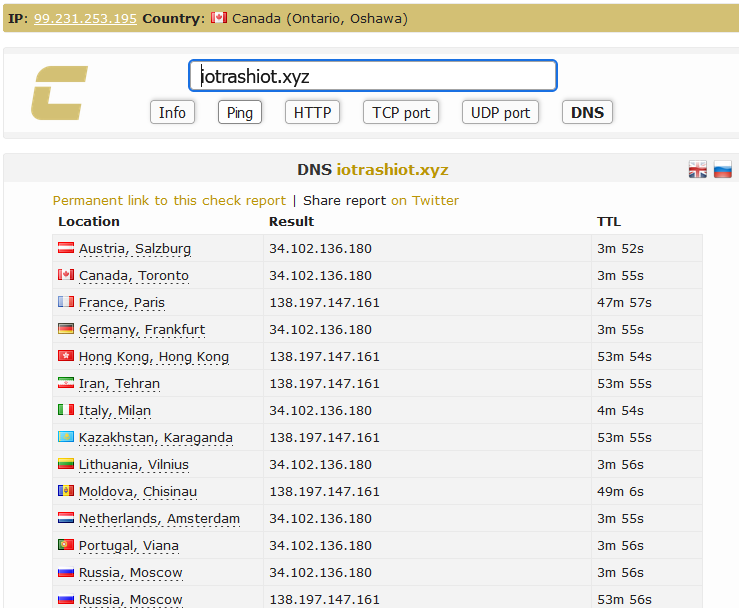
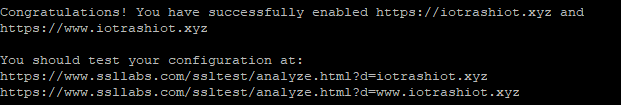


Image3: Waiting for nameserver and record changes to be applied



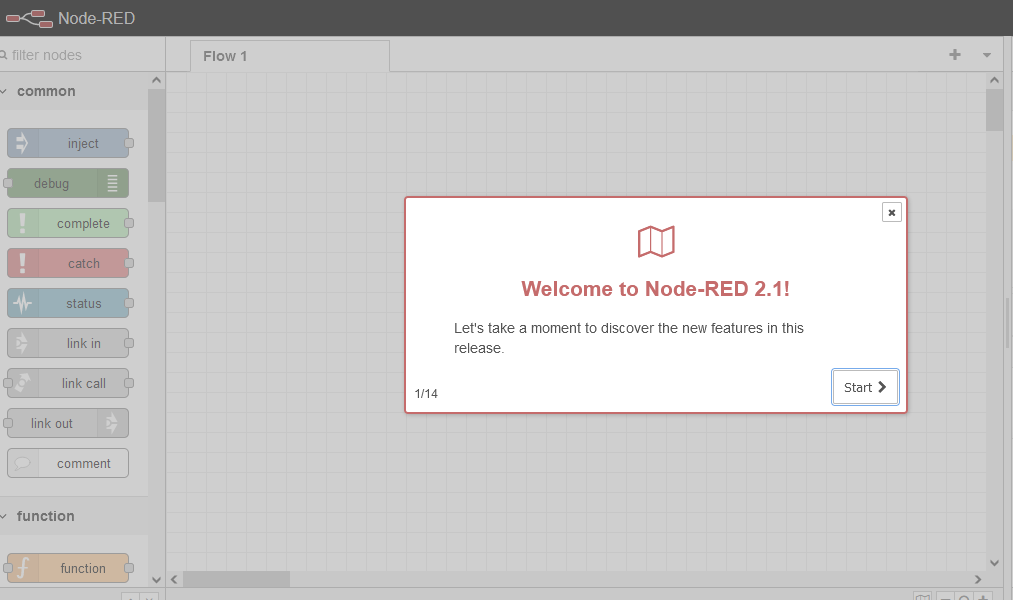
In order for the communication between the cloud and the microcontroller to be enabled. The end phase of the IoT project supposedly should have the capability to asynchronously update and communicate with the database and MQTT-client-server enabled connection.

SSL certificates were also installed using Cerbot.





This is where we had some issues in applying this implementation. Somewhere around the part of using NodeRED as a dashboard. Securing NodeRED was implemented by editing settings.js and adding parameters to allow log in based on set level



We werent able to establish a connection form our MQTT Broker with the

In the dedicated droplet, Django application was created and is used to be hosted on the Virtual Host port 80. Note that the Django application was also changed to use a mysql database instead of using its native sqlite.

## Stage 3: Application Layer

We faced errors with establishing connection with MQTT as the message broker. As it is one of the critical components for the project, our team was not able to move forward with achieving the end-goal IoT platform.