Proyecto COVID-19

(Álvaro Caño Soto y Claudia Trancón Jiménez)

ÍNDICE

---

1. ¿EN QUÉ CONSISTE?

Hace dos años, una de las pandemias más devastadoras que se recuerdan arrasó por todo el mundo provocando infinidad de daños, tanto materiales como personales. En España, los ciudadanos vivieron un confinamiento de tres meses aproximadamente, algo insólito y que reflejaba la magnitud de la situación. Actualmente en España, el virus ha contagiado a más de 12,5 millones de personas y se ha cobrado la vida de más de 107.000 fallecidos, y pese a que todo hace indicar que la enfermedad está dando sus últimos coletazos, el temor de que aparezca una nueva cepa que rompa con todos los esquemas está más que presente en la comunidad científica.

Como todo el mundo sabe, es mejor prevenir que curar, y ahí es donde entra en escena nuestro proyecto. Supongamos que estamos en una habitación con más personas, aunque aparentemente ninguno presente síntomas, pueden ser portadores del virus. Está demostrado que con la suficiente ventilación, el riesgo de contraer el virus es muy bajo, pero cuando esta situación no se cumple, un simple contacto puede volverse muy peligroso. Nuestra aplicación cuenta con un sensor de CO2 (cuanto más CO2, peor es la ventilación) y un led que se encenderá cuando se detecte que los valores de CO2 en la habitación pueden ser favorables para la propagación del virus.

2. ELEMENTOS FÍSICOS

- Tres Módulo ESP 8266 ESP-12F WiFi con CH340 (Cliente):

Permite a los microcontroladores conectarse a una red WiFi y realizar conexiones TCP/IP.

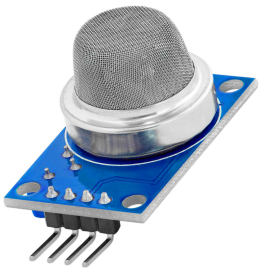
En nuestro caso, utilizaremos dos para sensores y una para actuador.

https://www.amazon.es/gp/product/B074Q27ZBQ/ref=ppx_yo_dt_b_asin_title_o03_s00?ie=UTF8&th=1

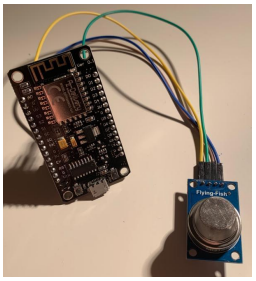
- Dos Módulo sensor CO2 MQ135:

El sensor utilizado es electroquímico, internamente contiene un calentador encargado de aumentar la temperatura interna, y dentro tiene un material que reacciona con los gases del exterior, provocando un cambio en la resistencia.

Debido al calentador, debemos esperar un tiempo de calentamiento para que la salida sea estable.



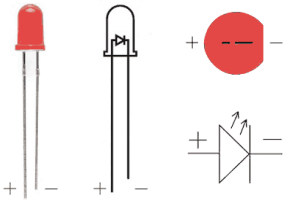
Montaje:



https://www.amazon.es/AZDelivery-sensor-Calidad-M%C3%B3dulo-Arduino/dp/B07CNR9K8P/ref=sr_1_1_sspa?__mk_es_ES=%C3%85M%C3%85%C5%BD%C3%95%C3%91&crid=3E1PTWEM8DNTA&keywords=mq135&qid=1655075334&sprefix=mq135%2Caps%2C88&sr=8-1-spons&smid=A1X7QLRQH87QA3&spLa=ZW5jcnlwdGVkUXVhbGlmaWVyPUEyV0xWSjNBQThFU1hTJmVuY3J5cHRlZElkPUEwMTQxNTQxMjkxSk9GREpGV1VIViZlbmNyeXB0ZWRBZElkPUEwODkxNjk3MUFXSlNHOEc3WE5OTSZ3aWRnZXROYW1lPXNwX2F0ZiZhY3Rpb249Y2xpY2tSZWRpcmVjdCZkb05vdEx0NsaWNrPXRydWU&th=1

- Un led :

Utilizamos un led como actuador. Cátodo (-) a tierra y ánodo(+) al voltaje positivo.



- Dispositivo móvil, utilizado como módulo GPS, con sistema Android.

3. ESTRUCTURA

3.1. BASE DE DATOS

Estas tablas han sido definidas como clases en nuestro proyecto Java, a su vez en MySQL Workbench.

- Concentracion: Se van almacenando loa valores obtenidos por el sensor de CO2.

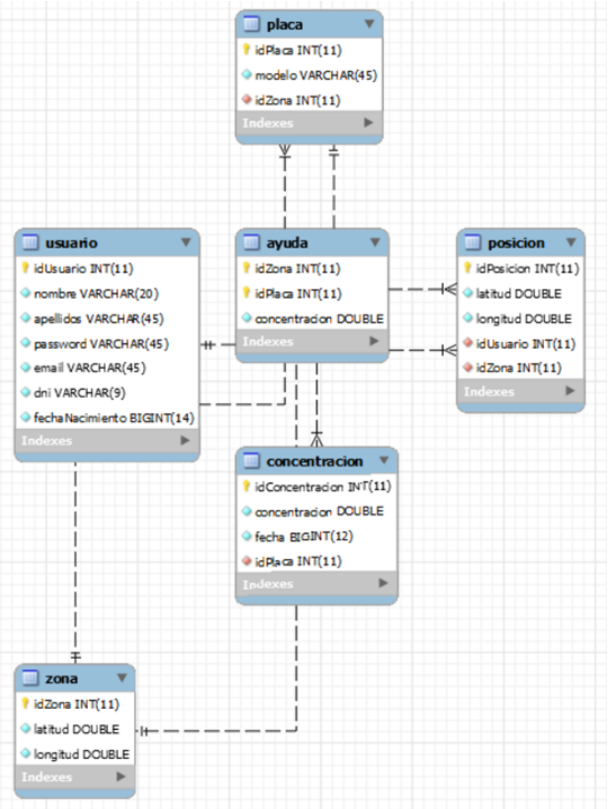
- Placa: Se almacena la información correspondiente a cada placa.

- Posicion: Tabla en la que se insertan los valores del sensor GPS.

- Usuario: Contiene la información de cada usuario.

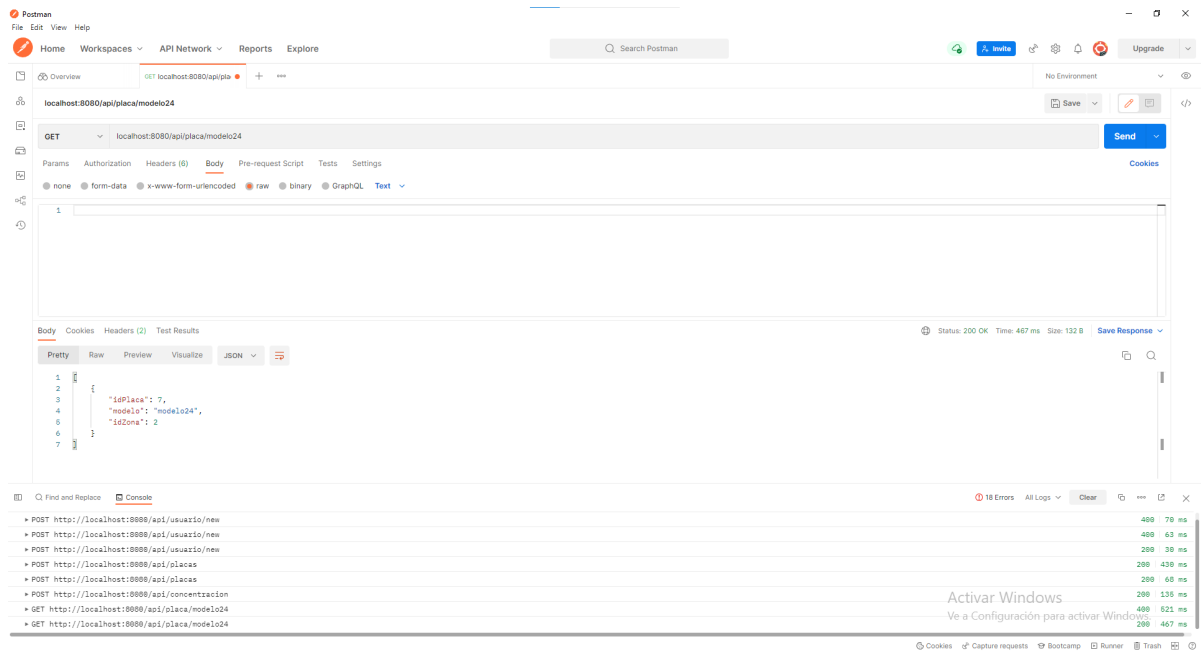
- Zona: Almacena los datos de cada zona.

- Ayuda: Tabla auxiliar que hemos creado para simplificar algunas consultas a la BD.

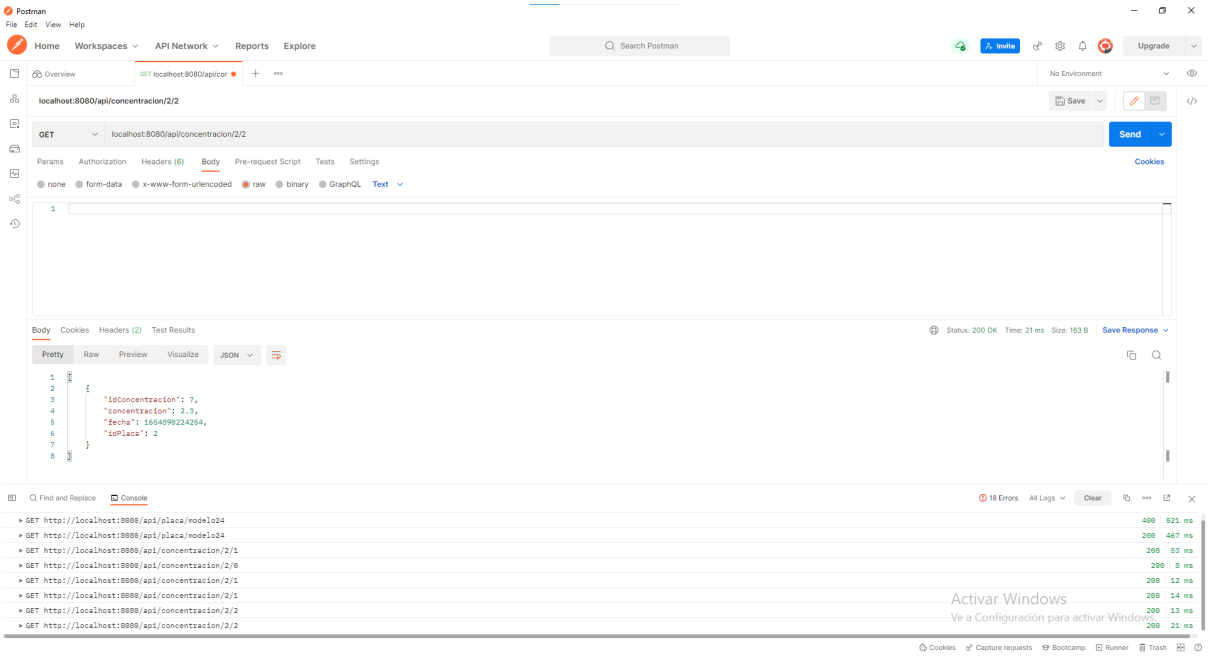


3.2. API REST (CAPTURAS REALIZADAS CON POSTMAN)

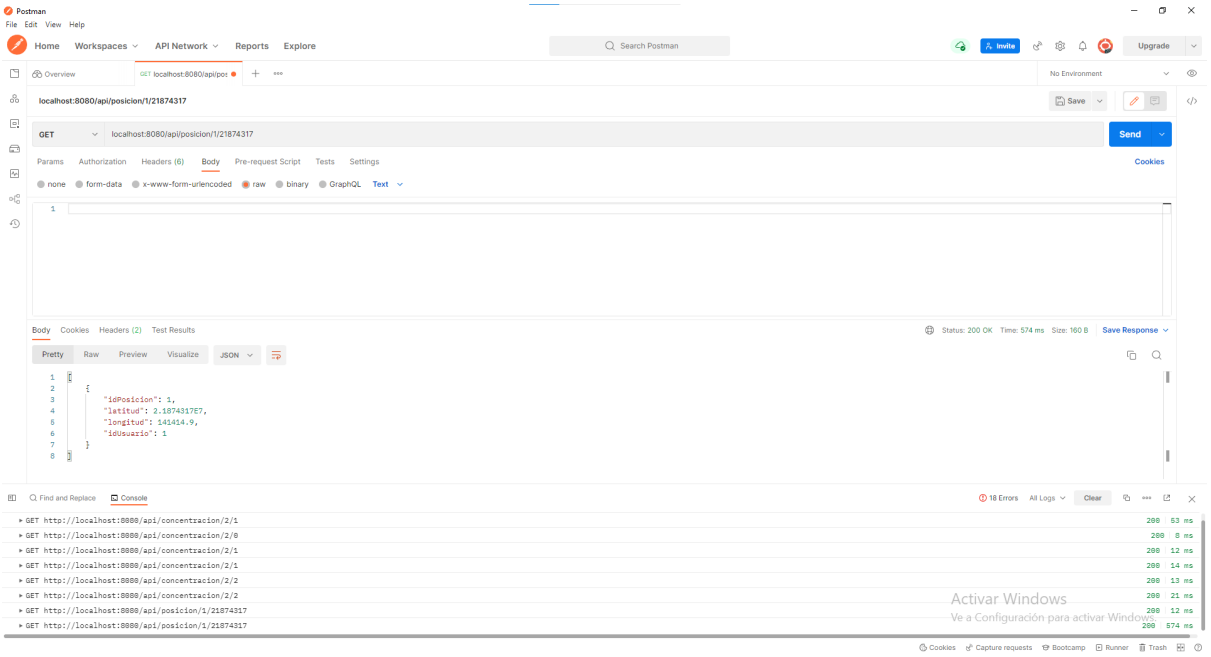
- Métodos GET:
    - getOnePlaca



    - getOneConcentracion

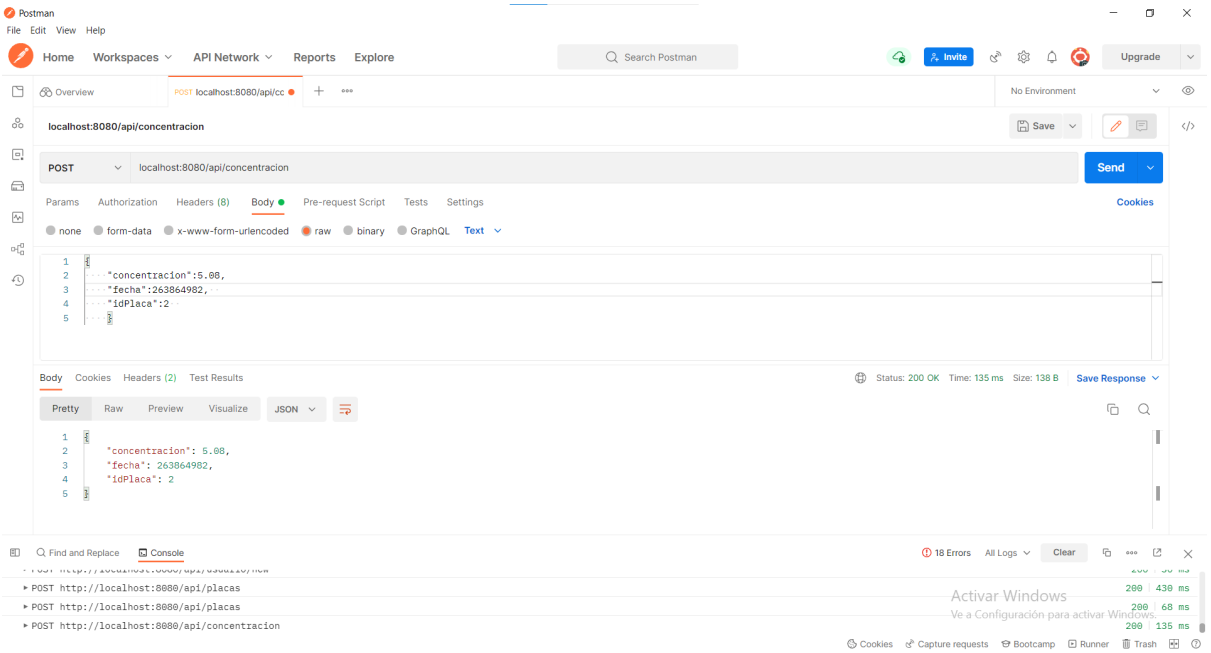


- getOnePosicion

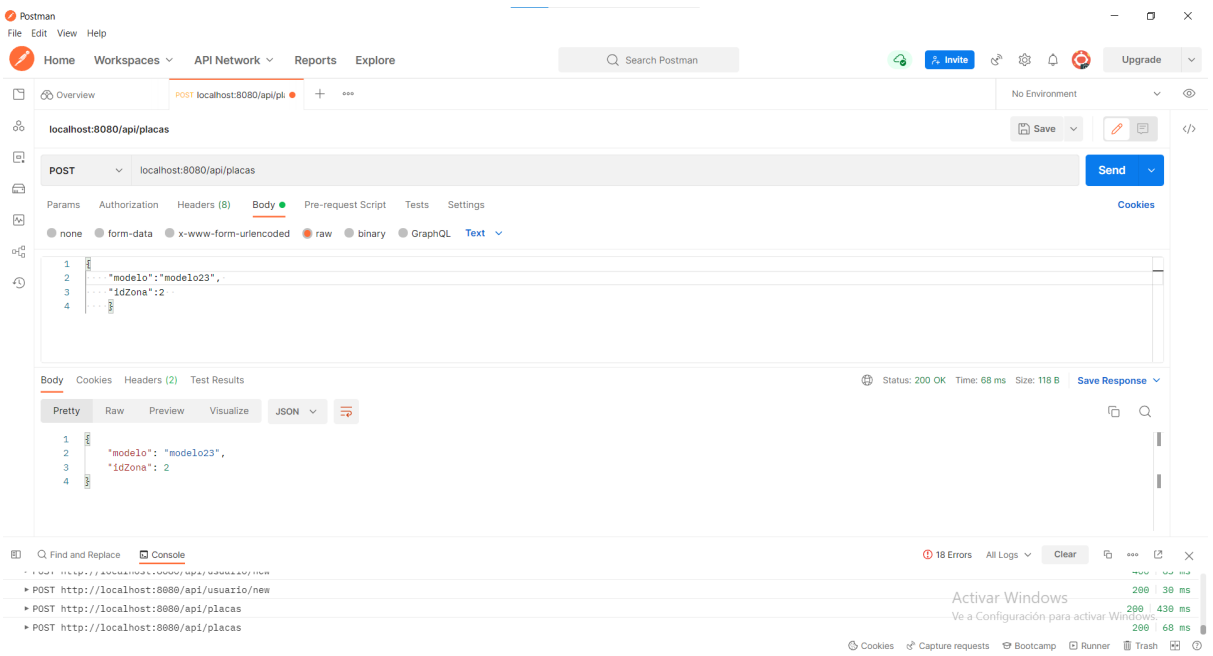


- Métodos POST:

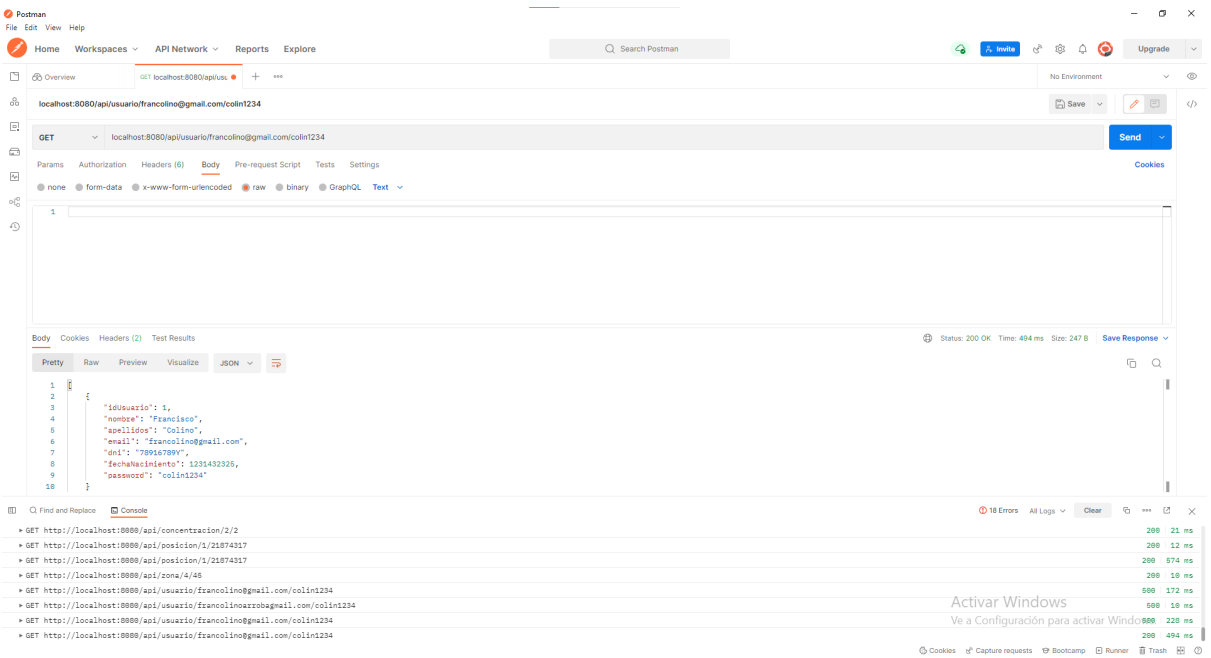
- addOneConcentracion



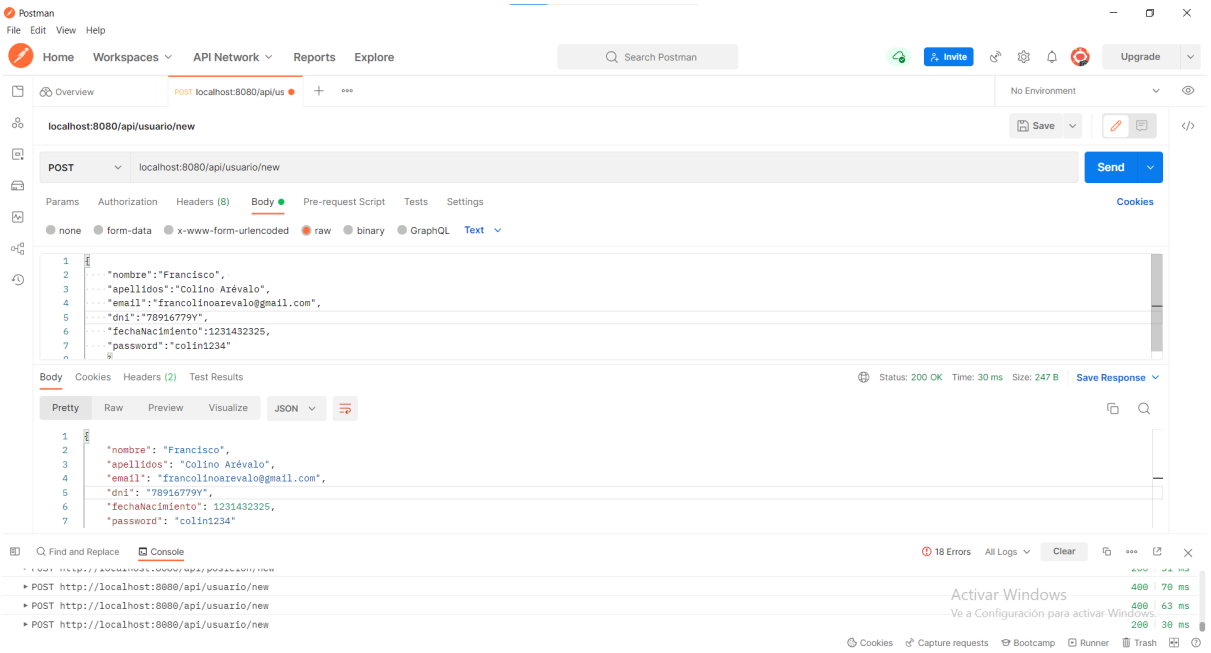
- addOnePlaca



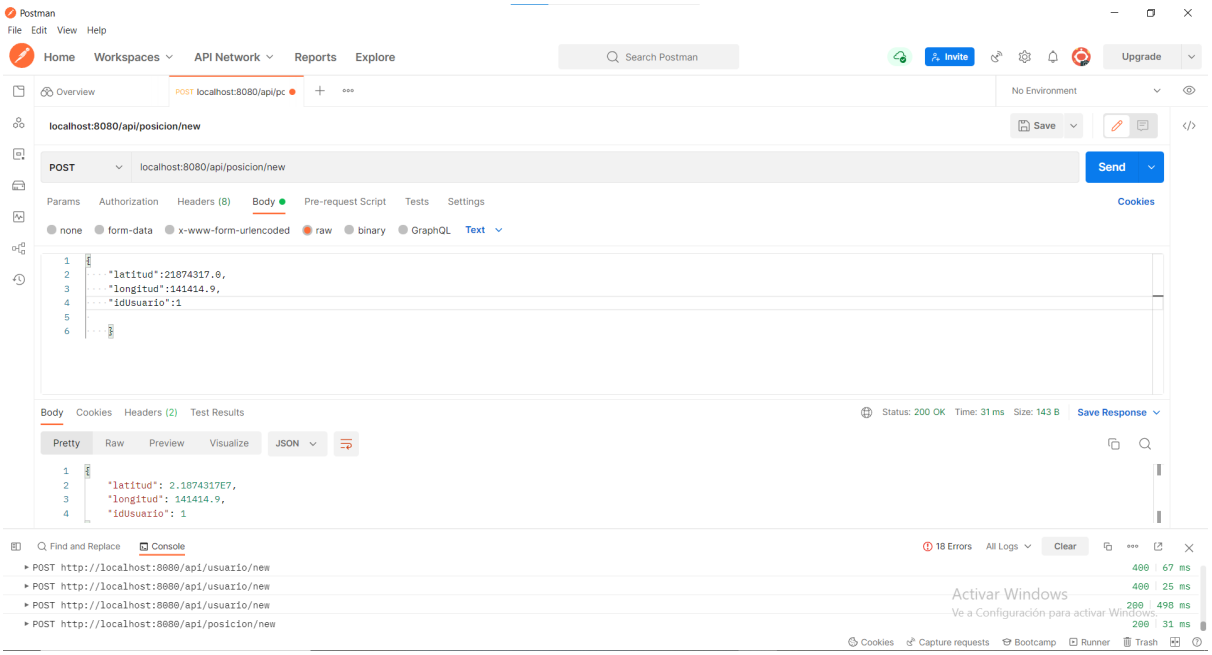
- getUserLogged



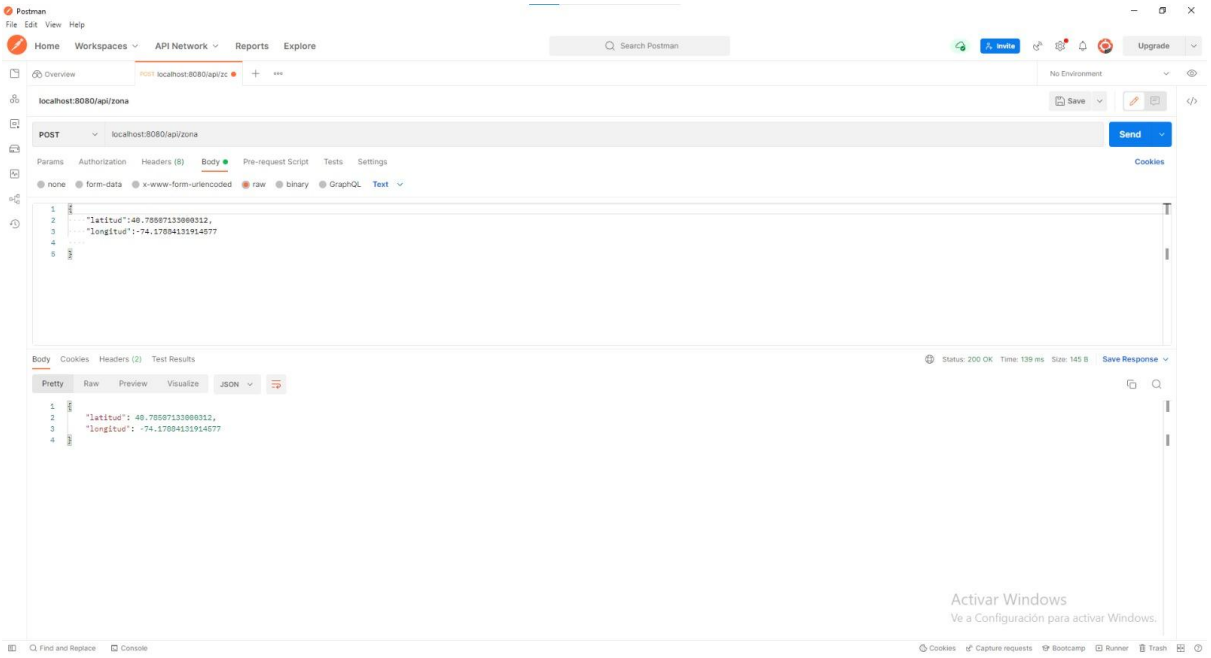
- postNewUser



- postNewPosicion

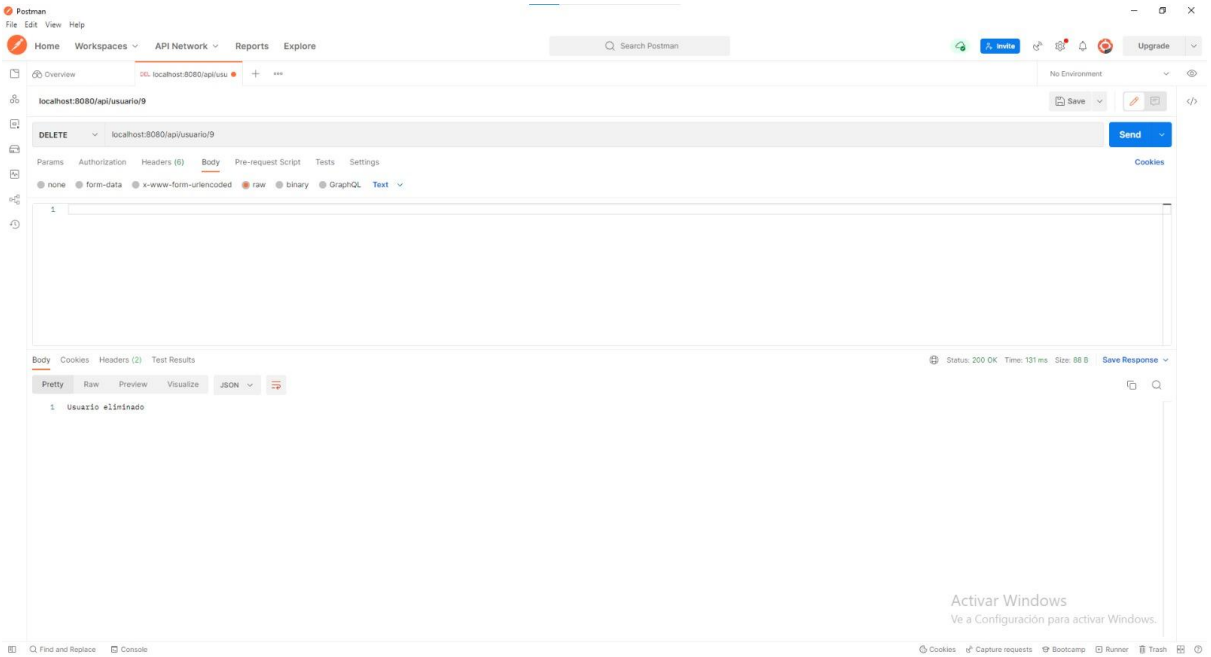


- postNewZona

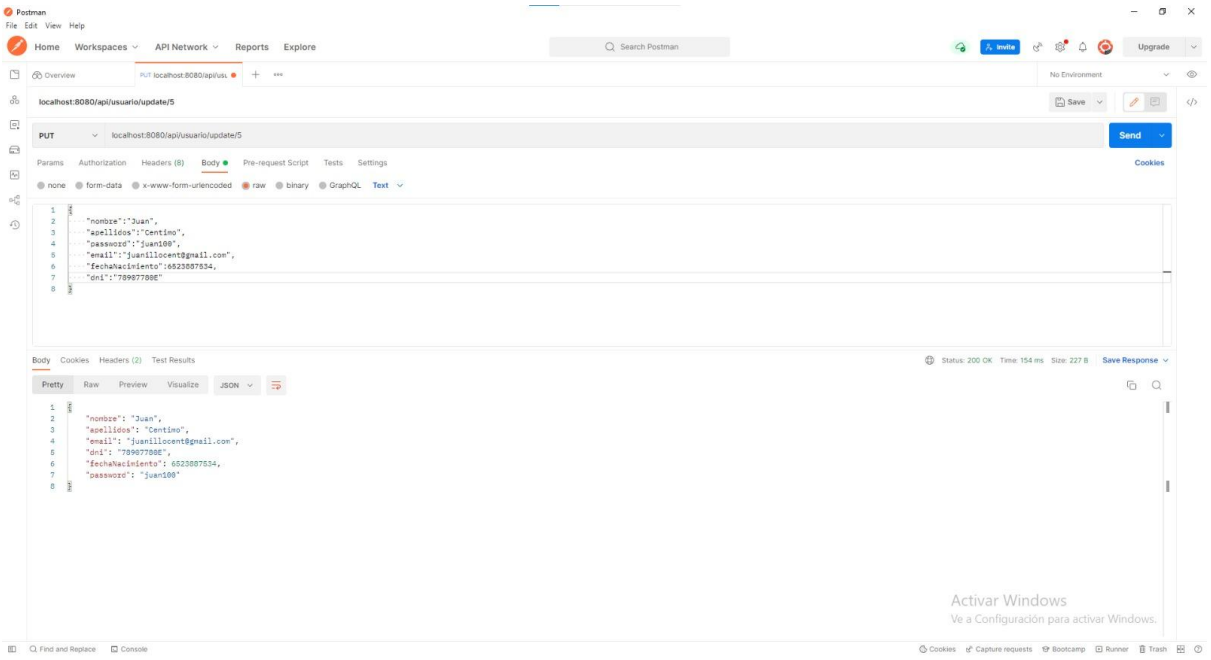


- Métodos DELETE:

- deleteUser



- Métodos PUT
    - putUser



3.3. CÓDIGOS CLIENTES

Realizado en el Visual Studio Code, con el plugins de Platform.io. Utilizando varias librerías externas.

- Main.cpp Sensor: Realiza la conexión entre la placa del sensor y la APIRest. Importar librería para él sensor MQ135 `<Adafruit_Sensor.h>`.
- Main.cpp Actuador: Realiza la conexión entre la placa del actuador y la APIRest.

3.4. MQTT

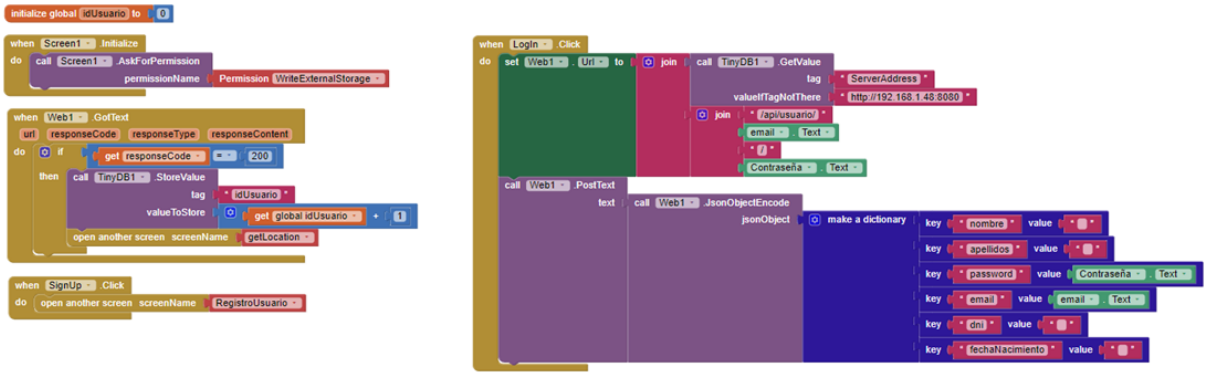
Utilizamos MQTT para hacer las conexiones cliente - servidor, y servidor - cliente. Definimos “topic_1” para enviar los mensajes entre el cliente y el servidor y así poder activar el actuador cuando corresponda, es decir, cuando los valores de concentración de una zona, superen el umbral establecido

3.5. APP ANDROID

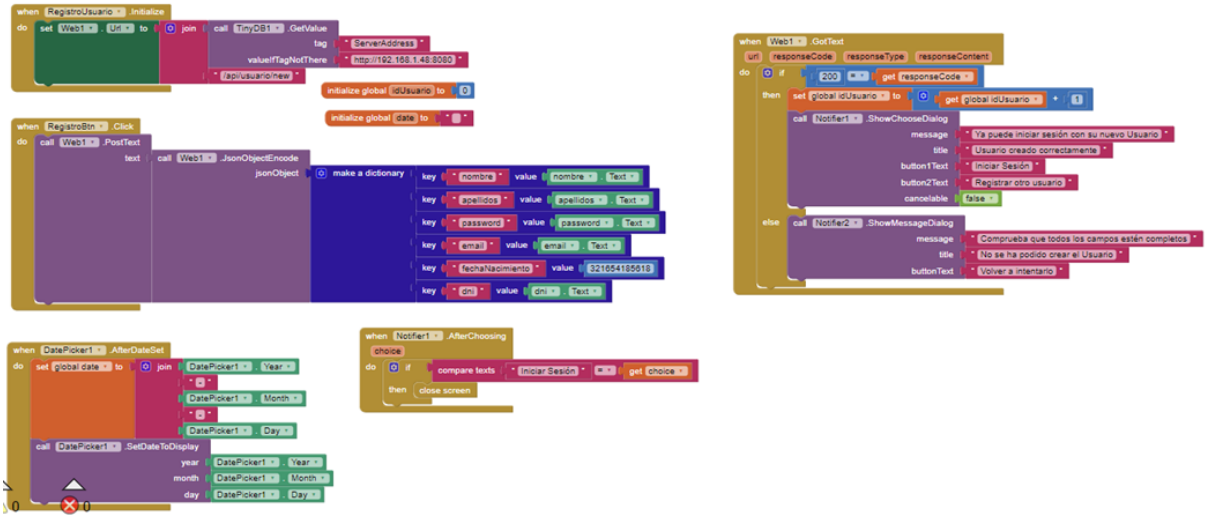
Para la funcionalidad del GPS y los usuarios hemos creado una aplicación Android con la herramienta MIT APP Inventor, utilizando tinyDB para almacenar la información (la cual se formatea como mensajes JSON) y un elemento web para hacer las peticiones POST enviando los datos almacenados en tinyDB.

Creado con MIT App Inventor, https://appinventor.mit.edu/ .

Login usuario: Realiza un POST pasando el email y la contraseña en el cuerpo y utiliza esos campos para acceder al usuario de la BD correspondiente



Registro usuario: Realiza un POST pasando todos los campos del usuario menos el idUsuario y lo inserta en la tabla usuario de la BD



Sensor GPS: Obtiene las coordenadas en la que se encuentra el usuario mediante POST las almacena en la tabla posición en la BD