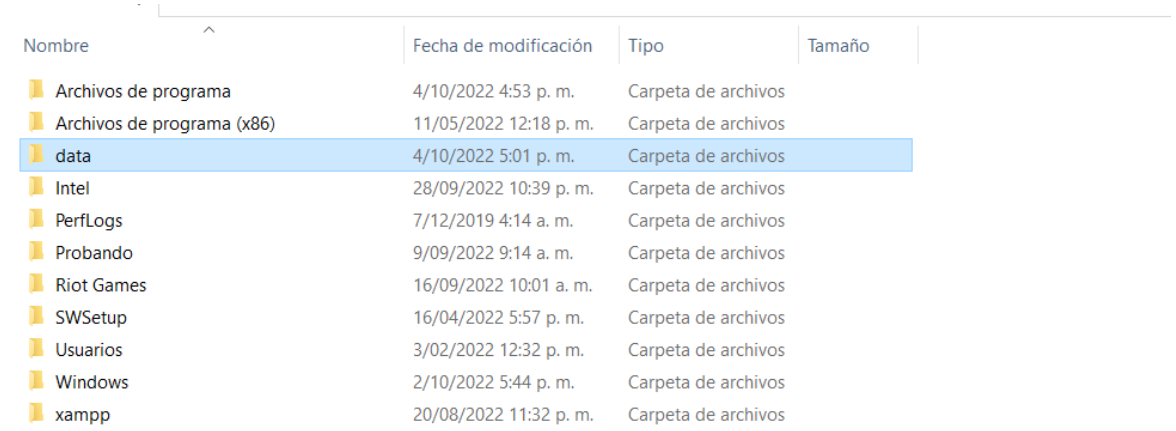
Taller 7b. Servidor IoT-MongoDB

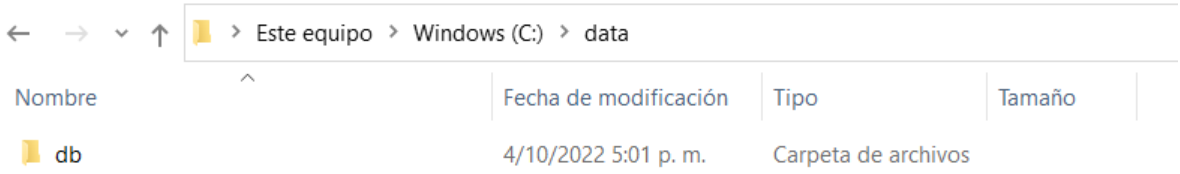
Diego Iván Perea Montealegre (2185751) diego.perea@uao.edu.co

Facultad de Ingeniería, Universidad Autónoma de Occidente

Cali, Valle del Cauca

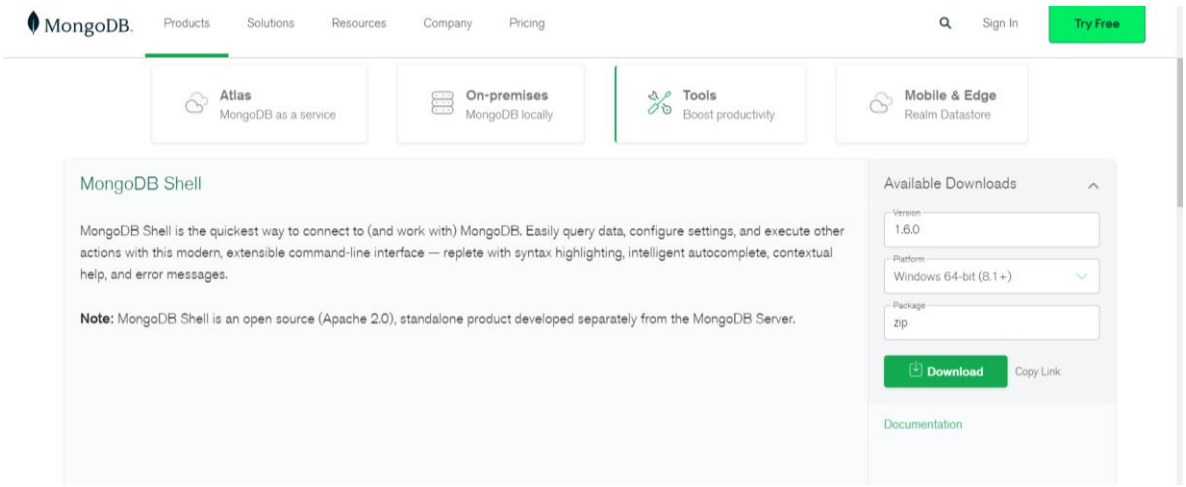


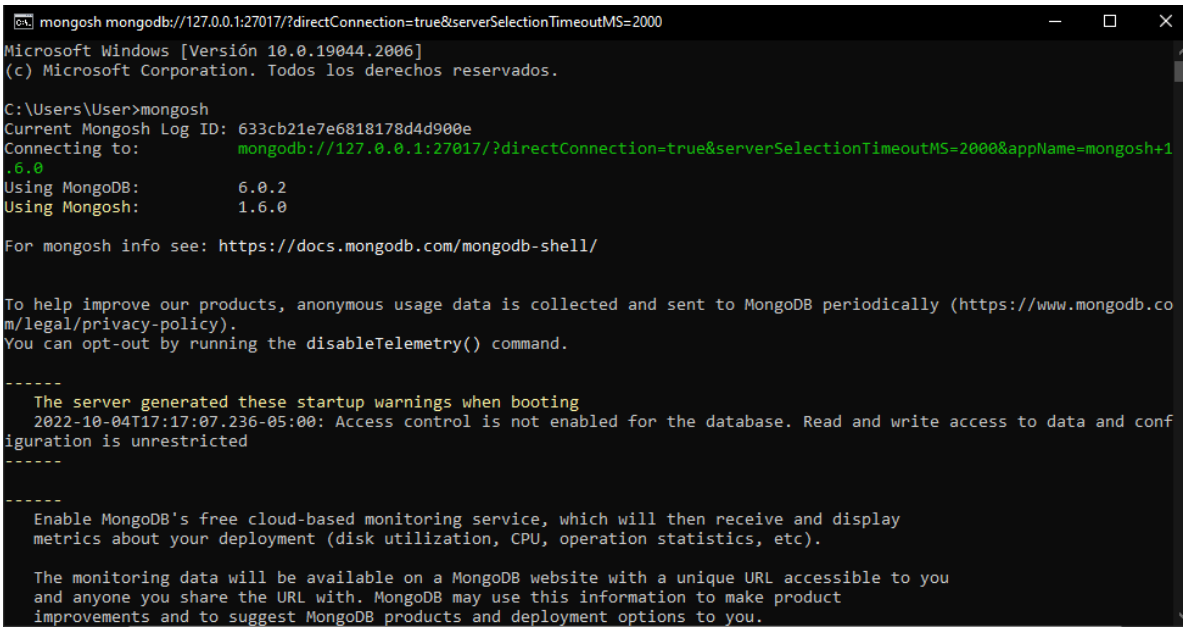
Nombre	Fecha de modificación	Tipo	Tamaño
Archivos de programa	4/10/2022 4:53 p. m.	Carpeta de archivos	
Archivos de programa (x86)	11/05/2022 12:18 p. m.	Carpeta de archivos	
data	4/10/2022 5:01 p. m.	Carpeta de archivos	
Intel	28/09/2022 10:39 p. m.	Carpeta de archivos	
PerfLogs	7/12/2019 4:14 a. m.	Carpeta de archivos	
Probando	9/09/2022 9:14 a. m.	Carpeta de archivos	
Riot Games	16/09/2022 10:01 a. m.	Carpeta de archivos	
SWSetup	16/04/2022 5:57 p. m.	Carpeta de archivos	
Usuarios	3/02/2022 12:32 p. m.	Carpeta de archivos	
Windows	2/10/2022 5:44 p. m.	Carpeta de archivos	
xampp	20/08/2022 11:32 p. m.	Carpeta de archivos	

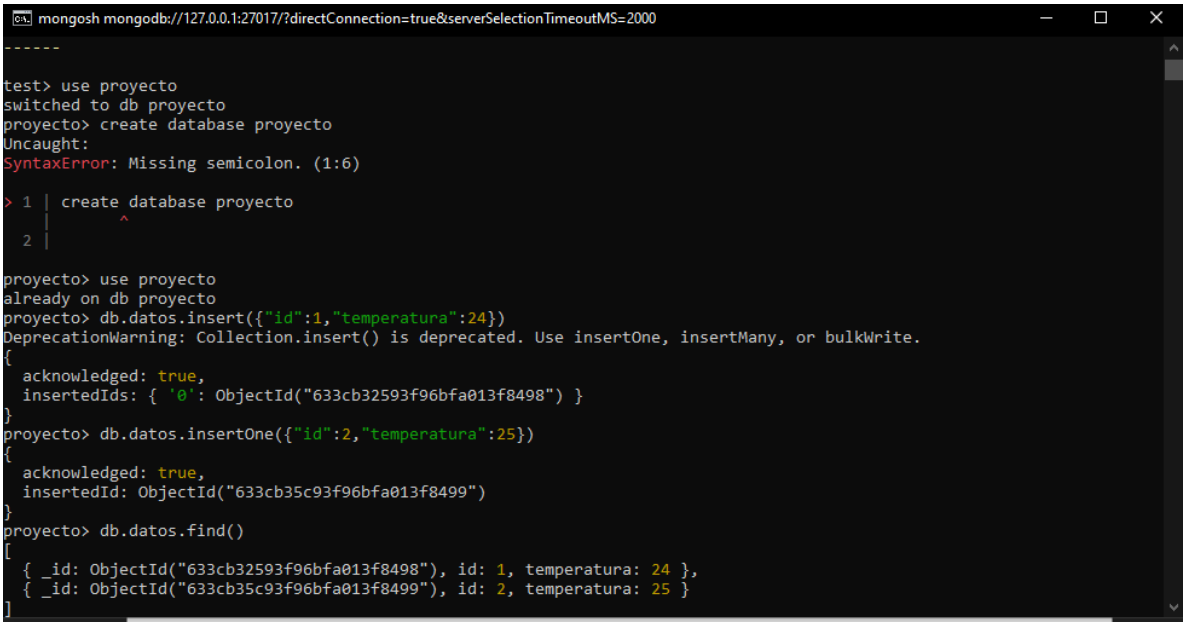


Nombre	Fecha de modificación	Tipo	Tamaño
db	4/10/2022 5:01 p. m.	Carpeta de archivos	

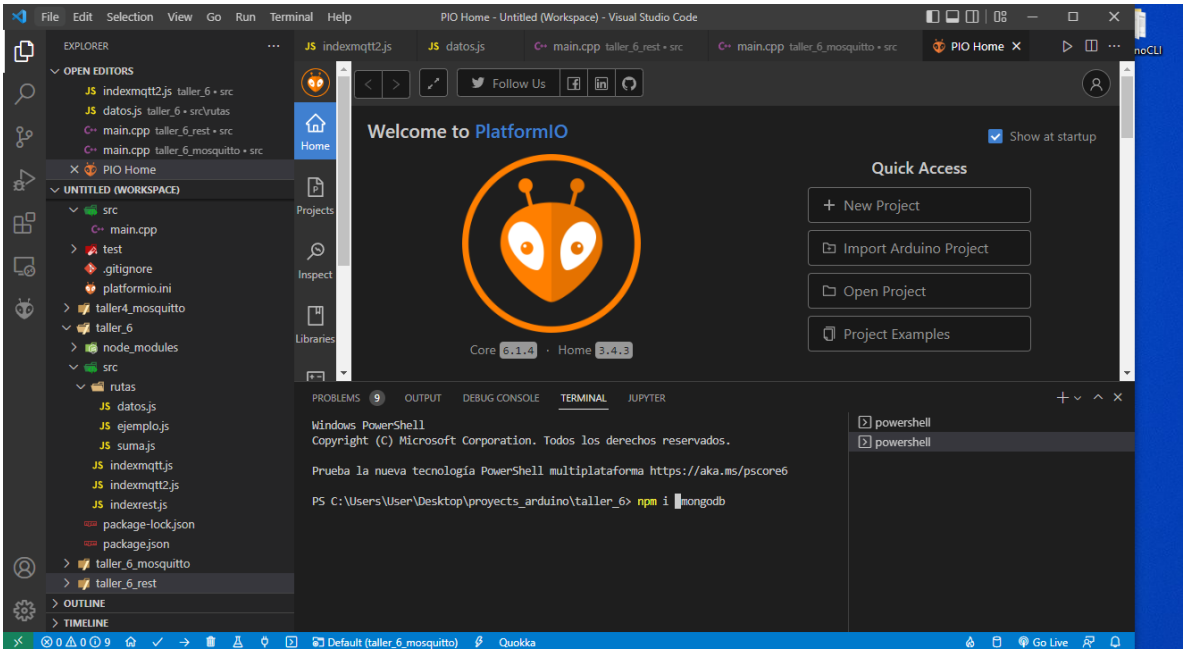
Ingresar a mongodb.com/try/download/shell

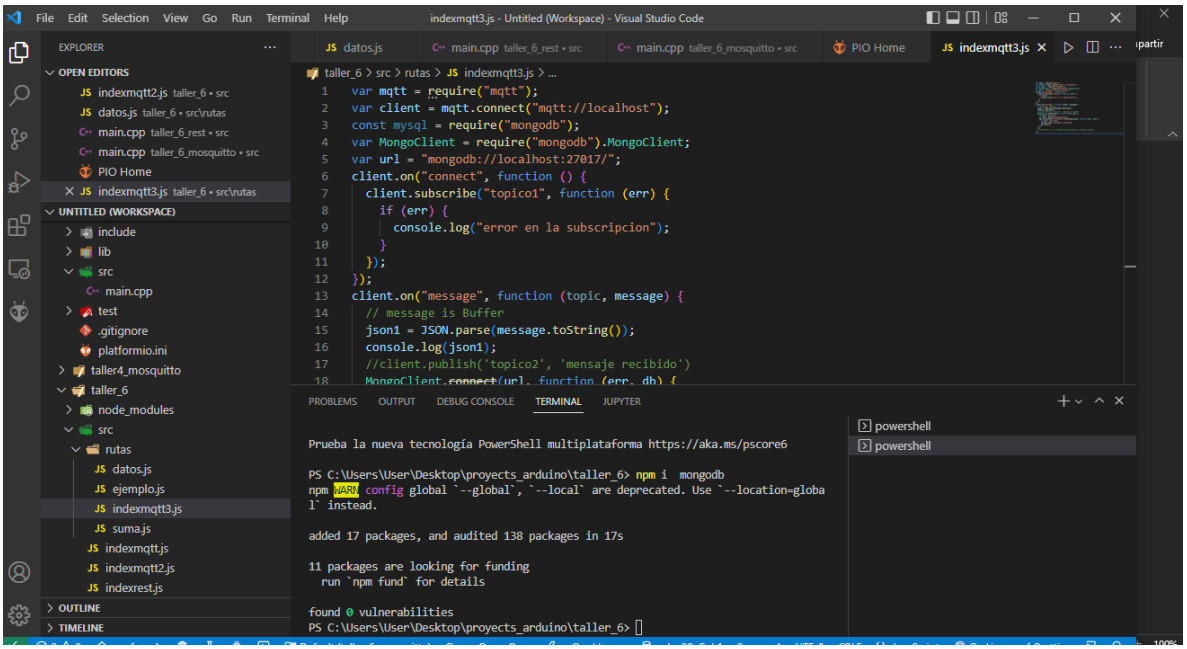


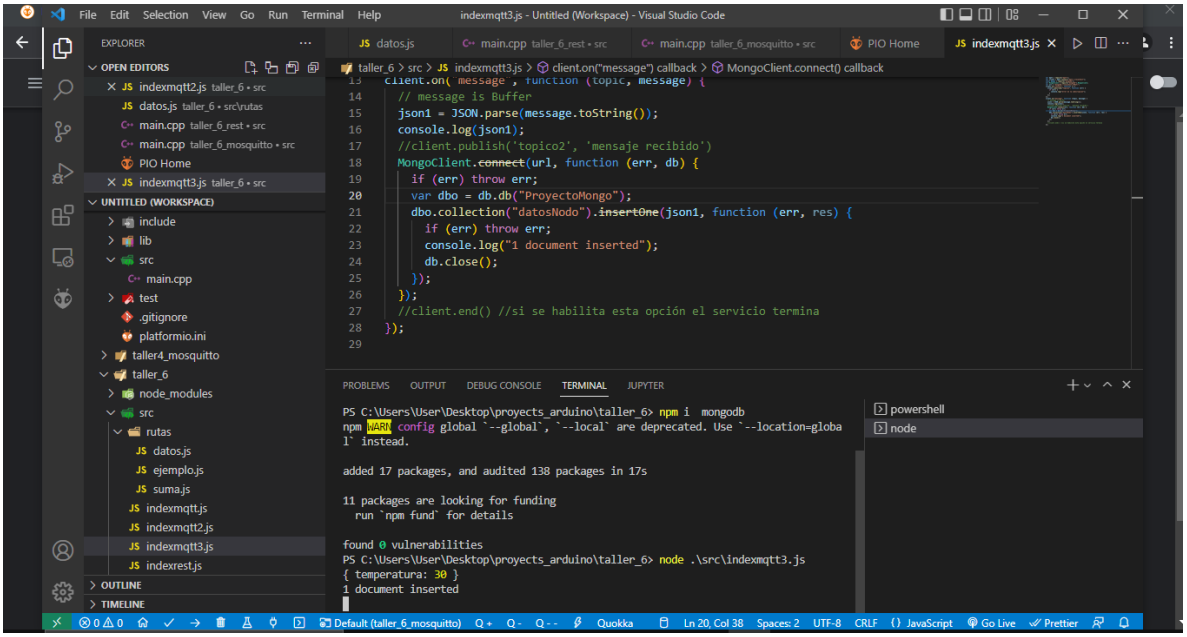

mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
Microsoft Windows [Versión 10.0.19044.2006]
(c) Microsoft Corporation. Todos los derechos reservados.
C:\Users\User>mongosh
Current Mongosh Log ID: 633cb21e7e6818178d4d900e
Connecting to: mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+1.6.0
Using MongoDB: 6.0.2
Using Mongosh: 1.6.0
For mongosh info see: https://docs.mongodb.com/mongodb-shell/
To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (https://www.mongodb.com/legal/privacy-policy).
You can opt-out by running the disableTelemetry() command.
The server generated these startup warnings when booting
2022-10-04T17:17:07.236-05:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
Enable MongoDB's free cloud-based monitoring service, which will then receive and display
metrics about your deployment (disk utilization, CPU, operation statistics, etc).
The monitoring data will be available on a MongoDB website with a unique URL accessible to you
and anyone you share the URL with. MongoDB may use this information to make product
improvements and to suggest MongoDB products and deployment options to you.

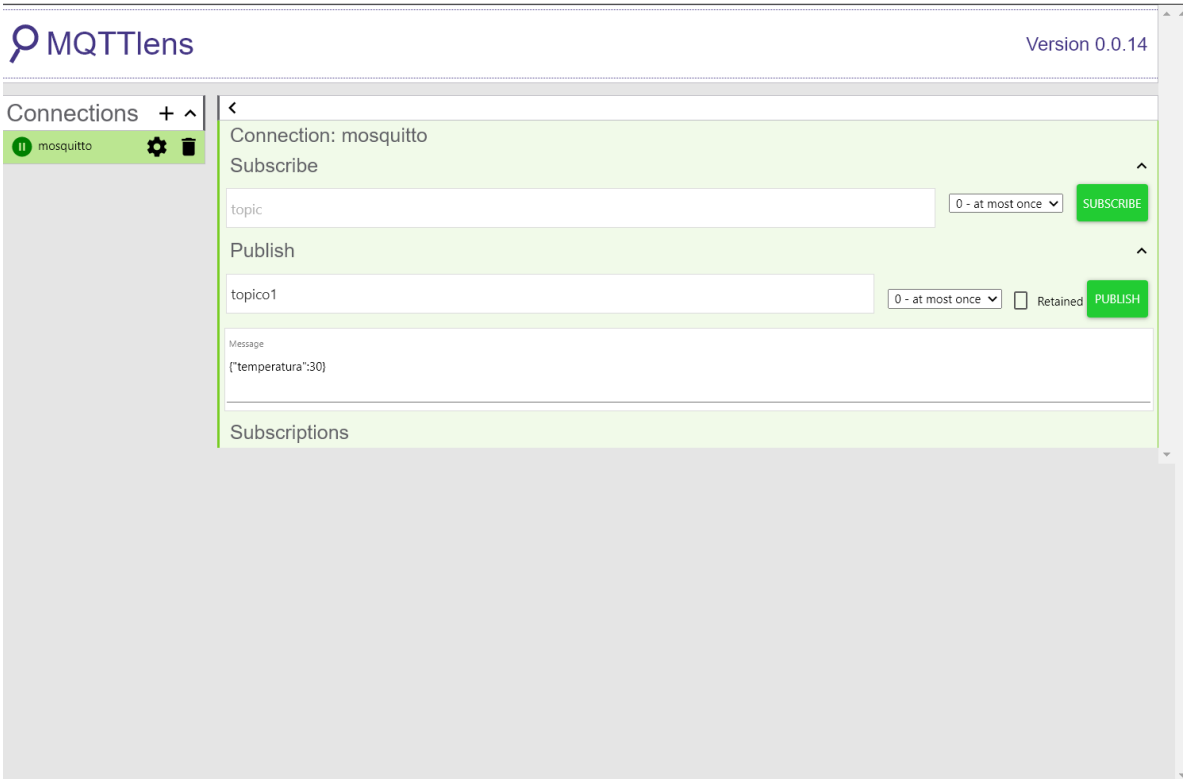

mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
test> use proyecto
switched to db proyecto
proyecto> create database proyecto
Uncaught:
SyntaxError: Missing semicolon. (1:6)
> 1 | create database proyecto
2 |
proyecto> use proyecto
already on db proyecto
proyecto> db.datos.insert({"id":1,"temperatura":24})
DeprecationWarning: Collection.insert() is deprecated. Use insertOne, insertMany, or bulkWrite.
{
acknowledged: true,
insertedIds: { '0': ObjectId("633cb32593f96bfa013f8498") }
}
proyecto> db.datos.insertOne({"id":2,"temperatura":25})
{
acknowledged: true,
insertedId: ObjectId("633cb35c93f96bfa013f8499")
}
proyecto> db.datos.find()
[
{ _id: ObjectId("633cb32593f96bfa013f8498"), id: 1, temperatura: 24 },
{ _id: ObjectId("633cb35c93f96bfa013f8499"), id: 2, temperatura: 25 }
]


mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000
acknowledged: true,
insertedIds: { '0': ObjectId("633cb32593f96bfa013f8498") }
}
proyecto> db.datos.insertOne({"id":2,"temperatura":25})
{
acknowledged: true,
insertedId: ObjectId("633cb35c93f96bfa013f8499")
}
proyecto> db.datos.find()
[
{ _id: ObjectId("633cb32593f96bfa013f8498"), id: 1, temperatura: 24 },
{ _id: ObjectId("633cb35c93f96bfa013f8499"), id: 2, temperatura: 25 }
]
proyecto> db.datos.find("id":1)
Uncaught:
SyntaxError: Unexpected token, expected "," (1:18)
> 1 | db.datos.find("id":1)
| ^
2 |
proyecto> db.datos.find({"id":1})
[
{ _id: ObjectId("633cb32593f96bfa013f8498"), id: 1, temperatura: 24 }
]
proyecto>

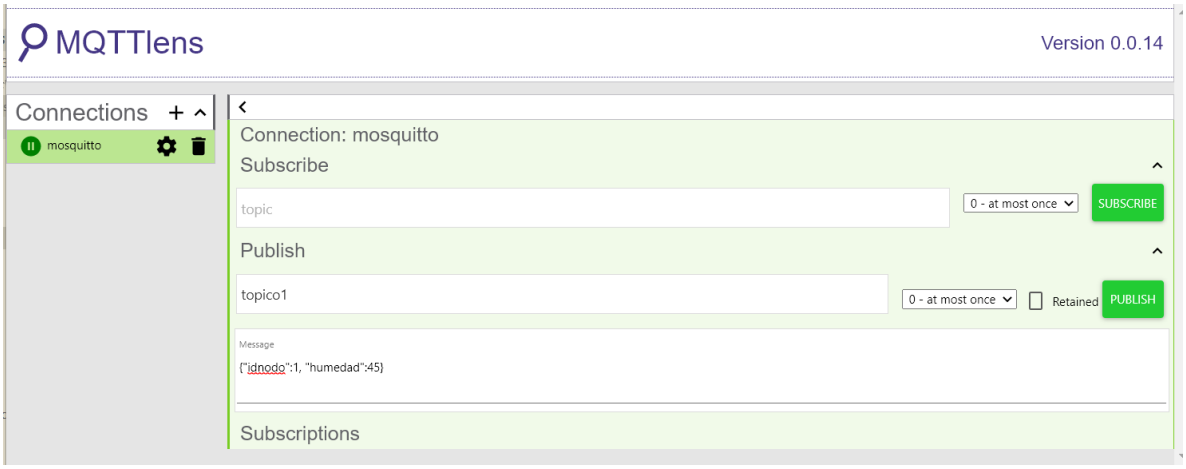

File Edit Selection View Go Run Terminal Help
PIO Home - Untitled (Workspace) - Visual Studio Code
EXPLORER
OPEN EDITORS
indexmqtt2.js taller_6 • src
datos.js taller_6 • src\rutas
main.cpp taller_6_rest • src
main.cpp taller_6_mosquitto • src
PIO Home
UNTITLED (WORKSPACE)
src
main.cpp
test
.gitignore
platformio.ini
taller4_mosquitto
taller_6
node_modules
src
rutas
datos.js
ejemplo.js
suma.js
indexmqtt.js
indexmqtt2.js
indexrest.js
package-lock.json
package.json
taller_6_mosquitto
taller_6_rest
OUTLINE
TIMELINE
Follow Us
Home
Projects
Inspect
Libraries
Welcome to PlatformIO
Show at startup
Quick Access
New Project
Import Arduino Project
Open Project
Project Examples
Core 6.1.4 · Home 3.4.3
PROBLEMS 9
OUTPUT
DEBUG CONSOLE
TERMINAL
JUPYTER
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.
Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6
PS C:\Users\User\Desktop\proyects_arduino\taller_6> npm i mongodb
powershell
powershell
Default (taller_6_mosquitto)
Quokka
Go Live


File Edit Selection View Go Run Terminal Help
indexmqtt3.js - Untitled (Workspace) - Visual Studio Code
EXPLORER
OPEN EDITORS
indexmqtt2.js taller_6 • src
datos.js taller_6 • src\rutas
main.cpp taller_6_rest • src
main.cpp taller_6_mosquitto • src
PIO Home
indexmqtt3.js taller_6 • src\rutas
UNTITLED (WORKSPACE)
include
lib
src
main.cpp
test
.gitignore
platformio.ini
taller4_mosquitto
taller_6
node_modules
src
rutas
datos.js
ejemplo.js
indexmqtt3.js
suma.js
indexmqtt.js
indexmqtt2.js
indexrest.js
OUTLINE
TIMELINE
datos.js
main.cpp taller_6_rest • src
main.cpp taller_6_mosquitto • src
PIO Home
indexmqtt3.js
taller_6 > src > rutas > indexmqtt3.js > ...
1 var mqtt = require("mqtt");
2 var client = mqtt.connect("mqtt://localhost");
3 const mysql = require("mongodb");
4 var MongoClient = require("mongodb").MongoClient;
5 var url = "mongodb://localhost:27017/";
6 client.on("connect", function () {
7 client.subscribe("topico1", function (err) {
8 if (err) {
9 console.log("error en la subscripcion");
10 }
11 });
12 });
13 client.on("message", function (topic, message) {
14 // message is Buffer
15 json1 = JSON.parse(message.toString());
16 console.log(json1);
17 //client.publish('topico2', 'mensaje recibido')
18 MongoClient.connect(url, function (err, db) {
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
powershell
powershell
Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6
PS C:\Users\User\Desktop\proyects_arduino\taller_6> npm i mongodb
npm WARN config global `--global`, `--local` are deprecated. Use `--location=globa
l` instead.
added 17 packages, and audited 138 packages in 17s
11 packages are looking for funding
run `npm fund` for details
found 0 vulnerabilities
PS C:\Users\User\Desktop\proyects_arduino\taller_6>

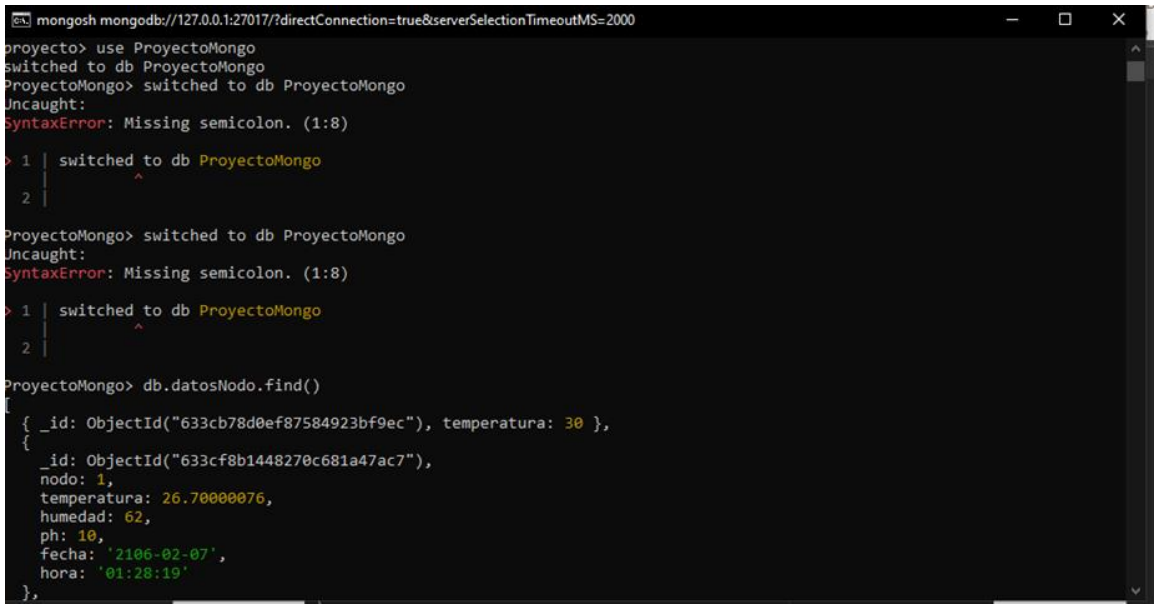

File Edit Selection View Go Run Terminal Help
indexmqtt3.js - Untitled (Workspace) - Visual Studio Code
EXPLORER
OPEN EDITORS
indexmqtt2.js taller_6 • src
datos.js taller_6 • src\rutas
main.cpp taller_6_rest • src
main.cpp taller_6_mosquitto • src
PIO Home
indexmqtt3.js taller_6 • src
UNTITLED (WORKSPACE)
include
lib
src
main.cpp
test
.gitignore
platformio.ini
taller4_mosquitto
taller_6
node_modules
src
rutas
datos.js
ejemplo.js
suma.js
indexmqtt.js
indexmqtt2.js
indexmqtt3.js
indexrest.js
OUTLINE
TIMELINE
taller_6 > src > indexmqtt3.js > client.on("message") callback > MongoClient.connect() callback
// message is Buffer
json1 = JSON.parse(message.toString());
console.log(json1);
//client.publish('topico2', 'mensaje recibido')
MongoClient.connect(url, function (err, db) {
if (err) throw err;
var dbo = db.db("ProyectoMongo");
dbo.collection("datosNodo").insertOne(json1, function (err, res) {
if (err) throw err;
console.log("1 document inserted");
db.close();
});
});
//client.end() //si se habilita esta opción el servicio termina
});
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
powershell
node
PS C:\Users\User\Desktop\proyects_arduino\taller_6> npm i mongodb
npm WARN config global `--global`, `--local` are deprecated. Use `--location=globa
l` instead.
added 17 packages, and audited 138 packages in 17s
11 packages are looking for funding
run `npm fund` for details
found 0 vulnerabilities
PS C:\Users\User\Desktop\proyects_arduino\taller_6> node .\src\indexmqtt3.js
{ temperatura: 30 }
1 document inserted
Default (taller_6_mosquitto) Quokka Ln 20, Col 38 Spaces: 2 UTF-8 CRLF JavaScript Go Live Prettier


MQTTlens
Version 0.0.14
Connections
mosquitto
Connection: mosquitto
Subscribe
topic
0 - at most once
SUBSCRIBE
Publish
topico1
0 - at most once
Retained
PUBLISH
Message
{"temperatura":30}
Subscriptions

Otro:

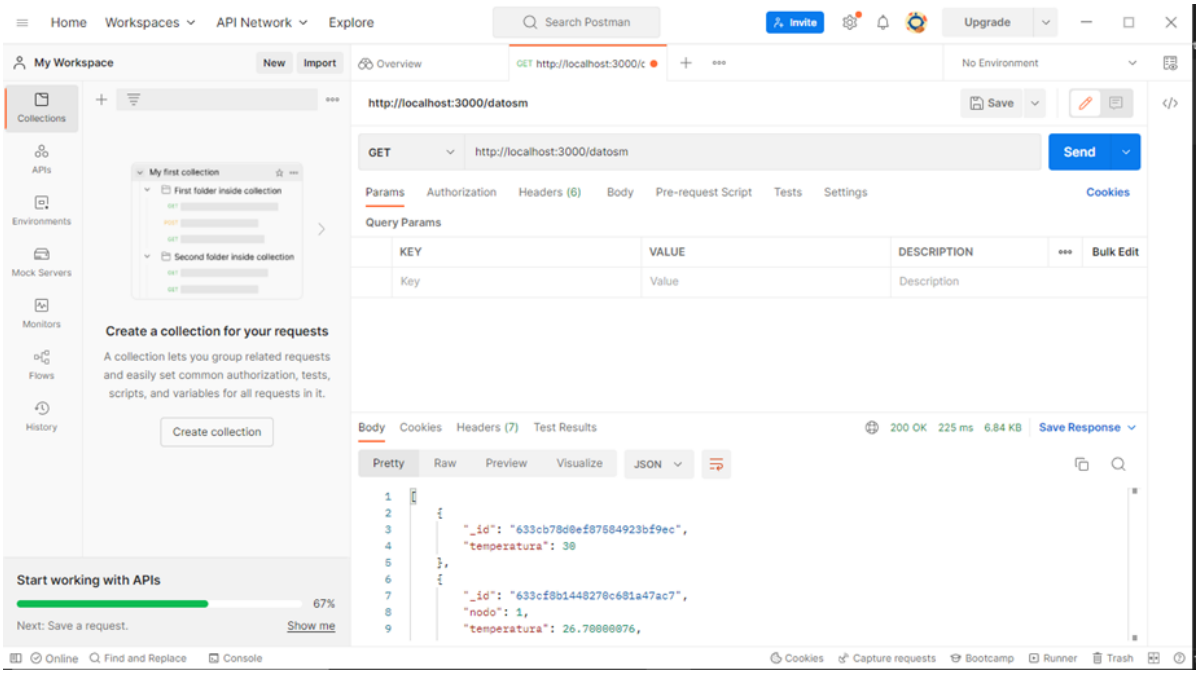

MQTTlens
Version 0.0.14
Connections
mosquitto
Connection: mosquitto
Subscribe
topic
0 - at most once
SUBSCRIBE
Publish
topico1
0 - at most once
Retained
PUBLISH
Message
{"idnodo":1, "humedad":45}
Subscriptions

Comprobación con objeto Iot Huertas Inteligentes

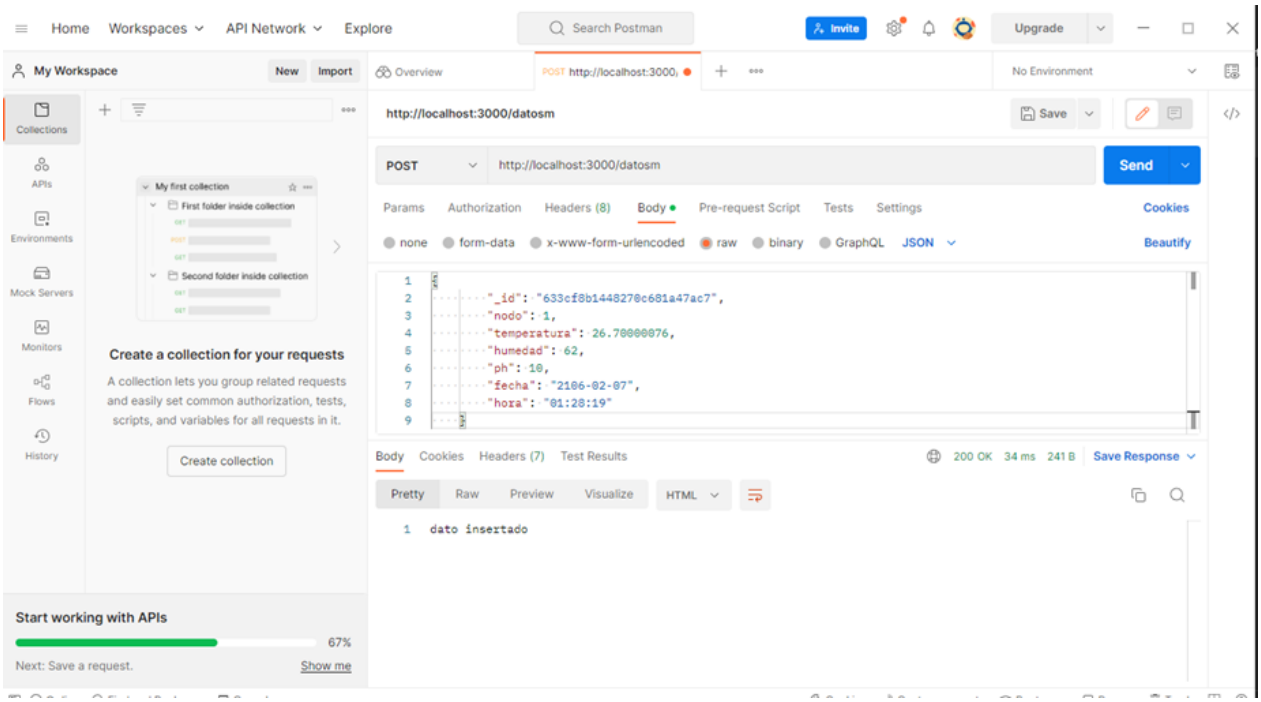


```
{
  _id: ObjectId("633cf8b7448270c681a47ac8"),
  nodo: 1,
  temperatura: 26.70000076,
  humedad: 62,
  ph: 0,
  fecha: '2106-02-07',
  hora: '01:28:25'
},
{
  _id: ObjectId("633cf8bc448270c681a47ac9"),
  nodo: 1,
  temperatura: 26.70000076,
  humedad: 62,
  ph: 0,
  fecha: '2106-02-07',
  hora: '01:28:30'
},
{
  _id: ObjectId("633cf8c1448270c681a47aca"),
  nodo: 1,
  temperatura: 26.70000076,
  humedad: 62,
  ph: 0,
```


Con postman GET



Postman con POST



Código en ESP32:

```
#include <Arduino.h>

#include <ArduinoJson.h>


#include <HTTPClient.h>

//LIBRERIAS PARA DHT11 (TEMPERATURA Y HUMEDAD)

#include <Adafruit_Sensor.h>

#include <DHT.h>

//LIBRERIAS PARA FECHA Y HORA

#include <WiFi.h>


//DEFINICION DE PINES DHT11

#define DHTPIN 14  // 4 = PIN D4

#define DHTTYPE    DHT11

DHT dht(DHTPIN, DHTTYPE);

//potenciometro ph

const int portPin=34;
```

```
int valorPh=0;




const char* ssid = "****your_wifi";//name wifi
const char* password = "***pasword*wifi"; // clave de wifi

void setup_wifi() {
  delay(10);
  // We start by connecting to a WiFi network
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);

  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
  delay(500);
  Serial.print(".");
}
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());

}
void setup() {
  Serial.begin(9600); //Serial connection
```

```
  setup_wifi(); //WiFi connection
  delay(1500);
}



void loop() {

  //temperatura y humedad
  float h= dht.readHumidity();
  float t =dht.readTemperature();
   //potenciometro ph
  valorPh=analogRead(portPin)/292.5;
  //----------------------
  String variable;
  int nodo_numero = 1;
  DynamicJsonDocument doc(1024); //creacion del json
  doc["nodo"] = nodo_numero;
  doc["temperatura"] = t;
  doc["humedad"] = h;
  doc["ph"]=valorPh;
  doc["fecha"] = "lunes";
  doc["hora"] = "3:00 pm";



  serializeJson(doc, variable);
  Serial.println("dato a enviar: "+ variable);
  HTTPClient http; //Declare object of class HTTPClient
  WiFiClient client;
```

```
  //Specify request destination
  //http.begin(client,"URL DEL SERVIDOR");
  //http.begin(client,"http://192.****:3000/"); //para mosquito o mqtt
  //http.begin(client,"http://192.****:3000/datos");// para rest mysql
   http.begin(client,"http://192.****:3000/datosm");// mongo
  http.addHeader("Content-Type", "application/json"); //Specify contenttype header
  int httpCode = http.POST(variable); //Send the request
  String payload = http.getString(); //Get the response payload
  Serial.println(httpCode); //Print HTTP return code
  Serial.println(payload); //Print request response payload
  http.end(); //Close connection
  delay(5000); //Send a request every 5 seconds
}
```