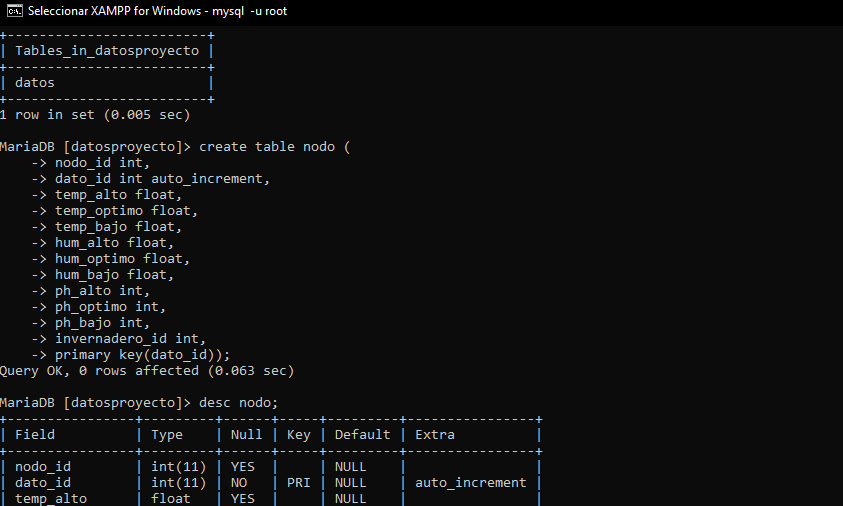
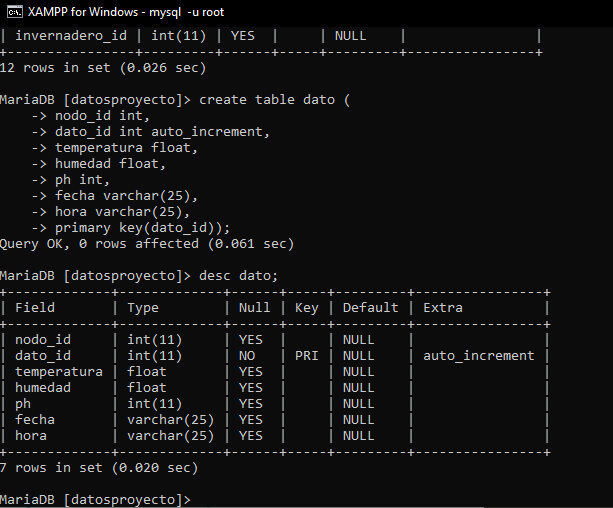
Taller 8. Servidor Remoto

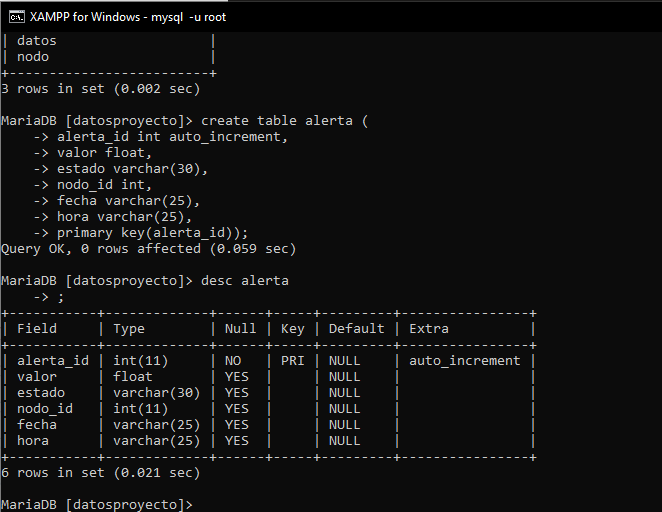
Diego Iván Perea Montealegre (2185751) [diego.perea@uao.edu.co](mailto:diego.perea@uao.edu.co)

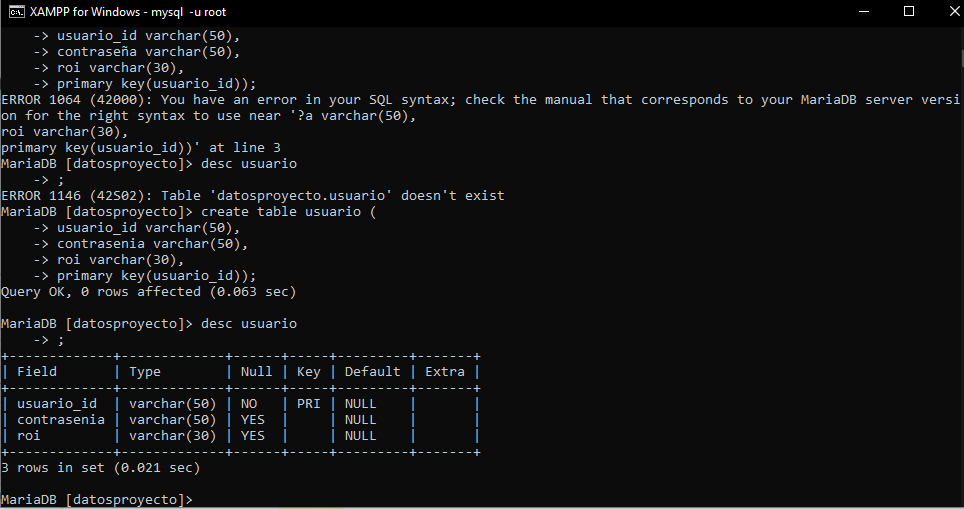
Facultad de Ingeniería, Universidad Autónoma de Occidente

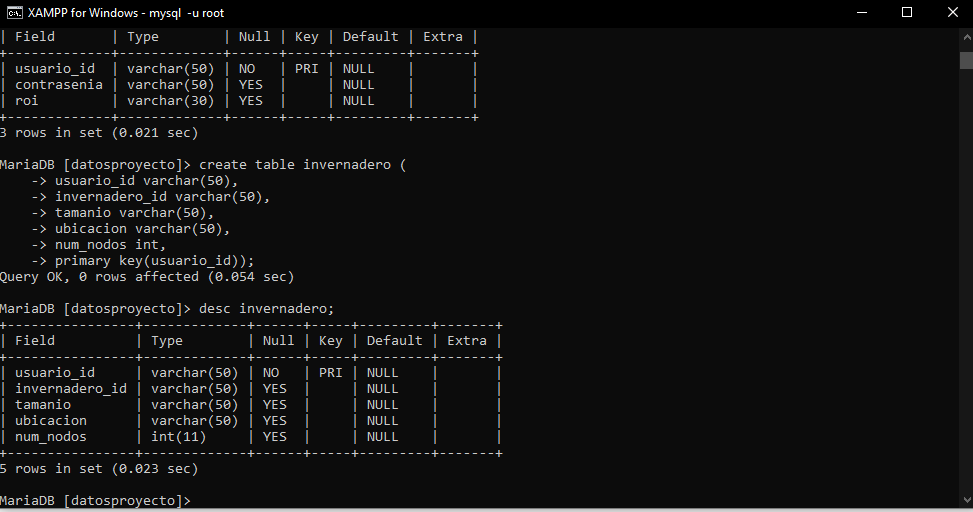
Cali, Valle del Cauca

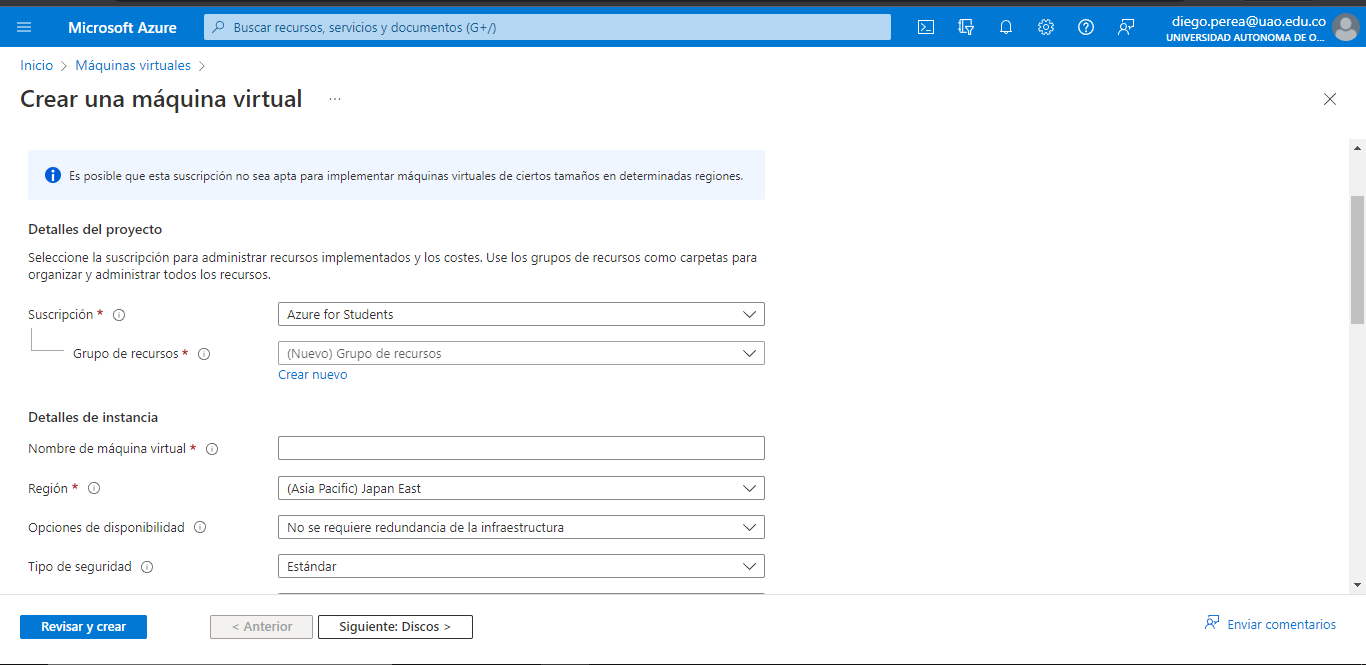


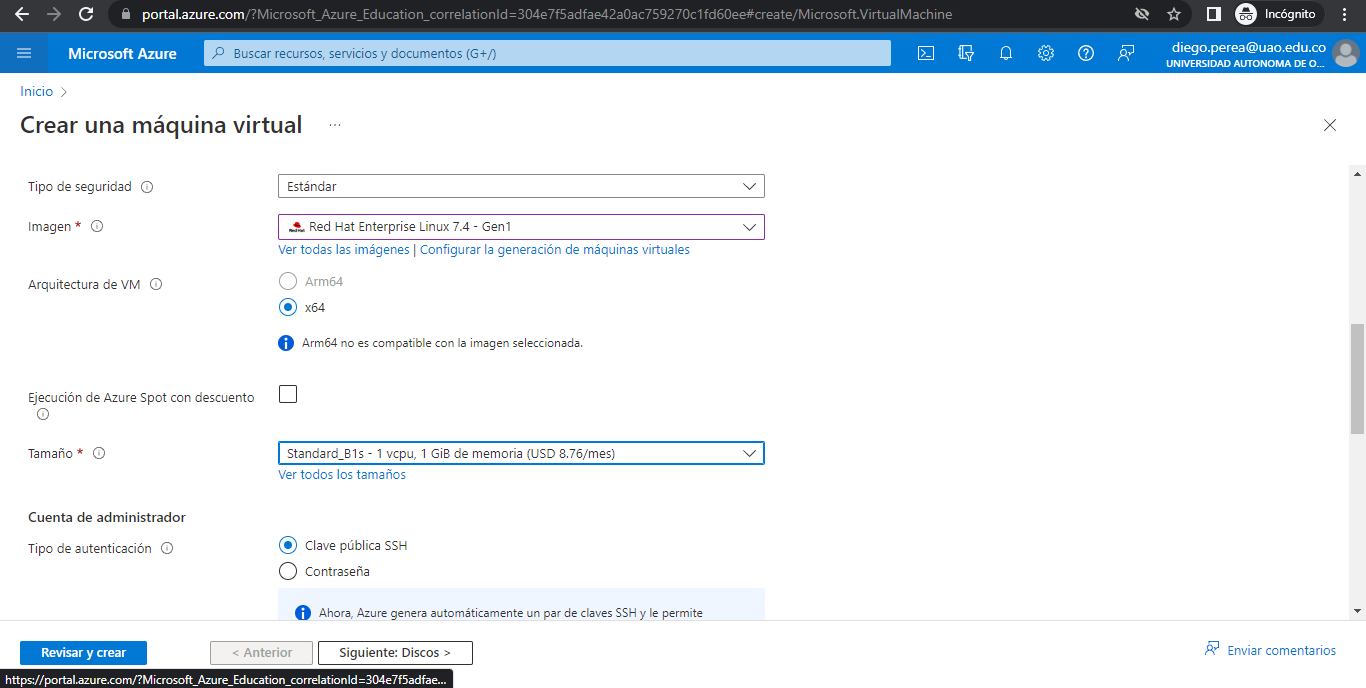


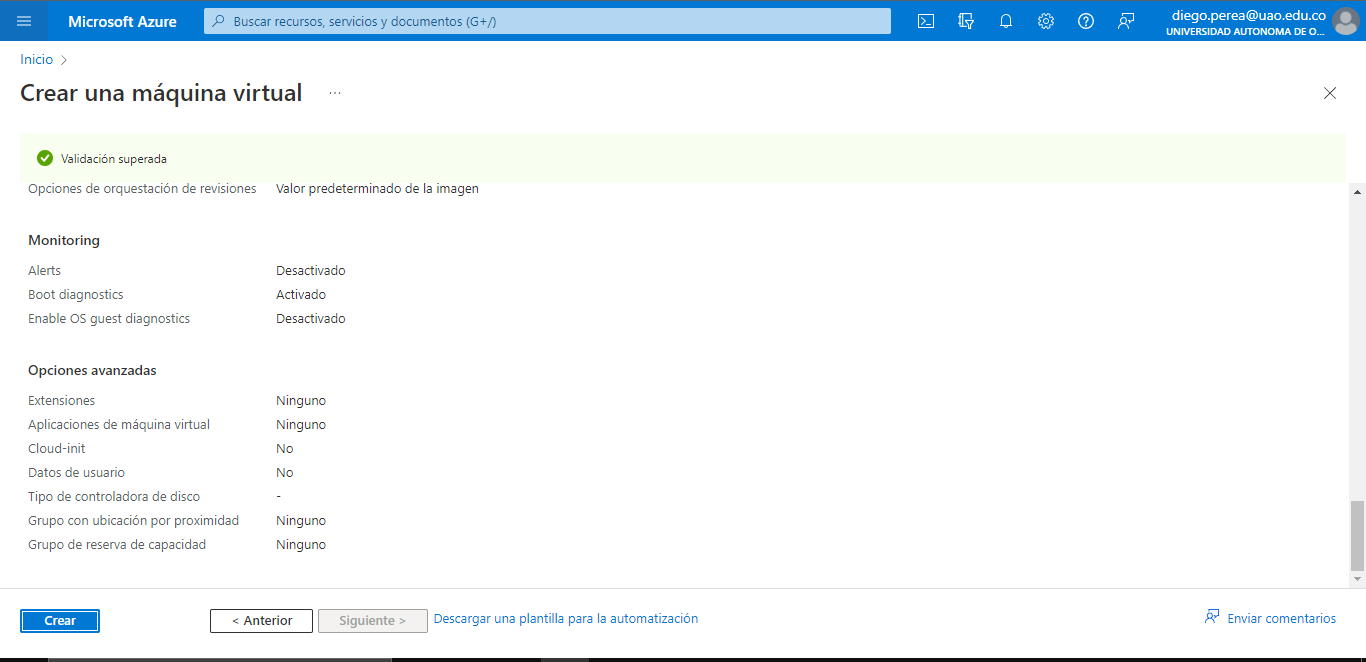


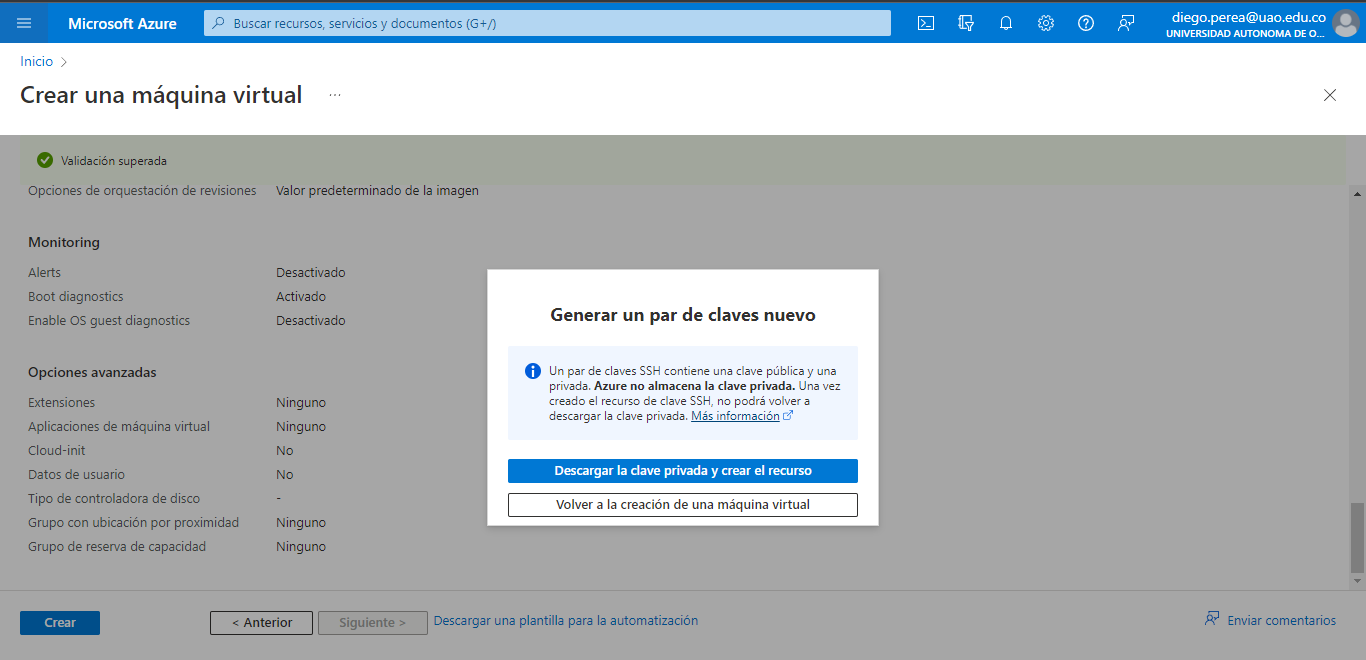


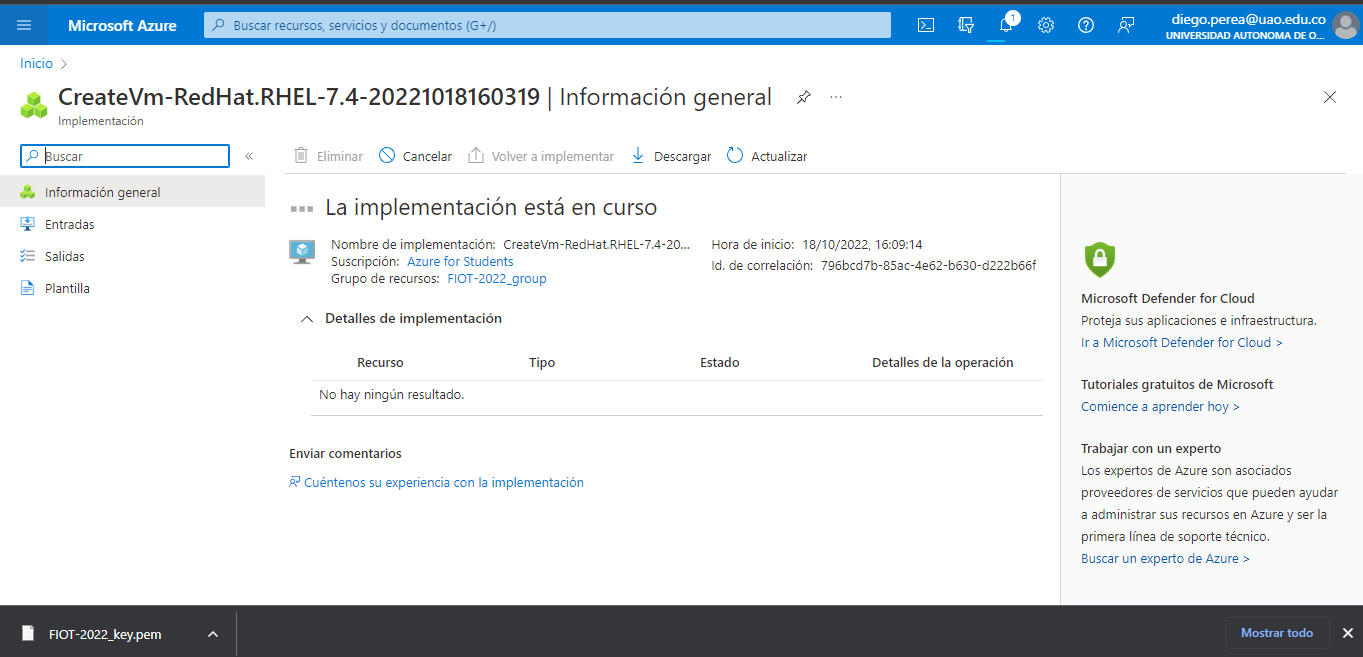


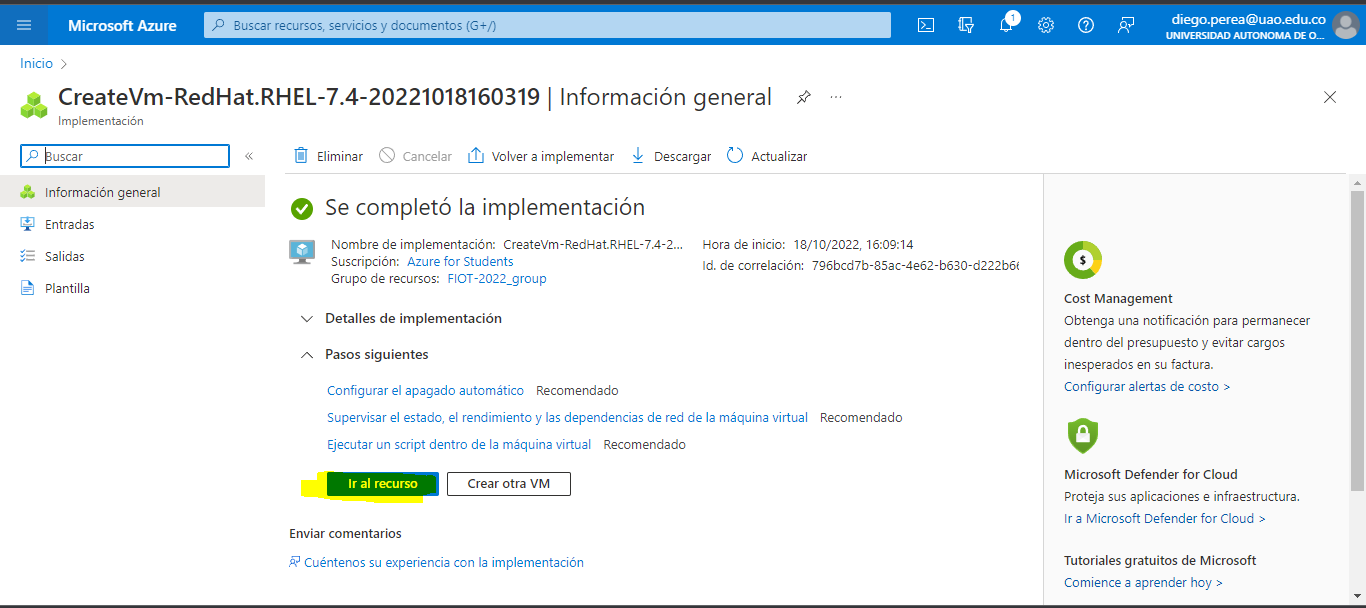


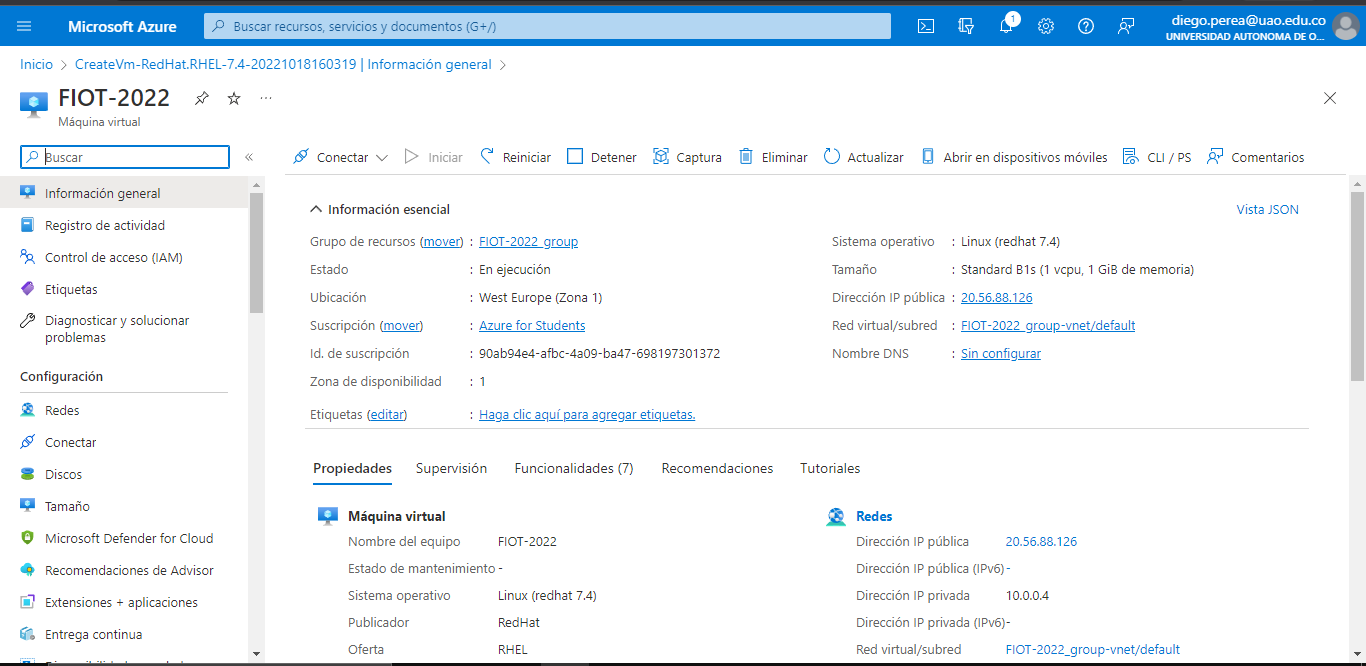


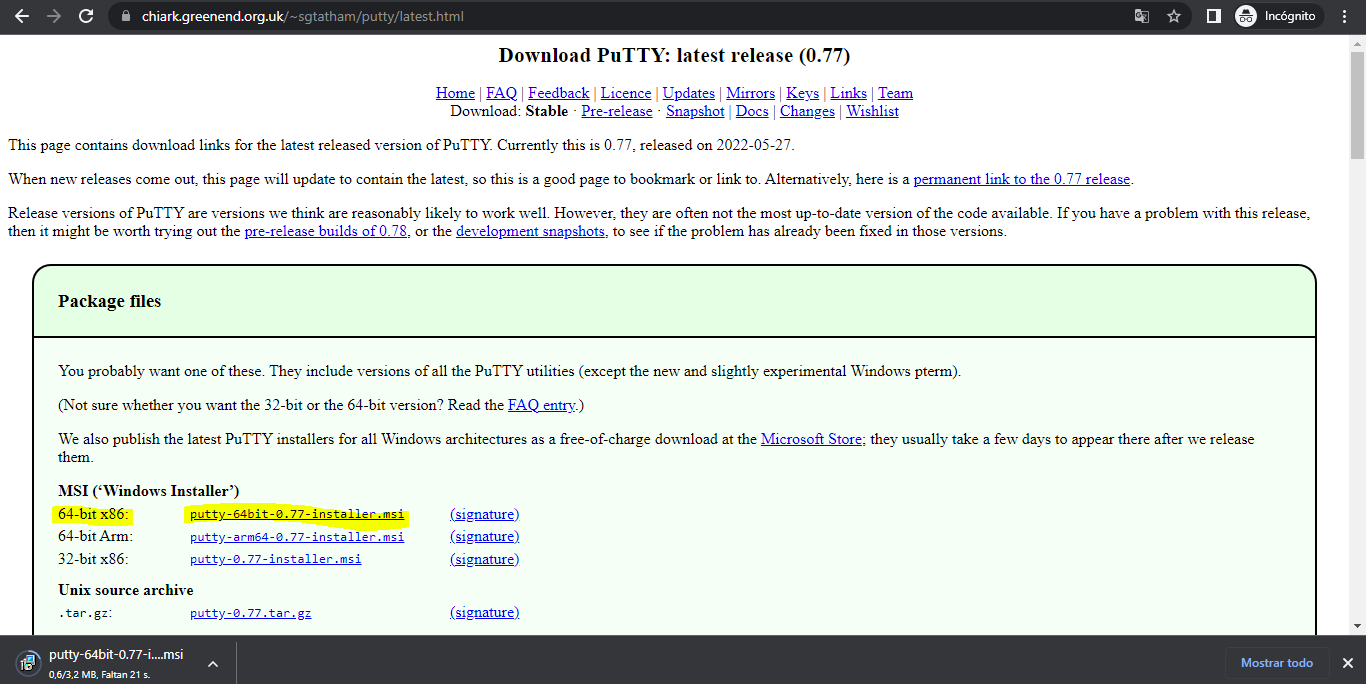


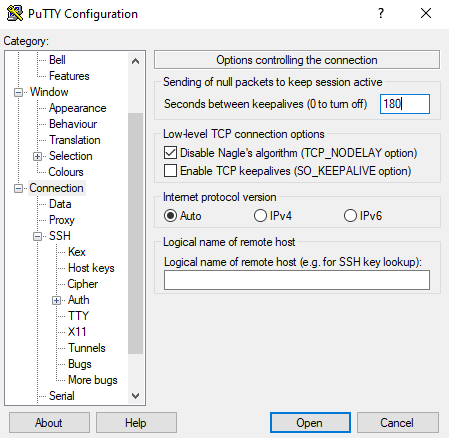


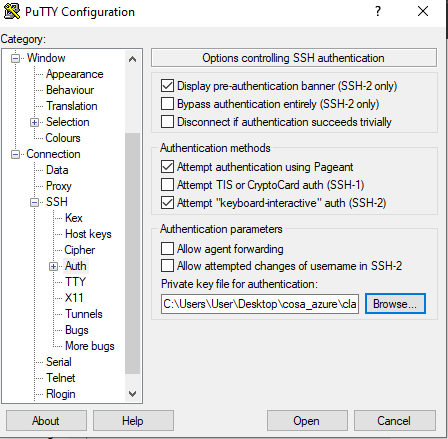






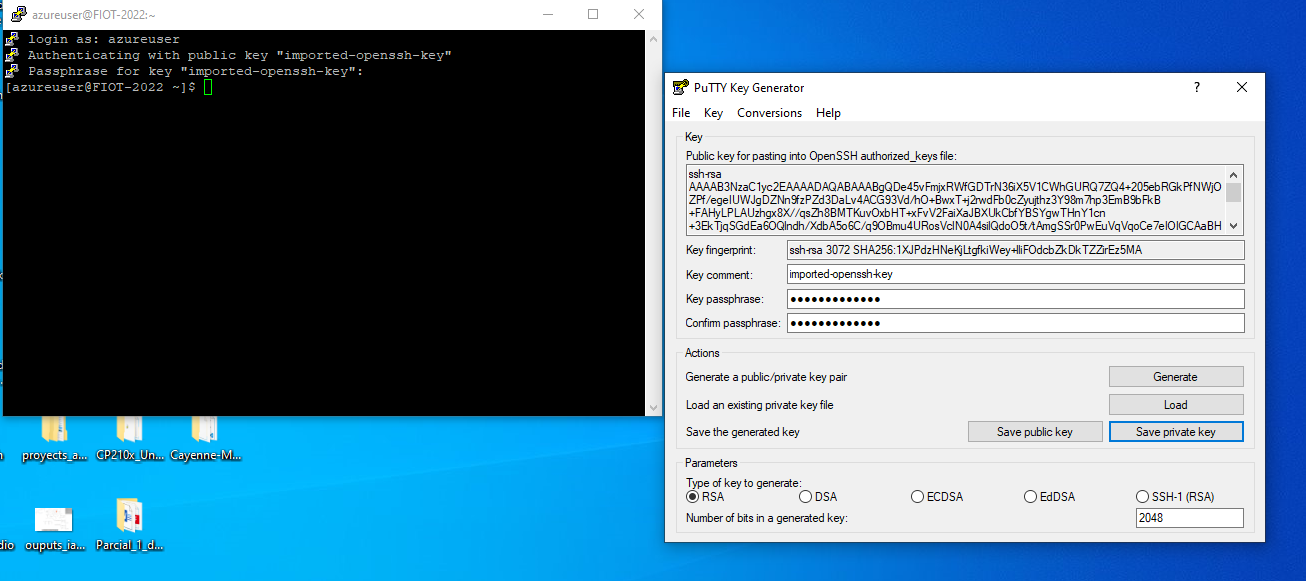


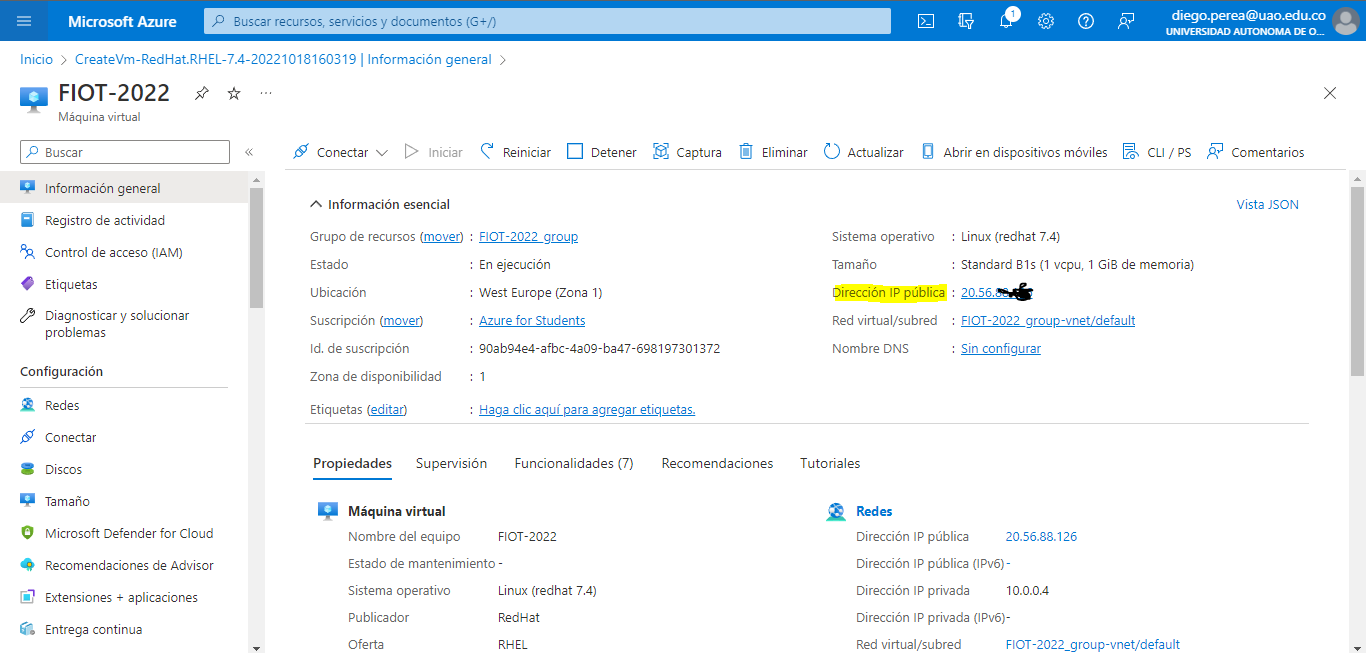




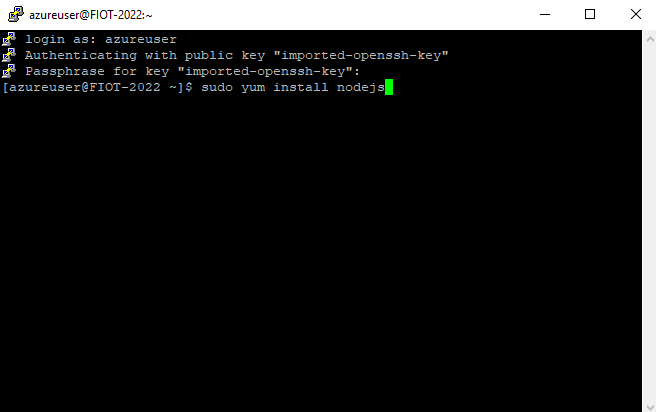
EN SESSION PONER IP

Darle a sabe private key y darle a open

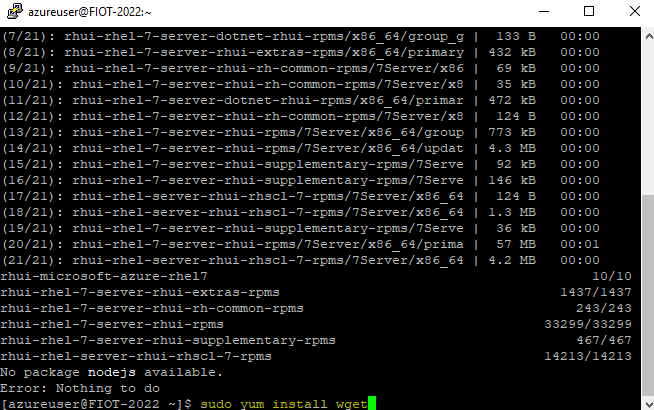




PARA INSTALAR NODE JS



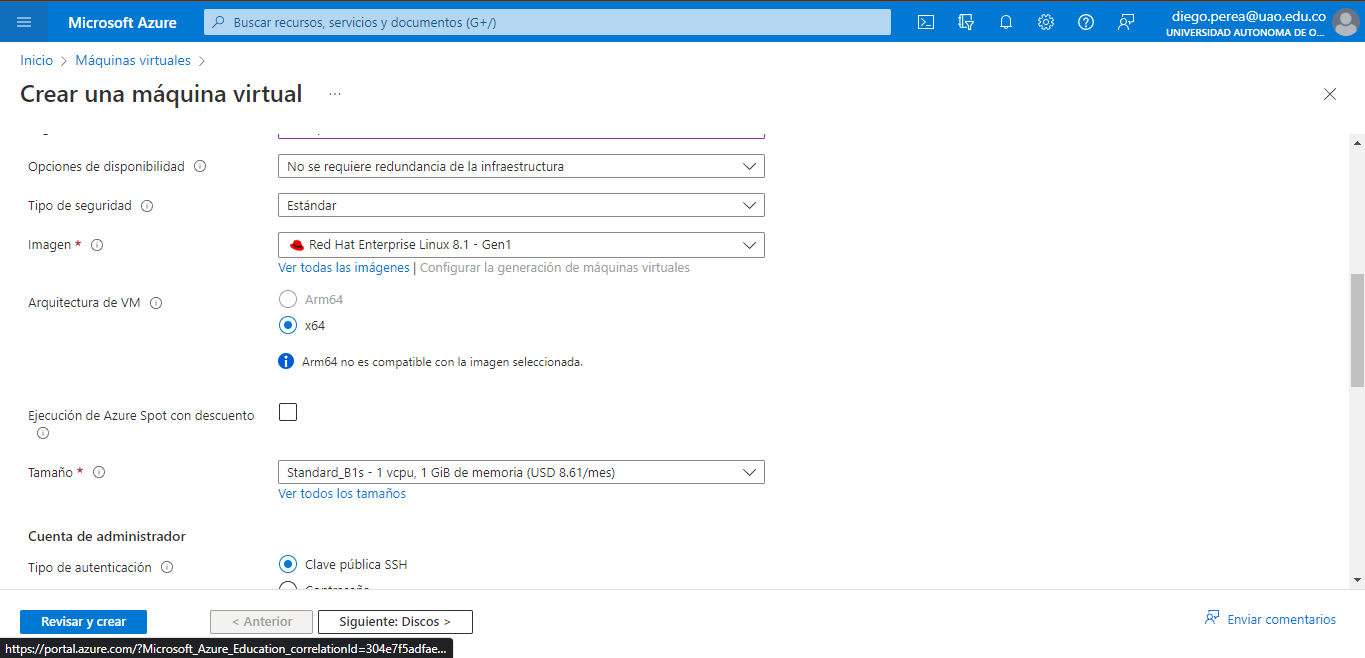
Instalar recursos de internet o web

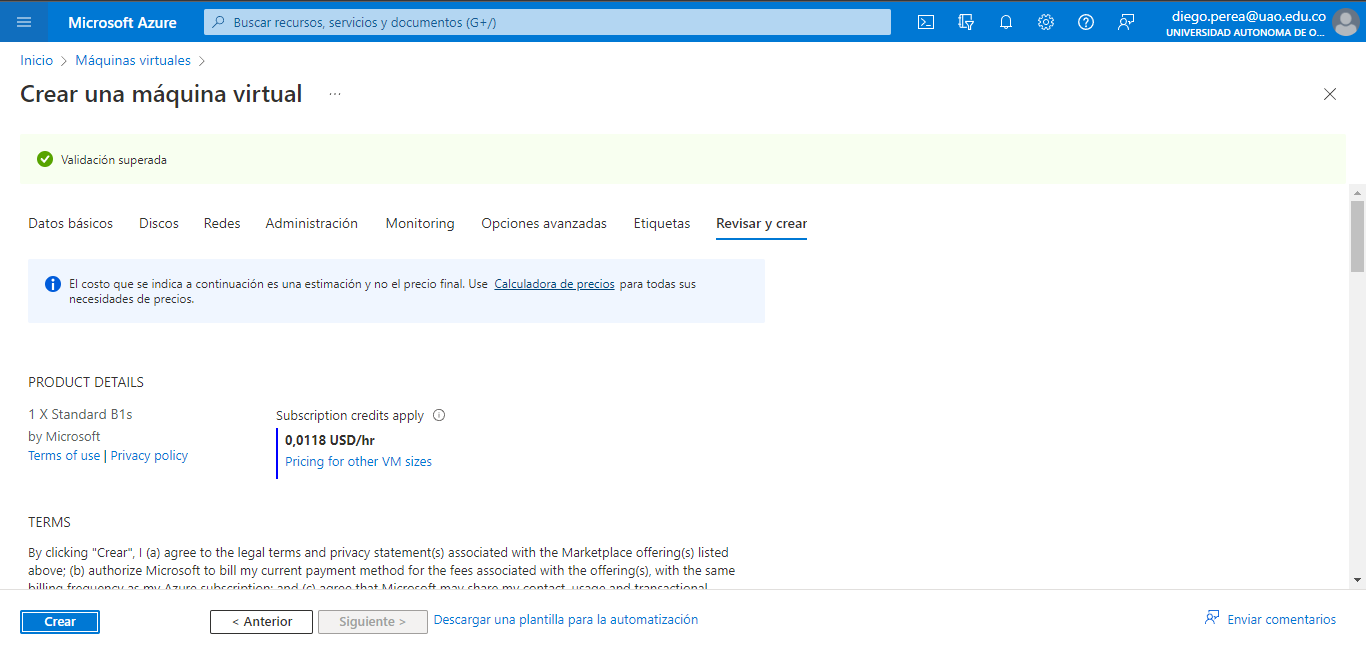


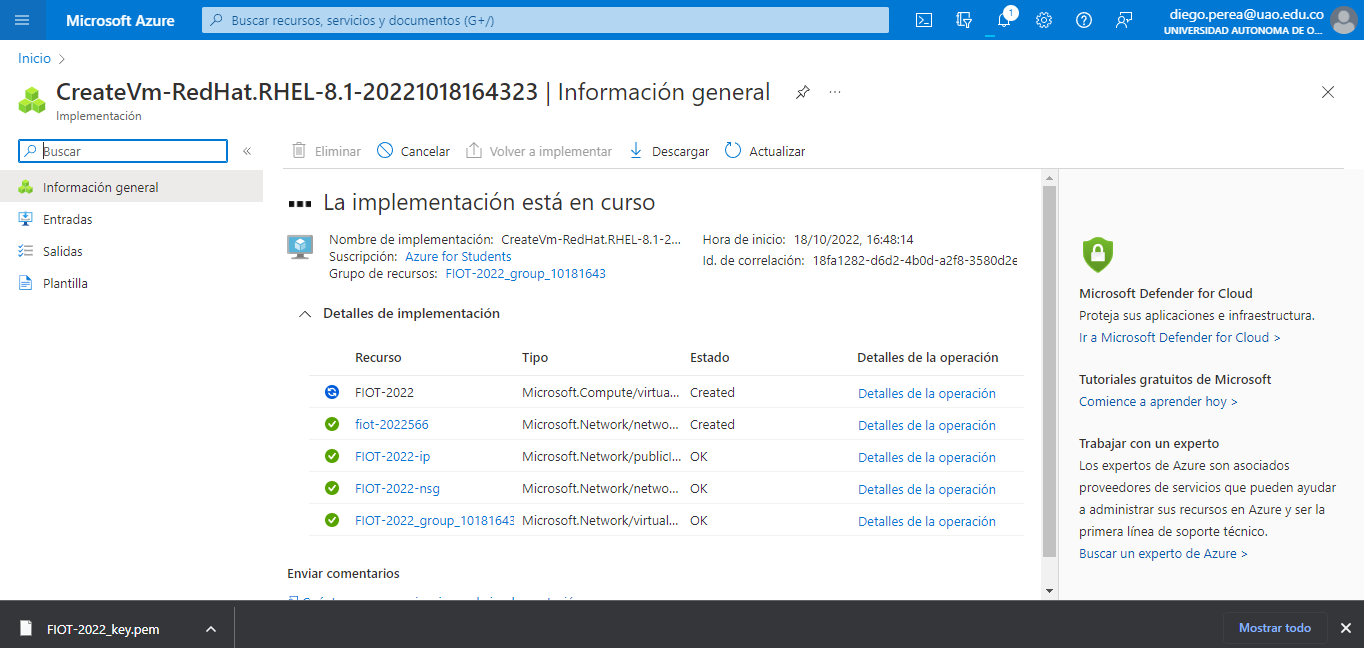
Maquina virtual fallida debido a la memoria

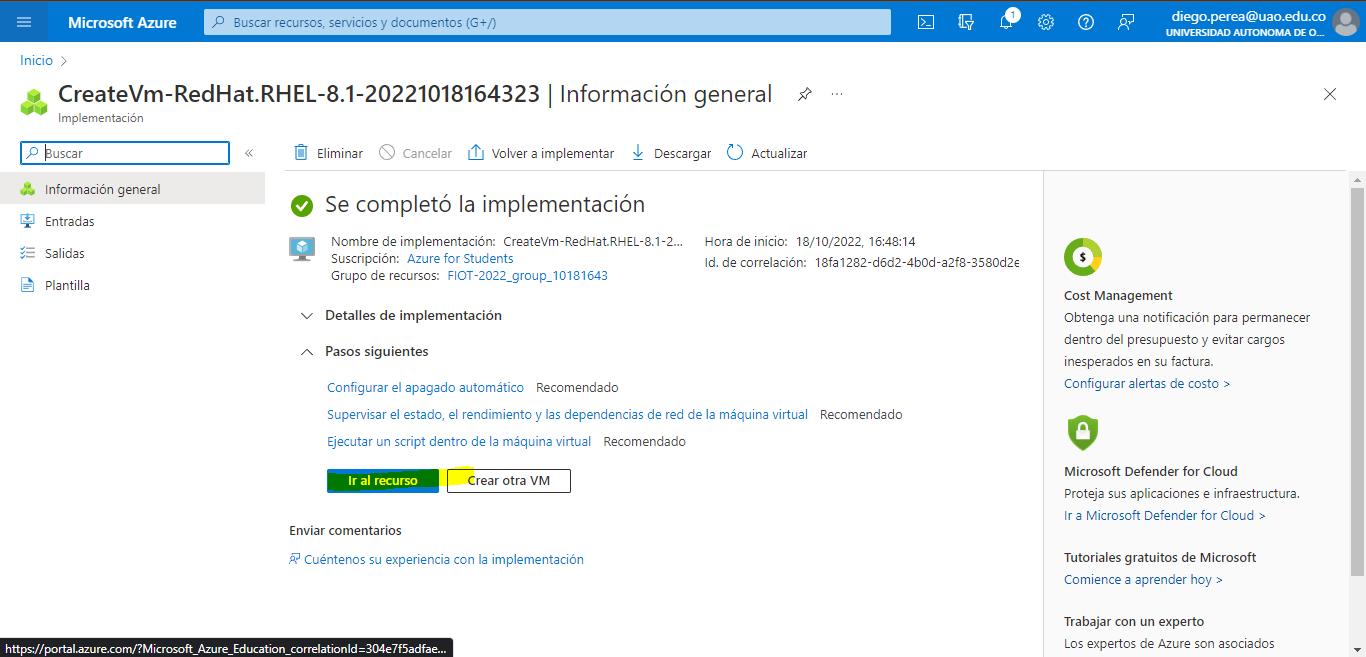
MUY IMPORTANTE TAMAÑO STARNDART B1S O QUE TENGO IGUAL O MAYOR MEMORIA!!!!!!

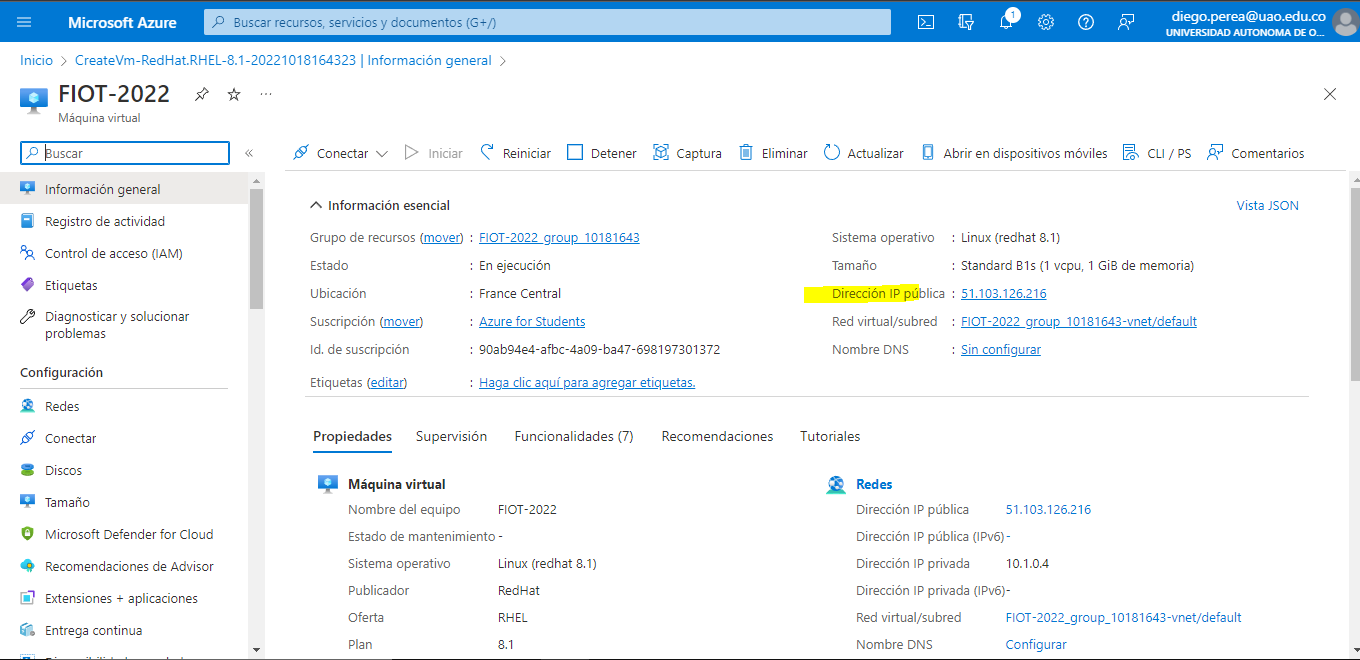
Y VERSION SEA LA MISMA O LOS CODIGOS DE INSTALACION CAMBIAN!!!

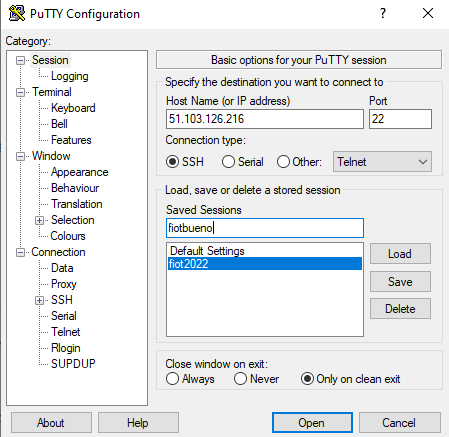




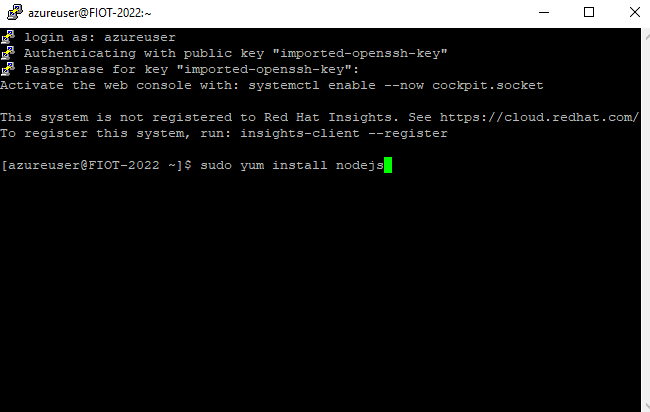




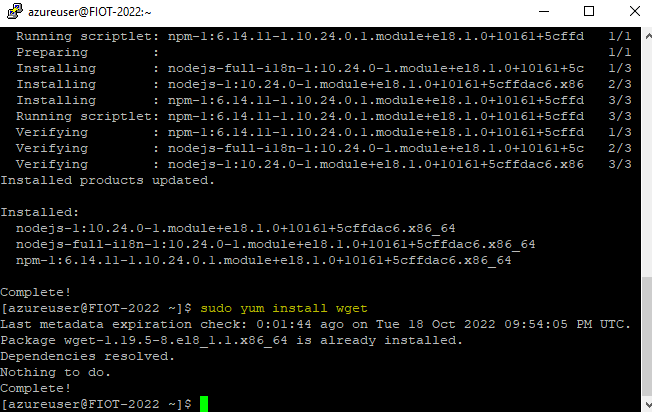




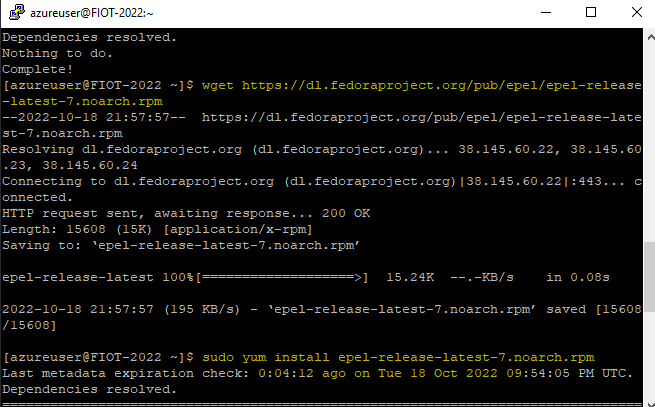
Instalar nodejs



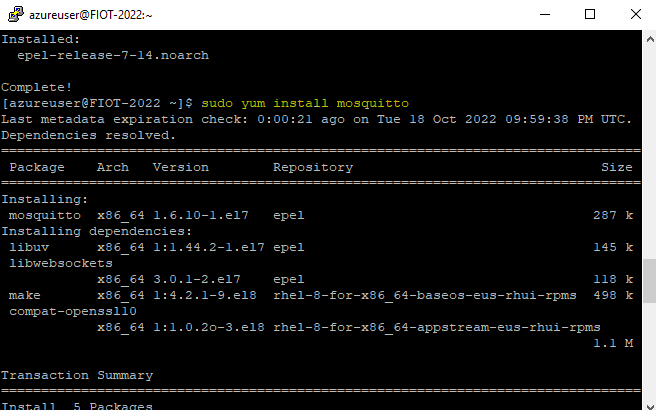
Instalar wget



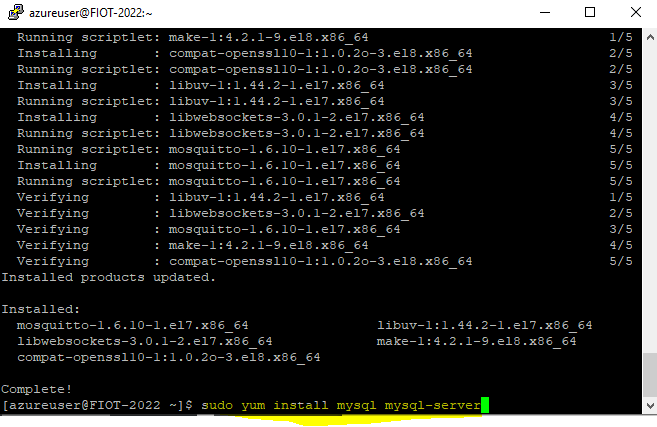
Otras cosas



Instalar mosquitoo



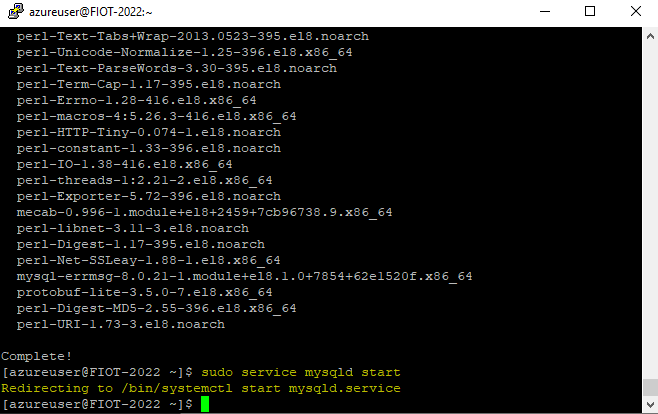
Instalar mysql



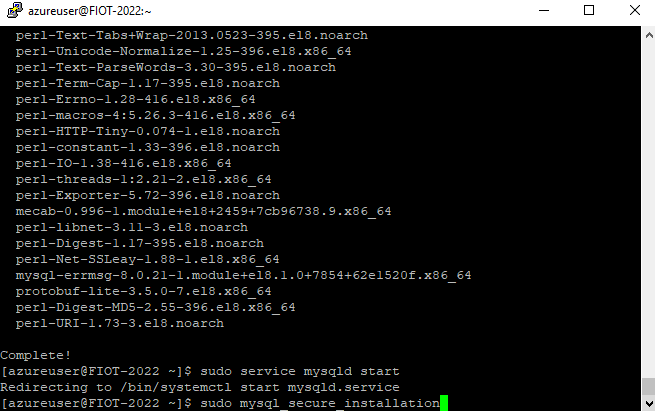
Iniciar servivio mysql

Si da error dar sudo yum remove mysql-server

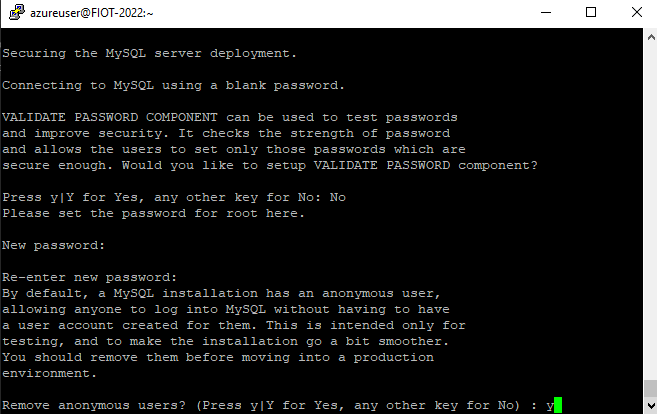
Instalarlo e iniciarlo de nuevo



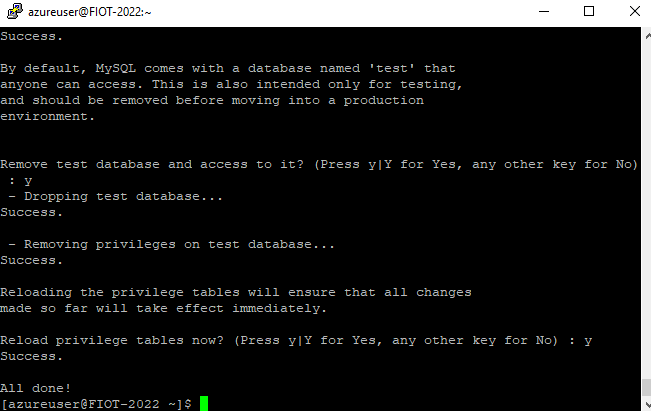
Ponerle una clave a mysql



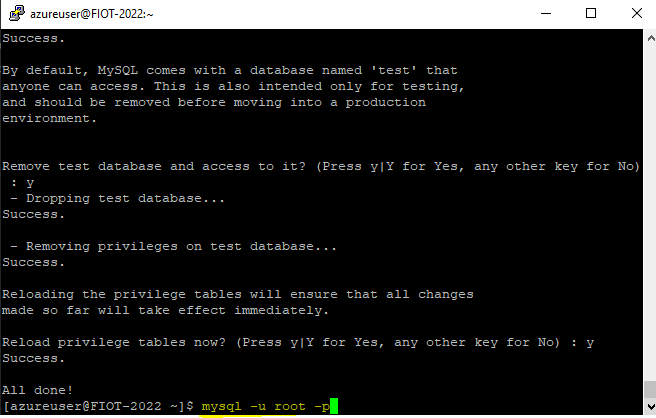
Ingresar clave



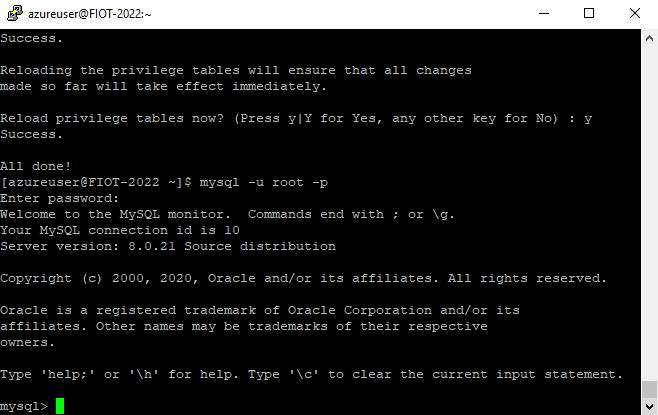
Ponerle y a todo desde ahí



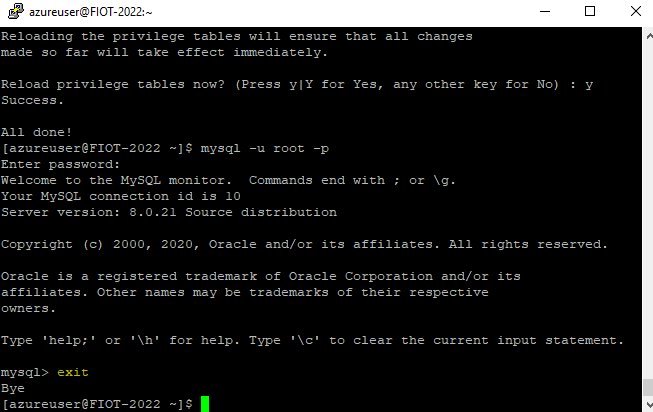
Ingresar a mysql



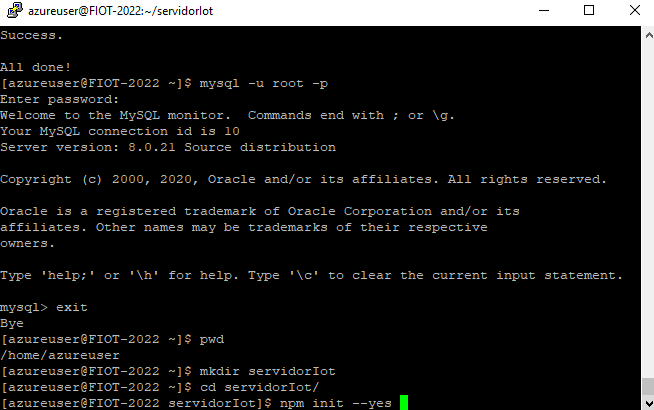
La -p es para que pida el password



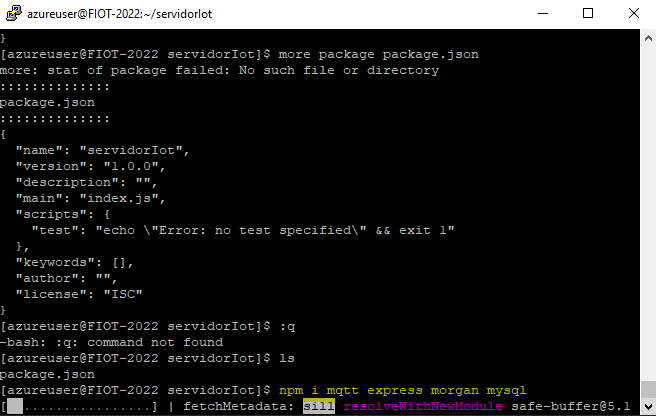
Para salir de mysql dar comando exit



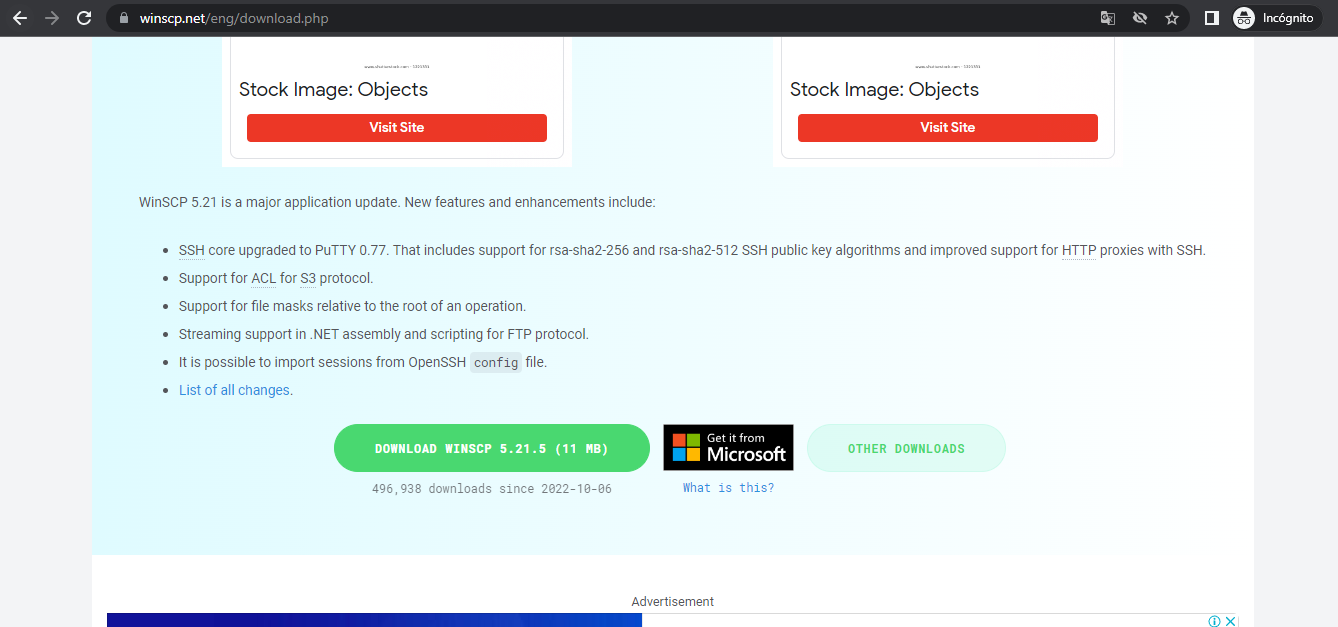
Crear carpeta y servidor nodejs

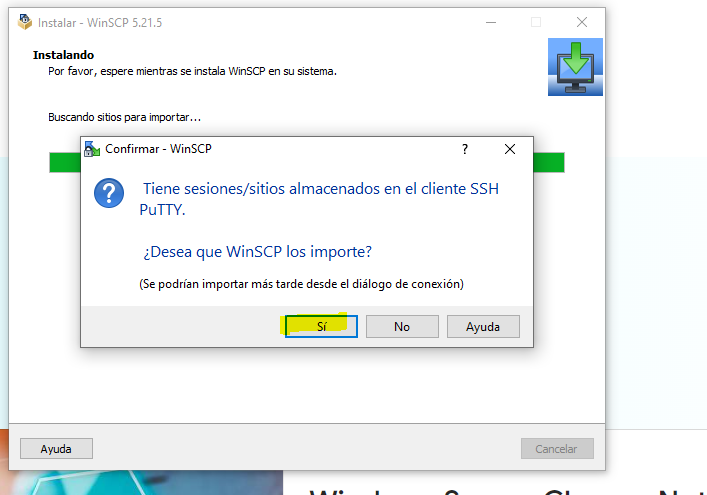


Creación de archivos para nodejs

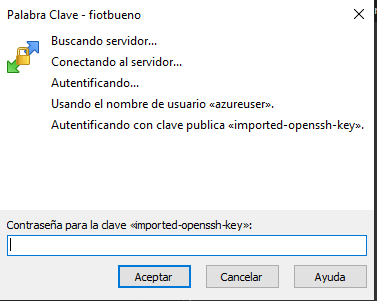


Descargar winscp

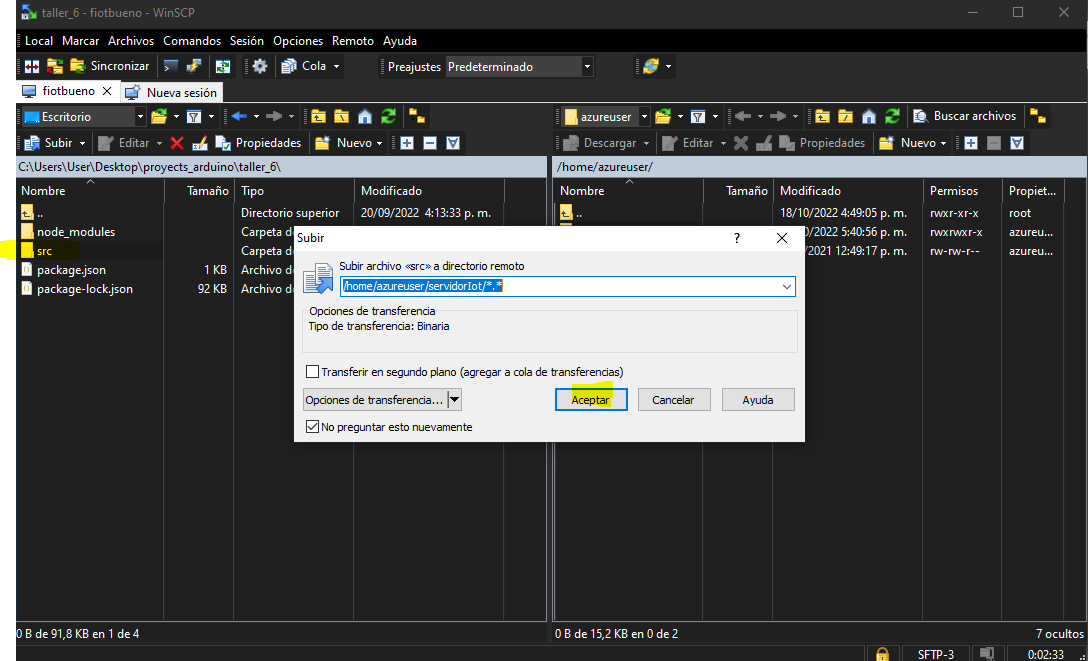


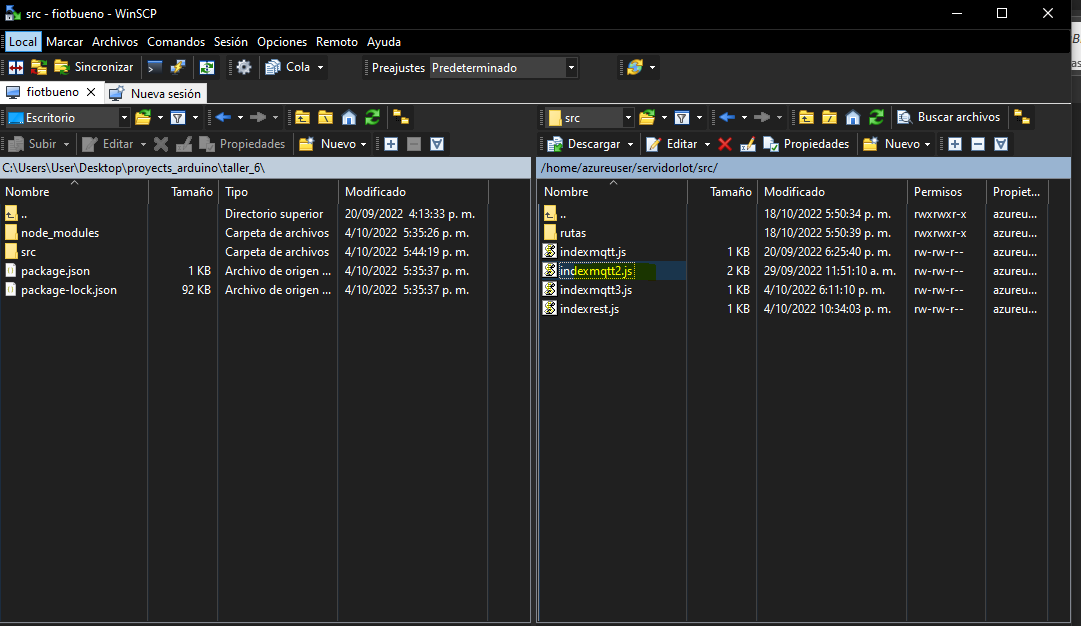


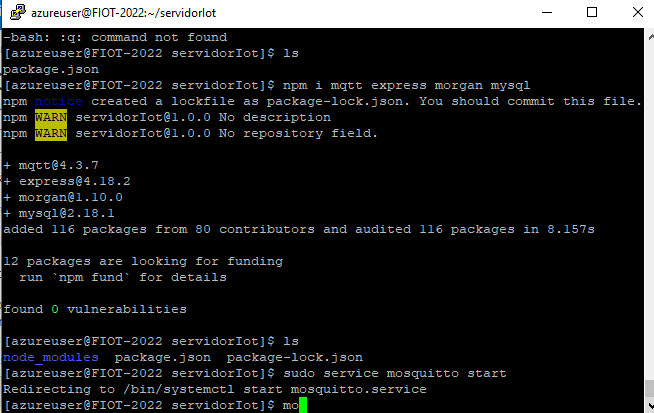
El usuario es azureuser



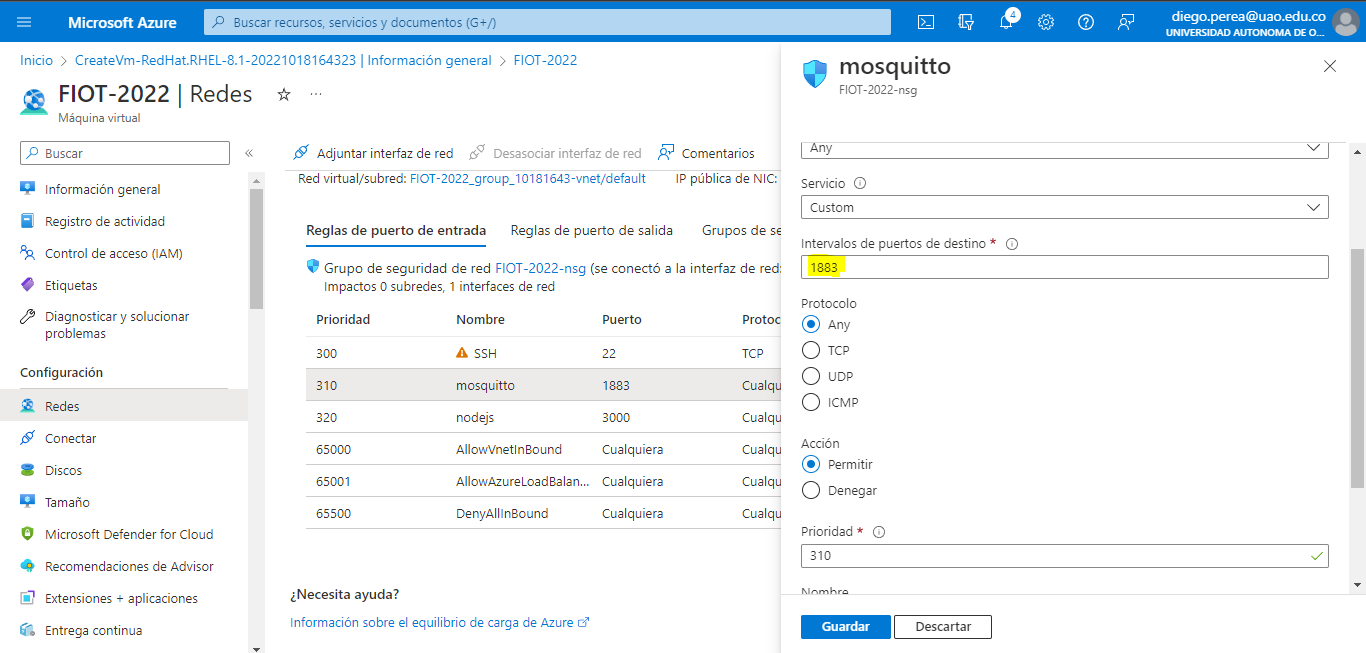
Arrastrar carpeta de trabajo src y ponerla en la nueva carpeta creada

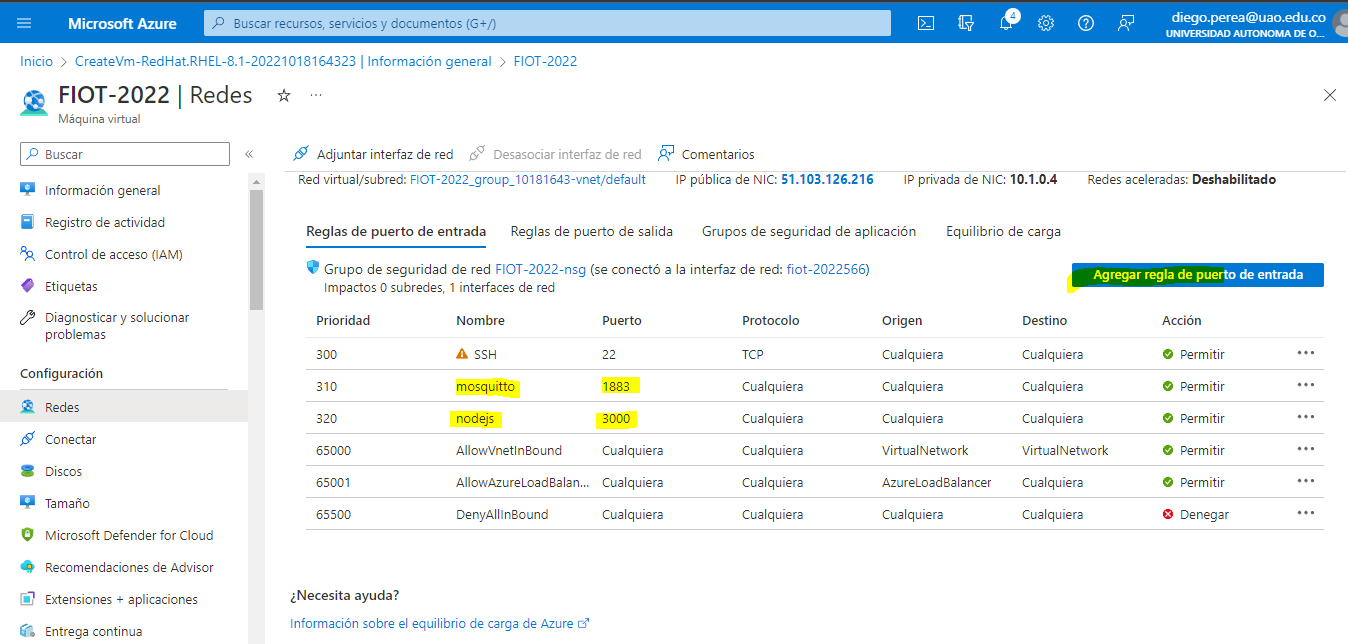


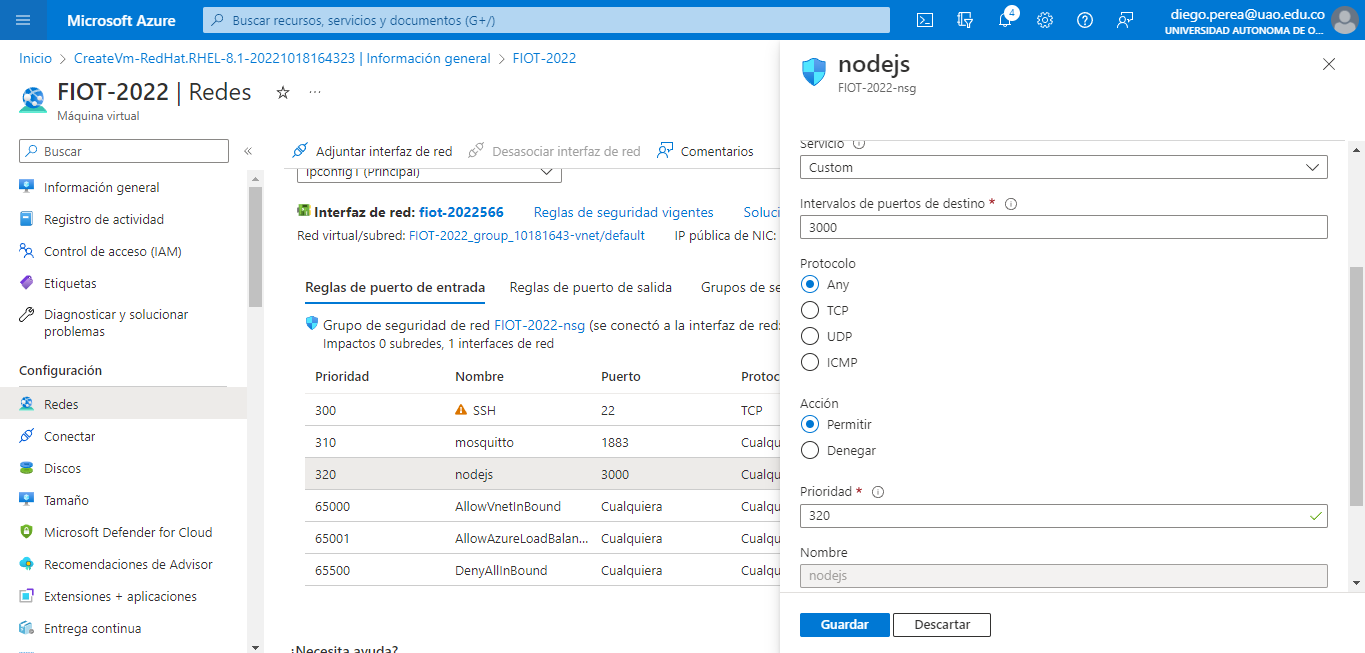




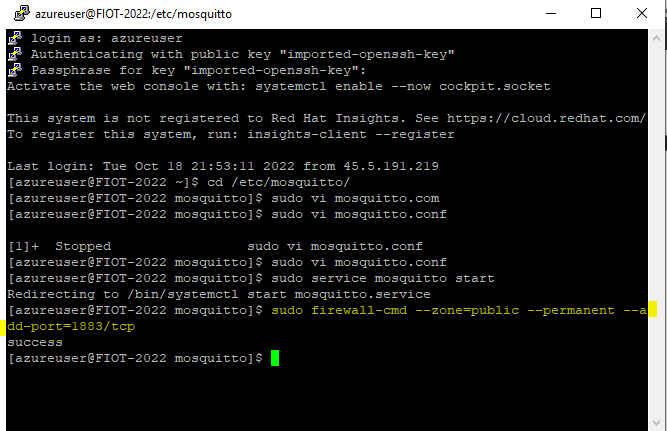
Darle en agregar regla para el puerto en mosquitto y nodejs



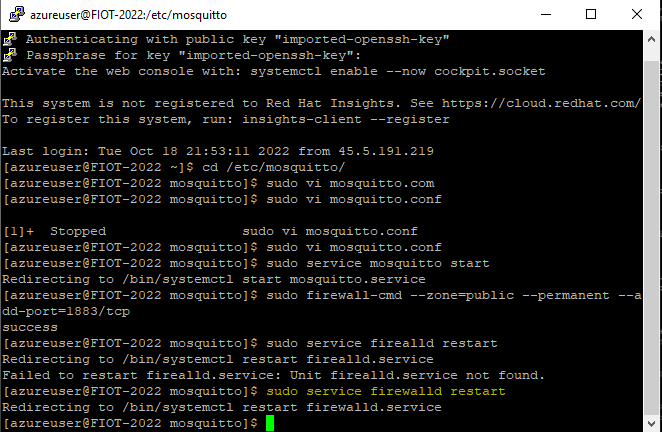




Se abre el firewall para que sirva empiecemosquito

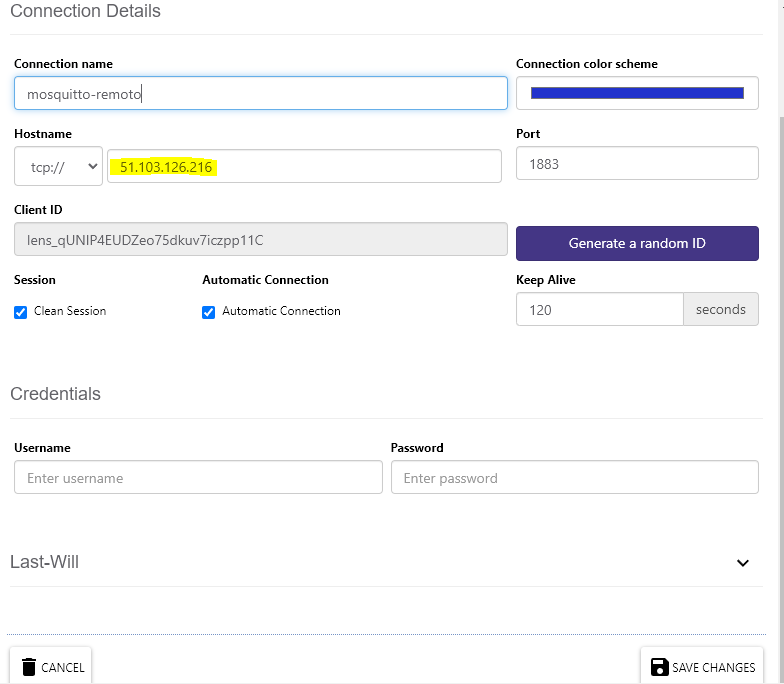


Después restart el firewall



En MQTTLENS dándola ip remota

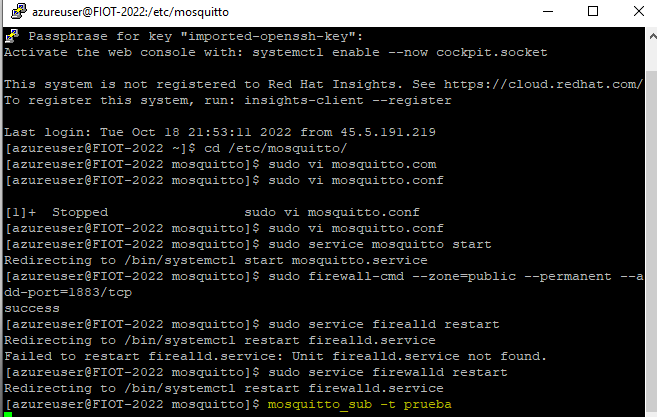




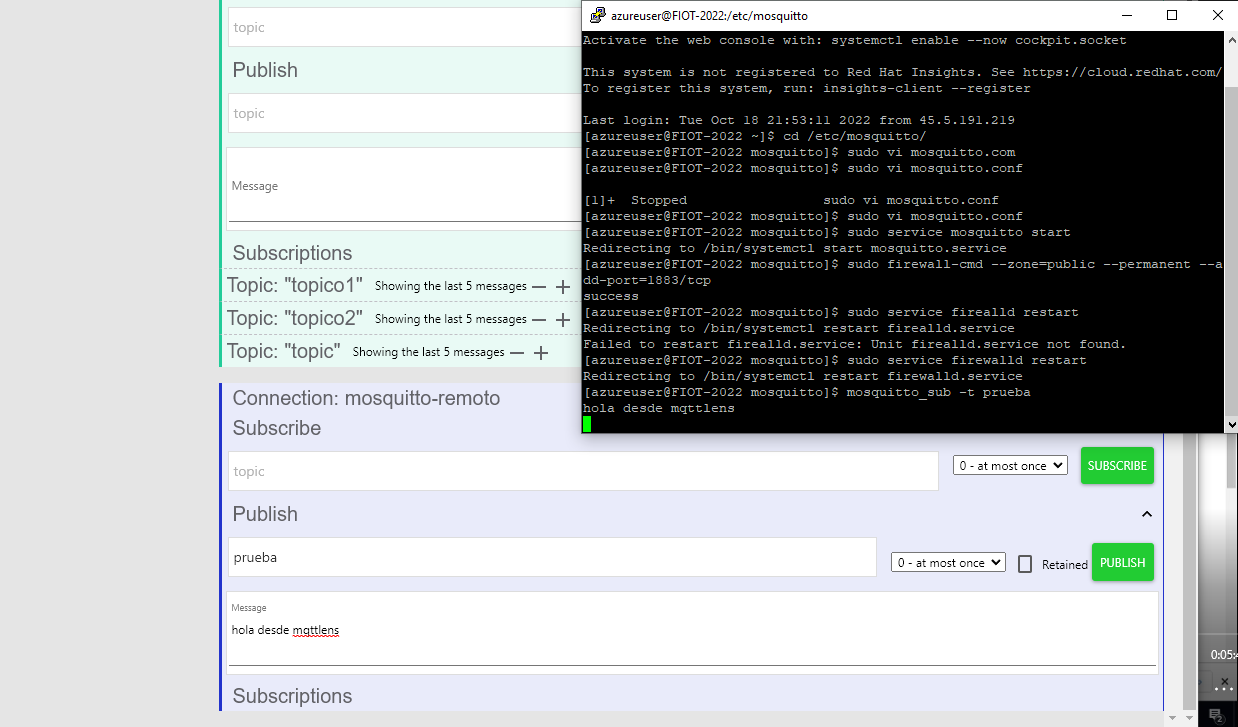
Se conecta mosquito



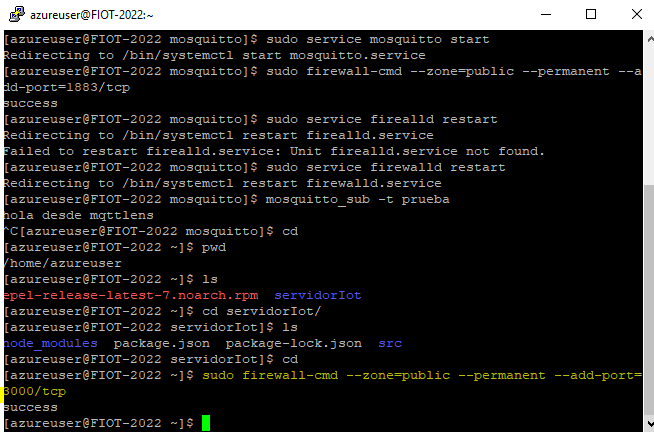
Se realiza suscripción de prueba en mosquito



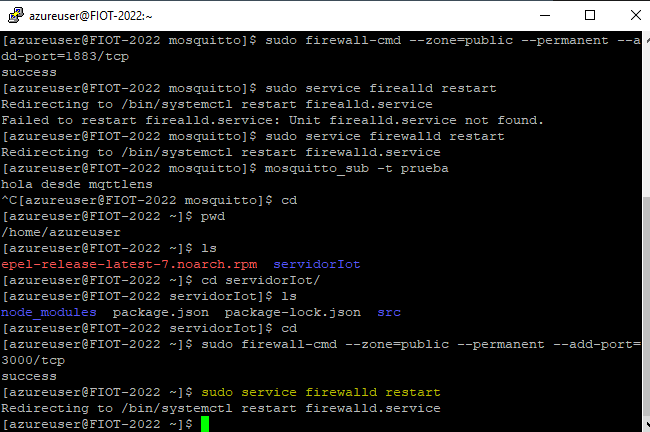
Publicación en el topico en mosquitto



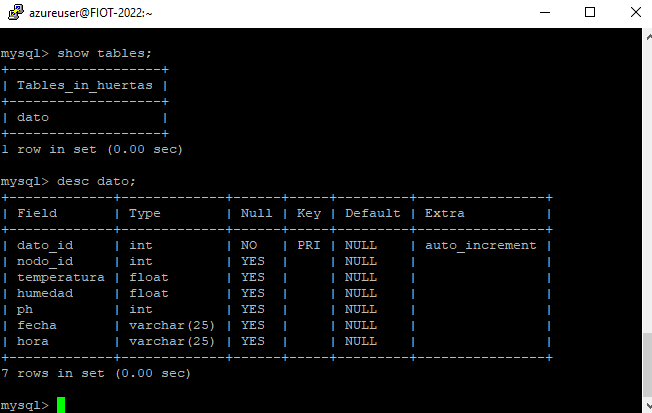
Ahora comenzar el puerto de nodejs



Y restart el firewall



Creación de tablas en mysql , y creación de base de datos llamada huertas



Códigos------------------en mqtt2

var mqtt = require("mqtt");

var client = mqtt.connect("mqtt://localhost");

const mysql = require("mysql");

// se crea la conexiÃ³n a mysql

const connection = mysql.createPool({

connectionLimit: 500,

host: "localhost",

user: "root",

password: "\*\*\*\*\*\*", //el password de ingreso a mysql

database: "huertas",//nombre del LA BASE DE DATOS CREADA en mysql

port: 3306,

});

client.on("connect", function () {

client.subscribe("topico1", function (err) {

if (err) {

console.log("error en la subscripcion");

}

});

});

client.on("message", function (topic, message) {

// message is Buffer

json1 = JSON.parse(message.toString());

console.log(json1);

//client.publish('topico2', 'mensaje recibido')

connection.getConnection(function (error, tempConn) {

//conexion a mysql

if (error) {

console.log("Problemas en la conexion"); //en caso de error en la

conexion;

} else {

console.log("Conexion correcta.");//tempConn.query( "INSERT INTO \*nombre de la TABLA\* VALUES(null, ?, ?,?,?)"

tempConn.query(

"INSERT INTO dato VALUES(null, ?,?,?,?,?,?)",

[json1.nodo\_id, json1.temperatura, json1.humedad,json1.ph, json1.fecha ,json1.hora],

function (error, result) {

//se ejecuta lainserciÃ³n

if (error) {

console.log("error al ejecutar el query");

} else {

tempConn.release();

console.log("datos almacenados"); //mensaje de respuesta en consola

}

//client.end() //si se habilita esta opciÃ³n el servicio termina

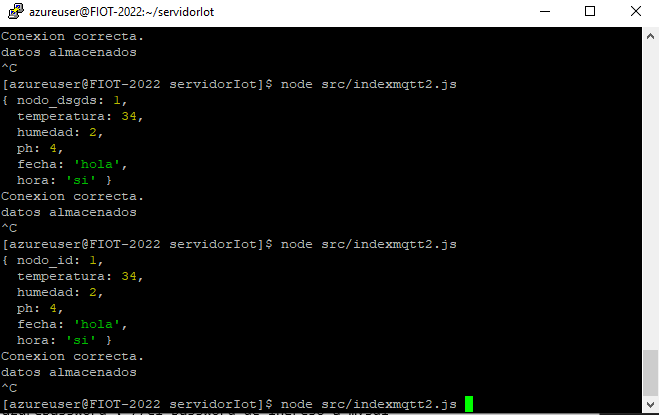
}

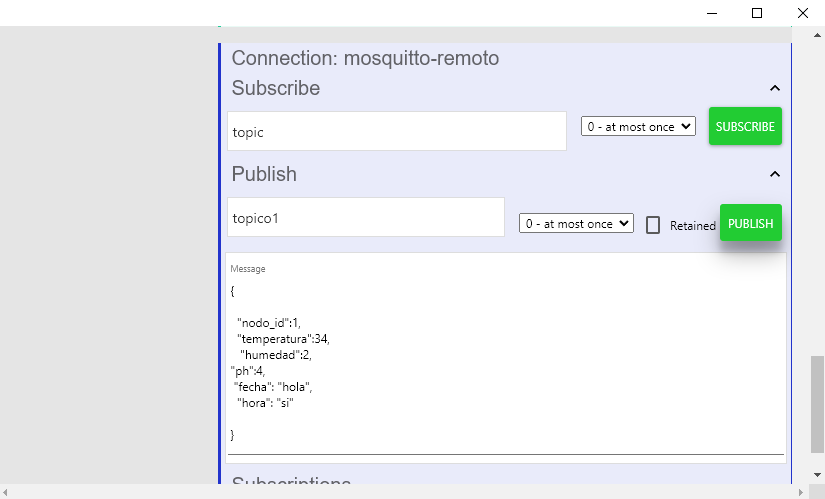
);

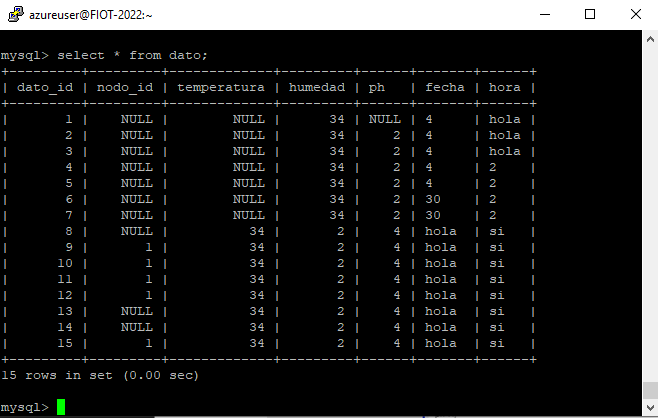
}

});

});







Ahora con postman

Código index.js:

const express = require('express'); //se indica que se requiere express

const app = express(); // se inicia express y se instancia en una constante de nombre app.

const morgan = require('morgan'); //se indica que se requiere morgan

// settings

app.set('port', 3000); //se define el puerto en el cual va a funcionar el

// Utilities

app.use(morgan('dev')); //se indica que se va a usar morgan en modo dev

app.use(express.json()); //se indica que se va a usar la funcionalidad para manejo de json de express

//Routes

//app.use(require('./rutas/ejemplo.js'));

//app.use(require('./rutas/suma.js'));

app.use(require('./rutas/datos.js'));

//app.use(require('./rutas/datosm.js'));

//Start server

app.listen(app.get('port'), ()=> {

console.log("Servidor funcionando");

}); //se inicia el servidor en el puerto definido y se pone un mensaje en la consola.



Código datos.js

const { Router } = require("express");

const router = Router();

const mysql = require("mysql");

// se crea la conexiÃ³n a mysql

const connection = mysql.createPool({

connectionLimit: 500,

host: "localhost",

user: "root",

password: "\*\*\*\*\*", //el password de ingreso a mysql

database: "huertas",// name del DATA BASE EN SQL

port: 3306,

});

//function get en la ruta /datos, que trae todos los datos almacenados en la tabla;

router.get("/datos", (req, res) => {

var json1 = {}; //variable para almacenar cada registro que se lea, enformato json

var arreglo = []; //variable para almacenar todos los datos, en formatoarreglo de json

connection.getConnection(function (error, tempConn) {

//conexion a mysql

if (error) {

throw error; //si no se pudo conectar

} else {

console.log("Conexion correcta.");

//ejecuciÃ³n de la consulta

tempConn.query("SELECT \* FROM dato", function (error, result) {

var resultado = result; //se almacena el resultado de laconsulta en la variable resultado

if (error) {

throw error;

res.send("error en la ejecuciÃ³n del query");

} else {

tempConn.release(); //se librea la conexiÃ³n

for (i = 0; i < resultado.length; i++) {

//se leeel resultado y se arma el json

json1 = {

"dato": resultado[i].dato,

"nodo": resultado[i].nodo,

"temperatura": resultado[i].temperatura,

"humedad": resultado[i].humedad,

"ph": resultado[i].ph,

"fecha": resultado[i].fecha,

"hora": resultado[i].hora,

};

console.log(json1); //se muestra el json en la consola

arreglo.push(json1); //se aÃ±ade el json al arreglo

}

res.json(arreglo); //se retorna el arreglo

}

});

}

});

});

//funciÃ³n post en la ruta /datos que recibe datos

router.post("/datos", (req, res) => {

console.log(req.body); //mustra en consola el json que llego

json1 = req.body; //se almacena el json recibido en la variable json1

connection.getConnection(function (error, tempConn) {

//conexion a mysql

if (error) {

throw error; //en caso de error en la conexion

} else {

console.log("Conexion correcta.");

tempConn.query(

"INSERT INTO dato VALUES(null, ?,?,?,?,?,?)",

[json1.nodo\_id, json1.temperatura, json1.humedad,json1.ph, json1.fecha,json1.hora],

function (error, result) {

//se ejecuta la inserciÃ³n

if (error) {

res.send("error al ejecutar el query");

} else {

tempConn.release();

res.send("datos almacenados"); //mensaje de respuesta

}

}

);

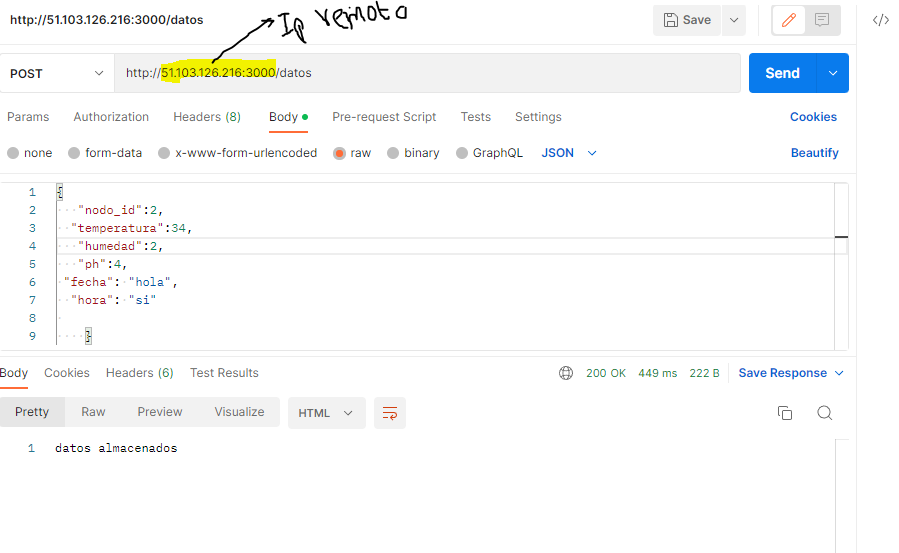
}

});

});

module.exports = router;

con postman :



Visualización en mysql:

