

“Año del Fortalecimiento de la Soberanía Nacional”
UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
(Universidad del Perú, Decana de América)
FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMATICA

SISTEMA DE RIEGO AUTOMATIZADO

PROYECTO FINAL



G8

INTERNET DE LAS COSAS

Docente: Cueva Rosas, Yessica

- Ccolcca Avalos, Mariluz - 19200177
- Dominguez Acosta, Prish Antony - 19200054
- Linares Castañeda, Bruno Luis - 19200140
- Malca Ramirez, Jhonattan David - 19200030
- Ramirez Teran, Brithany Antonella - 19200168
- Solis Rivera, Angel Jeanpierre - 19200194

I INTRODUCCIÓN

Actualmente, la comunicación inalámbrica es considerada y aceptada como medio de enlace entre diferentes dispositivos a través de una conexión WIFI, como en sistemas industriales, salud y agrícola.



Con la comunicación inalámbrica, la interrelación del puerto serie y algunos dispositivos, se pudo crear el proyecto de sistema de riego automatizado utilizando Internet de las Cosas.

III

REVISIÓN DEL ESTADO DEL ARTE

Diseño e implementación de un sistema de regado con funciones smart

AUTOR(ES) - AÑO	Roca Roca , Jose Manuel (2020)
UNIVERSIDAD O ENTIDAD	Universidad Politécnica de Valencia
PRINCIPAL OBJETIVO	Crear un sistema de riego que permita un funcionamiento autónomo y hacer un seguimiento de factores y control de estos con tecnología.
INSTRUMENTOS	Se utilizó Git para permitir la sincronización del código entre distintos dispositivos y respecto al hardware se tiene el Raspberry pi, Arduino, sensores, electroválvulas de riego, relés electromagnéticos y dentro del software se tiene a Node -red
RESULTADO	Creación de un sistema de riego autónomo con conexión a internet que dentro de sus acciones tiene la de bloquear riego en caso de previsión de lluvia
CONCLUSIÓN	El proyecto es ampliamente extensible, sobre todo en la parte referida a la comunicación entre dispositivos

III

PROBLEMÁTICA



El riego manual es el riego más común que existe, sobre todo por su simpleza; sin embargo, requiere de mucho tiempo y en caso de tener un área de cultivo grande esta actividad se puede convertir en algo muy incómodo. Es por ello que instalar un sistema de riego autónomo le da al cultivador un control mucho más preciso sobre la cantidad de agua y nutrientes que reciben sus plantas en comparación con el riego a mano.

IV

OBJETIVOS

Desarrollar un Sistema de Riego Automatizado basado en IOT para un mejor control de agua en los cultivos.

Diseñar el sistema de control teniendo en cuenta las condiciones del terreno y las variables ambientales.

Implementar el riego automático cuando sea necesario, según los datos arrojados por los sensores a usar.

Crear una comunicación entre los diferentes tipos de dispositivos empleando el protocolo MQTT.

Mostrar la información de los datos arrojados por los sensores en dashboards usando la plataforma node red.

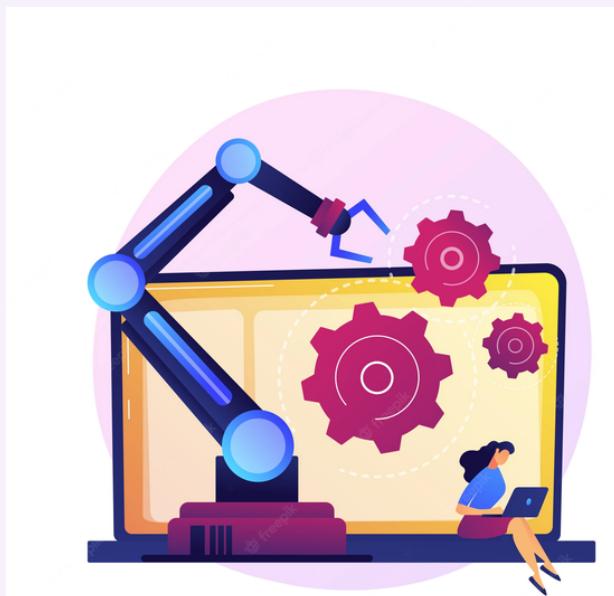
Almacenar en una base de datos, los datos arrojados por los sensores.

V MARCO TEÓRICO

CONCEPTOS



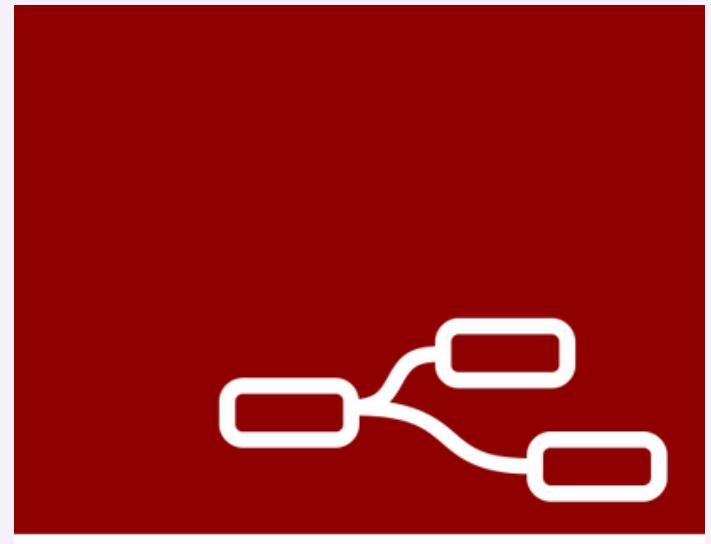
IOT



Automatización

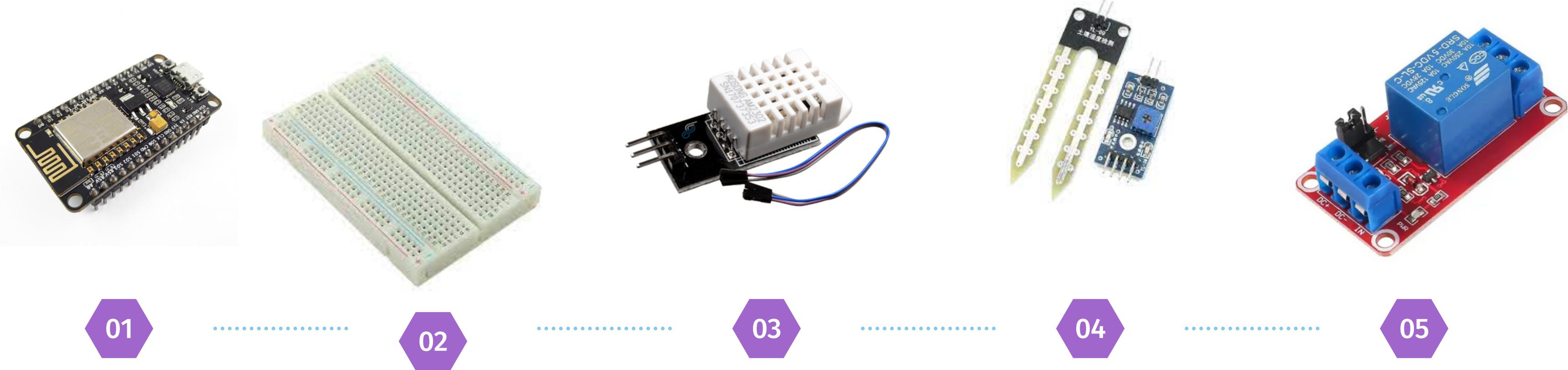


Sistemas de riego
automatizado



Node-RED

COMPONENTES DEL SISTEMA DE RIEGO



01

ESP8266

Controlador entre los componentes electrónicos y para el intercambio de información

02

PROTOBOARD

2 protoboard para realizar el proyecto

03

DHT22

Medir la temperatura y humedad

04

SENSOR HUMEDAD DEL SUELO

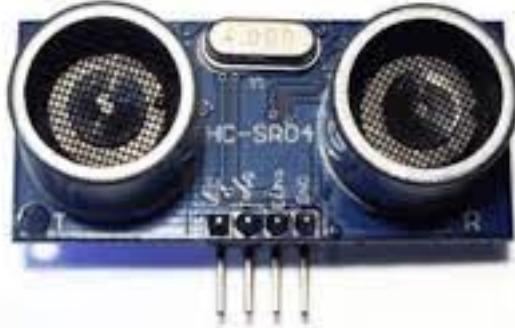
Permite medir la humedad del suelo

05

RELÉ

Componente eléctrico que permite abrir o cerrar un circuito en función de una bobina.

COMPONENTES DEL SISTEMA DE RIEGO



06



07



08



09



10

SENSOR ULTRASONIDO

Medir el nivel
del agua

LED

Emite la luz según la condición del sistema.

RESISTENCIA

Componente ubicado en casi todos los circuitos eléctricos.

BOMBA DE AGUA

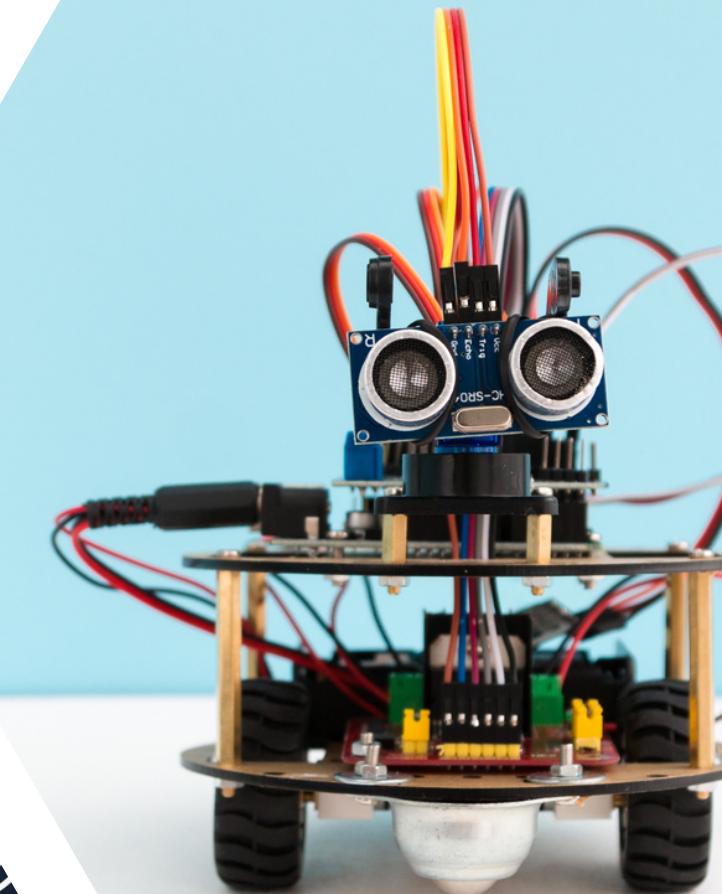
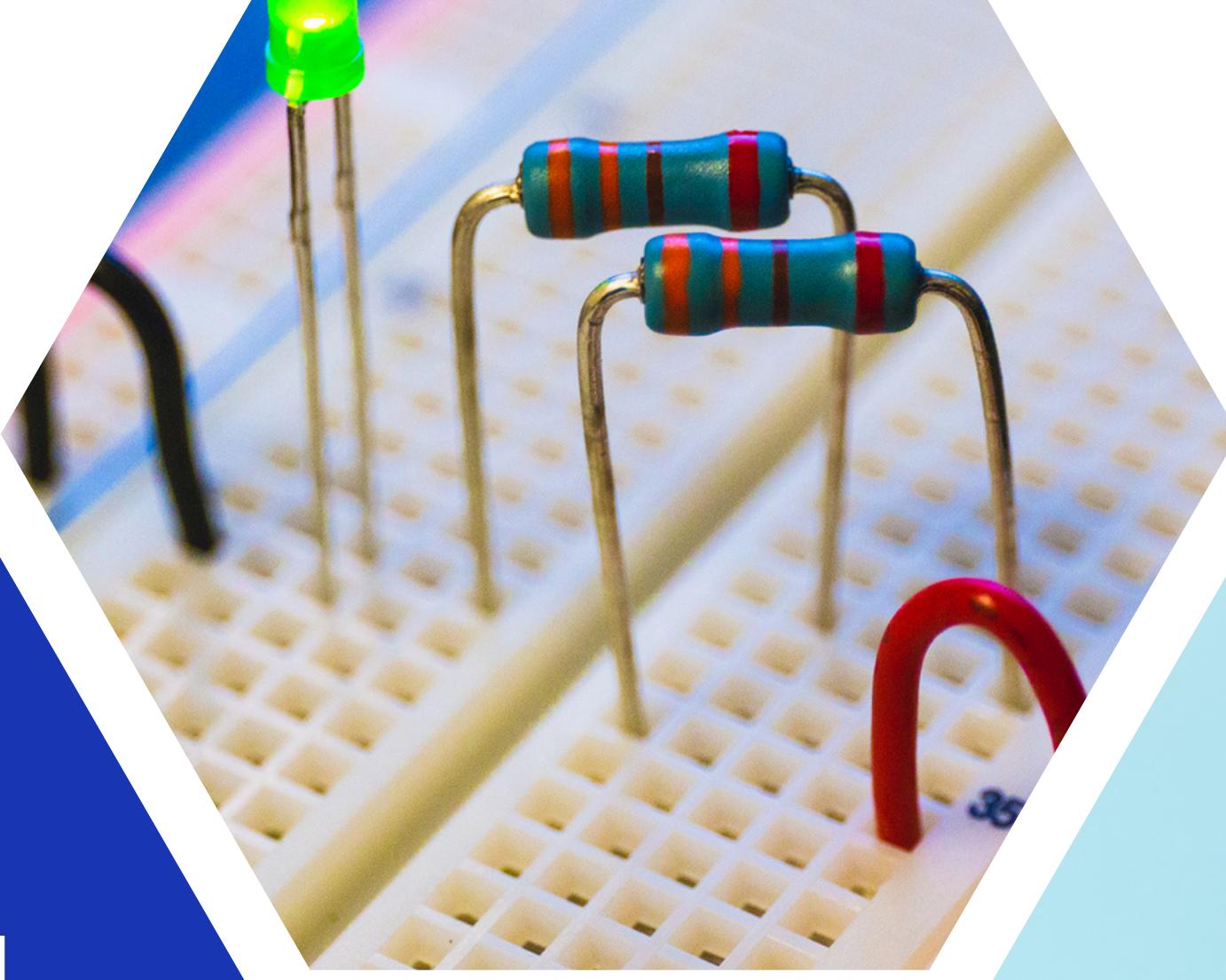
Permite el flujo del agua para el regado.

CABLES PARA PROTOBOARD

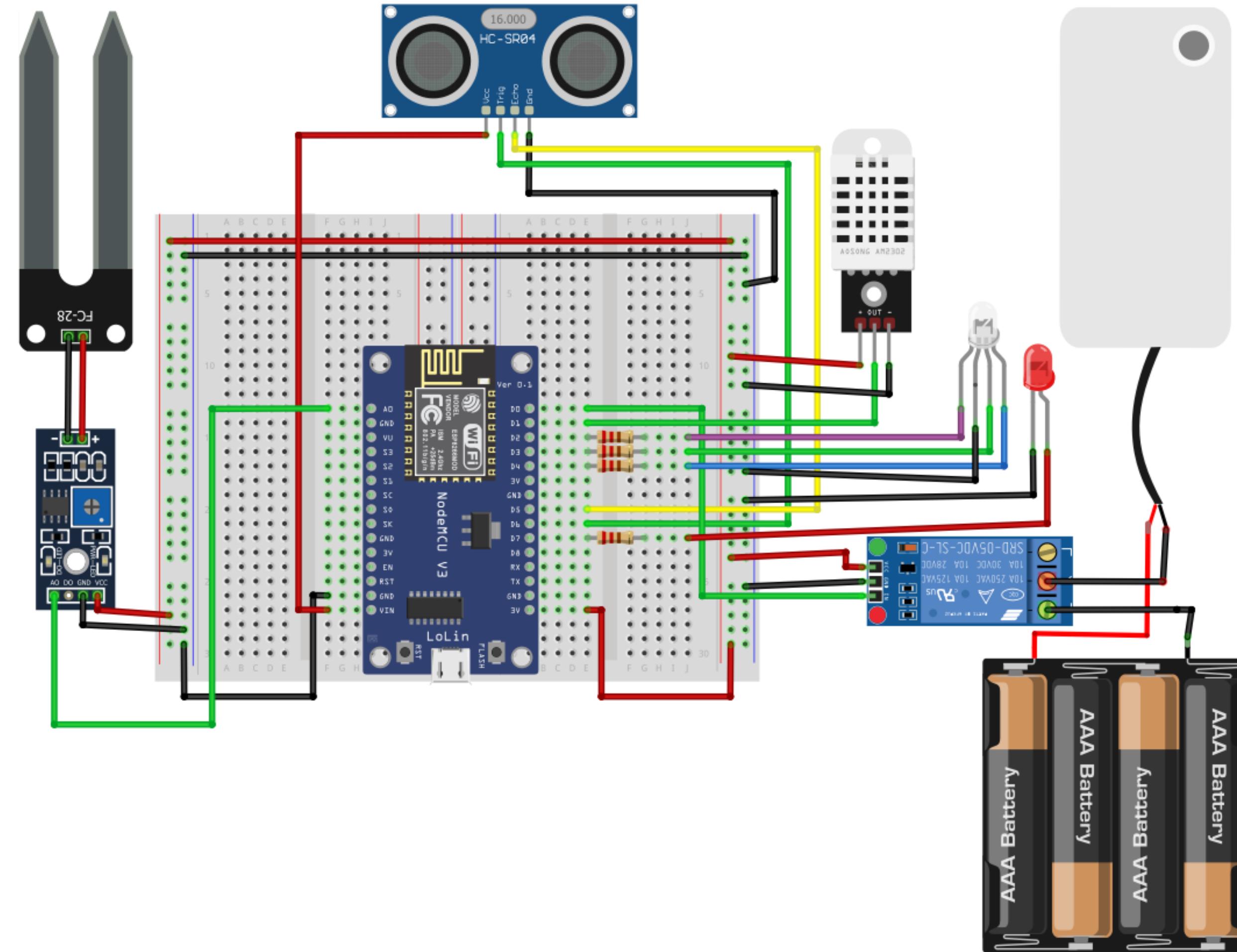
Cables jumper de 15cm de largo terminados como macho a hembra.



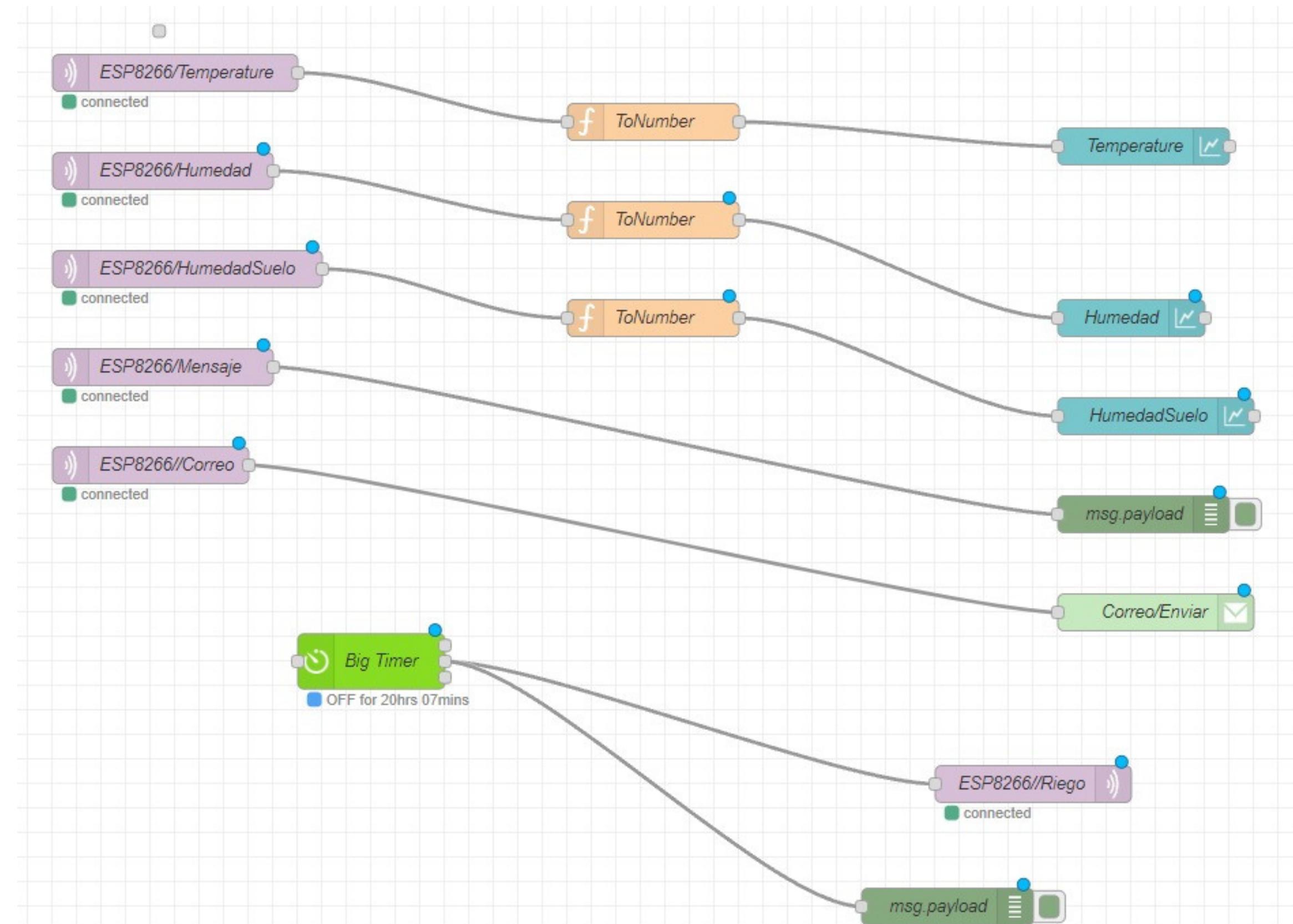
VI IMPLEMENTACIÓN DEL SISTEMA



D - E - L - A - G - R - A - M - A



NODE RED



BASE DE DATOS

PYTHON

```

import serial
import xlwt
from datetime import datetime

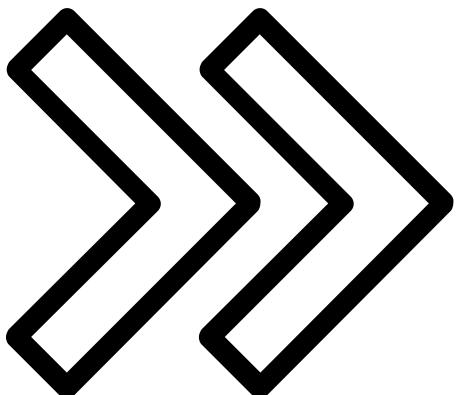
class SerialToExcel:

    def __init__(self,port,speed):
        self.port = port
        self.speed = speed

        self.wb = xlwt.Workbook()
        self.ws = self.wb.add_sheet("Data from Serial",cell_overwrite_ok=True)
        self.ws.write(0, 0, "Data from Serial")
        self.columns = ["Date Time"]
        self.number = 100
    def setColumns(self,col):
        self.columns.extend(col)
    def setRecordsNumber(self,number):
        self.number = number
    def readPort(self):
        ser = serial.Serial(self.port, self.speed, timeout=1)
        c = 0
        for col in self.columns:
            self.ws.write(1, c, col)
            c = c + 1
        self.fila = 2
        i = 0
        while(i<self.number):
            line = str(ser.readline())
            if(len(line) > 0):
                now = datetime.now()
                date_time = now.strftime("%m/%d/%Y, %H:%M:%S")
                print(date_time,line)
                if(line.find(",")>0):
                    c = 1
                    self.ws.write(self.fila, 0, date_time)
                    columnas = line.split(",")
                    for col in columnas:
                        self.ws.write(self.fila, c, col)
                        c = c + 1

            i = i + 1
            self.fila = self.fila + 1
    def writeFile(self,archivo):
        self.wb.save(archivo)

```

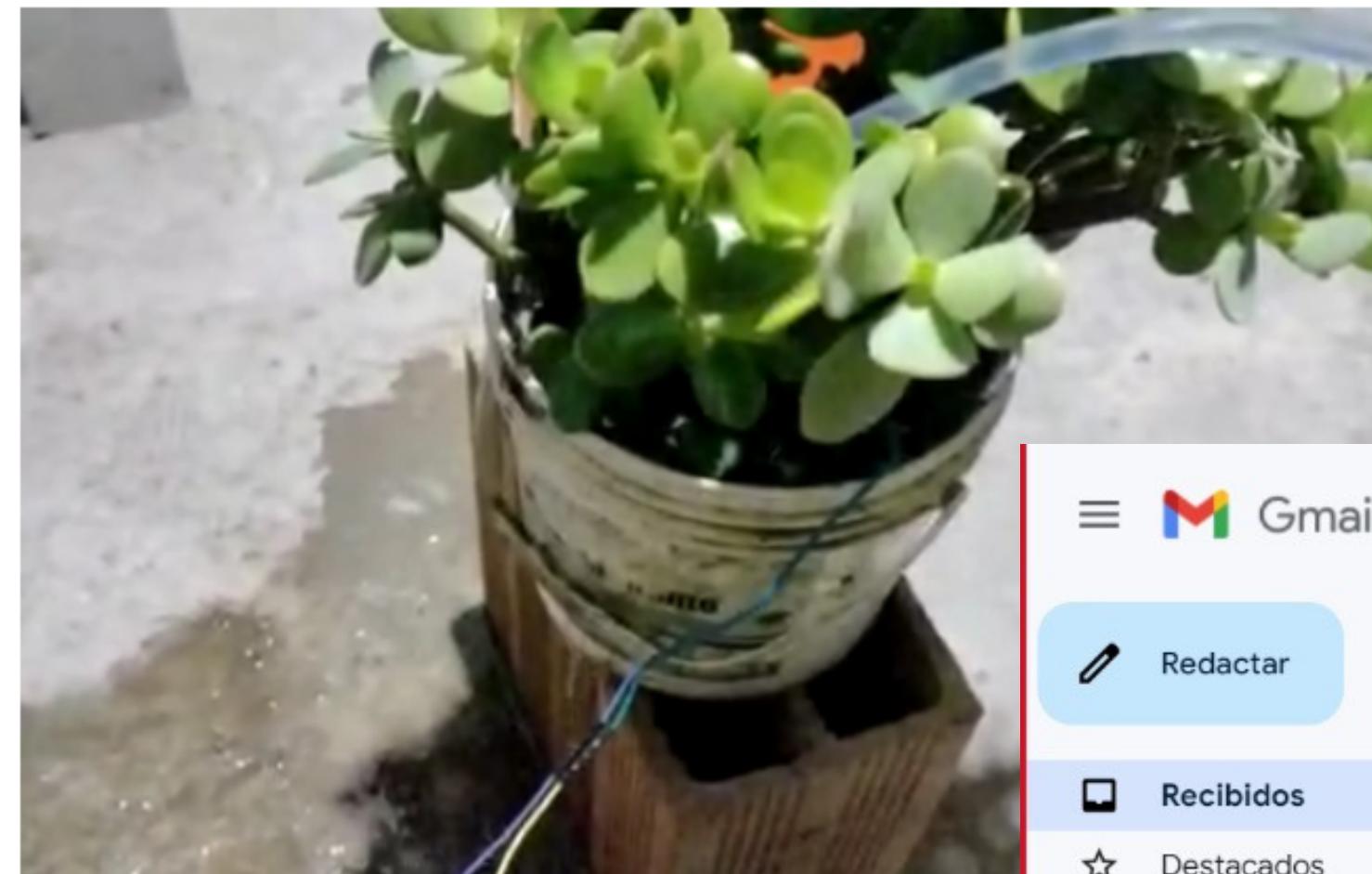
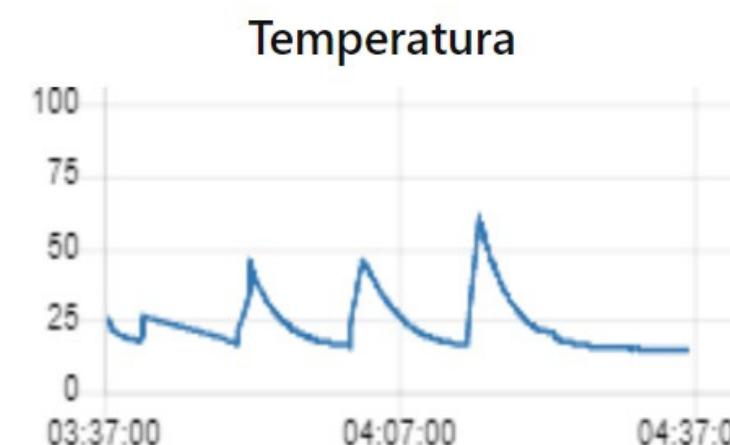
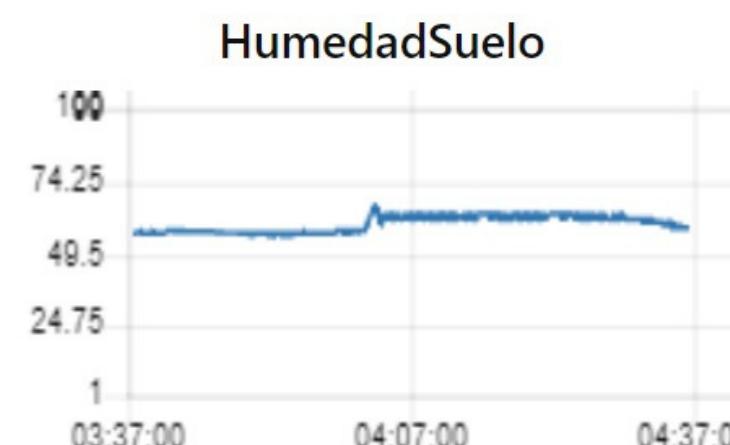
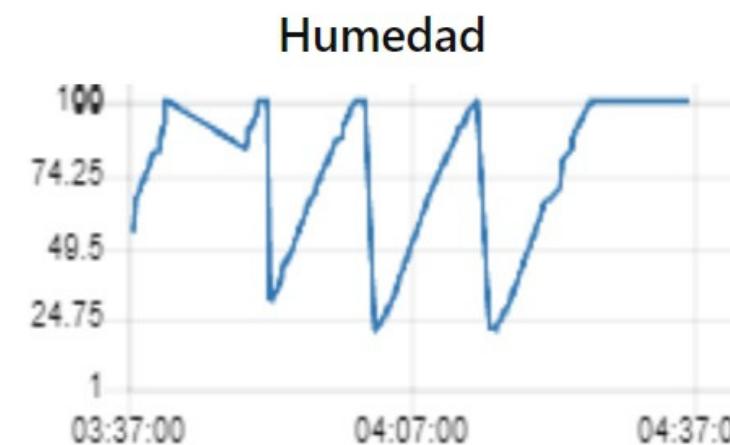


VIII

RESULTADOS

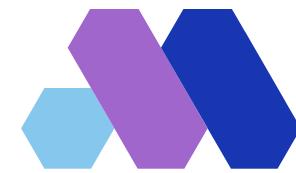
DASHBOARD

Gráficos-Sensores



Gmail

A screenshot of a Gmail inbox. The sidebar on the left shows navigation links: Redactar, Recibidos (509), Destacados, Pospuestos, Enviados, Borradores (2), and Más. The main area shows an incoming email from "jhonattan.malca@unmsm.edu.pe" with the subject "Llenar tanque". The email content includes a link to "45 mensajes suprimidos en esta conversación. Puedes ver lo". At the bottom right are "Responder" and "Reenviar" buttons.



CONCLUSIONES

- El desarrollo del sistema logró implementar correctamente el riego automático para regar la maceta solo en situaciones que sea necesario, según los datos arrojados por los sensores.
- La generación de dashboard logró mostrar la información arrojada por los sensores, lo cual hace una forma sencilla de interpretar por los usuarios.
- Los datos arrojados por los sensores se logran almacenar en una base de datos, lo cual beneficia a los usuarios, ya que podrán realizar informes según las fechas correspondientes.
-

¡GRACIAS!

