

# Desarrollo de un sistema de monitorización y automatización de un invernadero

Autor: Víctor Rincón  
Tutor: David Roldán



# Índice

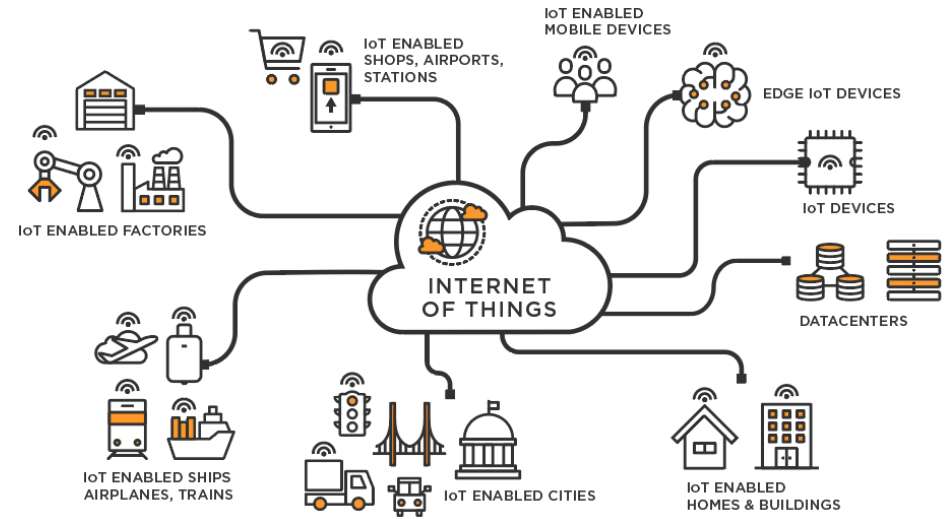
1. Introducción
2. Objetivos
3. Herramientas
4. Diseño e implementación
5. Casos de uso
6. Conclusiones

# INTRODUCCIÓN



# Internet de las Cosas (IoT)

- **IoT**: interconectividad de objetos a través de la red.
- Tratamiento masivo de datos, Smart Cities...



# Tecnologías web

- Tecnologías del lado del **cliente** (*front-end*).
- Tecnologías del lado del **servidor** (*back-end*).
- **Bases de datos.**

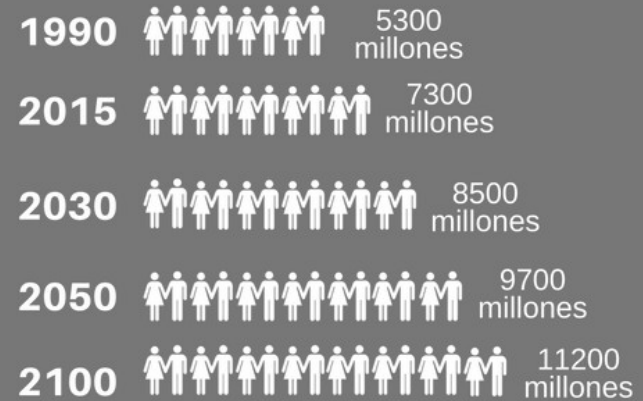


# Agricultura autónoma

- Objetivos de desarrollo sostenible.
- Agricultura de precisión y digitalización de la agricultura.
- Optimización y aprovechamiento de los recursos.
- Agricultura a pequeña escala y consumo local de alimentos.

## Población mundial

*Población mundial proyectada hasta 2100*



Fuente: Revisión de 2015 de la publicación World Population Prospects (Perspectivas demográficas mundiales)  
División de Población del Departamento de Asuntos Económicos y Sociales de las Naciones Unidas.  
Producción: Departamento de Información Pública

# OBJETIVOS

# Objetivos generales

1. Montaje **físico** del invernadero.
2. Montaje de la **electrónica**. Sensores y actuadores.
3. Scripts para realizar la **recolección y almacenamiento** de datos a partir de sensores.
4. Desarrollo de **aplicaciones de monitorización y visualización**.
5. Desarrollo de scripts de **automatización**.



# Plan de trabajo

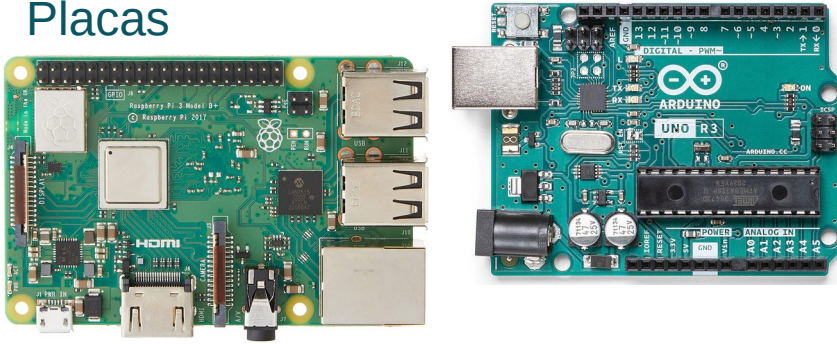
1. **Estudio previo:** invernaderos inteligentes, sensores y actuadores...
2. **Diseño del sistema distribuido** a nivel software y hardware.
3. **Iniciación e investigación** sobre las tecnologías utilizadas.
4. **Implementación de los sensores**, testeo, recolección de datos y subida a la base de datos.
5. Desarrollo de la **aplicación web**.
6. Desarrollo del **bot de Telegram**.
7. **Inclusión de los actuadores** y desarrollo de los scripts de **automatización** del invernadero.
8. **Control de errores:** scripts capaces de notificar al usuario vía Telegram sobre estos.

# HERRAMIENTAS UTILIZADAS



# Herramientas hardware

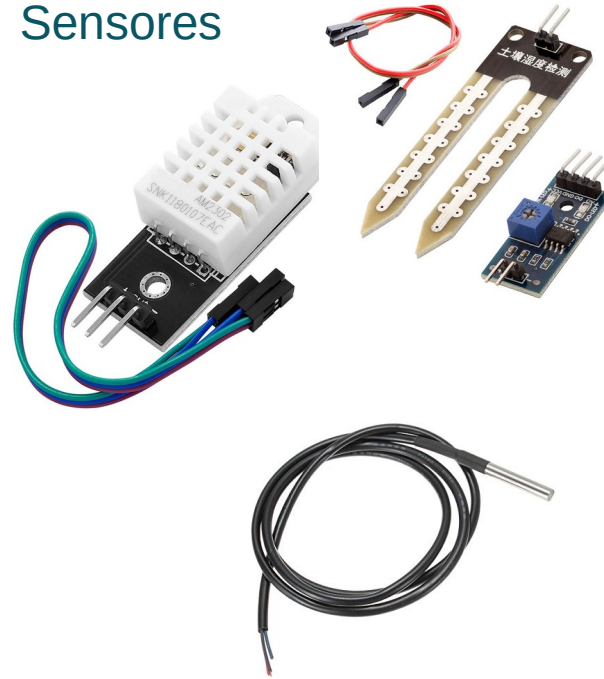
## Placas



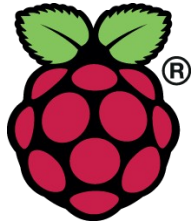
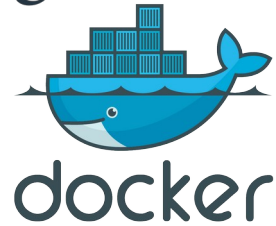
## Actuadores



## Sensores



# Herramientas software

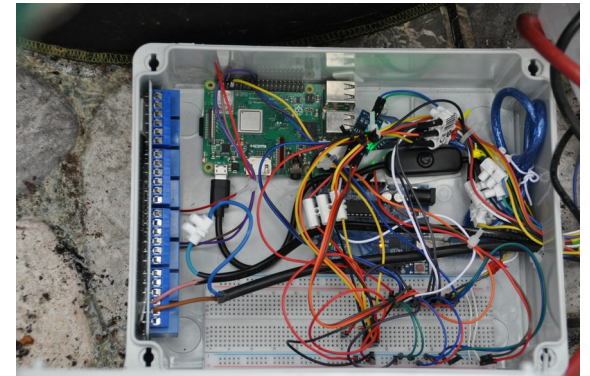


# DISEÑO E IMPLEMENTACIÓN



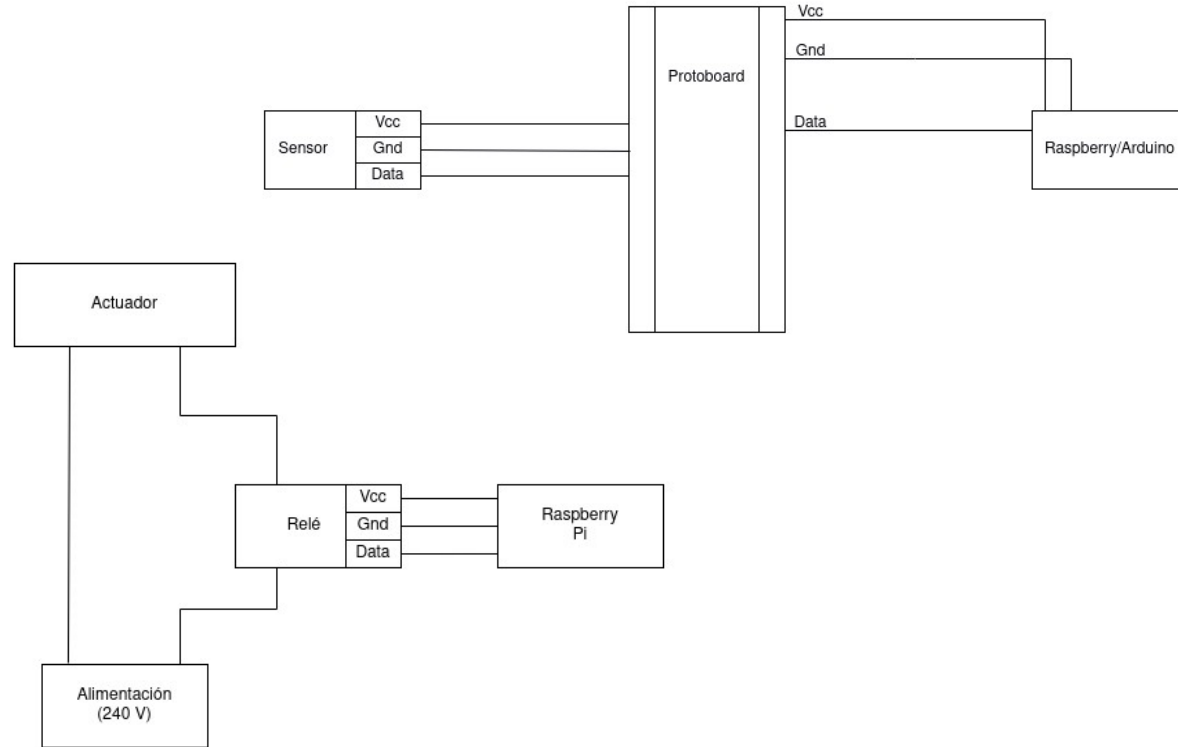
# Montaje físico del invernadero

- Invernadero y sacos de cultivo.
- Depósito de agua, bomba y mangueras de riego.
- Caja para proteger la electrónica.



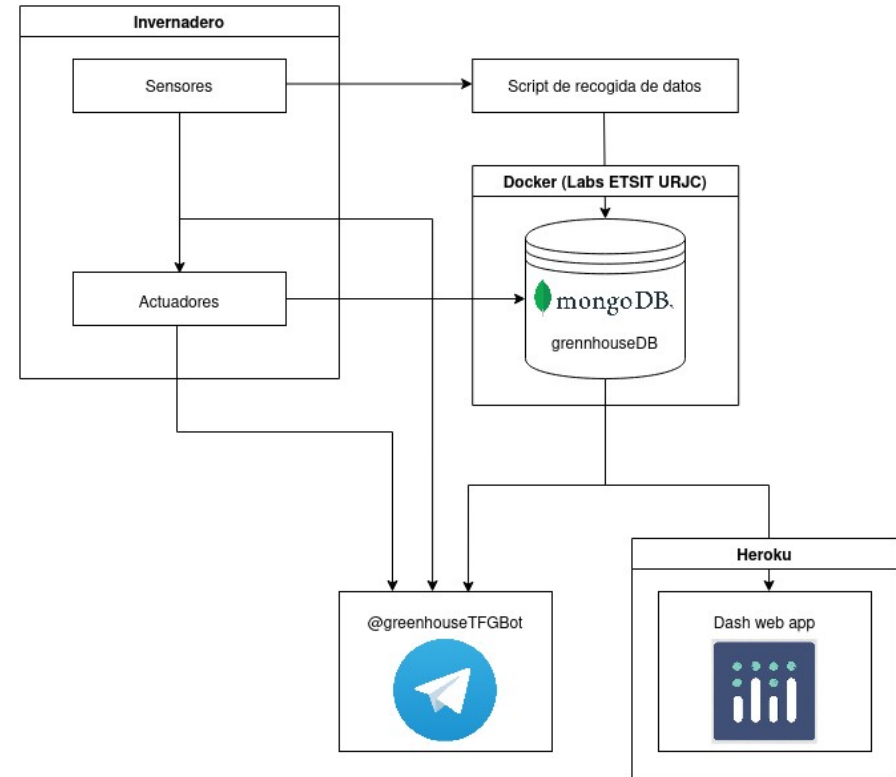
# Diseño hardware

- DHT22 1, GPIO 17
- DHT22 2, GPIO 27
- DS18B20 1 y DS18B20 2, GPIO 04
- YL-69, A0
- YL-69, A1
- Bomba, GPIO 24



# Diseño software

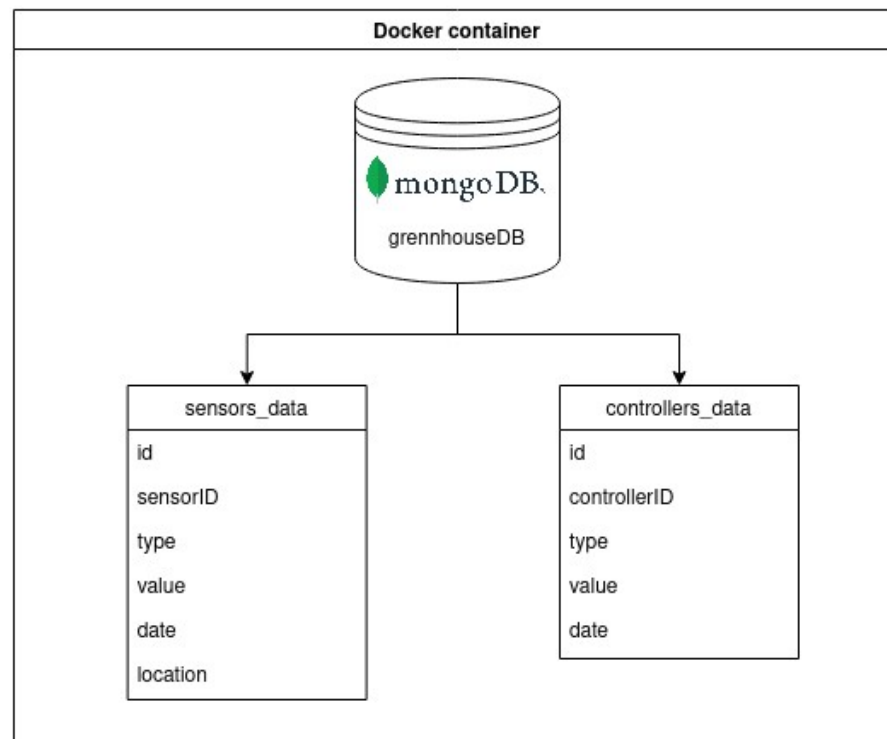
La **base de datos** es el **centro neurálgico** del sistema.





# Diseño software: base de datos

- Alojada en los **labs** de la **ETSIT** usando **Docker**.
- **Copia de seguridad** una vez al día.



# Diseño software: recogida de datos

- Se realiza en la **Raspberry** usando **Cron**.
- **Diferentes módulos:** Adafruit, w1thermsensor, pyfirmata2.
- Si hay un **error**: se almacena “-1”.

# Diseño software: aplicación web

- Basada en **Dash**, Plotly.
- Desplegada usando **Heroku**.



## Menu

HOME

SENSORS DATA ▾

CONTROLLERS DATA ▾

DOCUMENTATION

## My Final Project Degree

Develop of an automation and monitoring distributed system for a greenhouse

This is a data visualization web app developed as a part of the system that has been built as a final project degree for the Telematics Engineering degree. The project consists of an autonomous distributed system for monitoring and care of a greenhouse.

This app is based on Python, Dash and Plotly for the data visualization. All data is retrieved from the different sensors and processed using a Raspberry Pi 3B+ and an Arduino and sended to a MongoDB database hosted in a Docker container in the ETSIT labs from Universidad Rey Juan Carlos. You can have a look to their webpage [here](#)

The project has four clearly differentiated main parts, this visualization web application deployed using Heroku, the aforementioned data collection system, a Python app for the autonomous control of the greenhouse and a Telegram bot that is used not only for monitoring the whole system but also to obtain real-time data as well as getting it from the database, this bot is deployed in the Raspberry Pi 3B+.

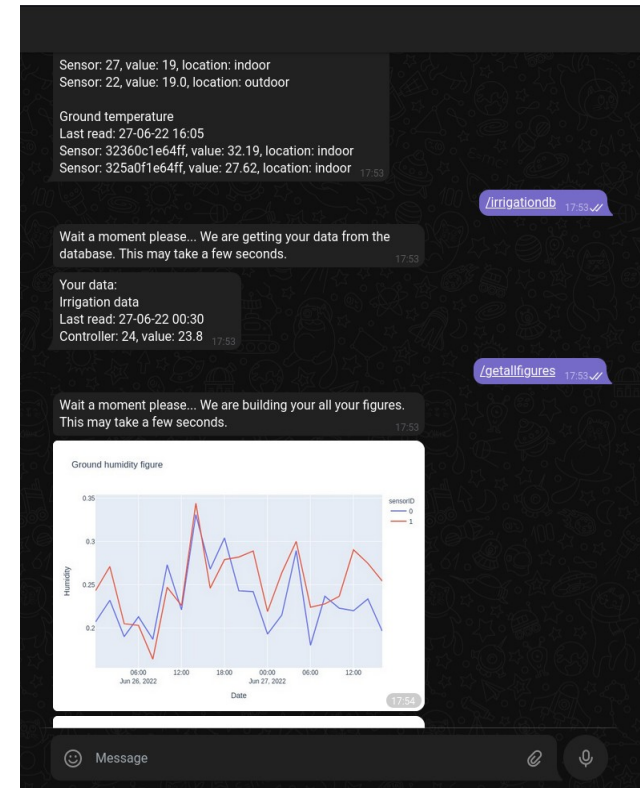
If you are interested you can have a look to the documentation section to learn more about it.

Developed by: [V́ctor Rincón Yepes](#)

Advised and tutored by: [David Roldán Alvarez](#)

# Diseño software: bot de Telegram

- Desplegado en la **Raspberry**.
- Puede obtener:
  - Datos en **tiempo real** y desde la **base de datos**.
  - Formato datos: **textual** y **gráficas**.
  - **Notificación** de errores y acciones.



# Diseño software: aplicación de riego

- Ejecutandose en la **Raspberry**.
- **Notificación** de acciones vía **Telegram**.
- Mantener valores en: 25-45 % de humedad.

Irrigation in bomb 24 activated. 13.3 liters used. 00:30







# CASOS DE USO



# Ejemplo: humedad de la tierra

- Recogida y almacenamiento de datos.
- Visualización en Dash y en Telegram.

# Ejemplo: humedad de la tierra

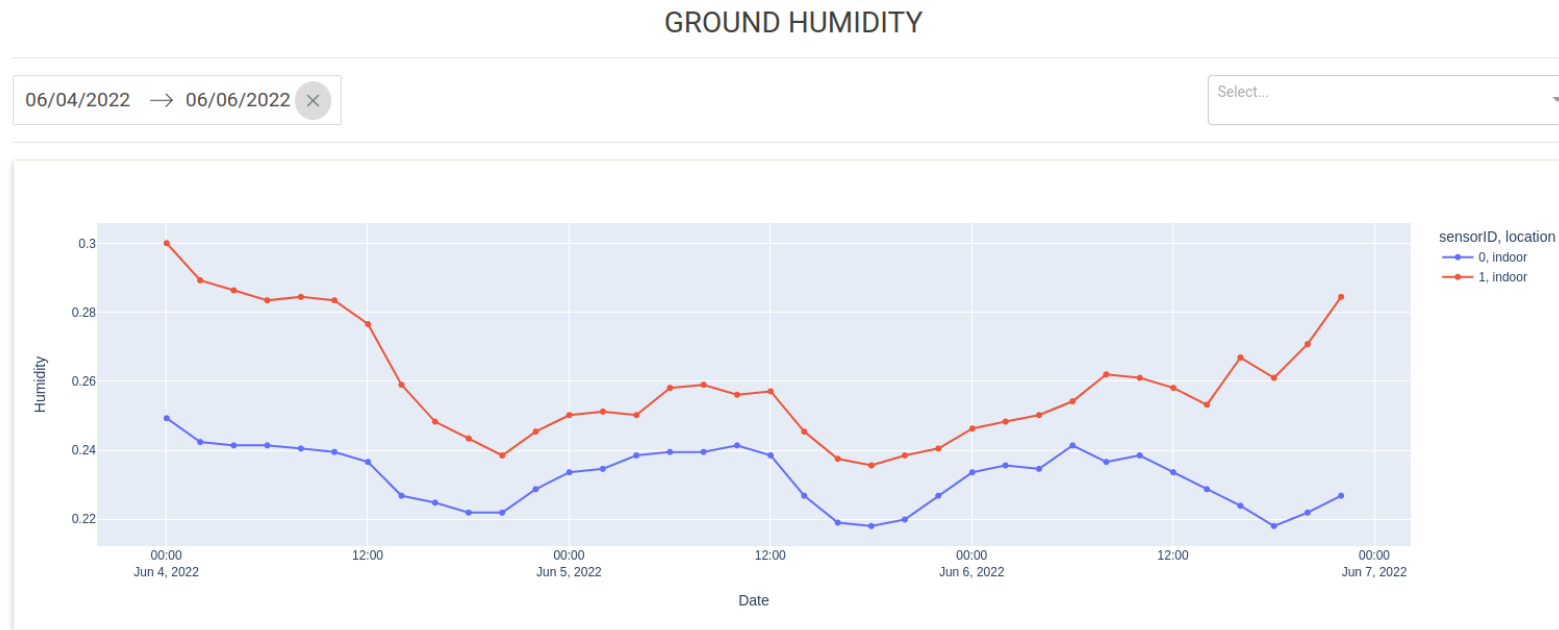
 \_id  
 type  
 value  
 sensorID  
 location  
 date

ObjectId("62b17ac35bacf6cbda4edbf6")  
ground humidity  
0.25  
0  
indoor  
2022-06-21 10:01:07.804Z

ObjectId  
String  
Double  
Int32  
String  
Date

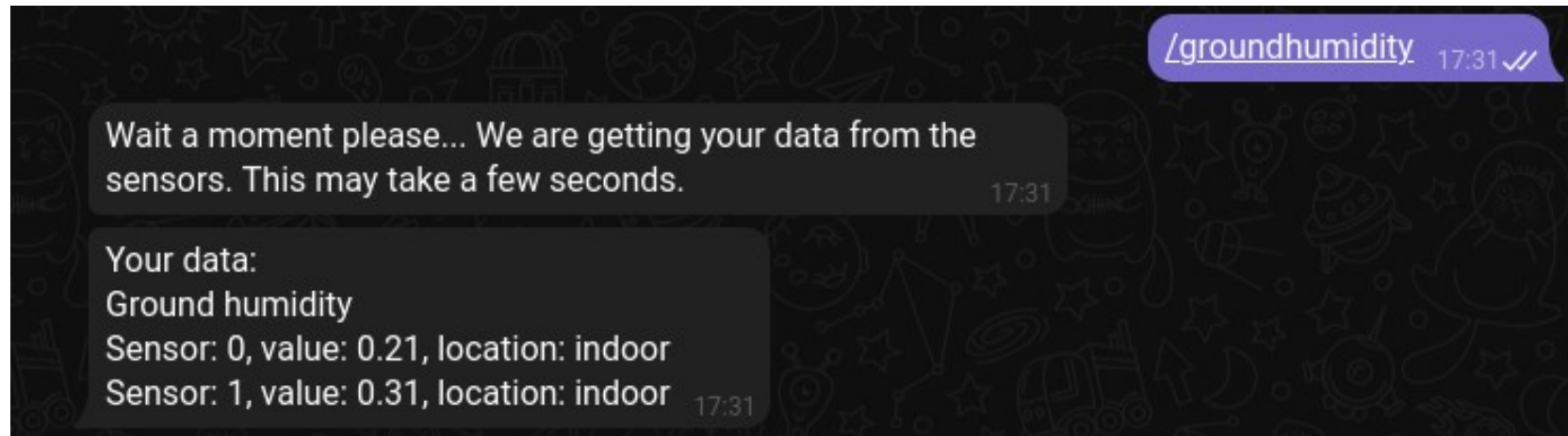


# Ejemplo: humedad de la tierra



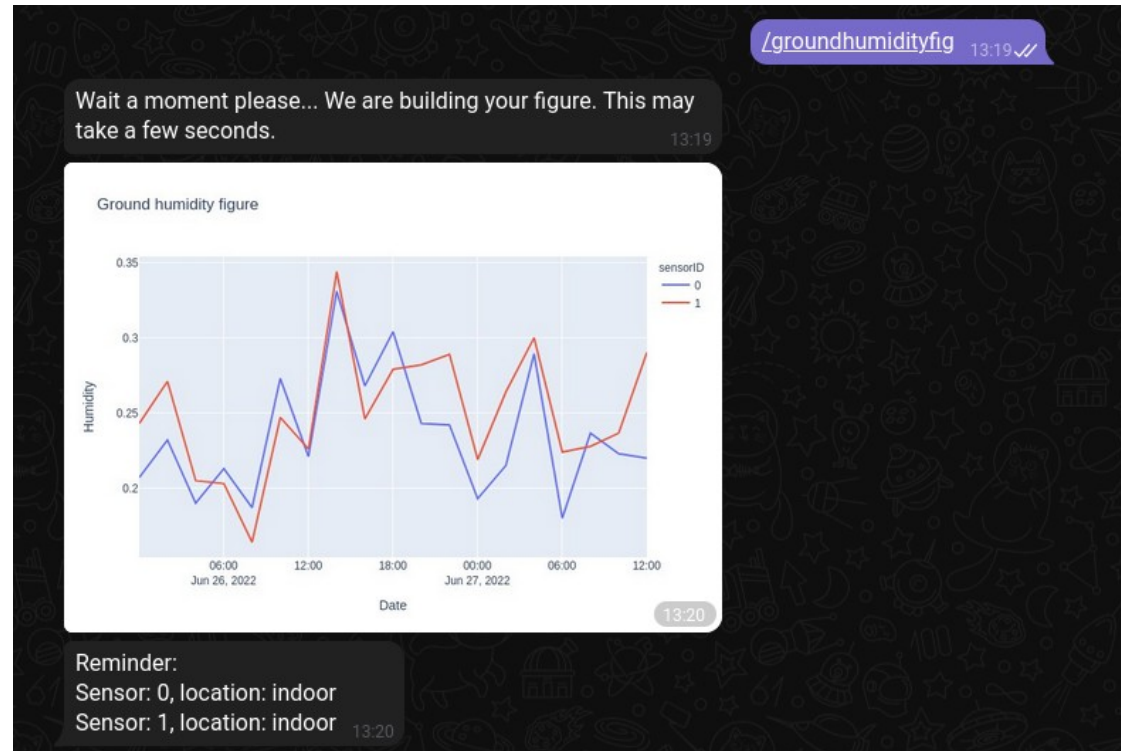
Ground humidity graph.

# Ejemplo: humedad de la tierra

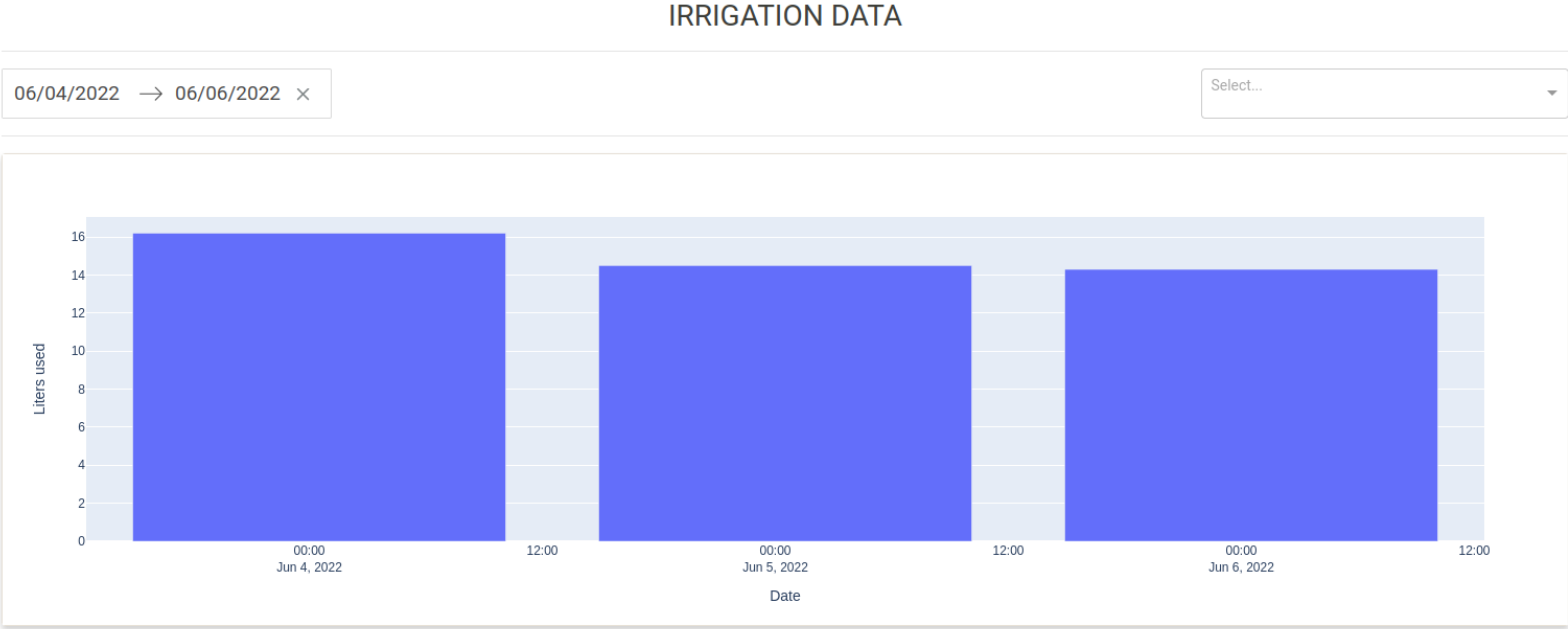


# Ejemplo: humedad de la tierra

- Posibilidad de filtrar mediante parámetros opcionales:  
**--startdate**  
**--enddate**



# Ejemplo: humedad de la tierra

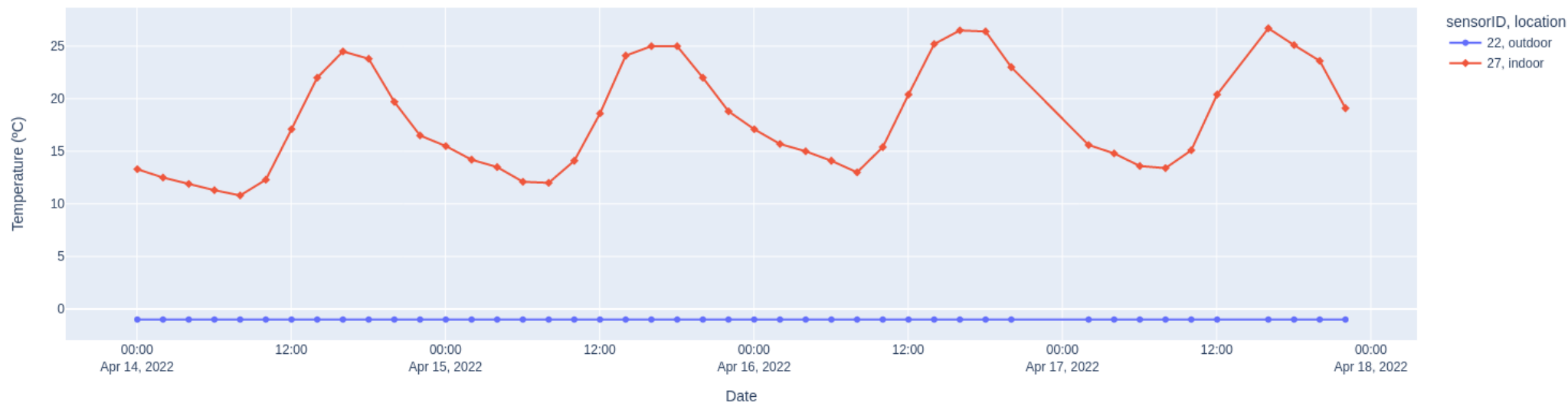


Irrigation data graph.

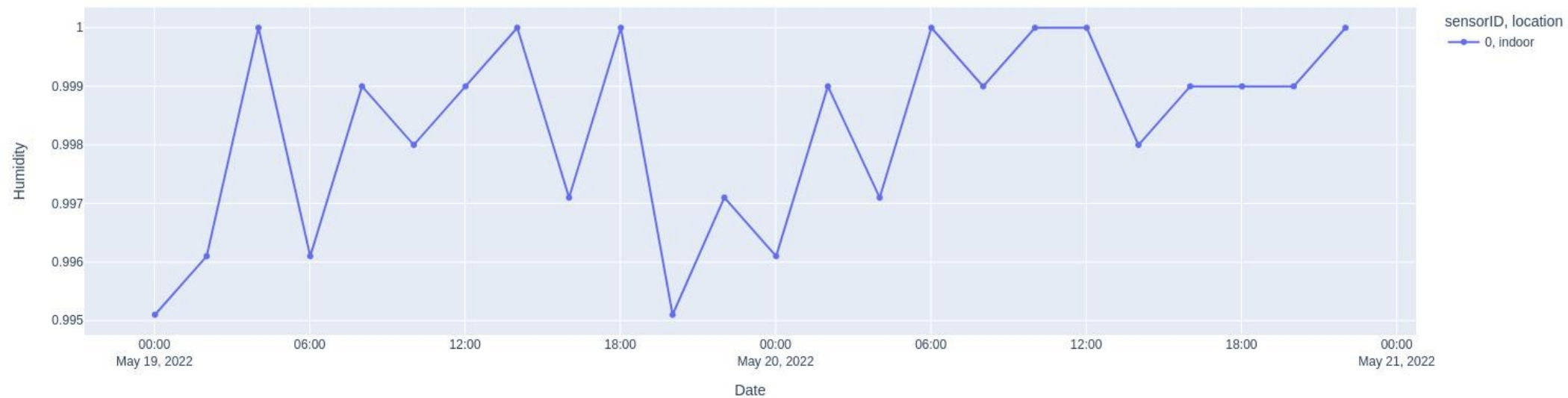
# Ejemplo: humedad de la tierra



# Detección de errores



# Detección de errores



# Detección de errores

Warning: Data collected between humidity sensors might be wrong. Sensor: 0->0.9238, Sensor: 1->0.1349

00:30

Error in sensor reading

Sensor 27, type: ambient temperature, location: indoor, failed at  
11-06-22 18:01:24

18:10



# CONCLUSIONES

# Coste total del proyecto

1. Caja hermética para electrónica: 5 €
2. Raspberry Pi 3B+ y fuente de alimentación: 45€
3. Sensores (DHT22, YL-69 y DS18B20): 25€
4. Actuadores (bomba CESFONJER): 15€
5. Módulo de relés: 11€
6. Resto de componentes electrónicos (cables y Protoboard): 20€

**Total: 244€**

# Trabajos futuros

1. Soporte para nuevos tipos de sensores.
2. Añadir nuevos actuadores al sistema.
3. Mejora de los componentes.
4. Hacer el sistema completamente autónomo.

# Vídeo demonstrativo

<https://www.youtube.com/watch?v=9Y3vQupzTOg>

**¡Muchas gracias!**