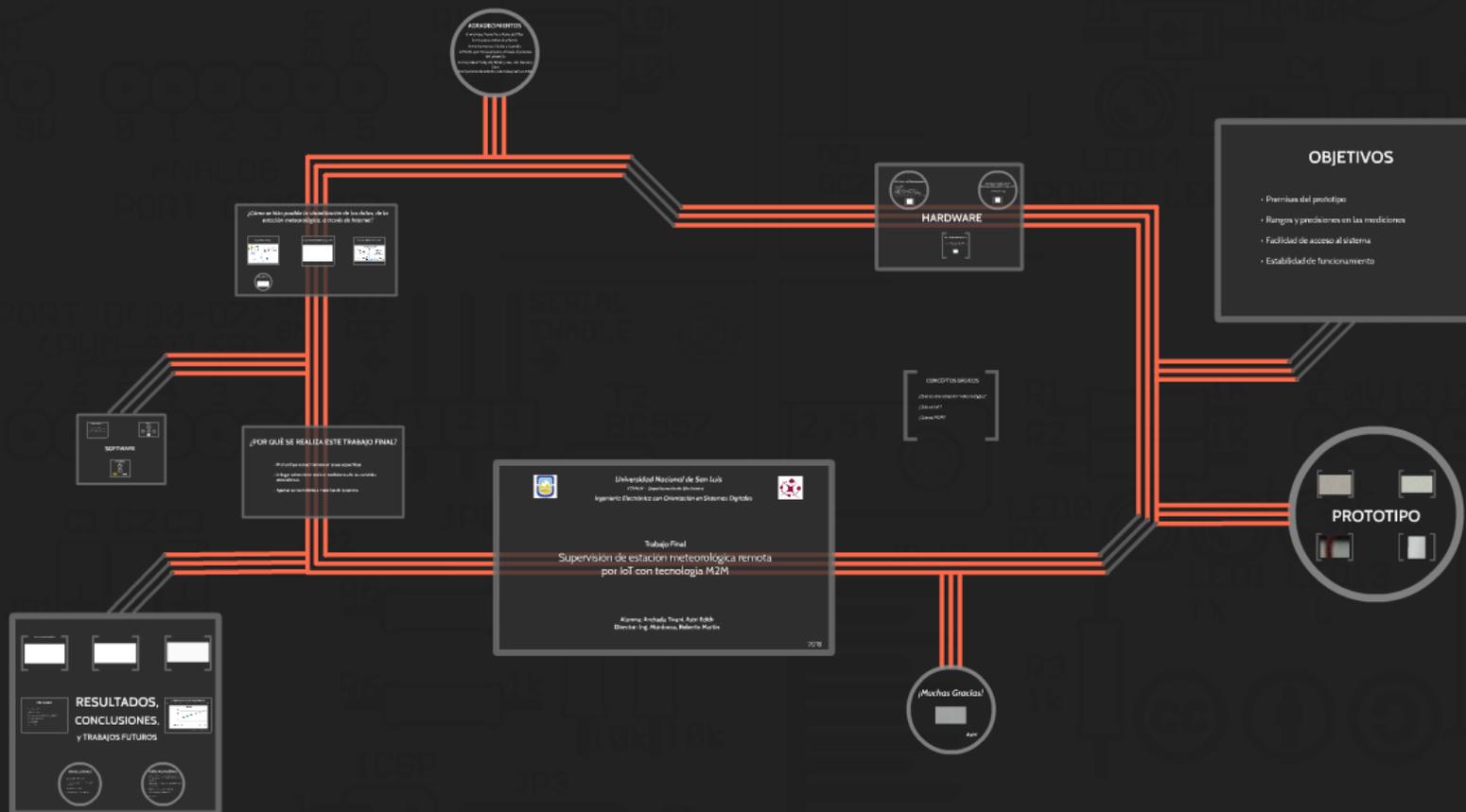
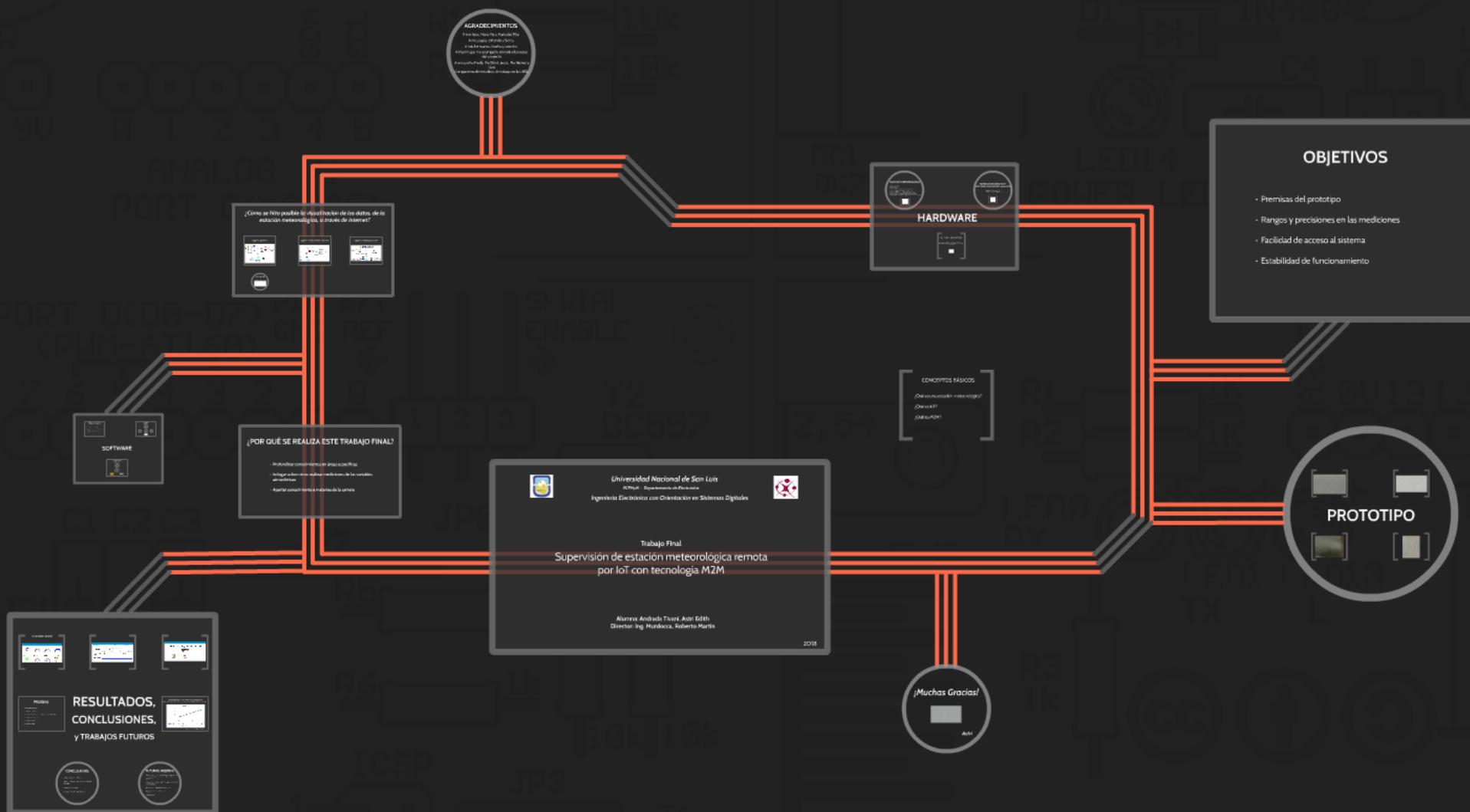


# Supervisión de estación meteorológica remota por IoT con tecnología M2M



# Supervisión de estación meteorológica remota por IoT con tecnología M2M





**Universidad Nacional de San Luis**

*FCFMyN - Departamento de Electrónica*

*Ingeniería Electrónica con Orientación en Sistemas Digitales*



**Trabajo Final**

**Supervisión de estación meteorológica remota  
por IoT con tecnología M2M**

**Alumna: Andrada Tivani, Astri Edith  
Director: Ing. Murdocca, Roberto Martín**

2018

# ¿POR QUÉ SE REALIZA ESTE TRABAJO FINAL?

- Profundizar conocimientos en áreas específicas
- Indagar sobre cómo realizar mediciones de las variables atmosféricas
- Aportar conocimiento a materias de la carrera

# OBJETIVOS

- Premisas del prototipo
- Rangos y precisiones en las mediciones
- Facilidad de acceso al sistema
- Estabilidad de funcionamiento

# CONCEPTOS BÁSICOS

¿Qué es una estación meteorológica?

¿Qué es IoT?

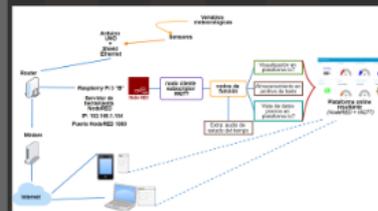
¿Qué es M2M?

# ¿Cómo se hizo posible la visualización de los datos, de la estación meteorológica, a través de Internet?

## ESQUEMA GENERAL



## ESQUEMA HERRAMIENTA Node-RED



## ESQUEMA PROTOCOLO MQTT

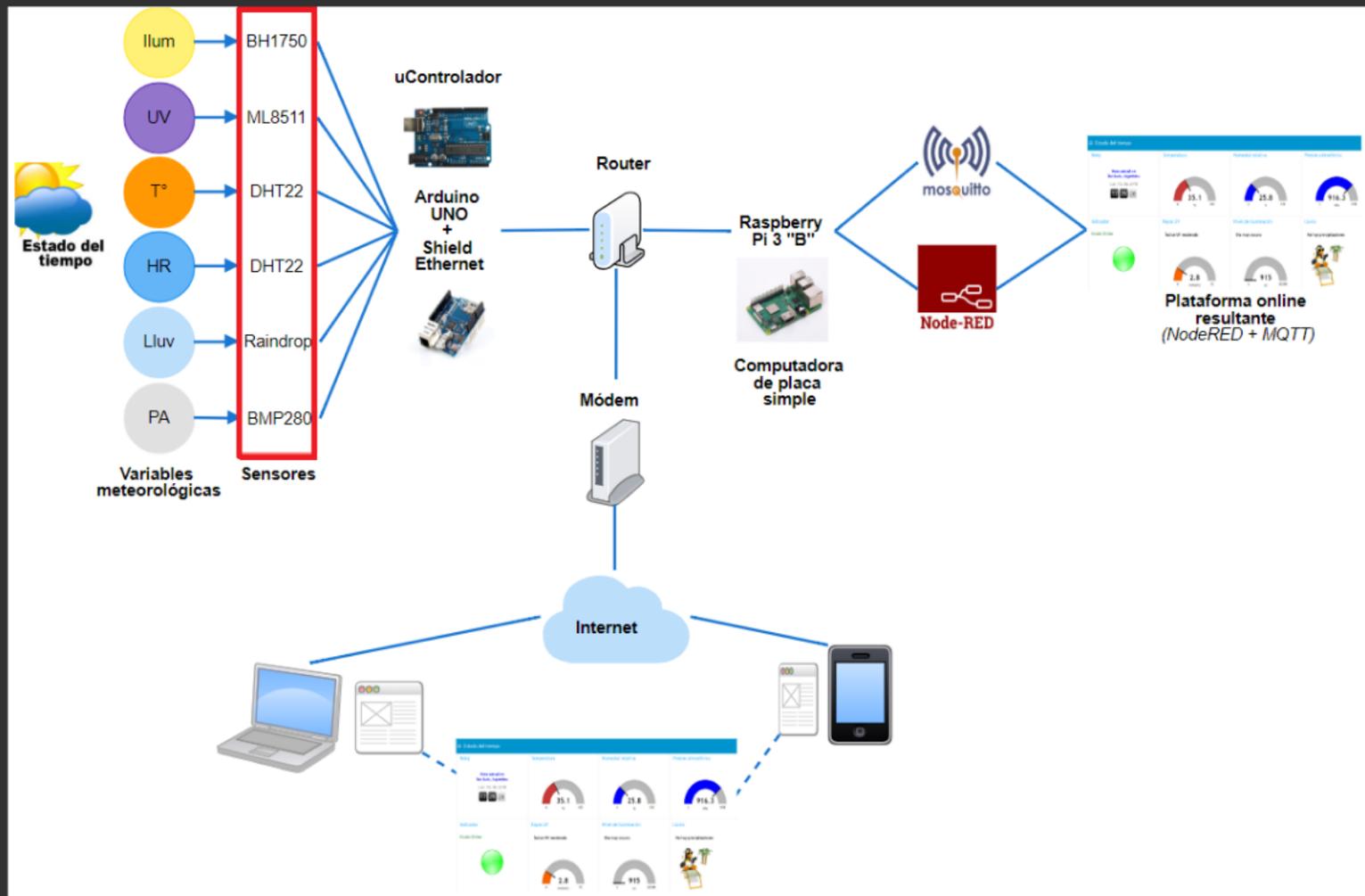


## DIRECCIÓN WEB

DOMS configurados en el router con el servidor NoIP

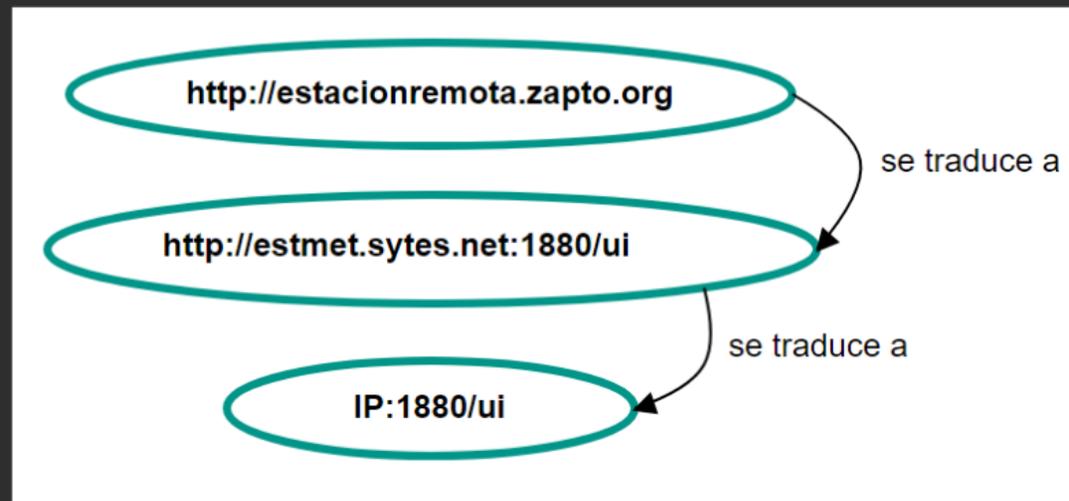


# ESQUEMA GENERAL



# DIRECCIÓN WEB

DDNS configurado en el router con el servidor NoIP



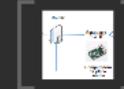
## ESTACIÓN METEOROLÓGICA

- Arduino UNO R3
- Shield Ethernet
- Sensor DHT22 - para temperatura y humedad
- Sensor BMP280 - para presión atmosférica
- Sensor BH1750 - para intensidad de luz ambiental
- Sensor ML8511 - para intensidad de luz ultravioleta
- Sensor YL-83 / Módulo Raindrop - para presencia de lluvia



## SERVIDOR (BROKER) MQTT SERVIDOR HERRAMIENTA Node-RED

- Raspberry Pi 3 modelo "B"
- Router TP-LINK TL-WR741ND



# HARDWARE

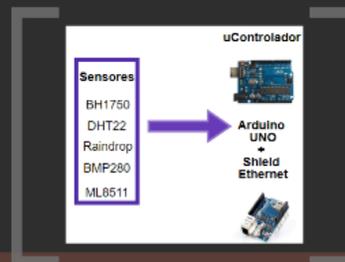
## PoE - POWER OVER ETHERNET

Permite que la alimentación eléctrica se suministre a un dispositivo de red usando el mismo cable que se utiliza para la conexión de red.



# ESTACIÓN METEOROLÓGICA

- Arduino UNO R3
- Shield Ethernet
- Sensor DHT22 - para temperatura y humedad
- Sensor BMP280 - para presión atmosférica
- Sensor BH1750 - para intensidad de luz ambiental
- Sensor ML8511 - para intensidad de luz ultravioleta
- Sensor YL-83 / Módulo Raindrop - para presencia de lluvia



## uControlador

### Sensores

BH1750

DHT22

Raindrop

BMP280

ML8511

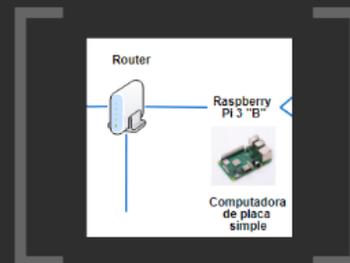


**Arduino  
UNO  
+  
Shield  
Ethernet**



# SERVIDOR (BROKER) MQTT SERVIDOR HERRAMIENTA Node-RED

- Raspberry Pi 3 modelo "B"
- Router TP-LINK TL-WR741ND



Router



Raspberry  
Pi 3 "B"



Computadora  
de placa  
simple

# PoE - POWER OVER ETHERNET

Permite que la alimentación eléctrica se suministre a un dispositivo de red usando el mismo cable que se utiliza para la conexión de red.





## FIRMWARE ARDUINO

- Es un software que maneja físicamente al hardware
- Las funciones que necesita realizar la placa, para cumplimentar su tarea dentro del proyecto:
  1. Lectura de valores provenientes de los sensores
  2. Realizar el procesamiento de medidas
  3. Enviar los datos procesados a Raspberry Pi 3 a través del protocolo MQTT

## MQTT



# SOFTWARE

## Node-RED

Programación basada en flujos para el Internet de las cosas



# FIRMWARE ARDUINO

- Es un software que maneja físicamente al hardware
- Las funciones que necesita realizar la placa, para cumplimentar su tarea dentro del proyecto:
  1. Lectura de valores provenientes de los sensores
  2. Realizar el procesamiento de medidas
  3. Enviar los datos procesados a Raspberry Pi 3 a través del protocolo MQTT

# Node-RED

Programación basada en flujos para el Internet de las cosas

## ¿QUÉ ES?

Node-RED es un editor de flujo basado en el navegador web donde se puede añadir o eliminar nodos y conectarlos entre sí con el fin de hacer que se comuniquen entre ellos.

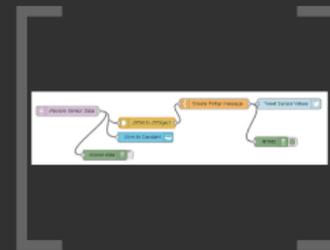
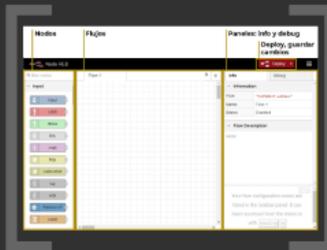
Con una gran cantidad de estos nodos para elegir, Node-Red facilita conectar los dispositivos de hardware, APIs y servicios en línea.

## ¿POR QUÉ USARLO?

Motor de flujos con enfoque IoT

Herramienta de programación visual

Herramienta de código abierto



# ¿QUÉ ES?

Node-RED es un editor de flujo basado en el navegador web donde se puede añadir o eliminar nodos y conectarlos entre sí con el fin de hacer que se comuniquen entre ellos.

Con una gran cantidad de estos nodos para elegir, Node-Red facilita conectar los dispositivos de hardware, APIs y servicios en línea.

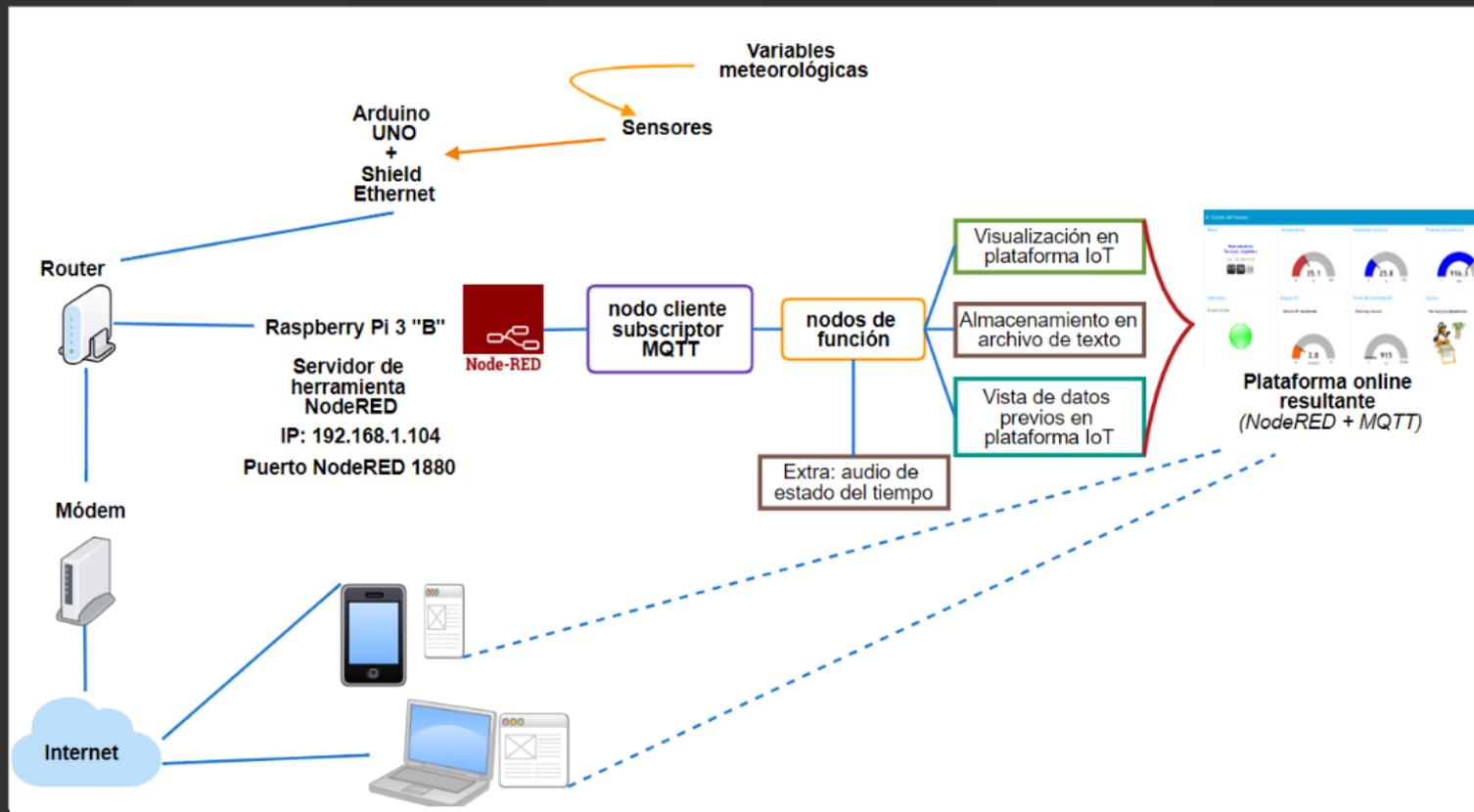
# ¿POR QUE USARLO?

Motor de flujos con enfoque IoT

Herramienta de programación visual

Herramienta de código abierto

# ESQUEMA HERRAMIENTA Node-RED

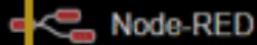


## Nodos

## Flujos

## Paneles: info y debug

### Deploy, guardar cambios

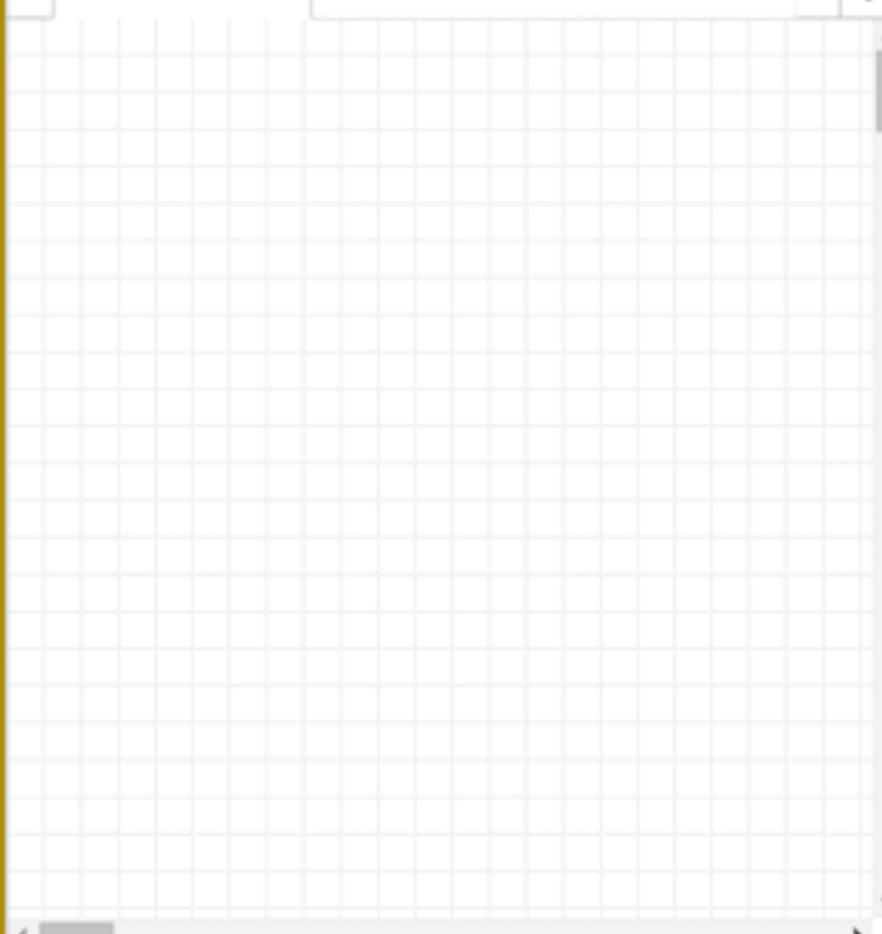


filter nodes

### input

- inject
- catch
- status
- link
- mqtt
- http
- websocket
- tcp
- udp
- Watson IoT
- serial

Flow 1



info

debug

### Information

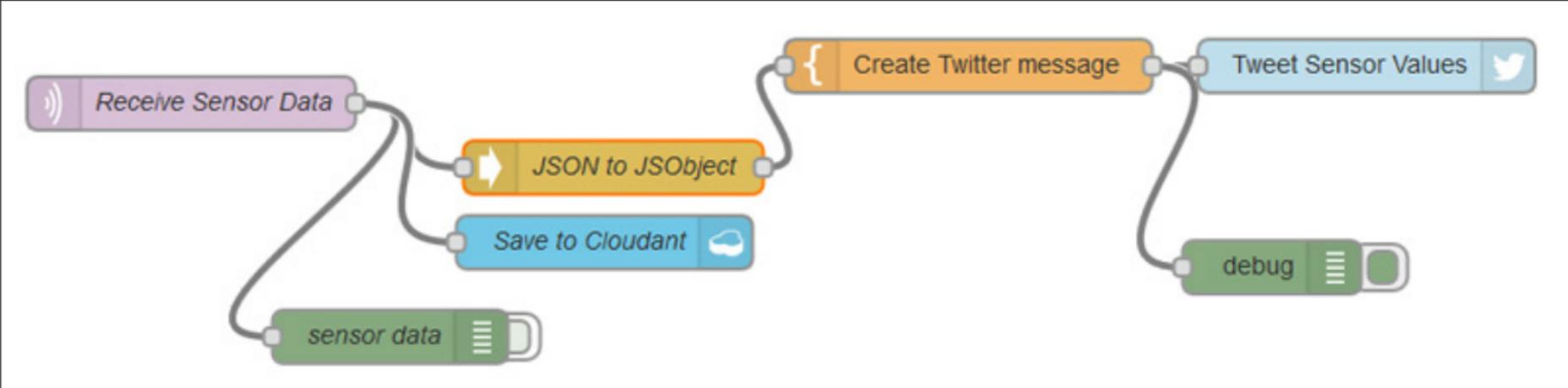
Flow	"52f90337.e450cc"
Name	Flow 1
Status	Enabled

### Flow Description

None

Your flow configuration nodes are listed in the sidebar panel. It can be accessed from the menu or

with `ctrl-g` `c`



# MQTT

## ¿QUÉ ES?

Es un protocolo de mensajería asíncrona  
Desarrollado por IBM  
Enfocado a la conectividad Machine-to-Machine (M2M)  
Utiliza un modelo de publicación y suscripción

## IMPLEMENTACIÓN

**MQTT Broker**  
Se instala en Raspberry Pi 3  
Se elige Mosquitto de la Fundación Eclipse

**MQTT Cliente Arduino**  
Se utiliza una librería para implementar el cliente  
Arduino Client for MQTT

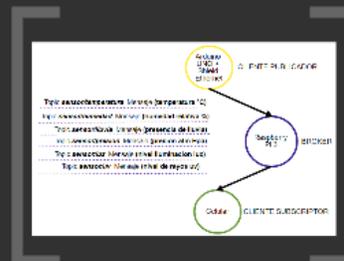
## ARQUITECTURA

**Servidor / Broker MQTT**  
- Gestiona la red  
- Transmite los mensajes.

**Cliente MQTT**  
- Publicar y/o un suscriptor de mensajes en topics  
- Siempre establece conexión con el servidor (Broker)

## TOPIC

Identificador que define el contenido del mensaje  
Emisores y receptores deben estar suscritos a un  
topic común para poder establecer la comunicación  
Se organizan según una estructura jerárquica en  
árboles



# ¿QUÉ ES?

Es un protocolo de mensajería asíncrona

Desarrollado por IBM

Enfocado a la conectividad *Machine-to-Machine* (M2M)

Utiliza un modelo de publicación y suscripción

# ARQUITECTURA

## Servidor / Broker MQTT

- Gestionar la red
- Transmitir los mensajes.

## Cliente MQTT

- Publicador y/o un suscriptor de mensajes en *topics*
- Siempre establece conexión con el servidor (Broker).

# TOPIC

Identificador que define el contenido del mensaje

Emisores y receptores deben estar suscritos a un topic común para poder establecer la comunicación

Se organizan según una estructura jerárquica en árbol

# IMPLEMENTACIÓN

## **MQTT Broker**

Se instala en Raspberry Pi 3

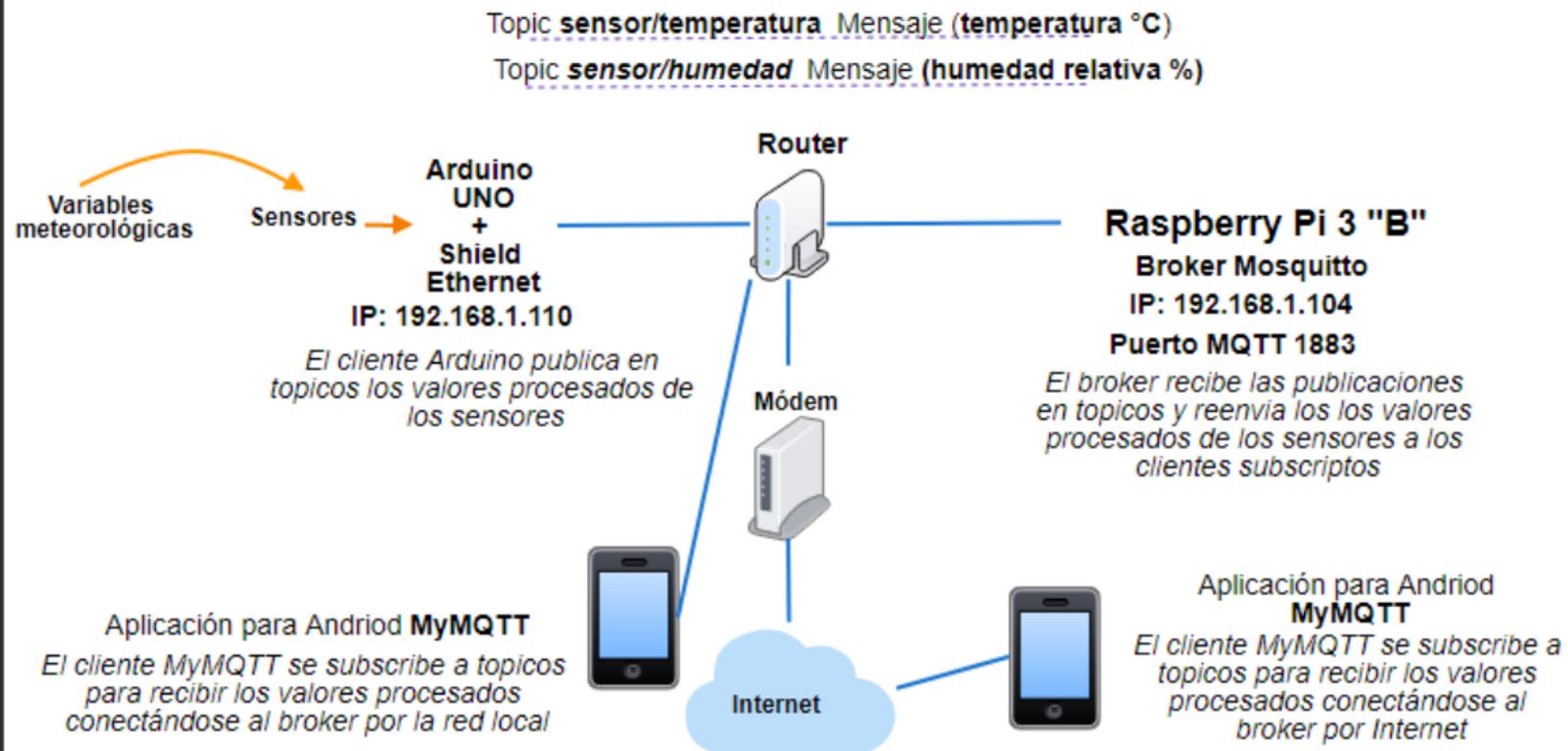
Se elige Mosquitto de la Fundación Eclipse

## **MQTT Cliente Arduino**

Se utiliza una librería para implementar el cliente

*Arduino Client for MQTT*

# ESQUEMA PROTOCOLO MQTT





CLIENTE PUBLICADOR

Topic **sensor/temperatura** Mensaje (**temperatura °C**)

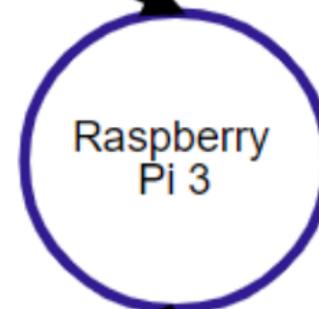
Topic **sensor/humedad** Mensaje (**humedad relativa %**)

Topic **sensor/lluvia** Mensaje (**presencia de lluvia**)

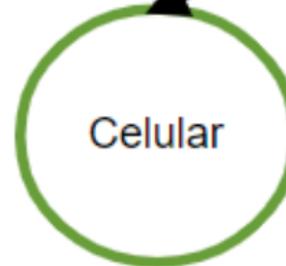
Topic **sensor/presion** Mensaje (**presion atm Hpa**)

Topic **sensor/luz** Mensaje (**nivel iluminacion lux**)

Topic **sensor/uv** Mensaje (**nivel de rayos uv**)



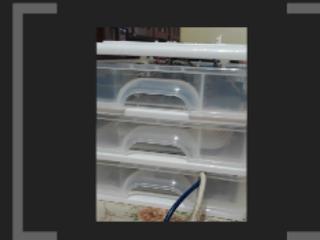
BROKER



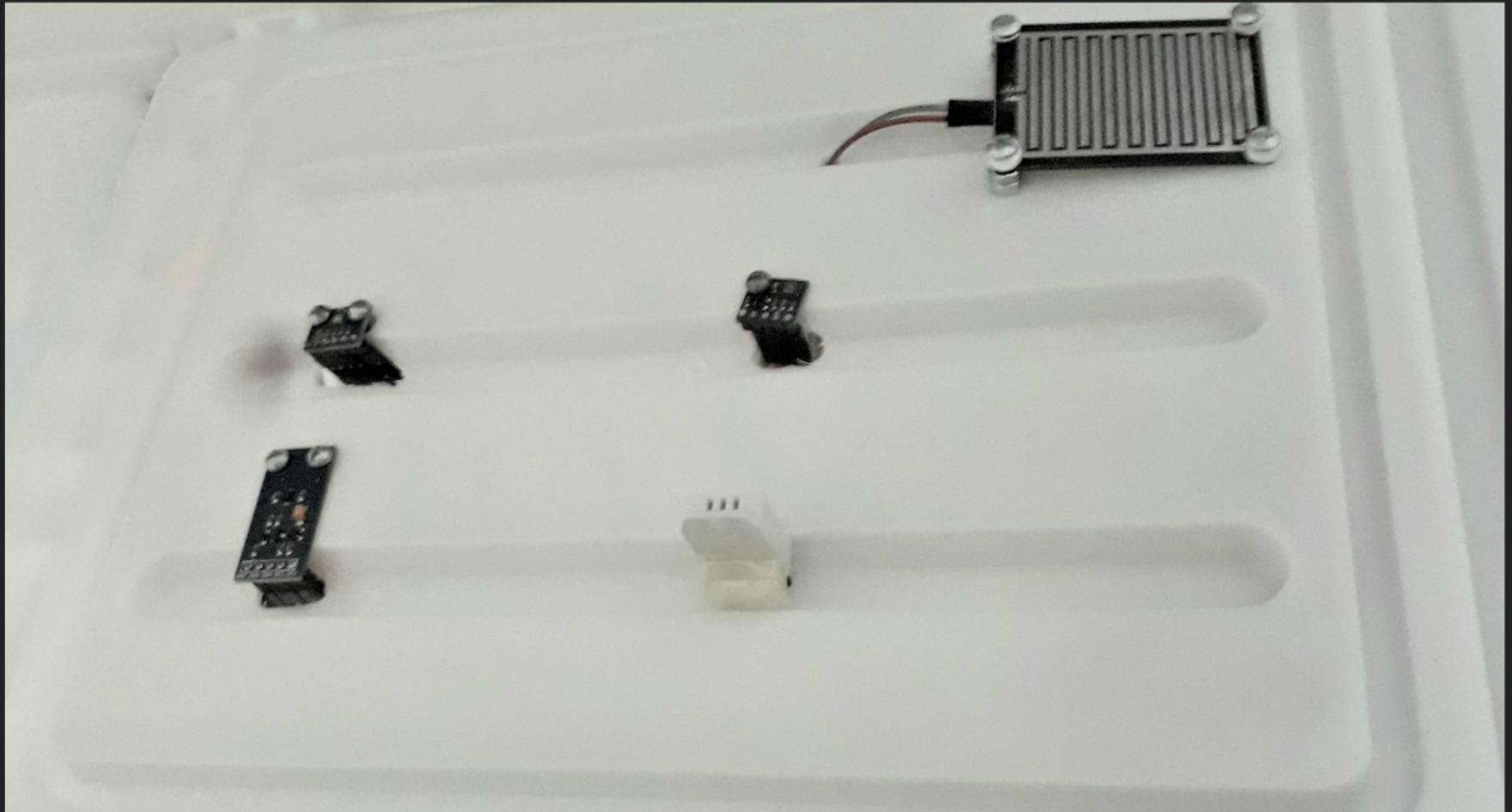
CLIENTE SUBSCRIPTOR

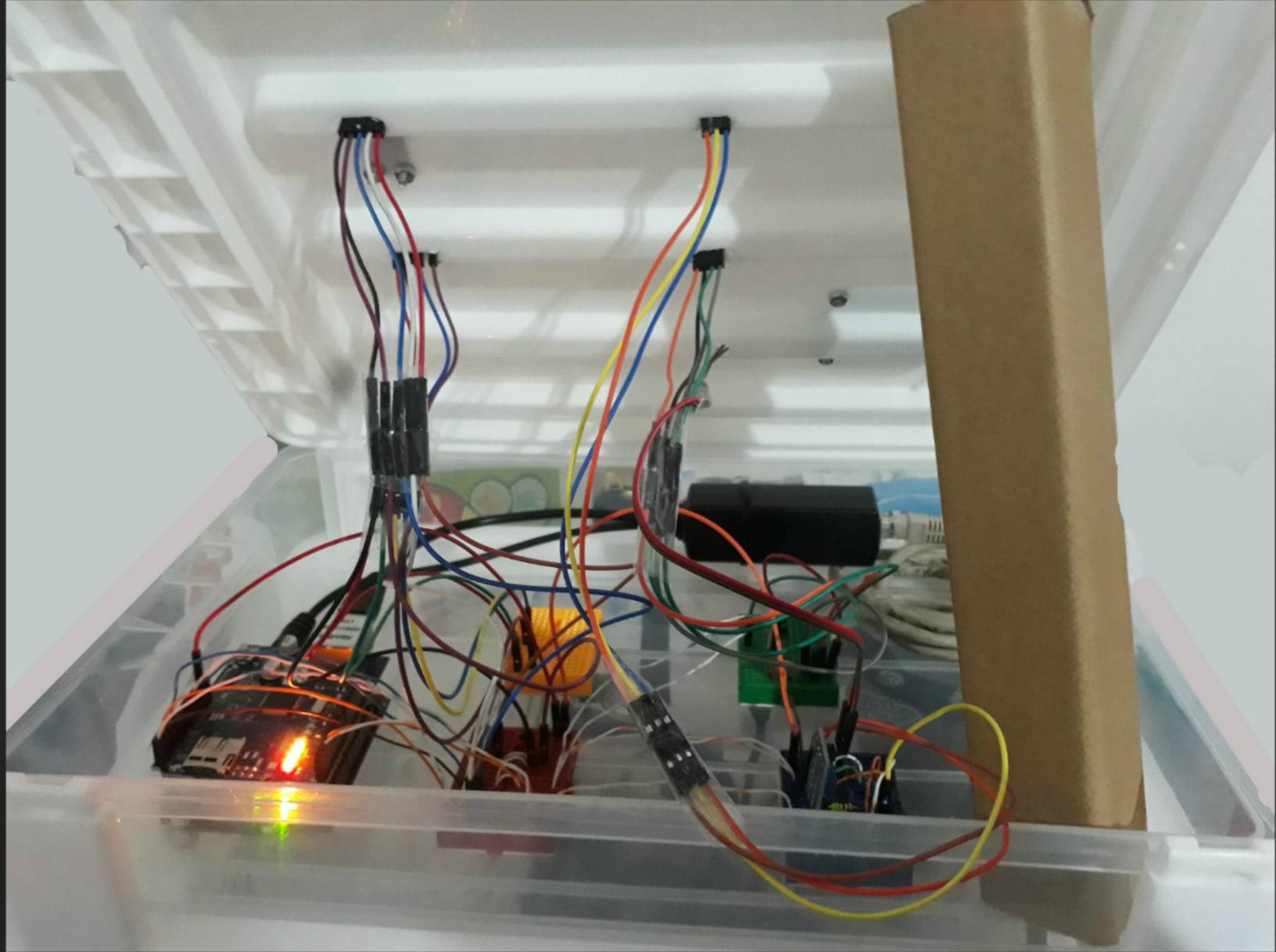


# PROTOTIPO











## PLATAFORMA ONLINE



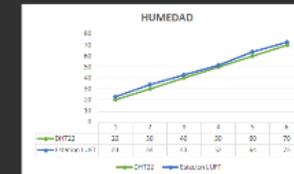
## PRUEBAS

- Sensores por separado
- Conjunto de sensores
- Envío y recepción de datos desde Arduino a Raspberry Pi
- Corte de suministro eléctrico
- Corte de red local
- Corte de red global

# RESULTADOS, CONCLUSIONES, y TRABAJOS FUTUROS

## COMPARACIÓN CON PATRÓN DE MEDICIÓN

Estación meteorológica Luff modelo Central Luff y higrómetro digital Proxman modelo MS6602



Objetivo  $\pm 1\%$  Resultado  $\pm 1.4\%$  (max)

## CONCLUSIONES

- Logros a nivel de hardware
- Objetivos conseguidos con el armado del prototipo
- Logros con el software
- Incorporación de conocimientos

## FUTURAS MEJORAS

- Sumar estaciones meteorológicas y generar con un mismo broker una red que abarque más zonas geográficas
- Sensores con mayor precisión y que puedan estar a la intemperie
- Realizar el envío de datos almacenados
- Agregar autenticación de usuarios
- Broker online

# PLATAFORMA ONLINE

☰ Estado del tiempo

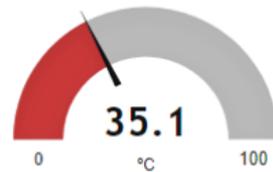
Reloj

Hora actual en  
San Luis, Argentina

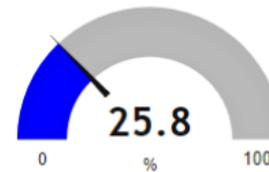
Lun, 10. Dic 2018

17:28:26

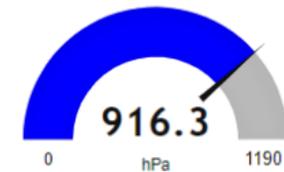
Temperatura



Humedad relativa



Presion atmosférica



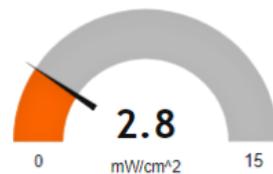
Indicador

Estado Online



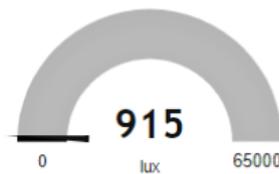
Rayos UV

Índice UV moderado



Nivel de Iluminación

Día muy oscuro



Lluvia

No hay precipitaciones



## ☰ Datos previos

### Información importante

Para la correcta interpretación, tener en cuenta:

- El **gráfico** muestra los valores obtenidos durante las últimas 24 hs.
- Los valores calculados **máximo, mínimo y promedio** se obtienen con los valores mostrados durante la hora anterior a la actual.
- El botón de **Reset datos almacenados** borra el historial de datos almacenados en archivo.

### Temperatura



Valor máximo **17.40**

Valor mínimo **17.40**

Valor promedio **17.40**

RESET DATOS ALMACENADOS

### Humedad Relativa



Valor máximo **47.30**

Valor mínimo **47.30**

Valor promedio **43.20**

RESET DATOS ALMACENADOS

### Presión atmosférica



Valor máximo

Valor mínimo

Valor promedio

RESET DATOS ALMACENADOS

### Rayou UV

### Nivel de Iluminacion

Ingeniería Electrónica con Orientación en Sistemas Digitales

# Supervisión de estación meteorológica remota por IoT con tecnología M2M

Trabajo Final de Carrera

Alumna Andrada Tivani Astri Edith  
Director Ing. Murdocca Roberto Martin

Universidad Nacional de San Luis



Facultad de Ciencias Físico, Matemáticas y Naturales



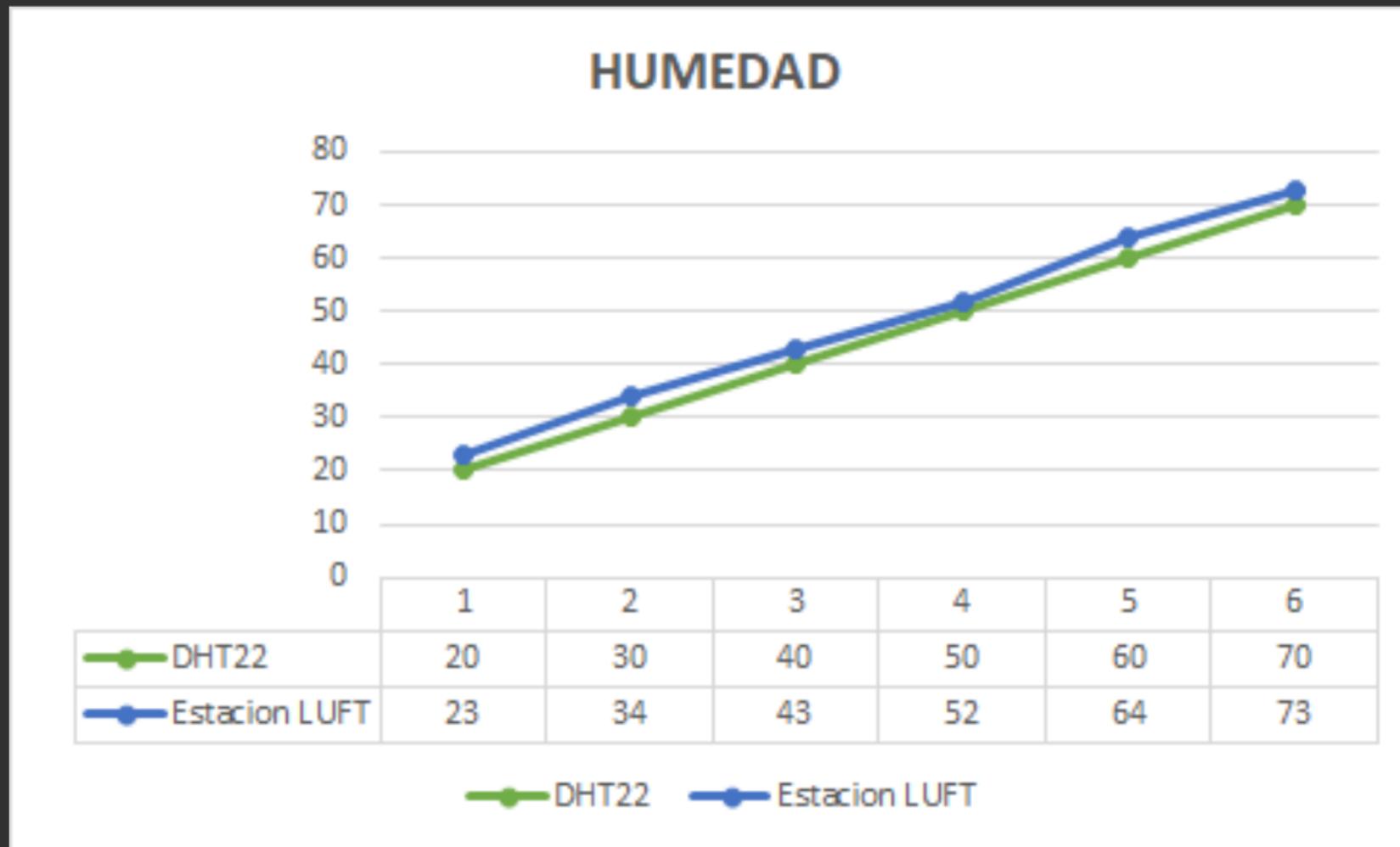
Departamento de Electrónica

# PRUEBAS

- Sensores por separado
- Conjunto de sensores
- Envío y recepción de datos desde Arduino a Raspberry Pi
- Corte de suministro eléctrico
- Corte de red local
- Corte de red global

# COMPARACIÓN CON PATRÓN DE MEDICIÓN

Estación meteorológica Luft modelo Central Luft y luxómetro digital Protomax modelo MS6610



Objetivo +/- 6% Resultado +/-4% (max)

# CONCLUSIONES

- Logros a nivel de hardware
- Objetivos conseguidos con el armado del prototipo
- Logros con el software
- Incorporación de conocimientos

# FUTURAS MEJORAS

- Sumar estaciones meteorológicas y generar con un mismo broker una red que abarque más zonas geográficas
- Sensores con mayor precisión y que puedan estar a la intemperie
- Realizar el envío de datos almacenados
- Agregar autenticación de usuarios
- Broker online

# AGRADECIMIENTOS

A mis hijas, María Pía y María del Pilar

A mis papás, Orlando y Sonia

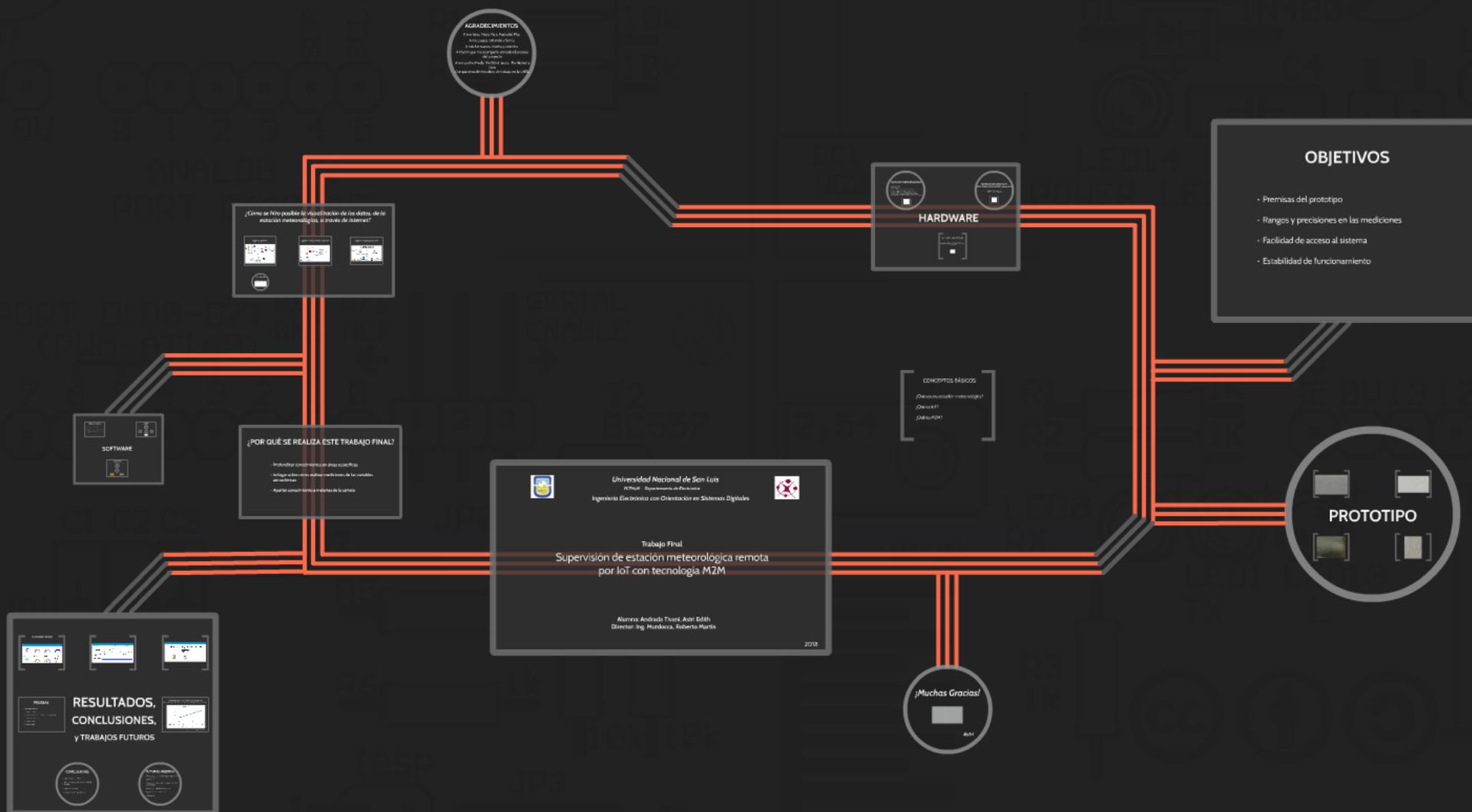
A mis hermanos, Noelia y Leandro

A Martín que me acompañó en todo el proceso  
del proyecto

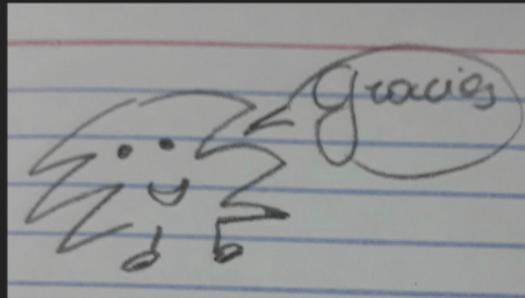
A mis profes Fredy, Ale Silnik, Jesús, Ale Nuñez y  
Curu

Compañeros de estudio y de trabajo en la UNSL

# Supervisión de estación meteorológica remota por IoT con tecnología M2M



***¡Muchas Gracias!***



***Astri***