

# **Memoria de proyecto**

## **Silvanus Bonsai**



I Concurso de hardware – cat. Máster  
Universidad complutense de Madrid  
Antonio Vázquez Pérez

# Índice

<b>RESUMEN.....</b>	<b>3</b>
<b>INTRODUCCIÓN.....</b>	<b>4</b>
<b>DESCRIPCIÓN.....</b>	<b>5</b>
Nodo principal (Servidor).....	6
Nodo de riego.....	7
Estructura del sistema de comunicación.....	8
Diagrama nodo de riego.....	9
Casos de uso.....	9
<b>ESTADO DEL ARTE.....</b>	<b>10</b>
<b>DESARROLLO.....</b>	<b>12</b>
Consumo energético.....	18
Cálculos.....	20
Acabados.....	21
Fabricación de la PCB.....	21
Cajas.....	27
<b>APLICACIÓN REAL.....</b>	<b>29</b>
Viabilidad económica.....	29
Precio servidor.....	29
Precio nodo de riego.....	29
Público objetivo.....	30
<b>CONCLUSIONES.....</b>	<b>31</b>
Mejoras a futuro.....	31

# RESUMEN

---

Se ha imaginado una solución que pueda hacer un riego periódico de una planta y que nos permita ver en qué condiciones se encuentra haciendo uso de sensores de humedad/temperatura ambiental, luminosidad y humedad del sustrato.

Todo ello con un nodo central que actuará de servidor el cual creará un punto de acceso WiFi para la comunicación y una página web a la que podremos acceder desde un terminal móvil o PC para configurar los nodos y ver los datos transmitidos por el resto de nodos.

El proyecto incluye también la fabricación de una PCB y un montaje en caja para mejorar los acabados finales.

# INTRODUCCIÓN

---

El proyecto que se desea acometer resulta ser una solución para dar cierta comodidad al cuidado de un Bonsai o planta que pudiera quedar desatendida durante un periodo vacacional, o también para poder monitorizar todos los parámetros que nos ofrecen una visión acerca de las condiciones ambientales del mismo.

Con estos requisitos, se buscó una solución a un problema real, y es que las plantas son de mantenimiento sencillo pero requieren de un riego periódico que no le podemos dar si no nos encontramos en el lugar.

Estos cuidados se hacen incluso más complicados cuando se trata de una planta como un bonsái, que no es otra cosa que un árbol en miniatura, los cuales requieren una cantidad de iluminación determinada y de un control un poco más exhaustivo.

Por esta razón, se ha imaginado una solución que pueda hacer un riego periódico de la planta y que nos permita ver en qué condiciones se encuentra haciendo uso de sensores de humedad/temperatura ambiental, luminosidad y humedad del sustrato. Todo ello con un nodo central que actuará de servidor el cual creará un punto de acceso WiFi para la comunicación y una página web a la que podremos acceder desde un terminal móvil o PC para configurar los nodos y ver los datos transmitidos por el resto de nodos.

Se trata de un proyecto muy completo que agrupa conocimientos aplicados acerca de redes, servicios web, interfaces, electrónica, fabricación de PCBs y microcontroladores.

# DESCRIPCIÓN

---

El dispositivo a desarrollar busca ser un sistema automatizado que de un cuidado autónomo a una o varias plantas, permitiéndonos descuidarla y que se mantenga en óptimas condiciones.

El sistema consistiría en un nodo de riego o más que poseerá los sensores y una bomba de riego y otro que actuará como monitor para poder observar el estado de todos los nodos.

El agua sería almacenada individualmente por cada planta en forma de mochila.



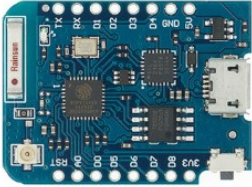


*Ilustración 1: Plantas varias*

A continuación se procede a detallar las capacidades individuales de los dos tipos de nodo que se van a desarrollar.

# Nodo principal (Servidor)

Este nodo creará un punto wifi que usará para que se conecten el resto de nodos de riego y será el encargado de recoger todos los datos generados. Además, también tendrá un monitor y un botón para ir observando los datos recogidos por los nodos.

Componentes utilizados:

Componente	Descripción
	Wemos D1 mini pro
	Pantalla OLED 0,66 pulgadas wemos shield
	Antena 2,4Ghz



Vendrá alimentado haciendo uso de un cargador estándar micro usb de móvil.



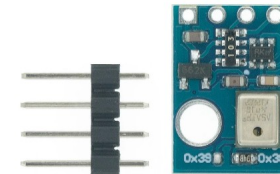
# Nodo de riego

Se puede contar con tantas de estas unidades como se desee, serán las dedicadas a cuidar de la planta sobre la que se coloque.

Su capacidad principal es la de regar automáticamente según lo necesite cada planta en función de la humedad de su sustrato, midiéndolo con un higrómetro de suelo; cuando se necesite activará una pequeña bomba de agua y la transferirá desde el tanque hasta la maceta.

Como capacidades adicionales destacar la medición de la temperatura y humedad del ambiente así como la luminosidad que está recibiendo tomando las medidas periódicamente.

Estos nodos contarán con una batería 18650, la cual haciendo uso del modo DeepSleep del microcontrolador.

Componente	Descripción
	Wemos D1 mini
	Micro bomba de agua sumergible
	Sensor de humedad y temperatura AHT10



	<p>Sensor capacitivo de humedad de suelo</p>
	<p>Cargador para 18650 y fuente de energía</p>
	<p>Sensor de luminosidad LDR</p>

## Estructura del sistema de comunicación

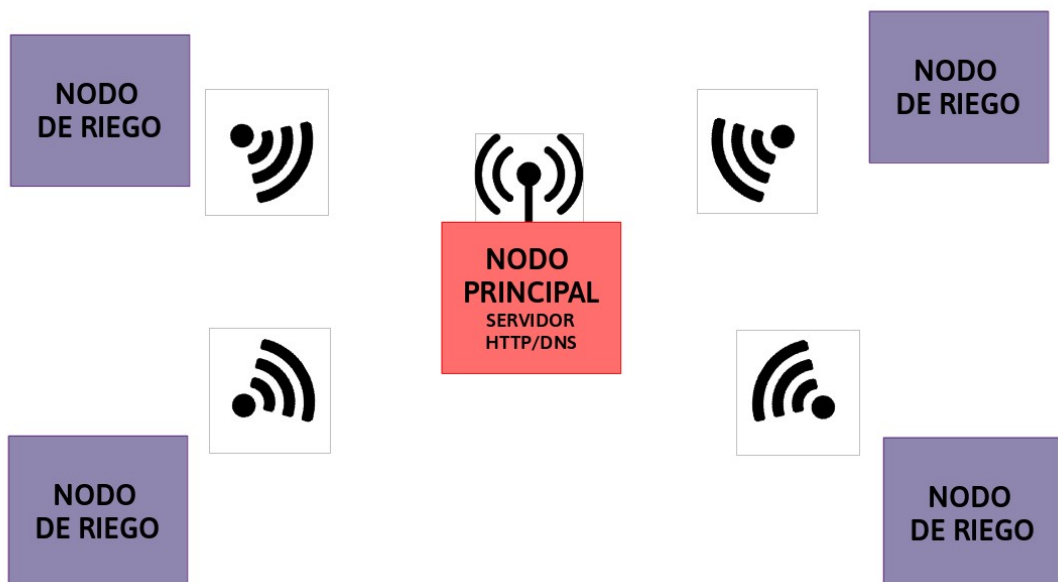


Figura 1: Sistema de comunicación



# Diagrama nodo de riego

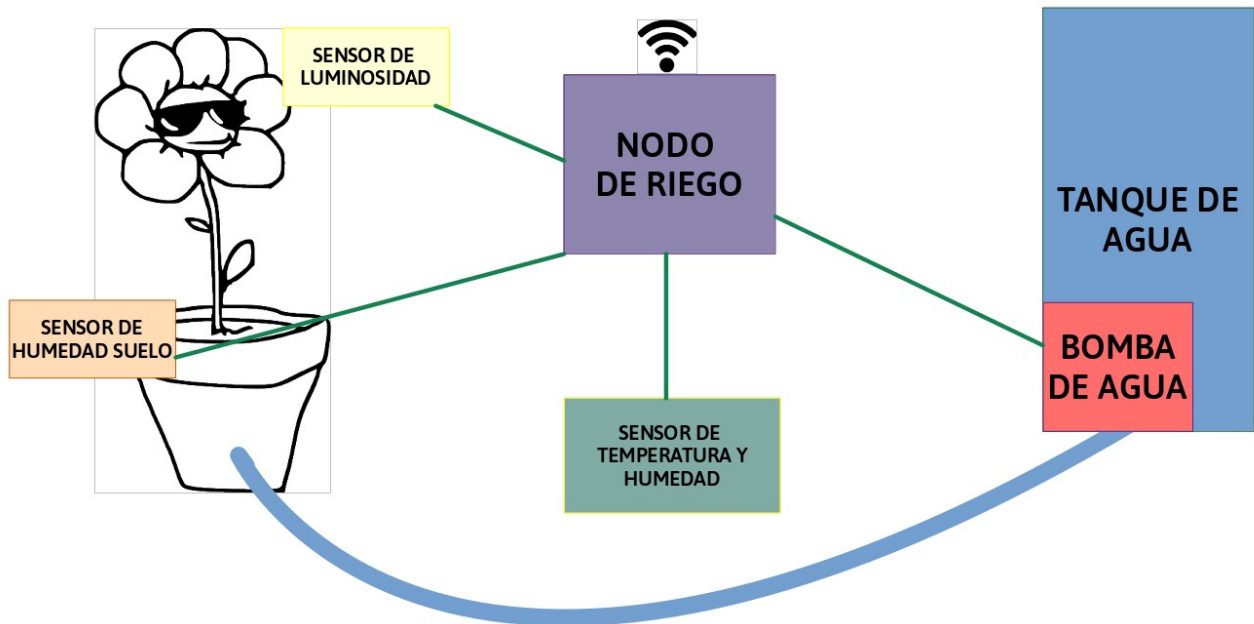


Figura 2: Diagrama del nodo de riego

## Casos de uso

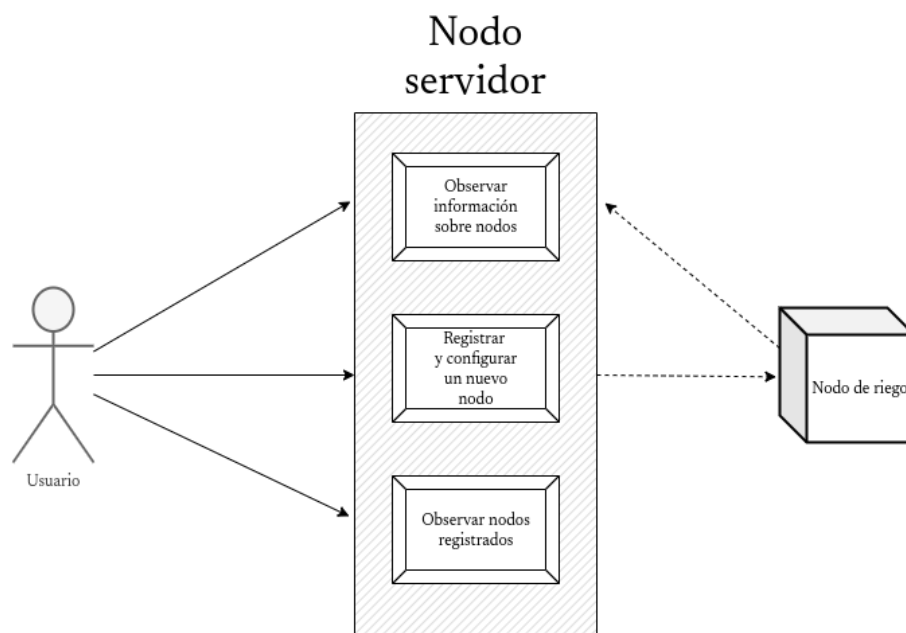


Figura 3: Caso de uso del nodo servidor

El nodo de riego no tendría casos de uso ya que se interacciona con él a través del servidor.

# ESTADO DEL ARTE

---

Actualmente en el mercado se pueden encontrar una gran variedad de soluciones con respecto al riego de las plantas. La gran mayoría de ellas van conectadas directamente a la boca de la manguera.

Aquí tenemos un ejemplo de los que hay disponibles junto a sus precios:



Figura 4: Captura de google shopping

El problema de estos sistemas es que se necesita la toma de agua para poder utilizarlos, y nuestro proyecto va más orientado a cultivo en interiores o en zonas urbanas donde no se puede siempre disponer de una boca de agua.

Con respecto a este punto sí que se pueden encontrar soluciones que son bombas de agua automatizadas y configurables.

Tenemos por ejemplo este, una bomba a la que se le pone un depósito de agua y se puede controlar con el móvil:

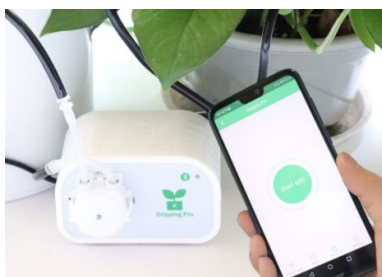


Figura 6: Bomba automatizada, control por móvil.

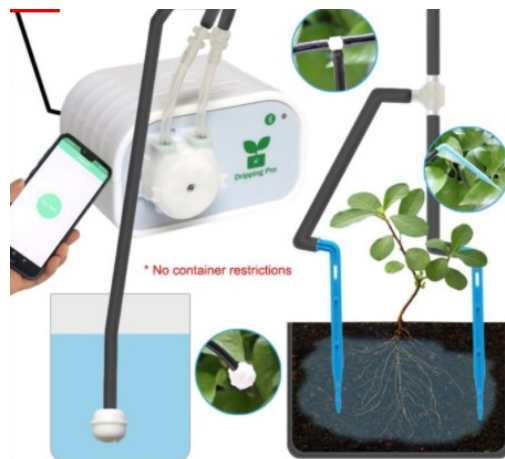


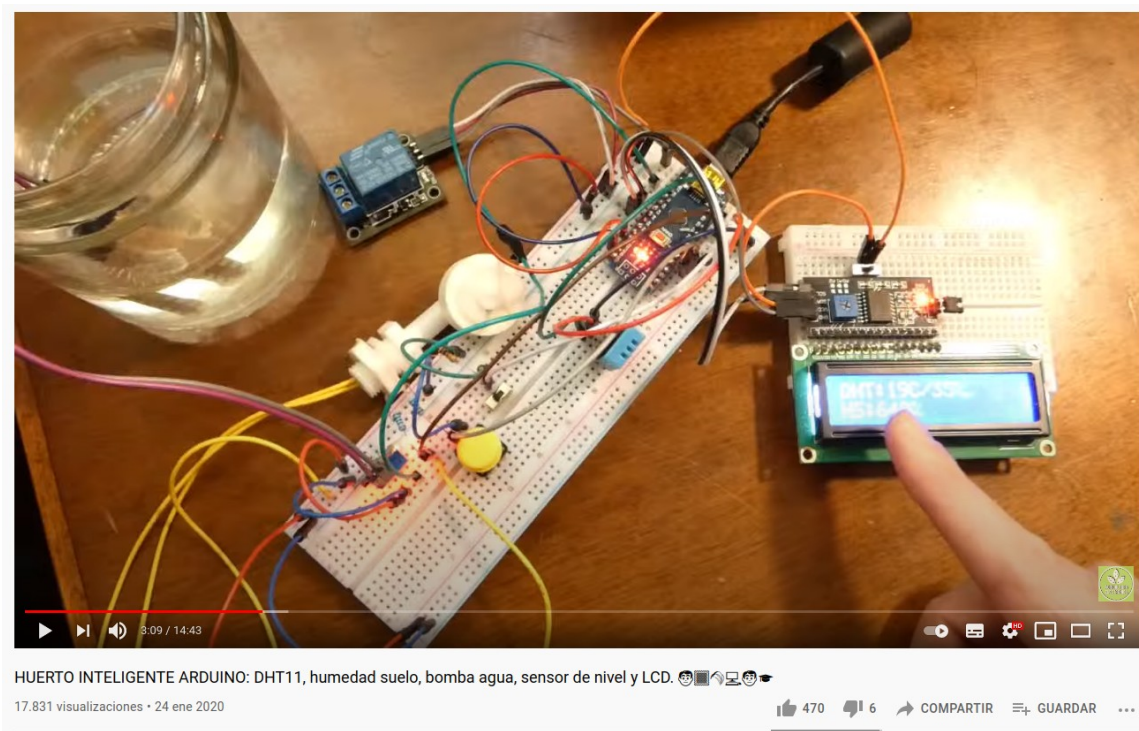
Figura 5: Diagrama de la bomba automatizada

Y como esta anterior existen otras cuantas de similares prestaciones y características. La diferencia de estas con nuestro sistema es que no poseen sensores de monitorización y no se puede montar una red de nodos. Hay que configurarlos individualmente.

Por eso, ese es el aspecto por el que puede destacar nuestro sistema, no existen otros tan complejos o especializados y de tan bajo consumo.

Por último, de forma casera o personal, se pueden ver muchas implementaciones de huertos urbanos o de control meteorológico haciendo uso de microcontroladores como arduino o sensores.

Aquí tenemos uno en youtube:



*Figura 7: Captura de vídeo de YouTube, sistema casero de monitorización de plantas*

Este último ejemplo es bastante parecido a nuestro caso con la diferencia que no tiene una interfaz muy amigable o no queda como una solución final cómoda de utilizar.

# DESARROLLO

El desarrollo comienza por lograr leer todos los sensores con el Wemos. Para ello, conectamos los periféricos uno por uno y se programan funciones que nos simplifiquen la tarea de acceder a ellos.

Continuamos conectándolos todos juntos y comprobando que funcionen todos a la vez. Para esto viene bien anotar los pines en los que van a ir conectados los sensores y periféricos.

En mi caso he utilizado el software *Xournal* para apuntarlo:

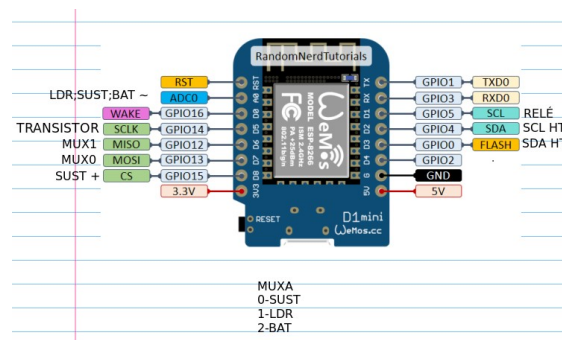
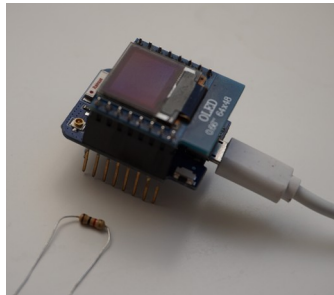


Figura 8: Notas de asignación de pines de la placa.

Los pines en los que se observa que pone *transistor* y *mux* es porque en el circuito del nodo se está utilizando un multiplexor para alternar entre todas las entradas analógicas y un transistor para que el microcontrolador pueda desconectar periféricos para ahorrar batería.

En el código, he añadido también un fichero de cabecera *conf(h)* que nos facilita la labor de configurarlo teniendo a mano la configuración de todas las bibliotecas. También se han desarrollado una biblioteca para la comunicación WiFi y otra para la comunicación con el servidor HTTP.

Después, se pasa a programar el nodo servidor, es un Wemos D1 mini pro al que le hemos añadido un *shield* con una pantalla OLED para visualizar los datos de los nodos.



*Ilustración 2: Nodo servidor utilizado*

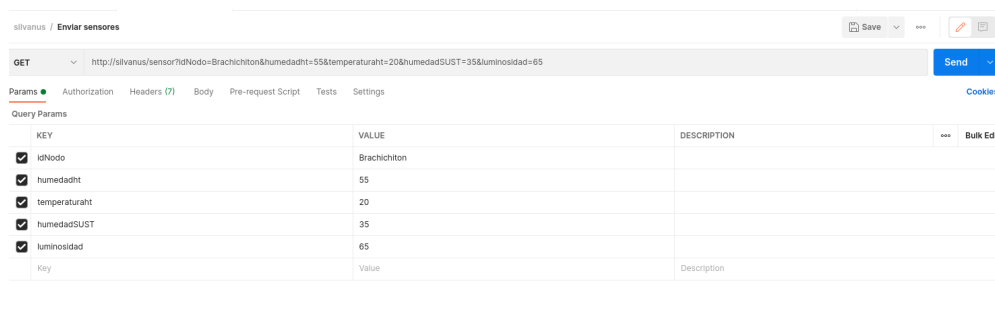
Se ha diseñado una pantalla de inicio para cuando arranca el servidor; se muestra por unos segundos.



*Ilustración 3: Nodo servidor; pantalla de arranque.*

Se ha programado un servidor web “a mano” con CSS, HTML y Javascript para ofrecer al usuario opciones de configuración y visualización, y otro DNS para que responda a la dirección <https://silvanus>.

El servidor creará un punto de acceso wifi al que se deberá estar conectado para acceder a la página web. Se ha utilizado el programa Postman para comprobar que todas las comunicaciones e intercambio de mensajes funcionan correctamente.



*Figura 9: Captura del software postman; envío de información por parte de un nodo.*

La página web ha sido diseñada con Web Responsive Design, por lo que le permite adaptarse hasta a dispositivos móviles. Esto es muy útil ya que los usuarios de hoy en día suelen conectarse más a través de este tipo de dispositivos y es más cómodo para configurar los nodos.

Vista de la web en PC y Móvil:

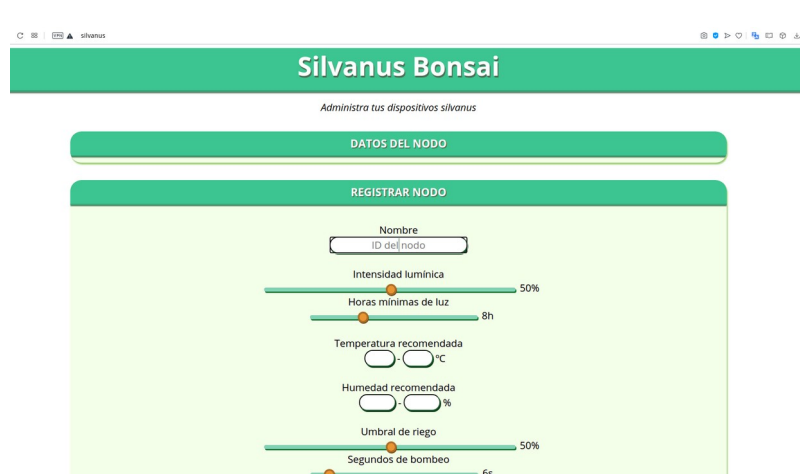


Figura 11: Vista de la web en PC

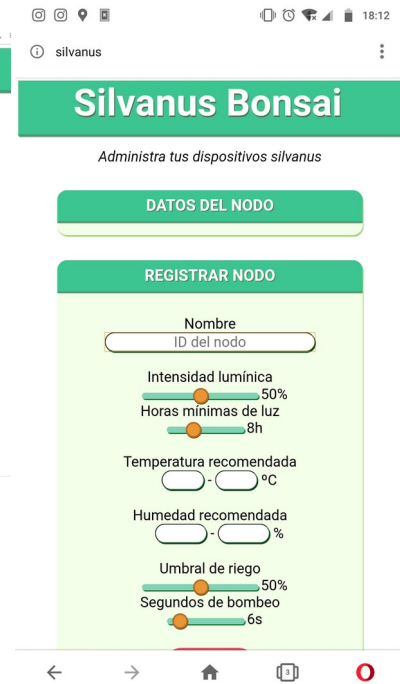


Figura 10: Vista de la web en móvil moto G5 plus

La web tiene 3 apartados:



Figura 12: Apartados de la aplicación web

Los apartados se despliegan haciendo click en ellos.



El primer apartado nos permite seleccionar entre los nodos de los que hemos recibido datos y si le damos a consultar nos lo muestra.

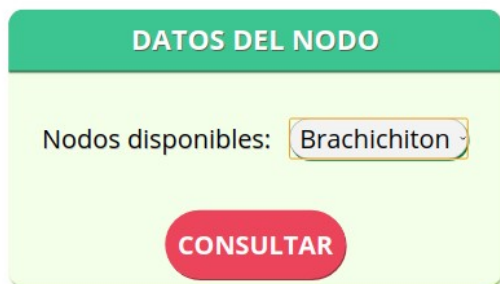


Figura 13: Menú de consulta de datos

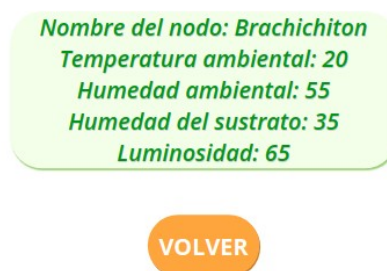


Figura 14: Resultado una vez habiendo pulsado consultar en consulta de datos

En el ejemplo tenemos un nodo que se llama *Brachichiton* y los últimos datos que ha mandado. Además, también podemos ver los datos en la pantalla del dispositivo.



Ilustración 4: Pantalla mostrando información del nodo

En el segundo apartado podemos registrar un nodo. El registro de nodos funciona de la siguiente forma:

- Se rellena el formulario de registro de nodo.
- Se queda guardada la información acerca del nuevo nodo en el servidor..
- Cuando se conecta un nuevo nodo o se despierta alguno se comprueba si existe una nueva configuración, si es así, se la descarga y el servidor la elimina.

Este proceso nos permite que los nodos no tengan que estar despiertos (los nodos se encuentran la mayor parte del tiempo en reposo para ahorrar energía de la batería) para poder recibir la configuración, de esta forma la reciben cuando ellos pueden y ahorran energía.

Figura 15: Formulario de registro de nodo

El tercer apartado es uno extra, un enlace a una página que es una utilidad para detectar qué planta tienes y así poder encontrar sus requerimientos.

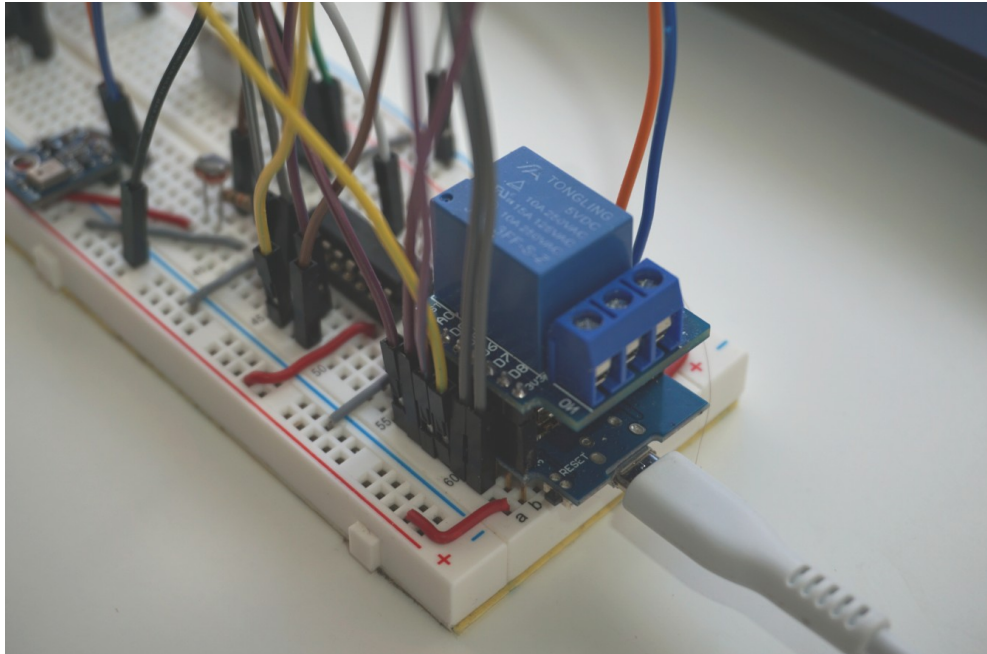
Figura 16: Sección de enlace a identificador de plantas.

Con todo esto ya tenemos la interfaz necesaria para controlar y visualizar todos los nodos.



Ahora volvamos al nodo de riego, vamos a ver de qué forma actúa este nodo ya teniendo la configuración.

El nodo tiene un shield de un relé en su parte superior para controlar la bomba de agua.



*Ilustración 5: Plano detalle del relé del nodo de riego.*

La bomba de agua se activará en el momento que el nivel de humedad de suelo disminuya por debajo del umbral establecido, en este momento funcionará durante el número de segundos que se le haya indicado.

Veamos a continuación como afectan todos los periféricos al consumo eléctrico.

# Consumo energético

Para ahorrar lo máximo posible y prolongar la vida de la batería, el nodo de riego va a estar en periodo de hibernación durante la mayor parte de su vida operativa.

Analicemos pues como podemos reducir el consumo lo máximo posible.

El microcontrolador *ESP8266*, el que incorpora la placa *Wemos*, tiene una función llamada *Deep Sleep*, cuyo resultado es que apaga todos los periféricos y componentes y entra en un estado de latencia de muy bajo consumo. Durante este estado solamente mantiene en funcionamiento un timer para que la reinicie transcurrido un tiempo.

Casi todos los periféricos, que van conectados a la salida de 3.3v de la placa, serán automáticamente desconectados a partir del momento en que ésta entre en el modo “deep sleep”(dormitancia o hibernación), por lo que a partir de ese momento no es necesario que nos preocupemos más de ellos, pues no van a consumir. El multiplexor también se incluye en este grupo.

El periférico que más preocupa en este sentido es el sensor de humedad de sustrato, que consume unos 18mA continuamente. Para solventar este problema se ha intercalado un transistor en su pin de alimentación para poder controlar cuándo va a estar encendido. La base del transistor va a un pin del microcontrolador.

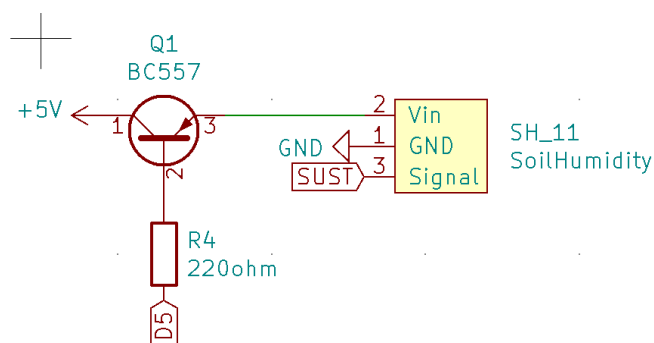
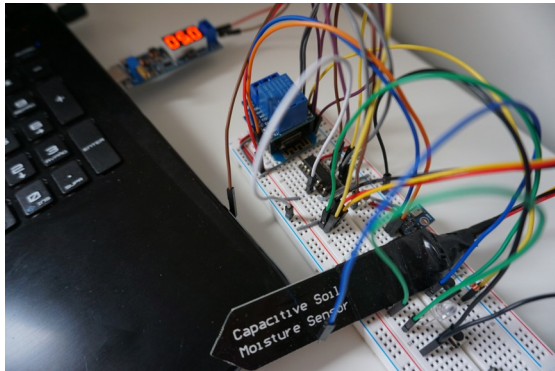
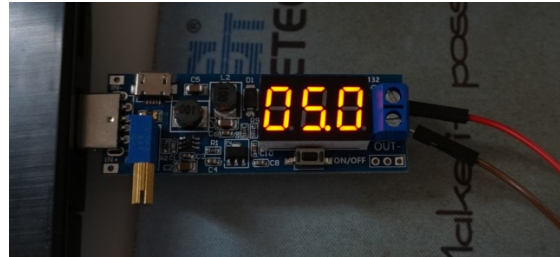


Figura 17: Captura del circuito que conecta con el sensor de sustrato.

Con todas estas optimizaciones veamos cuánto consume en activo y en reposo. Se ha preparado un banco de pruebas con la protoboard y una fuente de alimentación por USB, entre medias hemos colocado un amperímetro para calcular cuánta intensidad circula.



*Ilustración 6: Nodo de riego montado en protoboard*



*Ilustración 7: Fuente de alimentación USB.*

Intensidad en activo:



*Ilustración 8: Multímetro mostrando el consumo alto.*

Intensidad en reposo:



*Ilustración 9: Multímetro mostrando el consumo bajo.*

# Cálculos

Diferencia de amperaje activo-reposo:  $\frac{94,8mA}{3,2mA} = 30,875$  veces menos.

## Cálculo de consumo a la hora:

Se estima que el nodo va a encenderse una vez cada media hora, ese encendido durará unos 10s. La potencia de la bomba es de  $5V * 200mA = 1W$ , pero como probablemente se encienda unos segundos una vez cada dos días lo podemos declarar como irrelevante.

Calculamos qué porcentaje del tiempo de una hora se encuentra con consumo alto:

$$\frac{10s * 2}{3600s} = 0,0055... \rightarrow 0,55\% \text{ de una hora con alto consumo.}$$

Media ponderada de consumo:

$$3,2mA * 0,9945 + 94,8mA * 0,0055 = 3,7038mA \text{ de consumo a la hora}$$

## Cálculo de duración de la batería:

La batería de litio 18650 tiene una capacidad aproximada de 2000mah.

Se asumen pérdidas por descarga de batería del 10%.

$$\frac{2000mAh * 0,9}{3,7038mA} = 485,98h \text{ de autonomía} \rightarrow 20,24 \text{ días.}$$

**20 días aproximadamente de autonomía**

# Acabados

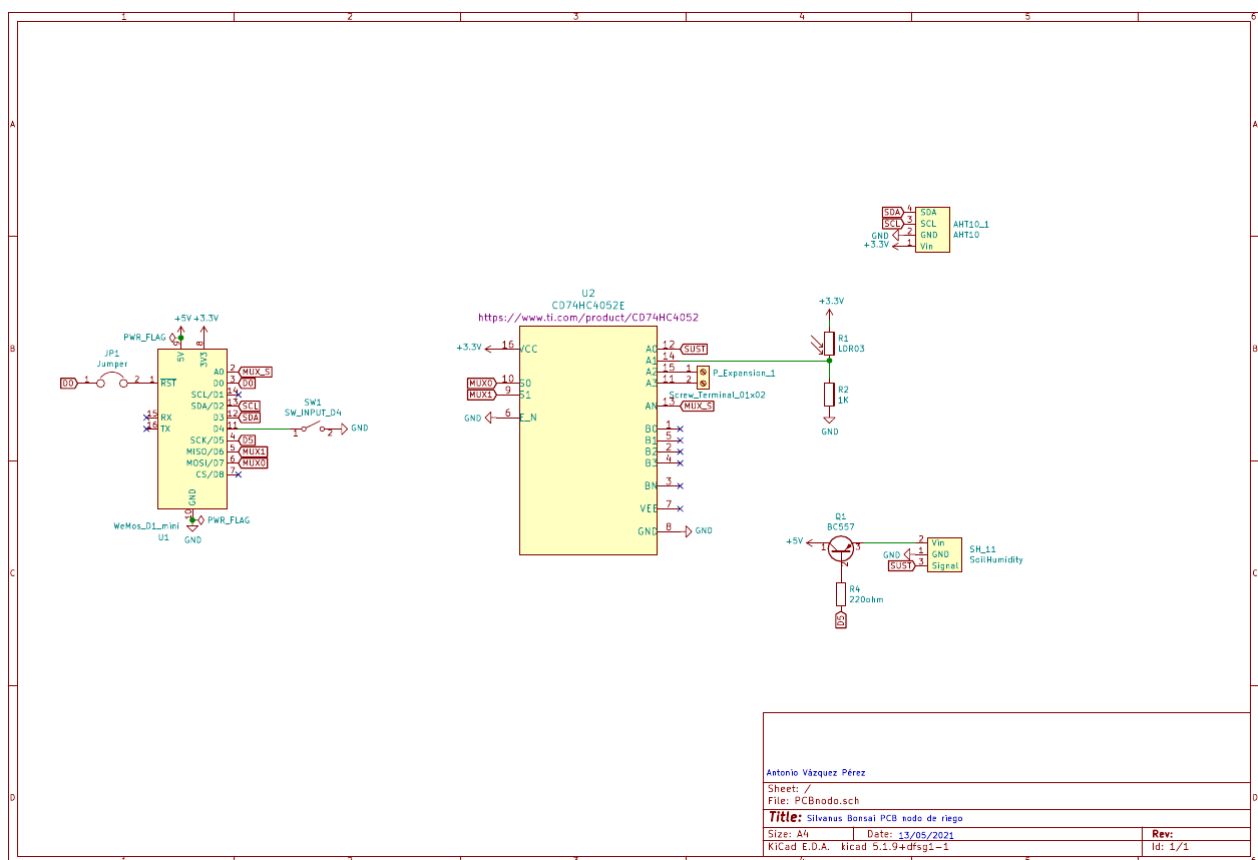
Finalmente, para dejar el proyecto como una solución final cómoda de utilizar se va a fabricar una placa PCB para el nodo de riego y ambos nodos, tanto el servidor como el de riego montarlos dentro de una caja.

## Fabricación de la PCB

La técnica de fabricación que se va a utilizar es la de baño por ácido y se va a proceder a ilustrar el proceso completo de fabricación.

Para elaborar los diseños se ha utilizado el software libre KiCad.

Primero comenzaremos creando el diseño de la electrónica del nodo, para ello se han importado los modelos necesarios de todos los componentes y se han cableado para obtener este diseño:



Dibujo 1: Planos de la electrónica del nodo de riego.

De este diseño se puede destacar que se ha incluido un *jumper* para conectar y desconectar el pin de reset, un puerto de expansión para dos entradas analógicas y otro puerto digital para el pin D4.

Se puede observar también un divisor de tensión para el LDR y un transistor para controlar la alimentación del sensor de humedad de suelo.

El elemento central es el multiplexor mencionado anteriormente para multiplexar la entrada analógica entre todos los sensores analógicos disponibles.

A continuación, el trazado de pistas de la placa:

*Primero establecemos las huellas sobre la placa y luego trazamos las conexiones..*

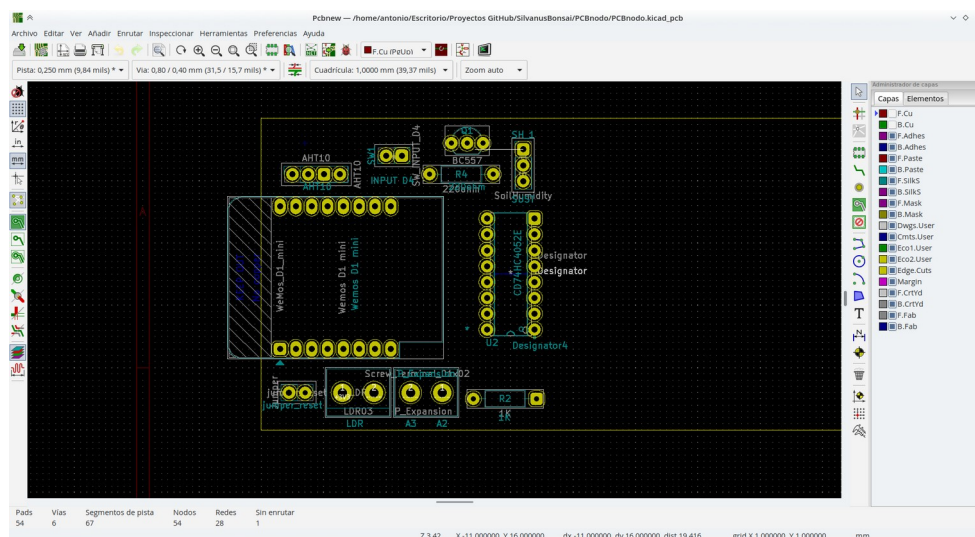


Figura 18: Captura software KiCad: Huellas de los componentes colocadas

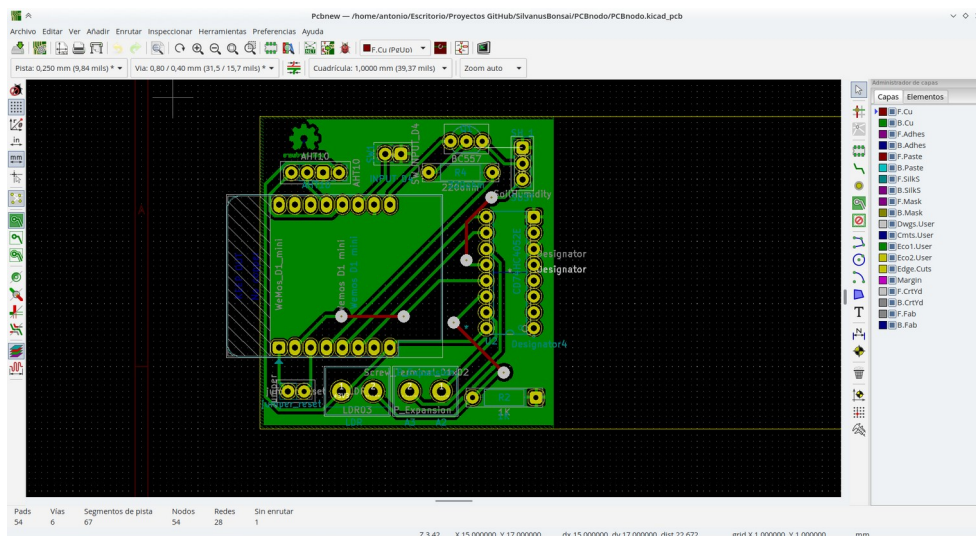
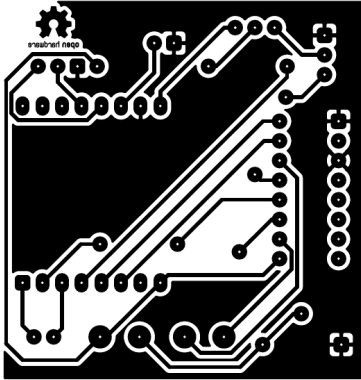


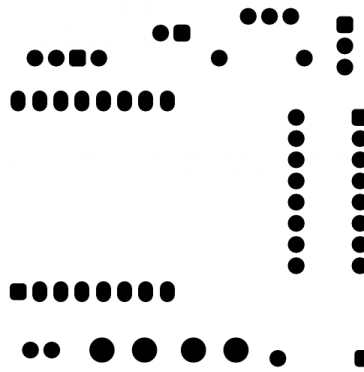
Figura 19: Captura del software KiCad: pistas trazadas. Las pistas mostradas en rojo son puentes por la cara delantera.



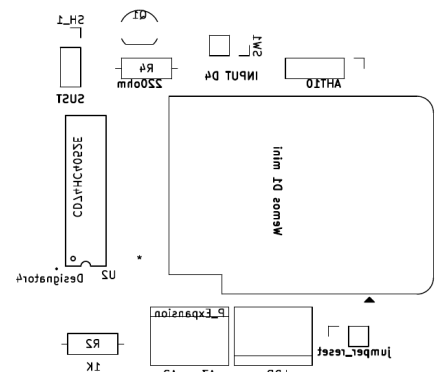
Una vez tenemos el diseño se extraen las imágenes que actuarán como máscara y serigrafía para la fabricación.



Dibujo 2: Capa de máscara de cobre.



Dibujo 3: Máscara para terminales.



Dibujo 4: Serigrafía frontal.

Los archivos de la capa de cobre y serigrafía los imprimiremos en un papel especial que se llama *papel transfer* o *papel tóner transfer*, como su nombre indica se deberán imprimir con una impresora de tóner.



Ilustración 10: Papel transfer. Este papel es utilizado para transferir la tinta adherida en él a otras superficies empleando calor.

Y la capa de esmalte imprimirla en papel de acetato (se asemeja a un plástico transparente formato A4).



*Ilustración 11: Papel de acetato.*

Pasamos a la parte física de la construcción:

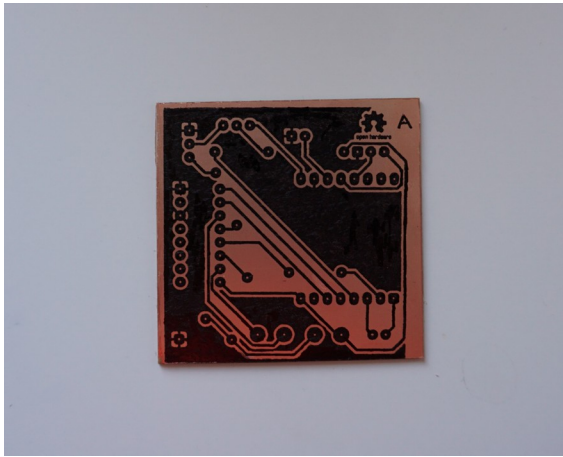


*Ilustración 12: Colocando la máscara de cobre.*



*Ilustración 13: Transferencia usando calor.*

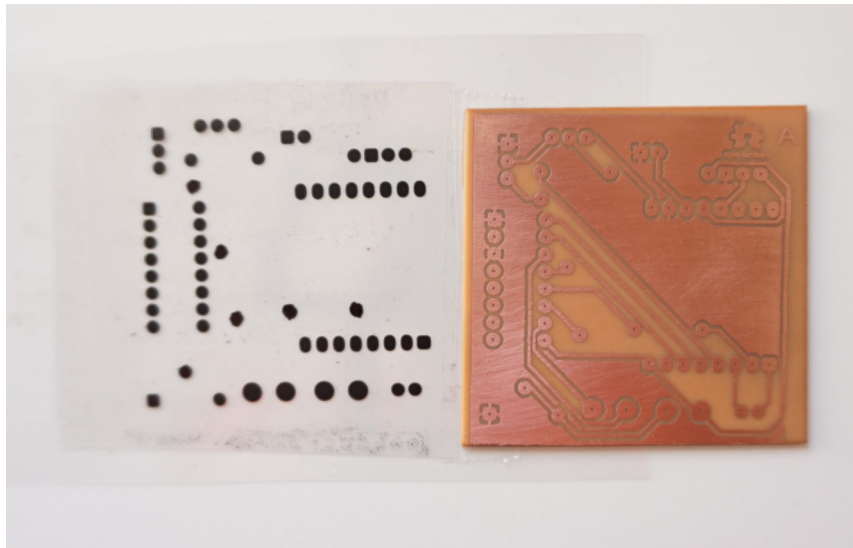




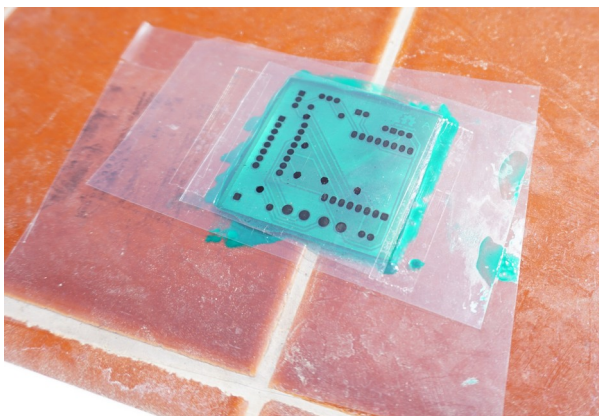
*Ilustración 14: Tinta transferida.*



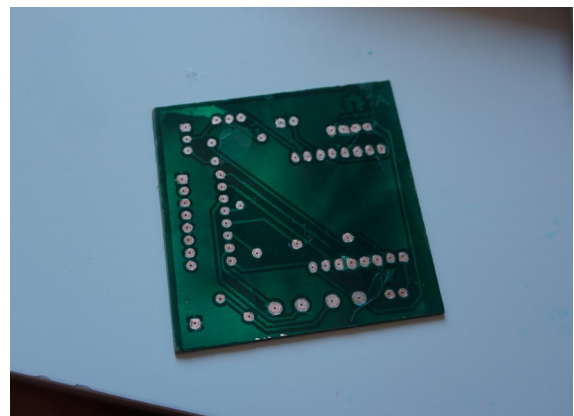
*Ilustración 15: Atacado con ácido para dejar solamente las partes cubiertas.*



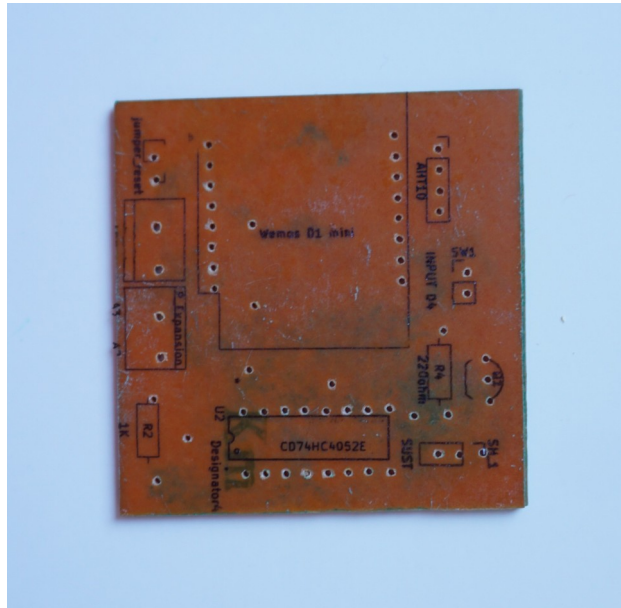
*Ilustración 16: Placa resultante junto a la máscara de esmalte.*



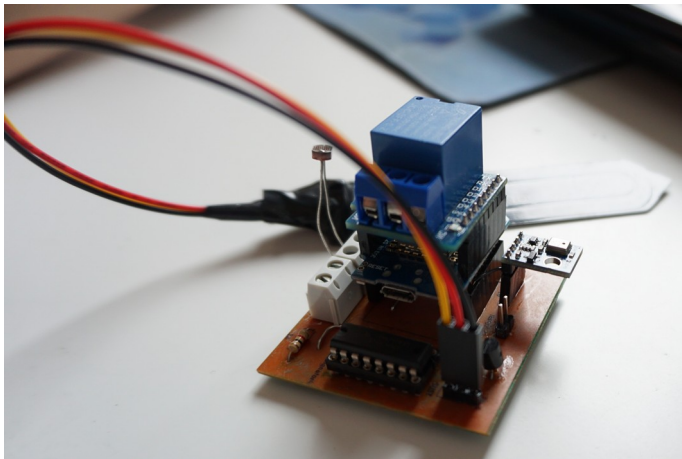
*Ilustración 17: Esmalte curando al sol junto a la máscara de terminales.*



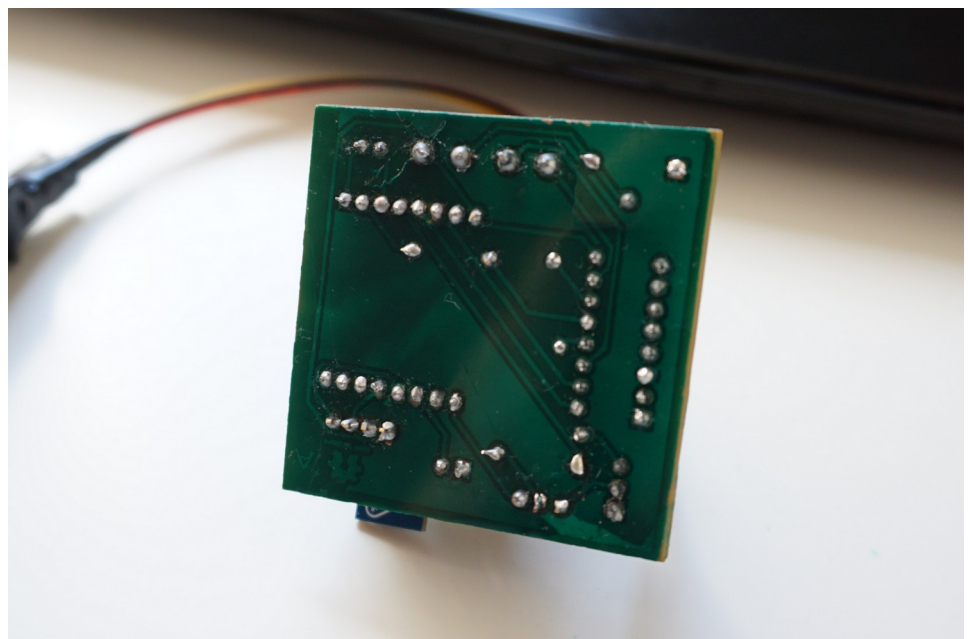
*Ilustración 18: Resultado de la aplicación del esmalte.*



*Ilustración 19: Serigrafía frontal aplicada.*



*Ilustración 20: Resultado final del nodo de riego.*



*Ilustración 21: Resultado final: soldaduras.*

## Cajas

Se ha comprado una caja de plástico, la cual se ha pintado y hecho el agujero para la pantalla del servidor. Posteriormente se le ha incorporado la antena.



*Ilustración 22: Caja del servidor con la pantalla*



*Ilustración 23: Caja del servidor con la antena introducida*

Tras pintarla un poco este es el resultado final:

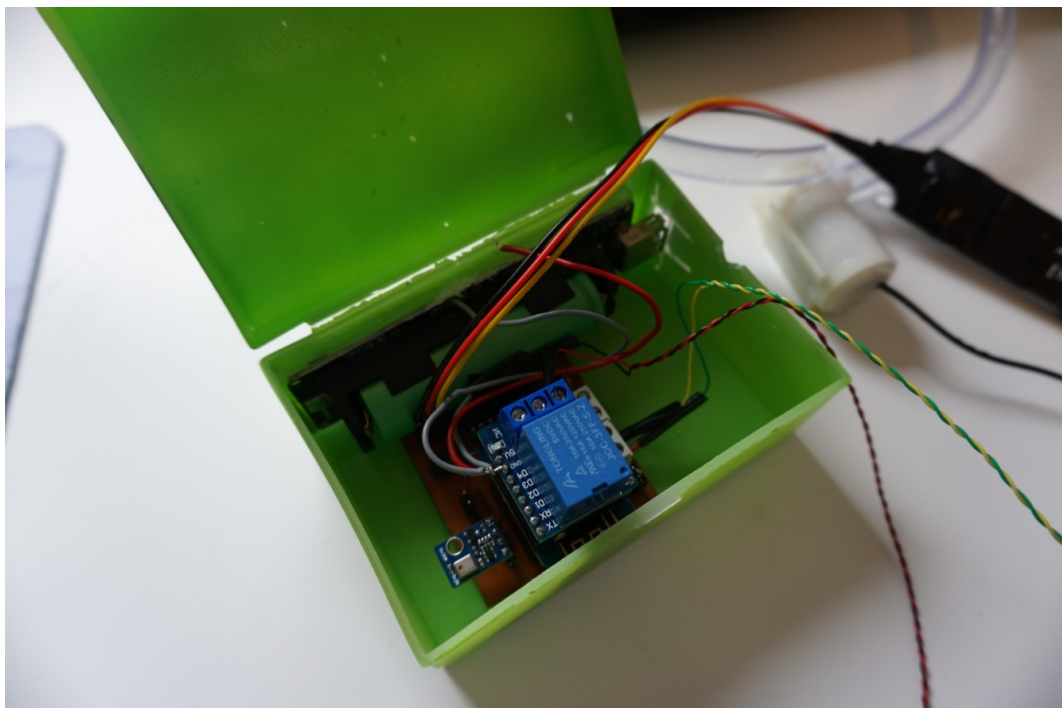


*Ilustración 24: Caja del servidor pintada*



*Ilustración 25: Plano detalle del arranque del servidor.*

Análogamente introducimos también los componentes en la caja del nodo de riego:



*Ilustración 26: Montaje de los componentes del nodo de riego dentro de la caja.*

Teniendo ya todo montado se puede proceder a la prueba. Se adjunta junto a este documento un vídeo de demostración en el que se observan los nodos y su funcionamiento.



# APLICACIÓN REAL

## Viabilidad económica

Desglose del precio final del prototipo elaborado.

### Precio servidor

Producto	Precio	Cantidad	Subtotal
Antena	0,85€	1	0,85€
Wemos D1 pro	2,60€	1	2,60€
Pantalla OLED	1,53	1	1,53€
Caja	2,5€	1	2,5€
<b>TOTAL</b>	---	---	7,48€

### Precio nodo de riego

Producto	Precio	Cantidad	Subtotal
Sensor sustrato	0,61€	1	0,61€
Wemos D1 mini	1,53€	1	1,53€
Relé	0,64€	1	0,64€
Caja	2,5€	1	2,5€
Batería	1,5€	1	1,5€
Sensor aht10	0,7€	1	0,7€
Bomba agua	0,89€	1	0,89€
LDR	0,41€	1	0,41€
<b>TOTAL</b>	---	---	8,78€

Como podemos apreciar, el precio al por menor del dispositivo no es para nada caro, y una solución comercial podría verse viable con un precio de unos 30€ para un nodo y un servidor.

# **Público objetivo**

Este producto podría ir orientado a personas aficionadas al Bonsai o a las plantas de interior que desean no descuidar su mantenimiento cuando se van de vacaciones o simplemente no se encuentran disponibles para atenderlas.

No solamente va dirigido a ese público, sino que también al avanzado que desea saber todos los parámetros de su planta a todo momento.

# CONCLUSIONES

---

Creemos que la solución ha conseguido toda la funcionalidad deseada y que, aspectos como la interfaz web con el usuario o el consumo energético del nodo (pudiendo aguantar durante 20 días en activo) son unos puntos muy fuertes a tener en cuenta en este primer prototipo.

Se ha intentado también incluir técnicas de fabricación para complementar el proyecto y lograr hacerlo lo más completo posible.

El producto final que fue desarrollado se podría considerar como un éxito y se han cumplido con las especificaciones impuestas en el diseño.

## Mejoras a futuro

Las posibles mejoras a futuro de este proyecto que se podrían implementar disponiendo de más tiempo son:

- Avisos de condiciones extremas detectadas por los sensores.
- Dotar de una mayor funcionalidad a la web añadiendo gráficos.
- Crear y almacenar perfiles de nodos personalizados.

Con esas mejoras el sistema podría convertirse en un producto totalmente utilizable en el día a día.

En cuanto a las referencias de este documento; ha sido una invención mayoritariamente mía y también he utilizado la experiencia de otros proyectos. He de dar las gracias a la cantidad de foros, librerías arduino y tutoriales que han servido en mi aprendizaje y me han ayudado en la implementación.

Con esto finalmente concluimos con este documento, muchas gracias por su atención.

Antonio Vázquez Pérez.

# ÍNDICE DE FIGURAS

Figura 1: Sistema de comunicación.....	8
Figura 2: Diagrama del nodo de riego.....	9
Figura 3: Caso de uso del nodo servidor.....	9
Figura 4: Captura de google shopping.....	10
Figura 5: Diagrama de la bomba automatizada.....	10
Figura 6: Bomba automatizada, control por móvil.....	10
Figura 7: Captura de vídeo de YouTube, sistema casero de monitorización de plantas.....	11
Figura 8: Notas de asignación de pines de la placa.....	12
Figura 9: Captura del software postman; envío de información por parte de un nodo.....	13
Figura 10: Vista de la web en móvil moto G5 plus.....	14
Figura 11: Vista de la web en PC.....	14
Figura 12: Apartados de la aplicación web.....	14
Figura 13: Menú de consulta de datos.....	15
Figura 14: Resultado una vez habiendo pulsado consultar en consulta de datos.....	15
Figura 15: Formulario de registro de nodo.....	16
Figura 16: Sección de enlace a identificador de plantas.....	16
Figura 17: Captura del circuito que conecta con el sensor de sustrato.....	18
Figura 18: Captura software KiCad: Huellas de los componentes colocadas.....	22
Figura 19: Captura del software KiCad: pistas trazadas. Las pistas mostradas en rojo son puentes por la cara delantera.....	22



# ÍNDICE DE ILUSTRACIONES

Ilustración 1: Plantas varias.....	5
Ilustración 2: Nodo servidor utilizado.....	13
Ilustración 3: Nodo servidor; pantalla de arranque.....	13
Ilustración 4: Pantalla mostrando información del nodo.....	15
Ilustración 5: Plano detalle del relé del nodo de riego.....	17
Ilustración 6: Nodo de riego montado en protoboard.....	19
Ilustración 7: Fuente de alimentación USB.....	19
Ilustración 8: Multímetro mostrando el consumo alto.....	19
Ilustración 9: Multímetro mostrando el consumo bajo.....	19
Ilustración 10: Papel transfer. Este papel es utilizado para transferir la tinta adherida en él a otras superficies empleando calor.....	23
Ilustración 11: Papel de acetato.....	24
Ilustración 12: Colocando la máscara de cobre.....	24
Ilustración 13: Transferencia usando calor.....	24
Ilustración 14: Tinta transferida.....	25
Ilustración 15: Atacado con ácido para dejar solamente las partes cubiertas.....	25
Ilustración 16: Placa resultante junto a la máscara de esmalte.....	25
Ilustración 17: Esmalte curando al sol junto a la máscara de terminales.....	25
Ilustración 18: Resultado de la aplicación del esmalte.....	25
Ilustración 19: Serigrafía frontal aplicada.....	26
Ilustración 20: Resultado final del nodo de riego.....	26
Ilustración 21: Resultado final: soldaduras.....	26
Ilustración 22: Caja del servidor con la pantalla.....	27
Ilustración 23: Caja del servidor con la antena introducida.....	27
Ilustración 24: Caja del servidor pintada.....	27
Ilustración 25: Plano detalle del arranque del servidor.....	27
Ilustración 26: Montaje de los componentes del nodo de riego dentro de la caja.....	28

# ÍNDICE DE DIBUJOS

Dibujo 1: Planos de la electrónica del nodo de riego.....	21
Dibujo 2: Capa de máscara de cobre.....	23
Dibujo 3: Máscara para terminales.....	23
Dibujo 4: Serigrafía frontal.....	23