

Memoria de proyecto

Silvanus Bonsai



Sistemas empotrados distribuidos

Universidad complutense de Madrid

Antonio Vázquez Pérez

Índice

ABSTRACT.....	3
INTRODUCCIÓN.....	4
DESCRIPCIÓN.....	5
Nodo principal (Servidor).....	6
Nodo de riego.....	7
Estructura del sistema de comunicación.....	8
Diagrama nodo de riego.....	9
Casos de uso.....	9
ESTADO DEL ARTE.....	10
DESARROLLO.....	12
Consumo energético.....	17
Cálculos.....	19
Acabados.....	20
APLICACIÓN REAL.....	21
Viabilidad económica.....	21
Precio servidor.....	21
Precio nodo de riego.....	21
Público objetivo.....	22
CONCLUSIONES.....	23
Mejoras a futuro.....	23

ABSTRACT

El proyecto que se desea acometer resulta ser una solución al enunciado de trabajo final presentado en la asignatura de Sistemas Empotrados Distribuidos.

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 al que podremos acceder 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.

El resultado fue satisfactorio y se logró implementar el sistema tal como se había imaginado a excepción de la placa PCB.

INTRODUCCIÓN

El proyecto que se desea acometer resulta ser una solución al enunciado de trabajo final presentado en la asignatura de Sistemas Empotrados Distribuidos.

Este enunciado busca una solución hardware distribuida que cumpla con los siguientes requisitos:

El objetivo es diseñar un sistema:

- *Empotrado.*
- *Distribuido (dos placas o más).*
- *De comunicación bidireccional entre los nodos.*

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 más complicados incluso 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 al que podremos acceder para configurar los nodos y ver los datos transmitidos por el resto de nodos.

DESCRIPCIÓN

El dispositivo que se pretende 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 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 almacenado individualmente por planta en forma de mochila



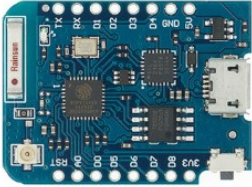


Figura 1: Plantas varias

A continuación se va a proceder 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.



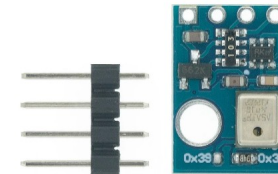
Nodo de riego

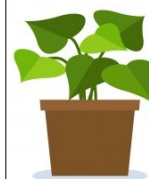
Se puede contar con tantas de estas unidades como se desee, serán las dedicadas a cuidar de la planta sobre la que se coloque.

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

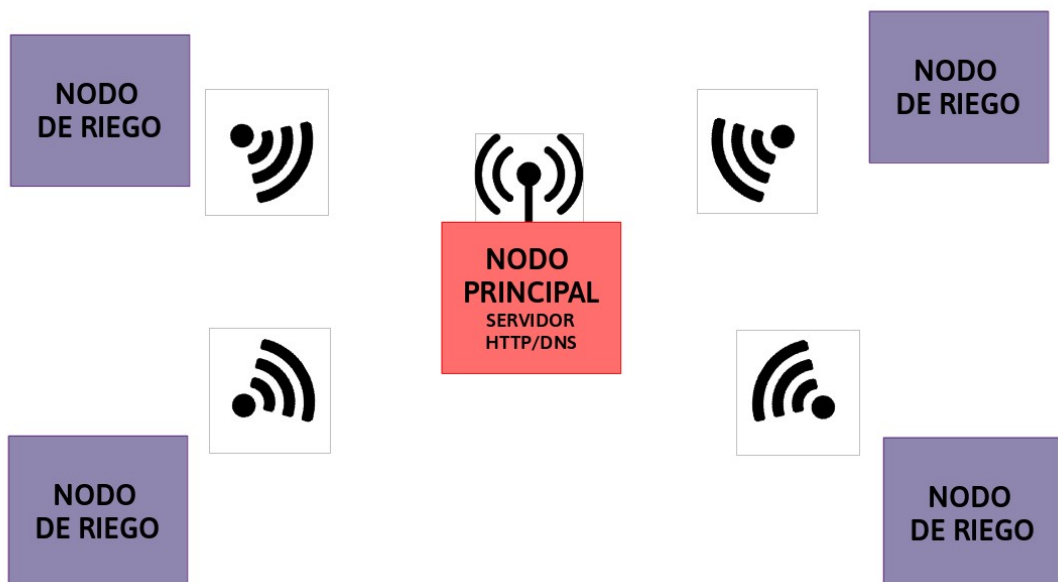


Figura 2: Sistema de comunicación

Diagrama nodo de riego

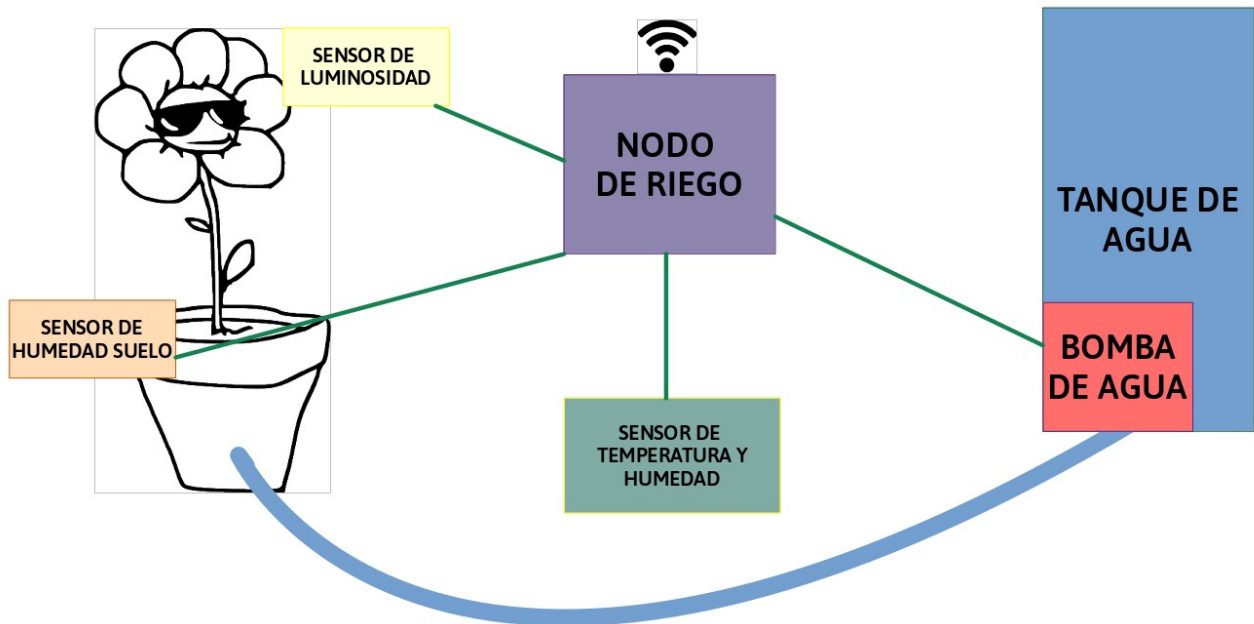


Figura 3: Diagrama del nodo de riego

Casos de uso

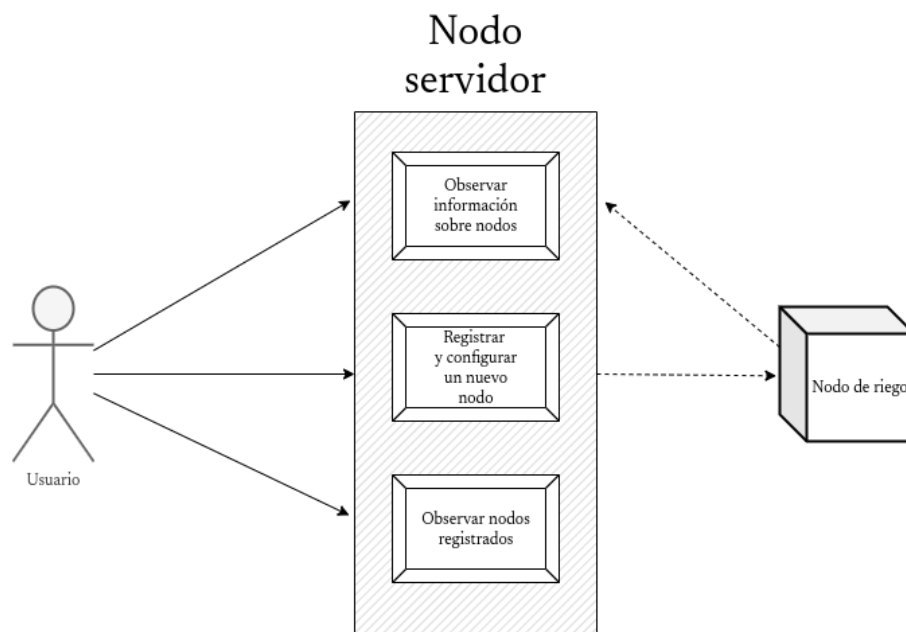


Figura 4: Caso de uso del nodo servidor

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 5: 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:

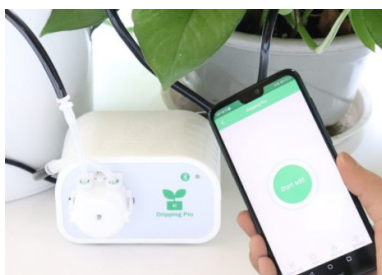


Figura 7: Bomba automatizada, control por móvil.

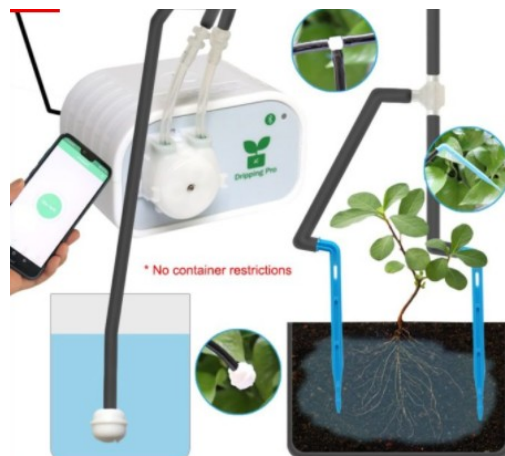


Figura 6: Diagrama de la bomba automatizada

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:

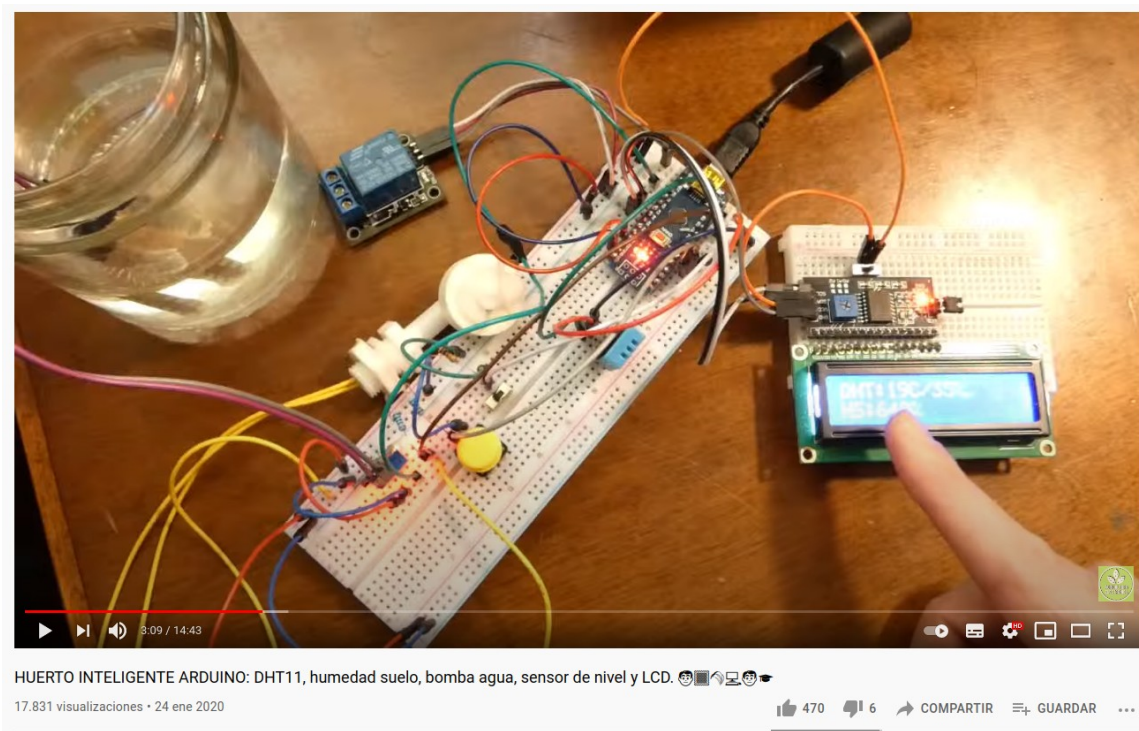


Figura 8: Captura de vídeo de YouTube, sistema casero de monitorización de plantas

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:

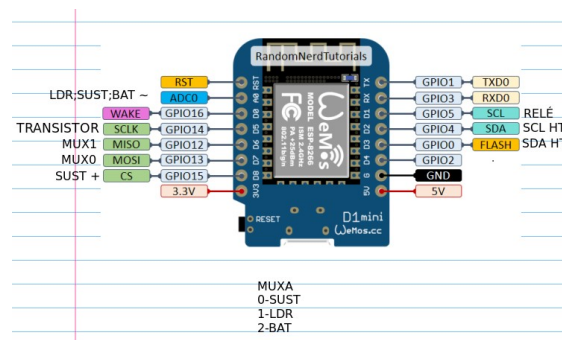


Figura 9: Notas de asignación de pines de la placa.

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.

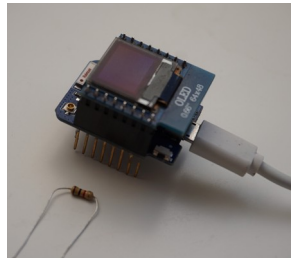


Figura 10: Wemos D1 mini pro utilizado.

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



Figura 11: Pantalla de inicio del servidor.

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.

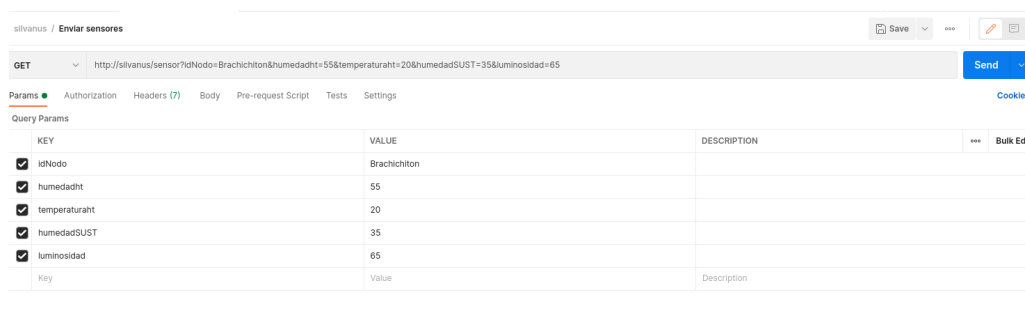


Figura 12: Captura del software postman; envío de información por parte de un nodo.

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:

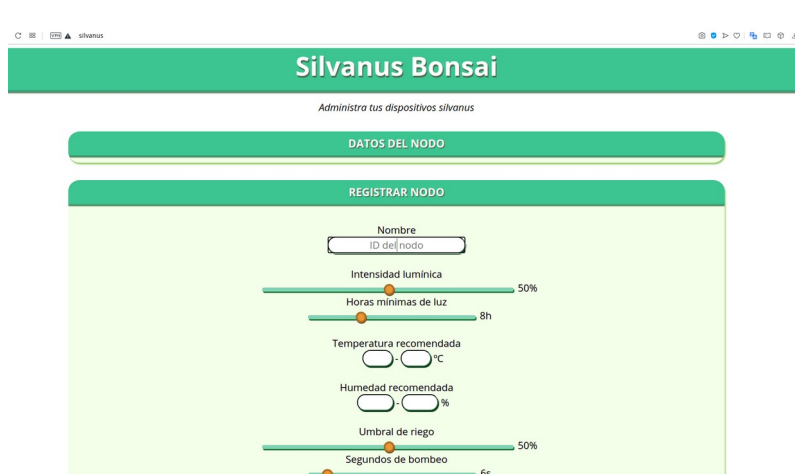


Figura 14: Vista de la web en PC

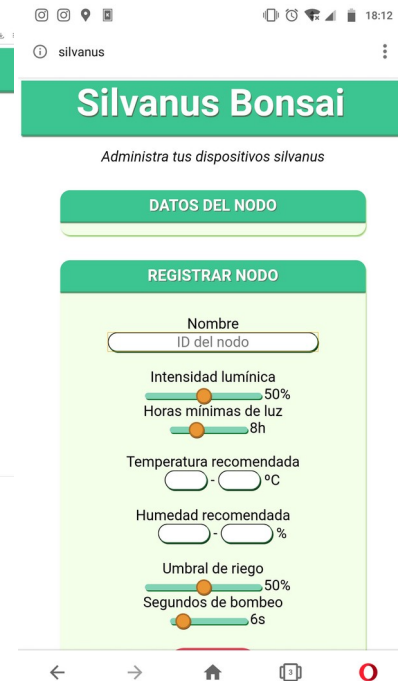


Figura 13: Vista de la web en móvil moto G5 plus

La web tiene 3 apartados:



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.

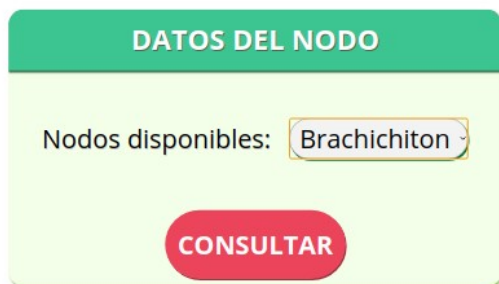


Figura 15: Menú de consulta de datos

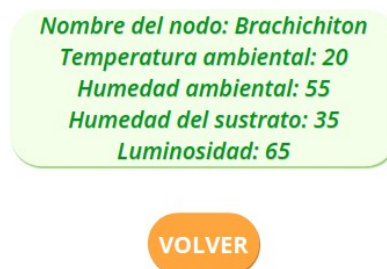


Figura 16: Resultado una vez habiendo pulsado consultar en consulta de datos

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.



Figura 17: 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.

Formulario de registro de nodo con los siguientes campos:

- Nombre:
- ID del nodo:
- Intensidad luminica: 50%
- Horas mínimas de luz: 8h
- Temperatura recomendada: - °C
- Humedad recomendada: - %
- Umbral de riego: 50%
- Segundos de bombeo: 6s
- Botón: REGISTRAR

Figura 18: 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.

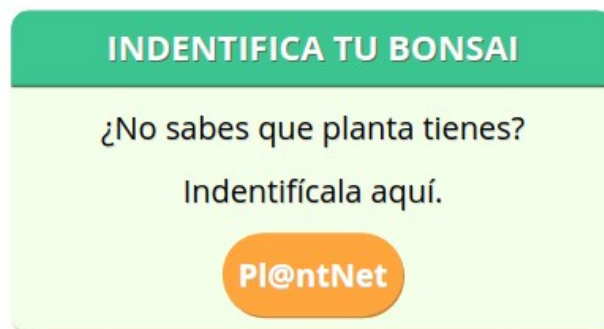


Figura 19: Sección de enlace a identificador de plantas.

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.

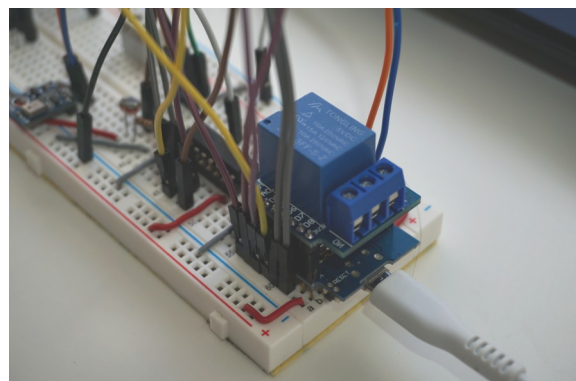


Figura 20: Nodo de riego en protoboard.

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.

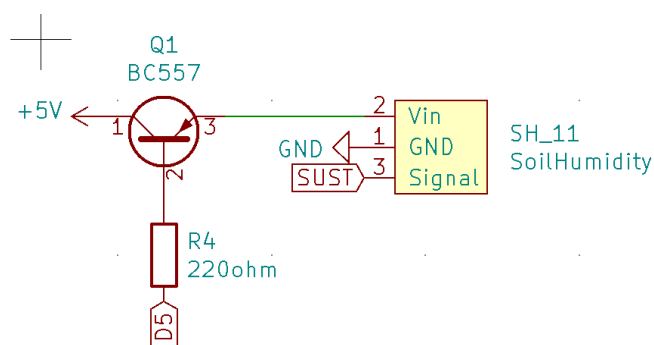


Figura 21: Captura del circuito que conecta con el sensor de sustrato.

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.

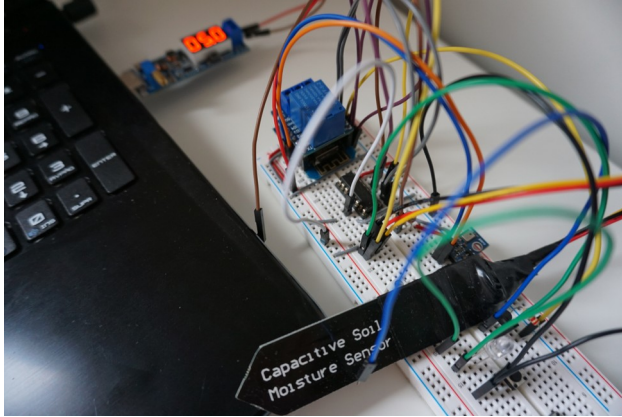


Figura 22: Banco de pruebas

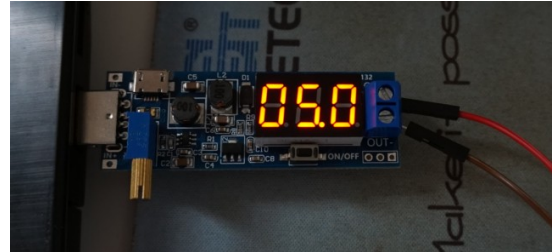


Figura 23: Fuente de alimentación.

Intensidad en activo:



Intensidad en reposo:



Cálculos

Diferencia de consumo 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, pero como probablemente se encienda 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.

En cuanto a la fabricación de la placa PCB se ha elaborado un documento (*introducción a la pcb*) en el que se explica detalladamente todo el proceso que se ha seguido para fabricar la placa. Adelantamos que la placa, debido a inconvenientes y problemas de plazo no hemos sido capaz de fabricarla, pero sí la caja del servidor.

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.



Figura 25: Caja del servidor



Figura 24: Caja del servidor con antena

Tras pintarla un poco este es el resultado final:



Figura 26: Caja final del servidor.



Figura 27: Pantalla del servidor.

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€
TOTAL	---	---	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€
TOTAL	---	---	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

El producto final que fue desarrollado se podría considerar como un éxito parcial al no lograr fabricar la placa para recoger el nodo de riego dentro de la caja.

Aun así, creemos que la solución ha conseguido toda la funcionalidad deseada y que, aspectos como la interfaz con el usuario por parte de la web o el consumo energético del nodo, pudiendo aguantar durante 20 días en activo son unos puntos muy fuertes a tener en cuenta en esta implementación.

Mejoras a futuro

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

- Corrección y fabricación de la PCB.
- Avisos de condiciones extremas detectadas por los sensores.
- Dotar de una mayor funcionalidad a la web añadiendo gráficos.

Con esas mejoras el producto ya se convertiría en un producto totalmente utilizable en el día a día.

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

Antonio Vázquez Pérez.