

# Sistema de tiempo real para el riego en la finca "Carmen Torres"

Christopher Andrés Potosí Carvajal<sup>1</sup>; Bryan Alexander Jinez Cedeño<sup>2</sup>;  
Christian Roberto Tapia Gaibor<sup>3\*</sup>

## Resumen

El sector agropecuario ha comenzado a ser implementado con tecnología, por lo que muchos gobiernos han comenzado a invertir en dichos sectores, por lo que el propósito de la siguiente investigación es la elaboración y optimización de un sistema de automatización en tiempo real para el riego de los cultivos de plátano para la finca "Carmen Torres" del Cantón El Carmen, mediante las herramientas de Arduino basado en un ESP32 y controlado por un sitio web el cual monitorea la humedad del suelo, y todo esto con el fin de optimizar el suministro hídrico, para reducir el desperdicio de agua destinado al riego de los cultivos de plátano. La metodología que se empleó para el desarrollo del proyecto es Extreme Programming. La población que se utilizó en desarrollo del proyecto fueron los trabajadores encargados de la finca. Las técnicas de recolección de datos aplicadas fueron la entrevista y observación sistemática. El sistema resultante está constituido en una placa ESP32, codificado con IDE de Arduino que a través del relay controla el encendido y apagado de la motobomba que permite el envío de agua por medio de las tuberías reparte a cada una de las zonas de riego del cultivo, el mismo que se realiza por aspersión.

**Palabras clave:** Automatización, arduino, riego, sensores, cultivos.

## Real time system for irrigation in the "Carmen Torres" farm

## Abstract

The agricultural sector has begun to be implemented with technology, so many governments have begun to invest in these sectors, so the purpose of the following investigation is the development and optimization of a real-time automation system for irrigation of plantain crops for the "Carmen Torres" farm in Cantón El Carmen, using Arduino tools based on an ESP32 and controlled by a website which monitors soil moisture, and all this in order to optimize the water supply, to reduce the waste of water for the irrigation of banana crops. The methodology used for the development of the project is Extreme Programming. The population that was used in the development of the project were the workers in charge of the farm. The data collection techniques applied were the interview and systematic observation. The resulting system is made up of an ESP32 board, coded with Arduino IDE that through the relay controls the on and off of the motor pump that allows it to send water through the pipes distributed to each of the irrigation areas of the crop, the same that is done by spraying.

**Keywords:** Automation, arduino, irrigation, sensors, crops.

**Recibido:** 17 de noviembre de 2022  
**Aceptado:** 20 de noviembre de 2023

<sup>1</sup> Ingeniero en Sistemas, Universidad Laica Eloy Alfaro de Manabí sede El Carmen

<sup>2</sup> Ingeniero en Sistemas, Universidad Laica Eloy Alfaro de Manabí sede El Carmen

<sup>3</sup> Magister en Gerencia Informática, Universidad Católica del Ecuador sede Ambato

\*Autor para correspondencia [crtg7@hotmail.com](mailto:crtg7@hotmail.com)

## I. INTRODUCCIÓN

La programación del riego en tiempo real basada en tiempo térmico (grados día) ha tenido gran aplicación en granos y hortalizas en el estado de Sinaloa, México, permitiendo manejar escenarios de disponibilidad hídrica y manejo, con ahorros de agua de 1 600 m<sup>3</sup> ha<sup>-1</sup> en maíz, sin reducción de rendimientos. (Sifuentes Ibarra et al, 2015).

La agricultura en el Cantón El Carmen en su mayoría constituye un sistema de manufactura y producción del cultivo de plátano, hoy en día este alimento ha venido creciendo exponencialmente debido a su consumo y adquisición o comercialización produciendo de esta manera empleo (García & Manchón, 2017). El principal motor económico gira entorno a la producción agrícola teniendo al plátano como cultivo principal, donde existe carencia de sistemas automatizado de riego en las plantaciones.

Los métodos tradicionales emplean mano de obra extra para su desarrollo y aplicación; además en las temporadas en que las lluvias disminuyen, puede afectar la producción del producto, siendo este el caso de la finca “Carmen Torres”, lugar donde se pretende automatizar el regadío de acuerdo a las necesidades de la plantas para mantenerlas con nivel de hidratación adecuado y con la menor cantidad de uso del recurso agua y así evitar una deficiencia hídrica que podría afectar la productividad de los cultivos. (Calvache *et al.*, 1997).

Según García & Manchón (2017), un sistema de tiempo real en la respuesta rápida a las interacciones del entorno, puesto que, si la respuesta se entrega después que haya sucedido el evento, ya no tiene importancia dicha respuesta. Mientras que el riego se puede definir como la aportación de suministro hídrico a cultivos por medio del suelo, satisfaciendo de esa manera las necesidades del cultivo, está actividad se suele utilizar en la hidroponía, la agricultura y la jardinería. (Yugueros, 2017).

La aplicación de un sistema de tiempo real para el riego de plátano en el sector agrícola debe ser constante y moderado para que la producción no se vea afectada o en el peor de los casos se arruine totalmente, la agricultura del cultivo de plátano es el objetivo inicial que desarrolla la finca “Carmen Torres” con el propósito de poder generar una

mayor producción e ingresos.

Para el desarrollo del proyecto se ha propuesto que el sistema de riego se base en un ESP32, que consiste en un chip SoC de bajo costo y consumo de energía, con tecnología Wi-Fi y Bluetooth de modo dual integrada con todos los requisitos fundamentales para su respectiva programación. (Bertoleti, 2019).

El sistema se centra en hacer más eficaz la utilización de la tecnología electrónica y planificación del sistema para una serie de procesos o tareas, en el cual este sistema de riego tendrá un mayor rendimiento y eficacia en el trabajo realizado por el sistema de riego en el cultivo de plátano.

La infraestructura y los requerimientos tecnológicos fueron determinados a través del método analítico – sintético, donde se utilizó la observación sistemática y la entrevista como instrumentos de recolección de información. Para cumplir con el objetivo del trabajo se aplicaron tareas científicas como la recolección de información de los sistemas de tiempo real, analizar la estructura del terreno de la finca, determinar los elementos y recursos necesarios para el sistema de tiempo real y finalmente elaborar el prototipo.

Para el desarrollo de la implementación y la generación del software para la aplicación se utilizó una adaptación de la metodología eXtreme Programming (XP).

El propósito de esta investigación fue el diseño de un sistema de tiempo real para el riego de cultivo de plátano por medio de aspersión programada y manejada de manera remota para la repartición del suministro hídrico de modo automatizado, manejando el cultivo de plátano por medio de sensores y temporizadores que se comunican para definir la aplicación del riego.

## II. METODOLOGÍA

Se desarrollo una investigación de tipo cualitativa donde se mostraron los resultados de la información obtenida a través de la aplicación de la entrevista a las personas encargadas de administrar el riego de la finca “Carmen Torres” y mediante la observación realizada al lugar de implementación.

Se realizó la entrevista a los encargados de la finca “Carmen Torres”, a los cuales se aplicó dos

entrevistas, una a personal administrativo y la otra al personal encargado del riego, con el fin de recolectar datos e información que facilite la elaboración del sistema de riego automatizado y definir los requerimientos de la finca. La entrevista se realizó a través de preguntas abiertas y cerradas, donde se confirmó que la finca posee los servicios de luz eléctrica e internet para el correcto funcionamiento del sistema de riego y también se pudo definir los días y horarios de riego, gracias a los datos recolectados. Se definió las actividades que serán controladas a través del microcontrolador implementado en el sistema de tiempo real, dando como resultado una integración funcional entre la interfaz del usuario y los componentes del sistema automático, también se contempló los recursos a utilizar mediante diferentes proveedores el costo de diversos elementos que se necesitaron, el método científico utilizado es el analítico sintético. Los resultados más relevantes son los siguientes:

Los cultivos de plátano alcanzan una extensión de 2 hectáreas ya que su desarrollo es de ciclo largo y necesitan 1 horas de riego por día, pero 4 veces por semana lo cual se encarga el trabajador de ejecutar ese trabajo con guía del administrador. El trabajador supo expresar que durante todo el día se sumista al cultivo de plátano entre 40,000 a 50,000 litros de agua. Cada vez que se va a iniciar la actividad de riego es necesario encender la motobomba, dicho encendido es un proceso tedioso y que requiere de esfuerzo físico. El trabajador expreso que debe suspender sus actividades cotidianas del cuidado del cultivo cuando llega la hora de activar el sistema de riego.

Se determinó la infraestructura más adecuada en base al sistema de riego existente en la finca, las cuales fueron comprobadas a través de la observación utilizando una lista de cotejo como muestra la tabla 1. Se comprobó los elementos básicos que formar parte de las instalaciones de riego en la finca “Carmen Torres”, donde se procedió al diseño de la automatización del sistema.

**Tabla 1.** Lista de Cotejo

Aspectos por verificar	Criterios en base a los componentes	
	Presente	Ausente
Fuente de suministro de agua	X	
Equipo de bombeo	X	
Red de distribución	X	
Tomas de Agua	X	
Bocas de riego	X	
Hidrantes	X	
Elementos Distribuidores	X	
Válvulas		X

El coste de implementación para el diseño del sistema de riego automatizado se detalla en la tabla 2, donde se analiza los elementos utilizados para la implementación sin tomar en cuenta los equipos del sistema de riego ya presentes en la finca, cabe indicar que estos presupuestos se tomaron de varios proveedores.

**Tabla 2.** Presupuesto para implementación del diseño

Descripción	Cantidad	Precio U	Valor Total
Microcontrolador ESP32	1	\$15,00	\$15,00
STARTER 497596			
V TWIN Motor de arranque para Briggs & Stratton 16 HP con engranaje	1	\$50,00	\$50,00
Protoboard	1	\$5,00	\$5,00
Modulo Relay de 2 vías	1	\$3,00	\$3,00
Sensor de humedad a tierra F-C 28	1	\$7.5	\$7.5
Fuente de alimentación 12VDC	1	\$4,00	\$4,00
Cable Dupont	1	\$3,00	\$3,00
Router Huawei Wifi	1	\$76,00	\$76,00
Caja de acrílico	1	\$20,00	\$20,00
Cable USB	1	\$5,00	\$5,00
Resistencia Homs	1	\$0,10	\$0,10
Luces LED	1	\$0,20	\$0,20
<b>Total</b>			<b>\$188,80</b>

En el desarrollo del prototipo se utilizó la metodología XP, ya que el conjunto de prácticas interrelacionadas permitió un enfoque de funcionalidad que se pretendía en el proyecto de software considerando las variables de coste, tiempo, calidad y alcance. Para ello fue necesario: comprender las necesidades del cliente, valorar el esfuerzo, generar la solución y entregar el producto finalizado.

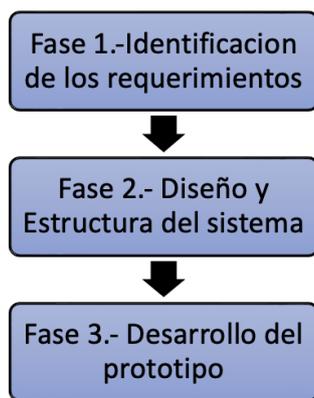


Imagen 1. Esquema de la propuesta

### III. DESARROLLO Y DISCUSIÓN

#### Identificación de los requerimientos (Fase 1)

La entrevista realizada a la Finca “Carmen Torres” ha podido establecer requerimientos indispensables para poder automatizar el riego del cultivo de plátano. Se ha podido establecer también que el riego es de forma manual.

- La persona encargada del estado del cultivo de plátano identificó problemas en el riego al cultivo, debido a que se realiza manualmente, existe problemas en la técnica provocando daño por frío, enfermedades y otra serie de efectos.
- Se observó por medio del encargado de la finca que cuenta con todos los servicios básico y de internet que favorecen el desarrollo del sistema de riego automatizado en el terreno que se desea implementar.
- Los cultivos de plátano alcanzan una extensión de dos hectáreas ya que su desarrollo es de ciclo largo y necesitan una hora de riego por día, pero cuatro veces por semana lo cual se encarga el trabajador de ejecutar ese trabajo con guía

del administrador.

- El trabajador supo expresar que durante todo el día se sumista al cultivo de plátano entre 40,000 a 50,000 litros de agua.
- Cada vez que se va a iniciar la actividad de riego es necesario encender la motobomba, dicho encendido es un proceso que toma tiempo y requiere de esfuerzo físico.

El trabajador expreso que debe suspender sus actividades cotidianas del cuidado del cultivo cuando llega la hora de activar el sistema de riego.

#### Observación del sistema de riego

El abastecimiento y el suministro de agua cuenta con un pequeño mecanismo automático el cual permitía que el contenedor de agua siempre se mantuviera lleno, haciendo uso de una bomba sumergible instalada en un pozo profundo de 70 metros.

El sistema de riego existente se encuentra enterrado y cubre la extensión de 2 hectáreas con una superficie plana empleando 50 aspersores por cada hectárea, el suministro de agua es absorbido por una motobomba capas de distribuir entre 40,000 a 50,000 litros de agua en una hora, esto es debido a que el terreno en el cual se aplica el riego tiene un poco de arcilla por lo que agua tiende a estancarse y a evaporarse.



Imagen 2. Sistema de riego de la finca “Carmen Torres”

Al momento de inspeccionar el equipo de bombeo se pudo apreciar que el encendido de la motobomba que distribuye el agua a través de la red de riego debe ser encendida manualmente, por lo tanto, es necesario incorporar un sistema de arranque eléctrico.

El sistema de riego por aspersión que se ejecuta en la finca “Carmen Torres” cuenta con 50 aspersores por hectárea que funcionan de manera

manual con la participación de 2 personas en la actividad del riego, de esa manera con ayuda de una motobomba la cual es encendida por un trabajador, se suministra el agua obtenida de una piscina que puede almacenar hasta 200 mil litros. Este proceso se repite 4 veces a la semana. En tiempo de sequía en el terreno no existe ninguna medida de seguridad que proteja el cultivo.

### Estructura y Diseño del Sistema (Fase 2)

Se llegó a la siguiente elección tomando en cuenta los siguientes factores: la herramienta de desarrollo, su documentación, su costo y las características fundamentales del microcontrolador.

El microcontrolador ESP32 provee de conectividad tanto WIFI como Bluetooth lo que la hace apreciada para diversos proyectos, esta ofrece potencia computacional: en CPU y en memoria, proyectos basados en comunicaciones, en seguridad (Bertoleti, 2019).



Imagen 3. Microcontrolador ESP32

Se puede apreciar que el punto fuerte de esta plataforma es su gran comunidad en internet que le brinda soporte y desarrolla constantemente nuevas herramientas para su uso. Posee una gran ventaja ya que cuenta con variedad de software, lenguajes de programación, frameworks, librerías, códigos, y otros recursos. También las tarjetas ESP poseen la capacidad de conectarse a Wifi de forma nativa.

Características del microcontrolador ESP32

- Voltaje de Alimentación (USB): 5V DC
- Voltaje de Entradas/Salidas: 3.3V DC
- CPU principal: Tensilica Xtensa 32-bit LX6
- Frecuencia de Reloj: hasta 240Mhz
- Desempeño: Hasta 600 DMIPS
- WIFI: 802.11 b/g/n/e/i (802.11n @ 2.4 GHz hasta 150 Mbit/s)
- Bluetooth: v4.2 BR/EDR and Bluetooth Low Energy (BLE)
- Memoria:

- 448 KByte ROM
- 520 KByte SRAM
- 16 KByte SRAM in RTC
- QSPI Flash/SRAM, 4 MBytes
- Pines Digitales GPIO: 24 (Algunos pines solo como entrada)
- Conversor Analógico Digital: Dos ADC de 12bits tipo SAR, soporta mediciones en hasta 18 canales, algunos pines soportan un amplificador con ganancia programable
- Chip USB-Serial: CP2102
- Antena en PCB

Un dato curioso sobre el microcontrolador es que la comunidad ha desarrollado un plugin que lo convierte además en un procesador compatible Arduino que incluye WIFI siendo además compatible de esta manera con las librerías oficiales.

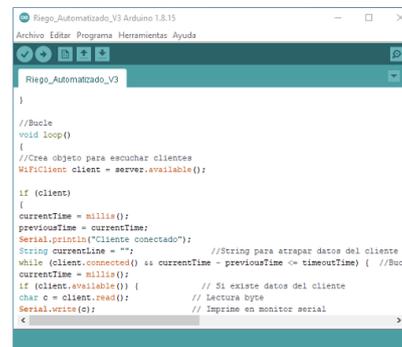


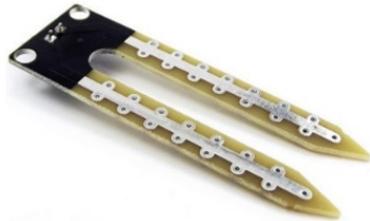
Imagen 4. IDE Arduino

El IDE Arduino es un software que se distribuye bajo licencia libre, de modo que es posible acceder al código fuente del IDE Y construir el instalador desde él o realizar las modificaciones que consideres necesarias. (Peña C. , 2020).

Los programas de Arduino están compuestos por un solo fichero con extensión “ino”, aunque es posible organizarlo en varios ficheros. El fichero principal siempre debe estar en una carpeta con el mismo nombre que el fichero, La plataforma Arduino se codifica por medio del uso de un lenguaje propio basado en el lenguaje de programación de alto nivel, usualmente se suele utilizar el lenguaje de programación C++, no obstante, es posible utilizar otros tipos de lenguajes de programación y aplicaciones populares en Arduino.

Sensor de humedad a tierra F-C 28, es un conector que calcula el bochorno general en un

espacio determinado. Se puede utilizar un sensor de humedad tanto en el interior como en el exterior. Los sensores de adherencia son accesibles en estructuras simples y avanzadas (RODRÍGUEZ, 2018).



**Imagen 5.** Sensor de humedad del suelo FC-28

La actividad del sensor depende de estimar la oposición entre 2 cátodos incrustados en la tierra, la obstrucción entre los terminales dependerá de la humedad del suelo. La actividad del sensor depende de estimar la obstrucción entre 2 electrodos incrustados en la tierra, la posición entre las terminales dependerá de la humedad del suelo. del cultivo de plátano, por lo que para un suelo muy húmedo tiene una resistencia muy baja (corto circuito) y para un suelo muy seco la resistencia es muy alta (circuito abierto).



**Imagen 6:** Módulo relay 2 canales

Es un dispositivo sensible a las sobreintensidades que circulan en su interior. Está dividido en un circuito electromagnético o electroimán y en uno

o varios contactos que cambiarán su posición si se sobrepasa el corriente umbral” (Diez et al., 2017).

Como parte de los actuadores el módulo relay de dos vías o canales cumplirá su función de relé como circuito electrónico que servirá para controlar circuito que trabajan a tensiones superiores al de los circuitos principal, siendo más específicos se encargara de controlar el motor de arranque STARTER 497596 V TWIN.



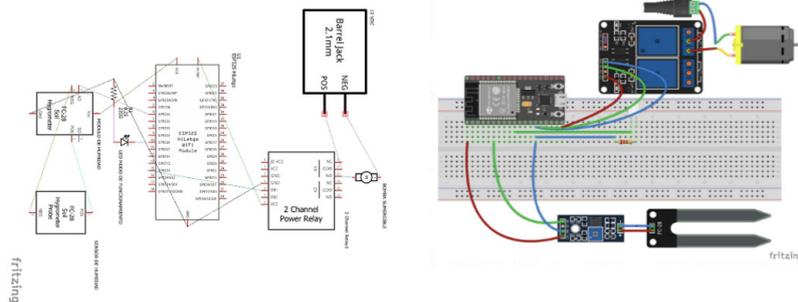
**Imagen 7:** Motor de arranque Starter

“El motor de arranque es un motor eléctrico con excitación en serie. El arrollamiento de excitación y el arrollamiento del inducido están conectados en serie” (Hamm, 2021).

El motor de arranque STARTER 497596 V TWIN en conjunto al relé mencionado anteriormente se encargará de encender y apagar una motobomba de 16 hp la cual se encarga de distribuir el recurso hídrico a través de toda la red de agua para final desembocar en los aspersores que posee la finca “Carmen Torres”.

**Software empleado para la descripción del circuito**

Fritzing es un software gratuito de diseño electrónico, para crear diagramas electrónicos y los circuitos impresos, en el que se desarrolló los diagramas del presente proyecto.



**Imagen 8:** Diseño del circuito del sistema y sus componentes

- Se necesita realizar la detección de humedad ya sea abundante o mínima en el terreno para evitar ciertos efectos dañinos para la planta de plátano por medio de sensores que monitoreen el estado del suelo.
- Al existir una gran cantidad de cultivos de plátano en un extenso terreno con superficie plana y cierta cantidad de arcilla

en suelo es necesario implementar un sistema automático que agilice el trabajo y monitoree del suelo en el que se va a aplicar.

- Para automatizar el proceso de encendido y pagado de la motobomba se deberá adaptar un arranque eléctrico el cual deberá ser controlado por un relé en base a los intervalos de tiempo ya mencionados (1 hora durante - 4 veces a la semana).

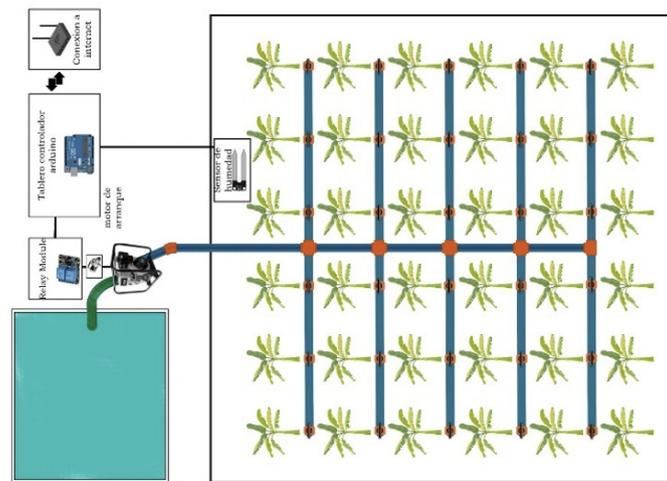


Imagen 9: Estructura del diseño del sistema de riego automatizado

Para permitir el control de estos procesos se implementó una página web para gestionar el tiempo de riego y la humedad del suelo desde cualquier dispositivo conectado a internet.

### Desarrollo del Prototipo (Fase 3)

El entorno de desarrollo integrado o IDE de Arduino es una aplicación multiplataforma que puedes utilizar para escribir y cargar programas en placas Arduino y también en aquellas que sean compatibles. Pero no solo eso, ya que, gracias a núcleos generados por terceros, también se pueden utilizar para cargar programas en placas de desarrollo de otros proveedores (Peña C. , 2020).

Se utilizaron las librerías necesarias para poder establecer la conexión Wifi y definir los procesos de automatización previamente interpretadas en base a los requisitos del usuario. Para obtener la librería de tarjeta de desarrollo ESP32 fue necesario utilizar el siguiente link obtenido de la página web “Prometec”: [https://dl.espressif.com/dl/package\\_esp32\\_index.json](https://dl.espressif.com/dl/package_esp32_index.json), el cual permite la

instalación del ESP32 en nuestro IDE Arduino, a través del gestor de tarjetas. La librería Wifi forma parte del pack de librerías que viene por defecto en el IDE Arduino y de igual manera debe ser instalado a por medio del gestor de tarjetas.

Para poder cargar el código es necesario conectar el Arduino a través de un cable USB que permite la comunicación con el ordenador.

Para el funcionamiento del sistema de la planta de riego automatizado el cerebro que controla los procesos es la placa de adquisición de datos ESP32 Wi-Fi + Bluetooth, que tiene una entrada micro-usb tipo B que va conectada con un cable USB tipo A al computador, ya sea para cargar el código programado a través del IDE de Arduino o para mantener encendida.

Se considera como sensor al módulo medidor de humedad en tierra FC-28 que se lo conecta a la salida de 3.3 VDC (3.3) y tierra (GND) de la tarjeta ESP32 para que varíe su salida analógica desde 0 a 3.3 VDC ya que las entradas analógicas de la ESP32 solo soportan hasta esa cantidad de voltaje.

El actuador usado es una bomba sumergible que se la está activando con 12 VDC por lo que se ha optado por usar un relay alimentado por la salida de 5 VDC (V5) y GND de la ESP32, que se activa por una de sus salidas digitales que cuando está en estado lógico BAJO (LOW) da 0 VDC que activan al relay y por ende a la bomba, en cambio al estar en estado lógico ALTO (HIGH) da 5 VDC que desactiva al relay y a la bomba. Se usa una fuente externa de 12 VDC para activar la bomba con el relay, donde se conecta directamente la salida negativa de la fuente a una de las entradas de la bomba y la salida positiva al pin común (COM del inglés Common) del relay, para que el pin Normalmente Abierto (NO del inglés Normally Open) quede conectado a la otra entrada de la bomba, no hay problema al cambiar la polaridad de la conexión con la fuente ya que la bomba trabaja siempre en un solo sentido.

El LED indica el estado en que se encuentra el modo de funcionamiento del sistema de la planta de riego automatizado, al encontrarse la salida digital de la ESP32 en un estado lógico ALTO (HIGH) se encenderá el LED indicando que el modo AUTOMÁTICO está activo, al cambiar al estado lógico BAJO (LOW) se apagará el LED indicando que se activó el funcionamiento MANUAL de la planta. Se conecta directamente una salida de la ESP32 al pin positivo del LED y el negativo a una resistencia de 220 Ω (ohms) en serie con la salida GND de la tarjeta ESP32.

Para un mejor entendimiento se puede observar en las Ilustraciones 8 y 9 así mismo la Tabla 3, donde se indica en que pines GPIO se encuentran funcionando los pines de entradas y salidas de la ESP32.

Tabla 3. Pines de módulos y ESP 32

DISPOSITIVO	PINES (DISPOSITIVO)	PINES (ESP32)
SENSOR DE HUMEDAD EN TIERRA FC-28	VCC	3.3
	GND	GND
	AO	GPIO35
MÓDULO RELAY DE 2 VÍAS	VCC	V5
	GND	GND
	IN1	GPIO13
LUZ LED ROJA	POSITIVO	GPIO26
	NEGATIVO	GND en series con resistencia de 220 Ω

**Programación del Arduino para el enlace con la página web**

Para crear un enlace de la tarjeta de adquisición de datos ESP32 con el dominio y host para la página web, primero se debe tener un localhost con la IP propia de la tarjeta, la cual se puede observar en el Monitor Serial del IDE de Arduino al dejarlo en 115200 baudios, al conectar la ESP32 cargada con la programación realizada, en la

sección de configuración void setup() { en la línea de código Serial.println(WiFi.localIP());}, se obtiene la IP local de la tarjeta 192.168.100.14 , como se observa en el ejemplo de la Ilustración 10, entonces se procede a colocar una dirección de puerto cualquiera excepto en la número 80, ya que el programa “ngrok” no funciona con ese puerto, entonces se coloca el puerto 8888, al inicio del código en la línea WiFiServer server(8888);.

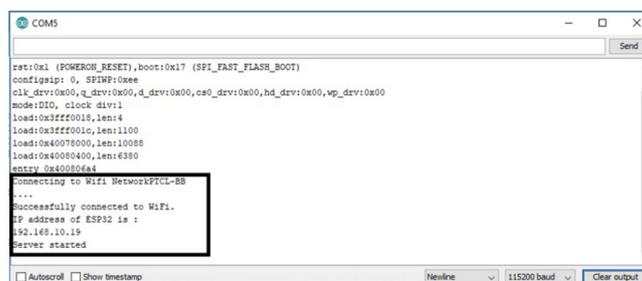


Imagen 10: Obtención de IP

Una vez obtenida la dirección local de la página web se procede a corroborar que funcione correctamente en nuestra red Wi-Fi, sin desconectar el ESP32, en cualquier navegador colocamos la dirección.

<http://192.168.100.14:8888>, que muestra la interfaz de la planta de riego automatizada, como se observa en la Ilustración 11.

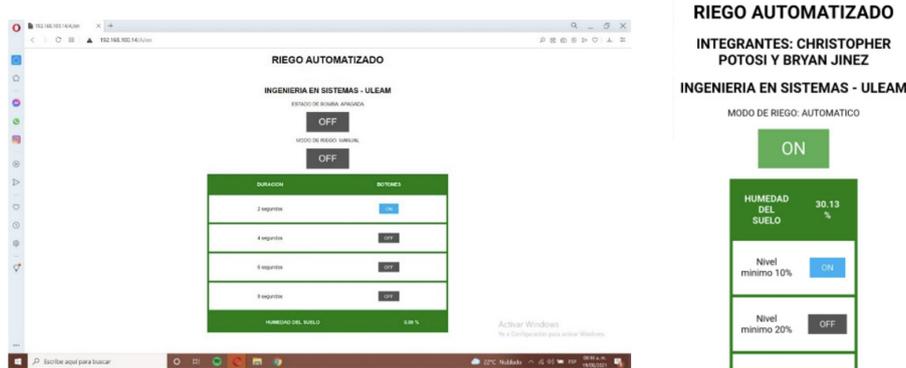


Imagen 11: Interfaz de la planta automatizada desde el ordenador y del móvil

Para usar el programa “ngrok” que sirve de túnel del localhost a la WEB, se crea una cuenta y usuario en su página web, se descarga el programa “ngrok.exe” de la misma página, se accede a la ubicación de dicho archivo entrando a CMD como Administrador, luego se regresa a la página web de ngrok para obtener un código de autenticación en la pestaña “Your Authtoken”, que se usará como “authtoken” reemplazando las “X” para el comando que se enviará por CMD de la siguiente forma:

```
ngrok.exe tcp 192.168.100.14:8888 --authtoken  
XXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

La página web generada es mostrada en la Ilustración 12, donde en lugar del tcp:// se usa http:// para acceder desde cualquier navegador ya sea en una computadora o celular desde cualquier parte del mundo ya sea con conexión Wi-Fi o datos, mientras se mantenga encendida la computadora donde se creó el túnel con el programa ngrok corriendo y la ESP32 conectada funcionando. En caso de cerrar el programa ngrok, no funcionará la dirección web generada por lo que se debe volver a realizar el proceso por CMD indicado anteriormente para que se vuelva a generar una dirección de página web nueva.

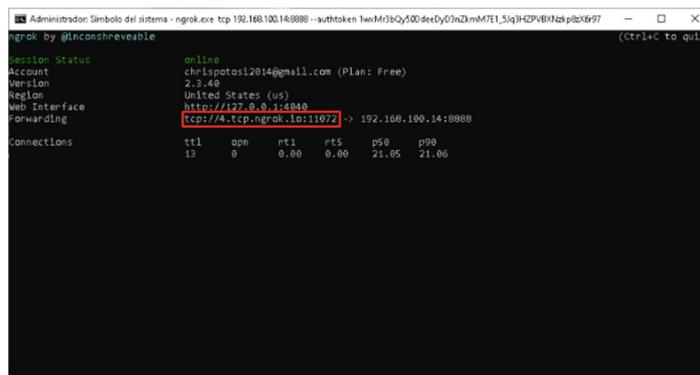


Imagen 12: Obtención de página web a través de Ngrok

### Programación de la página web

La programación de la página web se la realizo en HTML con CSS para darle un mejor aspecto, el código se lo incluye en el programa “.ino” que se carga en la tarjeta ESP32, desarrollado en el IDE

de Arduino en la sección del bucle void loop(), es decir que directamente se tiene la página web en la tarjeta ESP32 sin necesidad de subirla a un host. A continuación, se detalla brevemente la programación en las líneas de código que

se encuentra con “//” al inicio, para enviar la programación HTML de la página se usa el código “client.println();” de la librería de Arduino “WiFi.h”.

```
//Código HTML
client.println("<!DOCTYPE html><html>");
//Se establece configuración inicial de la
página
client.println("<head><meta name=\"viewport\"
content=\"width=device-width, initial-scale=1\">");
client.println("<link rel=\"icon\" href=\"data:,\">");

//Configuración de estilo en CSS para el
CUERPO y TABLA
//posición de alineación y fuente de letra del
cuerpo de la página
client.println("<style>body { text-align: center; font-
family: \"Arial\", Arial;}");
//tamaño y color de borde y margen de los
títulos de la tabla
client.println("<table { border-collapse: collapse;
width:40%; margin-left:auto; margin-right:auto;border-
spacing: 2px;background-color: white;border: 4px solid
green; }");
//Tamaño y color de las filas y columnas
client.println("<th { padding: 20px; background-color:
#008000; color: white; }");
client.println("<tr { border: 5px solid green; padding:
2px; }");
client.println("<tr:hover { background-color:yellow;
}");
client.println("<td { border:4px; padding: 12px; }");
client.println("<.sensor { color:white; font-weight:
bold; background-color: #bebebe; padding: 1px; }");

//Encabezado
client.println("</style></head><body><h1>RIEGO
AUTOMATIZADO</h1>");
client.println("<h2>INTEGRANTES:
CHRISTOPHER POTOSI Y BRYAN JINEZ</h2>");
client.println("<h2>INGENIERIA EN SISTEMAS -
ULEAM</h2>");

// Estilo en CSS para BOTONES
// Tamaños y colores de los botones de la
página web
client.println("<style>html { font-family: Helvetica;
```

```
display: inline-block; margin: 0px auto; text-align:
center;});
```

**//botones para el funcionamiento en modo AUTOMÁTICO**

```
//botón de ON
client.println("<.button { background-color: #4CAF50;
border: none; color: white; padding: 16px 40px;");
client.println("<text-decoration: none; font-size:
30px; margin: 2px; cursor: pointer;});");
```

```
//botón de OFF
client.println("<.button2 {background-color:
#555555;});");
```

**//botones para el funcionamiento en modo MANUAL**

```
//botón de ON
client.println("<.button3 {background-color:
#00B0F6; padding: 8px 20px; font-size: 15px; margin:
1px;});");
```

```
//botón de OFF
client.println("<.button4 {background-color:
#555555; padding: 8px 20px; font-size: 15px; margin:
1px;}</style>");
```

En el siguiente código se detalla la selección del estado de la bomba, con la ayuda de un botón en la página web desarrollada se enciende o se apaga la bomba realizando un llamado al servidor interno del ESP32.

**// Cambia el estado del boton de BOMBA debido al click**

```
if (outputMState=="MANUAL"){
//outputBState muestra si el estado del botón
está en ENCENDIDA o //APAGADA
client.println("<p>ESTADO DE BOMBA: " +
outputBState + "</p>");
```

**// Si se pulsa el botón en OFF cambia outputBState a ENCENDIDA y //muestra al botón en ON**

```
if (outputBState=="ENCENDIDA") {
//Envía la referencia “href” al final de la
dirección de la página web para //ser atrapado
el click en el botón por la ESP32, es decir que
mostrará //http://192.168.100.14/18/on, y lo
mismo hace en todos los “href” //siguientes.
client.println("<p><a href=\"/18/on\"><button
class=\"button\">ON</button></a></p>");
} else {
```

```

client.println("<p><a href=\" /18/off\"><button
class=\"button button2\">OFF</button></a></p>");
}

```

**//Tabla Tiempo Bomba**

```

client.println("<table><tr><th>DURACION</
th><th>BOTONES</th></tr>");

```

Código para el establecimiento del tiempo de encendido de la bomba, donde inicialmente se coloco 4, 6 y 8 segundos quedando la opción de aumentar más tiempos según la necesidad.

**// Duración de bomba establecido con botones**

**//Tiempo 2 segundos**

```

client.println("<tr><td>2 segundos</td><td>");

```

**//Según el estado del botón actúa para cambiar la duración del tiempo de //encendido de la bomba, en la variable “durB” con su valor en //milisegundos es decir que 2000 equivale a 2 segundos. Realiza lo //mismo para los demás botones del tiempo de la bomba**

```

if (outputD2State=="on") {
client.println("<p><a href=\" /D2/on\"><button
class=\"button button3\">ON</button></a></p>");
durB=2000;
} else {
client.println("<p><a href=\" /D2/off\"><button
class=\"button button4\">OFF</button></a></p>");
}
client.println("</td></tr>");

```

**//Tiempo 4 segundos**

```

client.println("<tr><td>4 segundos</td><td>");
if (outputD4State=="on") {
client.println("<p><a href=\" /D4/on\"><button
class=\"button button3\">ON</button></a></p>");
durB=4000;
} else{
client.println("<p><a href=\" /D4/off\"><button
class=\"button button4\">OFF</button></a></p>");
}
client.println("</td></tr>");

```

**//Tiempo 6 segundos**

```

client.println("<tr><td>6 segundos</td><td>");
if (outputD6State=="on") {
client.println("<p><a href=\" /D6/on\"><button
class=\"button button3\">ON</button></a></p>");
durB=6000;

```

```

} else{
client.println("<p><a href=\" /D6/off\"><button
class=\"button button4\">OFF</button></a></p>");
}
client.println("</td></tr>");

```

**//Tiempo 8 segundos**

```

client.println("<tr><td>8 segundos</td><td>");
if (outputD8State=="on") {
client.println("<p><a href=\" /D8/on\"><button
class=\"button button3\">ON</button></a></p>");
durB=8000;
} else{
client.println("<p><a href=\" /D8/off\"><button
class=\"button button4\">OFF</button></a></p>");
}
client.println("</td></tr>");

```

**//SE PUEDEN AUMENTAR MÁS TIEMPOS DE DURACIÓN**

```

}

```

**// Cambia el estado del boton de MODO debido al click**

```

client.println("<p>MODO DE RIEGO: " +
outputMState + "</p>");
if (outputMState=="AUTOMATICO") {
client.println("<p><a href=\" /A/on\"><button
class=\"button\">ON</button></a></p>");
} else {
client.println("<p><a href=\" /A/off\"><button
class=\"button button2\">OFF</button></a></p>");
}

```

Manejo del sensor de humedad para las mediciones en el suelo, manejado mediante el puerto analógico del ESP32.

**//Muestra valor de Humedad**

```

client.println("<table><tr><th>HUMEDAD
DEL SUELO</th><th>");

```

**//el puerto analógico de la ESP32 tiene 12 bytes = 4095 bites se hace una //regla de 3 para tener en porcentaje, y a 100 se le resta dicho valor para //obtener una humedad creciente de 0-100%**

```

humedad=100.0-(100.0/4095.0)*analogRead(
sensorhum);
client.println(humedad);
client.println(" %</th></tr>");

```

```

//Muestra botones para establecer el
nivel mínimo de humedad
if (outputMState=="AUTOMATICO"){
// Niveles minimos establecidos con
botones
//Nivel 10%
client.println("<tr><td>Nivel minimo 10%</
td><td>");
//Según el estado del botón actúa para
cambiar el nivel mínimo de //humedad,
en la variable "minH". Se realiza lo mismo
para los demás //botones del nivel.
if (outputH10State=="on") {
client.println("<p><a href=\"//H10/
on\"><button class=\"button button3\">ON</
button></a></p>");
minH=10;
} else {
client.println("<p><a href=\"//H10/
off\"><button class=\"button button4\">OFF</
button></a></p>");
}
client.println("</td></tr>");
//Nivel 20%
client.println("<tr><td>Nivel minimo 20%</
td><td>");
if (outputH20State=="on") {
client.println("<p><a href=\"//H20/
on\"><button class=\"button button3\">ON</
button></a></p>");
minH=20;
} else{
client.println("<p><a href=\"//H20/
off\"><button class=\"button button4\">OFF</
button></a></p>");
}
client.println("</td></tr>");
//Nivel 30%
client.println("<tr><td>Nivel minimo 30%</
td><td>");
if (outputH30State=="on") {
client.println("<p><a href=\"//H30/
on\"><button class=\"button button3\">ON</
button></a></p>");
minH=30;
} else{
client.println("<p><a href=\"//H30/
off\"><button class=\"button button4\">OFF</

```

```

button></a></p>");
}
client.println("</td></tr>");
//Nivel 40%
client.println("<tr><td>Nivel minimo 40%</
td><td>");
if (outputH40State=="on") {
client.println("<p><a href=\"//H40/
on\"><button class=\"button button3\">ON</
button></a></p>");
minH=40;
} else{
client.println("<p><a href=\"//H40/
off\"><button class=\"button button4\">OFF</
button></a></p>");
}
client.println("</td></tr>");
// SE PUEDEN AUMENTAR MÁS
NIVELES MÍNIMOS DE HUMEDAD
}
//fin HTML
client.println("</body></html>");

```

### Conclusiones

En base a los requisitos y necesidades del personal de la finca "Carmen Torres" se diseño el sistema de riego automatizado para el cultivo de plátano, atendiendo el tiempo que les toma el poder iniciar el riego en la finca, así como el uso de personal dedicado para dicha labor. Es bien sabido que los sistemas de riego por aspersión presentan un menor consumo de agua y es la mejor opción debido a que imita a la lluvia, la calidad de la entrega de agua a los cultivos es mucho mejor, por lo que la automatización implementada ayudo significativamente a las labores de la finca.

Se pudo determinar que el producto final no representa un valor económico elevado, además posee un bajo consumo eléctrico, esto significa que el desarrollo de un sistema de tiempo real para aplicación de riego en cultivos es viable y accesible para los productores agrícolas, donde los requerimientos de infraestructura se centran a un adecuado sistema eléctrico y de preferencia tener el servicio de Internet, aunque no es imprescindible.

### Referencias

Bertoletti, P. (2019). Proyectos con ESP32 y LoRa. Recuperado de <https://www.casadellibro.com/ebook-proyectos-com-esp32-y-lora-ebook/9788595680661/10085214>

- Caicedo , A. (2017). Arduino para Principiantes: 2ª Edición. Recuperado de [https://books.google.fr/books?id=Fw\\_RDgAAQBAJ&pg=PA14&lpq=PA14&dq=control+arduino+desde+ordenado&source=bl&ots=QVkzUmTa30&sig=ACfU3U35uCAxbajc5LA2QPX5JhPM-7QnKRQ&hl=es&sa=X&ved=2ahUKEwjRxKfqsYX5AhUagHMKHWZLBBkQ6AF6BAglEAM#v=onepage&q=control%20arduino%20desde%20ordenado&f=false](https://books.google.fr/books?id=Fw_RDgAAQBAJ&pg=PA14&lpq=PA14&dq=control+arduino+desde+ordenado&source=bl&ots=QVkzUmTa30&sig=ACfU3U35uCAxbajc5LA2QPX5JhPM-7QnKRQ&hl=es&sa=X&ved=2ahUKEwjRxKfqsYX5AhUagHMKHWZLBBkQ6AF6BAglEAM#v=onepage&q=control%20arduino%20desde%20ordenado&f=false)
- Candil, I. M. (2015). UF2121 - Programación del trabajo de campo de la investigación – España. Recuperado de [https://books.google.com.ec/books?id=HG-NWDwAAQBAJ&printsec=frontcover&hl=es&source=gbs\\_atb#v=onepage&q&f=false](https://books.google.com.ec/books?id=HG-NWDwAAQBAJ&printsec=frontcover&hl=es&source=gbs_atb#v=onepage&q&f=false)
- Chetto, M., & Queudet, A. (2020). Sistemas de tiempo real autónomos en energía. Reino unido. Recuperado de [https://books.google.com.pr/books?id=97zTD-wAAQBAJ&pg=PR5&lpq=PR5&dq=tiempo%20real&source=bl&ots=3IyGC3UoC&sig=ACfU3U-1DZY\\_Vrjd8XlOSXj7DjF1FCEsNbg&hl=es419&sa=X&ved=2ahUKEwjxkajrj\\_tAhXgVTABHXCx-CgAQ6AEwDHoECBYQAg#v=onepage&q=tiempo%20real&f=false](https://books.google.com.pr/books?id=97zTD-wAAQBAJ&pg=PR5&lpq=PR5&dq=tiempo%20real&source=bl&ots=3IyGC3UoC&sig=ACfU3U-1DZY_Vrjd8XlOSXj7DjF1FCEsNbg&hl=es419&sa=X&ved=2ahUKEwjxkajrj_tAhXgVTABHXCx-CgAQ6AEwDHoECBYQAg#v=onepage&q=tiempo%20real&f=false)
- Chen, W.-K. (2019). The VLSI Handbook. In Chiet. Recuperado de <https://www.sciencedirect.com/book/9780122341007/vlsi-handbook#book-description>
- Claudio, P. (2020). Arduino IDE: Domina la programación y controla la placa - Buenos aires. Recuperado de <https://books.google.com.ar/books?id=Xgv2DwAAQBAJ&pg=PT19&lpq=PT19&dq=instalar+openjfx+para+java+8+en+archlinux&source=bl&ots=vNCSJaTz11&sig=ACfU3U2QYpoJT-2bHUfiatiWkAwMbSM8Hvg&hl=es-419&sa=X&ved=2ahUKEwjcgYeWsa33AhUospUCHQJRDE-QQ6AF6BAgnEAM#v=onepage&q=instalar%20openjfx%20para%20java%208%20en%20archlinux&f=false>
- CURIEL MARIN, E. O. (2019). Observación sistemática y análisis de contexto para la innovación - Madrid.
- Denzin, N., & S. Lincoln, Y. [Ed. 4]. (2015). Métodos de recolección y análisis de datos: Manual de investigación - Barcelona: Gedisa S.A.
- Diez Suarez, A., González Martín, A., Gil, Á., De Simon, M., & Vega Barrallo, B. (2017). Selección de equipos y materiales en las instalaciones eléctricas de baja tensión en los entornos de edificios de viviendas - España.
- Galeano M., M. (2019). Diseño de proyectos en la investigación cualitativa - Medellín. Recuperado de <https://books.google.es/books?id=Xkb78OSRMI8C&printsec=frontcover&hl=es#v=onepage&q&f=false>
- García, A., & Escaño, J. (2018). Sistemas secuenciales programables - España: Pananinfo, S.A.
- García, L., & Manchón, R. (2017). Sistemas Informáticos en Tiempo Real: Teoría y Aplicaciones. España: Universidad Miguel Hernández.
- Giraldo, J. D. (2015). La investigación en los campos de la arquitectura - Colombia. Recuperado de <https://arquitectura.medellin.unal.edu.co/images/imagenes/pdf/lainvenloscampos.pdf>
- Goilav, N., & Loi, G. (2016). Arduino: Aprender a desarrollar para crear objetos inteligentes. Barcelona: ENI.
- Hartas, D. (2015). Educational Research and Inquiry: Qualitative and Quantitative Approaches - New York. Recuperado de <https://books.google.com.ec/books?id=2rJSBgAAQBAJ&dq=2+on+1+%2311>
- Jesús, T. (2016). Sistemas eléctricos en centrales. España: Pananinfo, SA.
- Kohei, A., Bhatia, R., & Kapoor, S. (2018). roceedings of the Future Technologies Conference (FTC) 2018: Volume 1. Springer.
- Moreno Vega, A. (2016). Instalaciones, maquinaria y equipos agrícolas – Madrid.
- Jesús, P. (2020). Internet de las cosas (IOT) con ESP. Manual práctico - Madrid.

- M.<sup>a</sup> José Romero, S., López López, J., & García, X. (2020). Estructura de computadores. Prácticas de laboratorio- Cuenca.
- Moaveni, S. (2015). Engineering Fundamentals: An Introduction to Engineering - Mexico.
- Mora, J. (2018). Montaje de los cuadros de control y dispositivos eléctricos y electrónicos. IC Editorial.
- Mukhopadhyay, S., Boby, G., Joyanta, K., & Tarikul, I. (2021). Interdigital Sensors: Progress Over the Last Two Decades - India.
- Muñoz, F. (2015). UF0861 - Montaje y verificación de componentes - España
- Nieto, E. (2015). Fuentes de alimentación conmutadas en la práctica. Fidestec.
- Peña, C. (2020). Programa tu Arduino: Sintaxis - Uso de Variables - Control de Flujo - R/S - Buenos Aires.
- Rodríguez, J. [Ed. 2]. (2018). Circuito cerrado de televisión y seguridad electrónica – España: Ediciones Paraninfo, S.A.
- Rodríguez Peña, V. (2018). UF2245 - Diagnóstico de averías en pequeños electrodomésticos y herramientas .
- Redusers. (2015). Electricidad - Fundamentos y seguridad: Herramientas para el desarrollo profesional - Buenos Aires.
- Sahdev. (2018). Electrical Machines - India. Recuperado de <https://www.biblio.com/book/electrical-machines-k-sahdev/d/1492864735>
- Sifuentes Ibarra, Ernesto, Samaniego Gaxiola, José Alfredo, Anaya Salgado, Antonio, Núñez Moreno, Jesús Humberto, Valdez Gascón, Benjamín, Gutiérrez Soto, Rosario Guadalupe, Ruelas Islas, Jesús del Rosario, & Macías Cervantes, Jaime. (2015). Programación del riego en nogal pecanero (Carya illinoensis), mediante un modelo integral basado en tiempo térmico. Revista mexicana de ciencias agrícolas, 6(8), 1893-1902. Recuperado en 06 de septiembre de 2022, de [http://www.scielo.org.mx/scielo.php?script=sci\\_arttext&pid=S2007-09342015000801893&lng=es&tlng=es](http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S2007-09342015000801893&lng=es&tlng=es).
- Vallejo Fernandez, D., Gonzales Morcillo, C., & Albusac Jimenez, J. (2016). Programacion concurrente y tiempo real.