

Diseño web con HTML5 para un sistema de monitoreo y control remoto por IoT

Web design using HTML5 for a remote monitoring and control system by IoT

Juan Pablo Caviedes Silva  Juan Martin Patuso Borda 

Johan Julian Molina Mosquera 

Universidad Surcolombiana, Colombia

OPEN  ACCESS

Recibido: 09/01/2023

Aceptado: 03/04/2023

Publicado: 10/05/2023

Correspondencia de autores:
julian.molina@usco.edu.co



Copyright 2020
by Investigación e
Innovación en Ingenierías

Resumen

Objetivo: Desarrollar una página web, cuyo propósito principal es la manipulación de forma manual o automática de un servomotor y, la medición en tiempo real de la temperatura y humedad proporcionada por el sensor DHT11. Esta aplicación se desarrolla en los entornos de HTML5, CSS y JavaScript y es aplicado a través del microcontrolador ESP8266, el cual permite la conexión a Wi-Fi para generar las comunicaciones necesarias entre el sistema a controlar, la página web y la base de datos creada en Firebase. **Metodología:** Para la elaboración de este prototipo, primero, se diseña el entorno de la página web y de la base de datos, segundo, se programan los códigos necesarios para implementar la página web, el sistema de control del servomotor y los datos que suministra el DHT11, tercero, se presenta la estructura del sistema embebido junto con el servomotor y el sensor, para finalizar se refleja el modelo cliente-servidor teniendo en cuenta cada una de las partes de este. **Resultados:** En los resultados se verifica las mediciones del sensor en tiempo real, y la respuesta del servomotor ante los datos de posición enviados por el usuario, además para facilitar al usuario la manipulación del prototipo, se desarrolla la aplicación para dispositivos Android. **Conclusiones:** Finalmente, se evidencia las ventajas del IoT y como facilita la vida de los usuarios con su infinidad de aplicaciones, la practicidad y el bajo costo de la placa ESP8266, los principales beneficios de la aplicación Android y la funcionalidad de la página web.

Palabras clave: Cascading Style Sheets (CSS), Diseño web, Hyper Text Markup Language version 5 (HTML5), Hosting, JavaScript (JS), Sistema embebido.

Abstract

Objective: Develop a web page, whose main purpose is the manual or automatic manipulation of a servomotor and the real-time measurement of temperature and humidity provided by the DHT11 sensor. This application is developed in the HTML5, CSS and JavaScript environments and is applied through the ESP8266 microcontroller, which allows Wi-Fi connectivity to generate the necessary communications between the system to be controlled, the web page and the database. created in Firebase. **Methodology:** For the development of this prototype, first, the environment of the web page and the database were designed, second, the necessary codes are programmed to implement the web page, the servomotor control system and the data it provides the DHT11, third, the structure of the embedded system is presented together with the servomotor and the sensor, finally the client-server model is reflected taking into consideration each of its parts. **Results:** The results verify the measurements of the sensor in real time, and the response of the servomotor to the position data sent by the user, in addition to making it easier for the user to manipulate the prototype, the application for Android devices is developed. **Conclusions:** Finally, the advantages of IoT are evidenced and how it makes life easier for users with its infinity of applications, the practicality and low cost of the ESP8266 board, the main benefits of the Android application and the functionality of the web page.

Keywords: Cascading Style Sheets (CSS), Design web, Hyper Text Markup Language version 5 (HTML5), Hosting, JavaScript (JS), embedded system.

Introducción

Actualmente, el mundo a lo largo de los años ha venido creando grandes potencias, desarrollo y crecimiento en la tecnología, así, es como se ha implementado un enfoque que permita la interacción de información o comunicación a través de dispositivos que consienten la detección de información de acuerdo con unos lineamientos o parámetros acordados [1].

Colombia ante esta dinámica a nivel global tiene como reto adoptar y desarrollar tecnologías en diferentes procesos industriales, entre otros; cuyos elementos proporcionen un progreso a la sostenibilidad y eficiencia de producción en el país para poder estar inmerso en avances y competitividad [2].

Dentro de este desarrollo, se encuentra el Internet de las cosas (IoT) por sus siglas en inglés) el cual ha permitido entre una de sus características, generar información posible para analizar y orientar toma de decisiones haciendo uso de la inteligencia artificial [3].

De esta manera, es menester reconocer la importancia de esta red tecnológica porque integra primordialmente, tecnologías que hagan uso de la electrónica [3]; una electrónica que permite a través de equipos diseñados con propósitos específicos, procesar información de sensores o dispositivos, obtener la capacidad de recolectar, transmitir comunicación en tiempo real, la adquisición de información y monitoreo entre dispositivos para llevar dicha información a servidores conectados a la red global [4].

En la investigación intitulada *Internet of Things (IoT) DISEÑO DE UNA RED DE IoT PARA EL HOGAR* expresa que “La evolución en Colombia de tecnologías de servicios de internet móvil e Internet Fijo genera la posibilidad de poder brindar servicios de IoT... IoT hace referencia a como se pueden interconectar a través de Internet múltiples dispositivos de uso cotidiano, como el televisor, cámaras de vigilancia, carros, lavadoras, neveras, sensores entre otros muchos objetos” [5].

Es por esto por lo que se presenta la tecnología IoT, ya que permite mediante dispositivos informáticos (página web) comunicar entre sí a través de una red cableada o inalámbrica, la recopilación y análisis de datos para generar resultado con o sin interrelación de persona a persona o de persona a computadora [6, 7].

Por tanto, se conoce que IoT es una red inteligente capaz de conectar todas las cosas a Internet por esto “la mayoría de las industrias manufactureras, los proveedores de servicios y las industrias de software están invirtiendo más y adoptar tecnologías IoT más rápidamente. Forbes estimó que la transformación de los conceptos hipotéticos de IoT a la realidad comenzó durante el año 2015 [6]. Según la encuesta, es probable que la cantidad de objetos conectados a Internet sea de 75,44 mil millones [7] y el crecimiento económico de la tecnología IoT oscilará entre \$ 2,7 y \$ 6,2 billones para 2025 [8] esto muestra el impacto de la tecnología IoT en la sociedad” [1].

De esta manera, se indica que Internet es más inmersivo porque permite ser una red de fácil acceso e interacción frente a variedades de equipos [8]. Por consiguiente, el presente artículo tiene como propósito desarrollar un prototipo de diseño de página web con aplicativo en CSS, JS y HTML5 que permite el monitoreo del sensor DHT11 para obtener variables climatológicas y el control del servomotor en respuesta a los datos obtenidos a través de la página web diseñada o desde la aplicación móvil. A su vez, demostrar mediante un sistema embebido la respuesta en tiempo real según la interacción entre la página web y el usuario.

Este artículo se divide en tres partes: en la primera parte, se muestra los materiales en donde se explican los software y hardware que se utilizaron para desarrollar el prototipo, asimismo, en la metodología se explica

todo el proceso que se emplea para alcanzar los objetivos que se proponen para el correcto desarrollo del prototipo. En la segunda parte, se expone los resultados alcanzados durante todo el proceso. Y, en la tercera parte, se concluye.

Metodología

Materiales

Software

Se emplearon lenguajes de programación como: JS, HTML, CSS, Firebase para la base de datos. GITHUB para enlazar la página web, Android Studio y Arduino para programar la placa.

JavaScript, HTML y CSS

Una aplicación web en el lado del cliente, se divide de tres elementos principalmente: en primer lugar, el código HTML que compone el bloque de construcción básico de la página web. En segundo lugar, están las hojas de estilo en cascada CSS usadas para el diseño, presentación y control de la disposición de cada elemento de la página web. Para finalizar, el código JavaScript, que está embebido en la página web o se puede importar como archivos por separado [9, 10].

Firebase

Facilita herramientas para la creación de informes, corrección de fallos en aplicaciones, seguimiento de analíticas, etc. Además, es un software de desarrollo de aplicaciones patrocinado por Google, que permite desarrollar en IOS, Android y aplicaciones web. De igual forma, Firebase ofrece varios servicios en los que, para el presente artículo se destaca:

Autenticación: ofrece una solución de identidad todo en una, soportando cuentas de correo electrónico y contraseña, autenticación telefónica, cuentas de Google, Facebook, GitHub, entre otras. De igual modo, permite a los desarrolladores mejorar la experiencia de inicio de sesión y “embarque de los usuarios”.

Base de datos en tiempo real: los datos son almacenados y sincronizados entre los usuarios y clientes en tiempo real, también estará disponible cuando la aplicación este sin conexión a internet [11].

ithub

A través de notificaciones y una interfaz permite la visibilidad de la actividad de un proyecto y proporciona un entorno de desarrollo abierto, lo que favorece a un mayor conocimiento con una menor comunicación entre colaboradores [12, 13].

Además, permite el alojamiento de páginas web de forma sencilla y sin ningún costo.

Android Studio

Es un entorno de desarrollo integrado (IDE) multiplataforma, y es el principal IDE para el desarrollo de aplicaciones Android basado en IntelliJ IDEA de JetBrains y es publicado de manera gratuita a través de la licencia Apache 2.0 [14].

Tiene consola de desarrollador, herramientas para detectar problemas de compatibilidad, rendimiento, etc.; plantillas para crear diseños comunes, renderiza en tiempo real, su construcción se basa en Gradle, entre otras cosas que facilitan la programación [15].

Arduino

Es una plataforma de código abierto [16], utilizada para programar y construir dispositivos electrónicos. Asimismo, puede enviar y recibir información a la mayor parte de dispositivos, también es posible hacerlo a través de la Internet [17]. Para el hardware utiliza una placa Arduino y en la parte de software usa un código C++ simplificado para programar la placa [18].

Hardware

ESP8266

Es un sistema embebido, que es uno de los chips con Wi-Fi integrado más usados, de dimensiones muy reducidas que varían dependiendo del modelo que se escoja, con un coste muy económico [9]. Integra el procesador Tensilica L106 de 32 bits, que presenta un consumo de energía muy bajo y un conjunto de instrucciones reducido de 16 bits (RSIC). Lo que le permite alcanzar una velocidad máxima de reloj de 160MHz. [19]. Además, posee una API avanzada para el control de entradas y salidas, lo que puede reducir drásticamente el trabajo para configurar y manipular hardware. El código de programación es similar a Arduino, pero de forma interactiva en Lua Script. Es posible mediante un plug in programarlo en IDE de Arduino [20].

DHT11

El sensor de humedad y temperatura DHT11 es un periférico asequible. Puede medir la humedad relativa entre el 20 y el 90% RH, con un rango de temperatura de funcionamiento entre 0 a 50°C, con una precisión de $\pm 5\%$ RH. Además, la temperatura se mide en el rango de 0 a 50°C, con una precisión de $\pm 2^\circ\text{C}$. Ambos valores tienen una resolución de 8 bits [21].

Se compone de dos partes, un termistor y un sensor de humedad capacitivo. De igual forma, tiene un circuito integrado en el interior que convierte la señal de analógico a digital y también es el encargado de enviar la señal de temperatura y humedad en digital [22].

Funciona con comunicación en serie. El sensor envía datos en forma de tren de pulsos de un período de tiempo especificado. Necesita un comando de inicialización con retraso de tiempo para poder enviar los datos a la placa. Para finalizar, la interfaz en serie de un solo cable hace que la respuesta del sistema sea rápida y fácil [23].

Servomotor SG90

Es un servo económico, pero de gran calidad y de dimensiones pequeñas, ideal para proyectos sencillos y prácticas con servos, ya que tiene requerimientos de energía muy bajos que, permite conectarlo a la misma fuente de alimentación que el circuito de control, de igual modo, funciona con las de tarjetas de control con microcontroladores más utilizadas, así como con la mayoría de los sistemas RC comerciales. Tiene un torque de 1.8Kg-cm a 4.8V, puede rotar 180° a una razón de 0.1s por cada 60°, con un voltaje de funcionamiento entre 3-7.2V [24].

Desarrollo

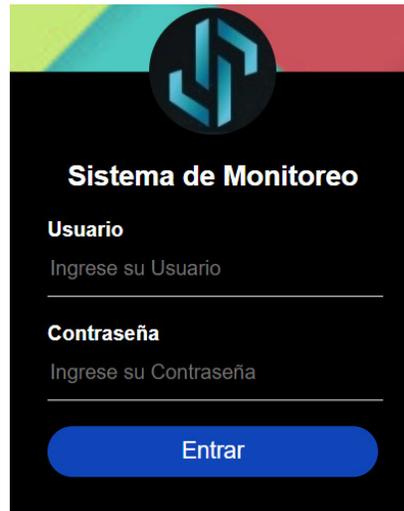
Para el desarrollo y finalidad de este artículo se ha realizado un estudio exhaustivo a cerca de la base datos en Firebase y los lenguajes de programación posibles (HTML5, CSS, JS, ARDUINO). Además, investigaciones sobre los diversos sistemas de seguimiento existentes en el mercado, se analiza sobre el funcionamiento del microcontrolador ESP8266 y sus aplicaciones.

El prototipo de control que se implementa se divide en cuatro etapas principales:

Diseño de interfaces

La primera etapa consiste en el diseño del entorno de la página web, el cual el usuario cliente va a identificar en un primer momento. A continuación, en la figura 1 se visualiza la interfaz de ingreso.

Figura 1. Interfaz de ingreso

La interfaz de ingreso tiene un fondo negro. En la parte superior hay un encabezado con un logo circular que contiene un símbolo de código de barras o similar, sobre un fondo de colores (verde, azul, rojo). Debajo del logo, el título "Sistema de Monitoreo" está en blanco. Hay dos campos de entrada con etiquetas "Usuario" y "Contraseña" en blanco, y el texto "Ingrese su Usuario" y "Ingrese su Contraseña" en un gris muy claro. Un botón azul con el texto "Entrar" en blanco está centrado al final.

Fuente: Elaboración propia

Por lo anterior, el cliente usuario ingresa de forma correcta y se despliega la interfaz de monitoreo y control como se muestra en la figura 2.

Figura 2. Interfaz de monitoreo y control

La interfaz tiene un fondo blanco. El título "Sistema de Monitoreo y Control" está en azul. Debajo, el título "Datos Actuales" está en gris. Hay dos líneas de texto: "Temperatura: °C" y "Humedad: %", ambas en gris. Un botón gris con el texto "Actualizar Datos" en blanco está centrado. Debajo, el texto "Grados Servo" está en gris, sobre un campo de entrada gris. Un botón gris con el texto "Mover" en blanco está centrado al final.

Fuente: Elaboración propia

También se encuentra la aplicación de la base de datos para la comunicación y almacenamiento de datos en la nube aplicando Firebase Realtime Database de la página web con el sistema embebido. Esta base de datos se encuentra alojada en la nube, Los datos se almacenan en formato JSON y se sincronizan en tiempo real con cada cliente conectado.

Software de control

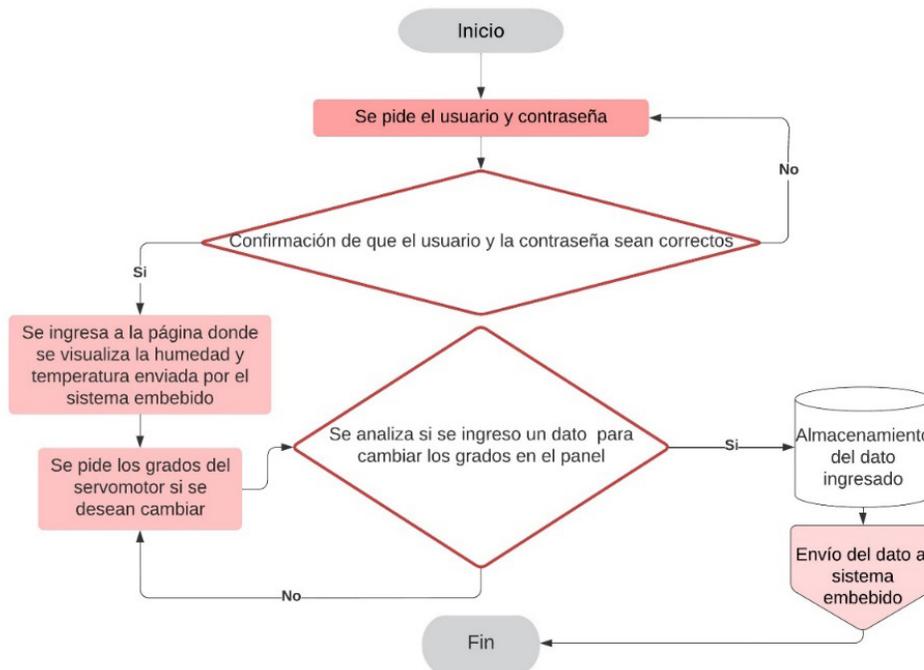
Para el correcto monitoreo constante de los parámetros de humedad y temperatura que se están presentando en el entorno, de igual forma, para determinar la posición deseada en el servomotor y garantizar un eficiente control de este, se implementa el diseño y funcionamiento de la página web, basados en el algoritmo de la figura 3.

Para el sistema de control se implementa el microcontrolador ESP8266, programado en el lenguaje de Arduino.

Este algoritmo de control principalmente manipula las librerías del micro ESP8266 (ESP8266WiFi.h), la de la base de datos Firebase (FirebaseESP8266.h), la del sensor DHT11 (DHT.h) y la de los servomotores (Servo.h) con el fin de resumir el código fuente y la complejidad de este. Como resumen del algoritmo del código, seguido de la definición de las librerías, se establecen las líneas de código para la comunicación de los datos para la conexión a la red wifi (nombre y contraseña), después se codifican los datos para la conexión de la Firebase (Nombre y URL).

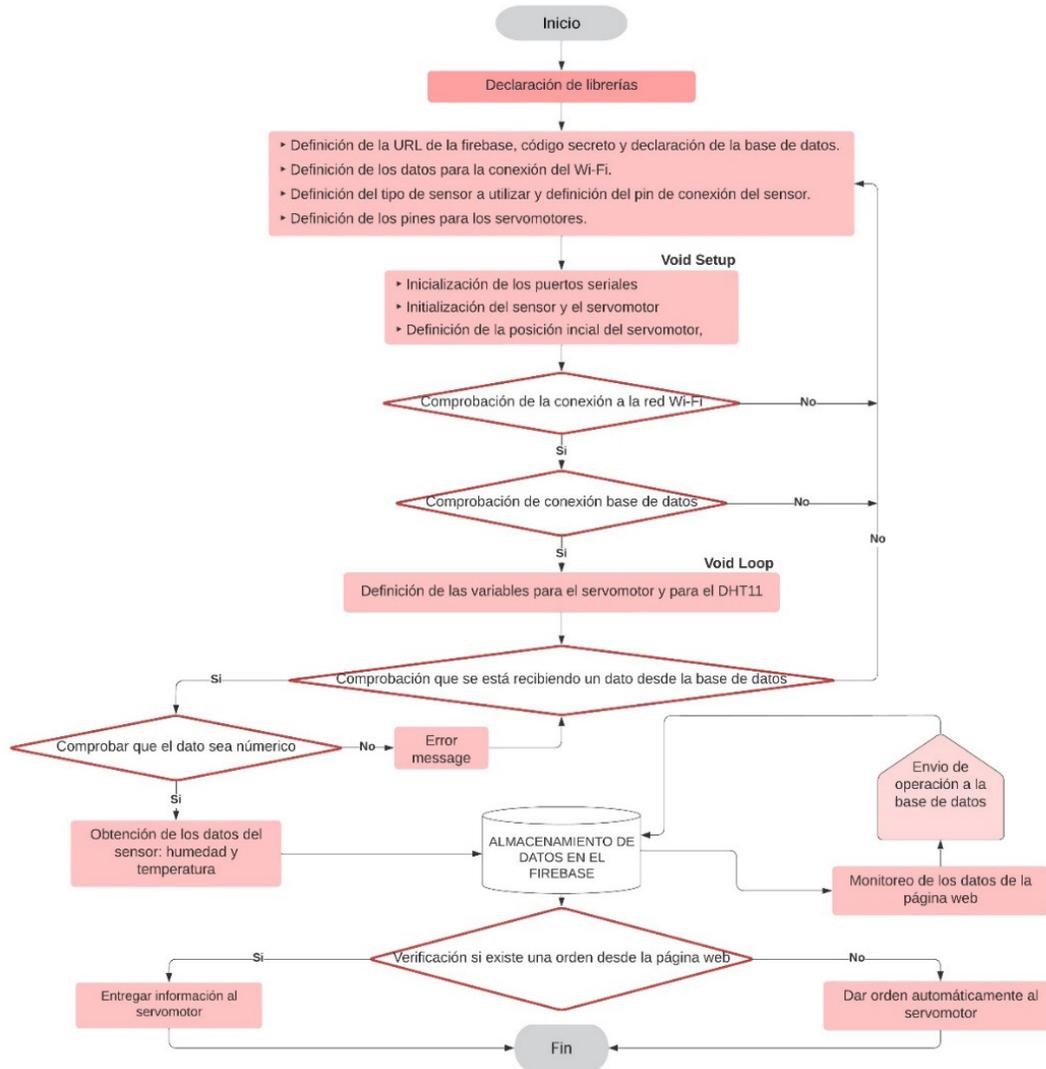
Y por último se define el tipo de sensor a utilizar (DHT11) y el pin de lectura de este (D1(GPIO5)) y se inicializa los servos. La descripción del código para lectura, envío y recepción de datos, se puede analizar el diagrama de flujo expuesto en la figura 4.

Figura 3. Diagrama de flujo de la página web



Fuente: Elaboración propia

Figura 4. Diagrama de flujo del algoritmo para el software de control

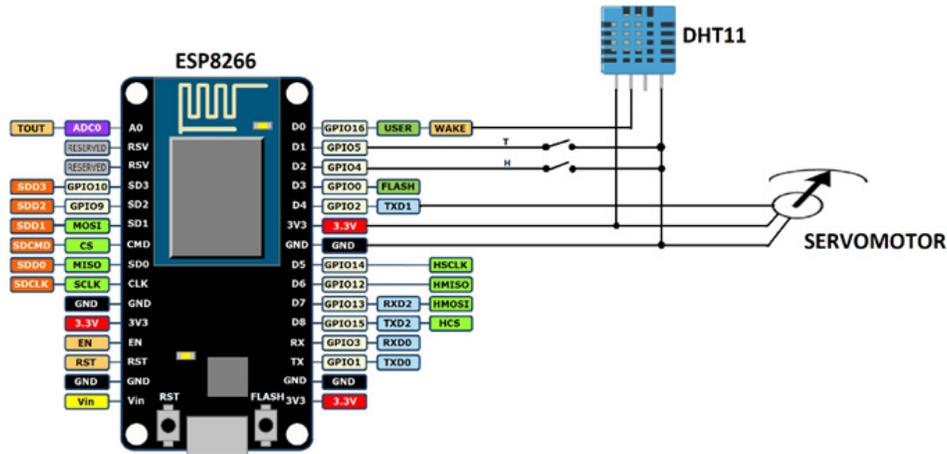


Fuente: Elaboración propia

Sistema embebido

El sistema embebido está conformado por el microcontrolador ESP 8266 encargado de recibir los datos de temperatura y humedad adquiridos mediante el sensor DHT 11 los cuales son enviados vía WI-FI a la base de datos en FireBase. Por lo tanto, el microcontrolador se encarga de recibir los datos de posición, suministrados por la interfaz del usuario desde la página web; de esta manera, ejecuta la posición del servomotor según la ubicación correspondiente; no obstante, al no recibir datos desde la interfaz del sistema de control remoto, el microcontrolador se encarga de ejecutar las instrucciones de forma autónoma para elegir una posición óptima basado en los parámetros e instrucciones en las líneas del código encargadas del control.

Figura 5. Estructura del sistema embebido



Fuente: Elaboración propia

Modelo cliente-servidor

Cliente

Ya sea en un dispositivo Android o en un PC, el cliente ingresa a la plataforma a través de la interfaz de ingreso que se muestra en la figura 1. Para este caso, la interfaz se programó para que se pudiera elegir cualquier usuario, pero para la contraseña se ingresa con el código "2405". Una vez hecho esto, se redirige a la interfaz de control mostrada en la figura 2 en la que se muestra la temperatura, la humedad y se puede controlar los grados del servomotor.

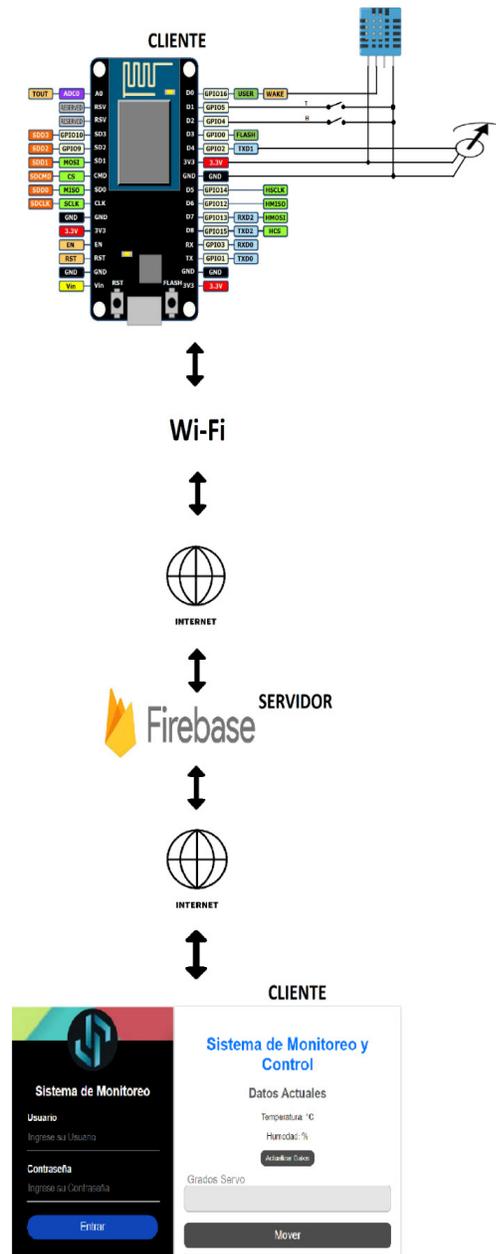
La placa ESP8266 tiene un chip Wi-Fi, que envía los datos hacia el servidor FireBase a través de Internet.

Servidor

En esta parte, el servidor recibe los datos que la placa ESP8266 envíe a través del chip Wi-Fi, para que el cliente pueda visualizarlo en su dispositivo Android o en la página web. De igual modo, en la base de datos se muestra los datos de temperatura, humedad y los grados del servomotor. Además, dichos datos se almacenan para que posteriormente el cliente pueda visualizarlos.

Se presenta el modelo cliente servidor final para el sistema de monitoreo y control en la figura 6.

Figura 6. Modelo cliente-servidor

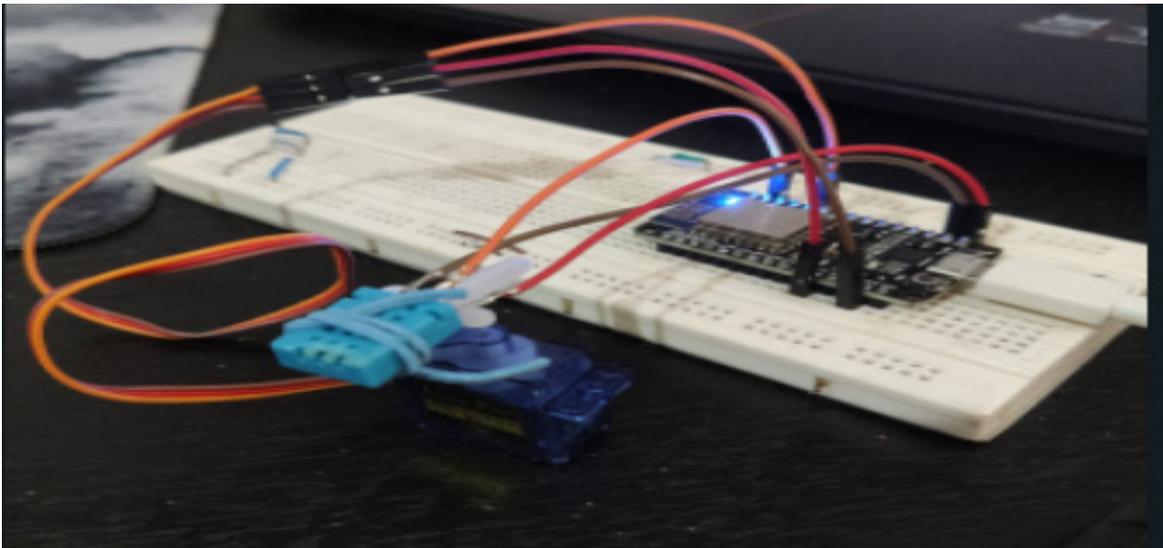


Fuente: Elaboración propia

Resultados

Se realiza el montaje del sistema embebido para ejecutar las pruebas de comunicación con la página web y el almacenamiento en la base de datos, seguidamente se realiza la verificación en la operación de los sensores y el correcto funcionamiento del servomotor para responder con precisión a los datos suministrados desde la página web, así mismo, se verifica que la comunicación establecida entre el sistema embebido que se muestra en la figura 7 y la página web diseñada para el monitoreo y control, no presente pérdidas a la hora de almacenar la información en la base de datos.

Figura 7. Circuito para las pruebas del sistema



Fuente: Elaboración propia

El aplicativo se desarrolla inicialmente como página web, además, para complementar, se despliega para visualizar y modificar desde cualquier dispositivo Android con la versión 5.0 o superior. Del mismo modo, el aplicativo se ajusta a la resolución y dimensiones de cualquier pantalla de los dispositivos móviles. Se hace de esta manera, ya que los dispositivos Android, además de ser muy comunes, también son muy portátiles y, es más rápido y cómodo usar el prototipo como aplicación Android que como aplicativo web. La visualización del aplicativo en el dispositivo Android se evidencia en la figura 8.

Por otra parte, se puede visualizar la respuesta en tiempo real entre el FireBase, el aplicativo web y el prototipo del sistema embebido al modificar los grados del servomotor partiendo de las lecturas obtenidas por el sensor DHT11. Igualmente, se almacenan en la base de datos de a FireBase todos los datos modificados de los ángulos aplicados al servomotor y los datos visualizados entregados por el sensor DHT11, el almacenamiento de los datos será empleado para posteriores análisis que garanticen un correcto monitoreo y control a partir de las variables obtenidas y manipuladas.

Figura 8.a. Ingreso en aplicación Android



Figura 8.b. Control en aplicación Android



Fuente: Elaboración propia

Conclusiones

El IoT ayuda a conectar de forma más fácil al usuario con todo lo que lo rodea, solo es necesario tener una conexión a internet para realizar infinidad de aplicaciones que, facilitan y simplifican la vida de las personas. Como es el caso de este sistema de control que monitorea la temperatura del entorno y controla las instrucciones del servomotor.

Al utilizar y aplicar la placa ESP8266 la cual presenta un bajo costo y de fácil adquisición en el mercado, se puede desarrollar como microcontrolador para la obtención de datos y manejo de instrucciones para diferentes actuadores, con el fin de crear sistemas que monitoreen y controlen de forma remota dispositivos a un bajo costo y alta eficiencia [25].

La aplicación Android, permite visualizar y modificar los valores medidos y controlados de forma sencilla, debido a su portabilidad y el uso general de los dispositivos con sistema operativo Android.

La creación e implementación de la página web funciona al realizar monitoreo de un sistema que requiere ser controlado. Este sistema desarrollado permite ser mayormente accesible al usuario-cliente ya que se puede ingresar desde cualquier parte. Además, la función y desarrollo como aplicativo en aparatos

tecnológicos como el celular móvil con conexión a Internet que también expresa fácil acceso desde el usuario cliente que, en su mayoría, cuentan con este servicio.

Referencias bibliográficas

1. K. S. Narasimha Swamy, «An Empirical Study on System Level Aspects of Internet of Things (IoT),» IEEE ACCESS, vol. XX, p. 55, 2020.
2. H. G. Hernandez Palma, J. Solórzano Movilla, y J. Jinete Torres, “La Teoría de restricciones para los procesos de gestión y control en las IPS del Caribe Colombiano”, Investigación e Innovación en Ingenierías, vol. 8, n.º 1, pp. 54–68, 2020. DOI: <https://doi.org/10.17081/invinno.8.1.3624>
3. C. M. M. N. Mojica, «El Internet de las cosas (IoT) impactando el sector de la logística,» Bogota, 2019.
4. V. B. C. Z. B. Zanella, «Internet of Things for Smart Cities,» IEEE INTERNET OF THINGS JOURNAL, vol. 1, n.º 1, pp. 22-32, 2014.
5. C. B. O. Enrique, Internet of Things (IoT) Diseño de una red IoT para el hogar, Bogota D.C, 2020.
6. H. D. B. a. H. Shanzhi, «A Vision of IoT: Applications, Challenges, and Opportunities With China Perspective,» IEEE INTERNET OF THINGS JOURNAL, vol. 1, n.º 4, pp. 349 -359, 2014.
7. M. R. K. Soumyalatha Naveen, «Key Technologies and challenges in IoT Edge Computing,» 2019.
8. E. A. Fallón Joshua, IoT-based Home Automation Using Android, 2020.
9. Yue, Chuan & Wang, Haining, “Characterizing insecure JavaScript practices on the Web,” WWW- Proceedings of the 18th International World Wide Web Conference. 2009, pp. 961-970, 2009
10. Ocariza, Frolin & Pattabiraman, Karthik & Zorn, Benjamin, “JavaScript Errors in the Wild: An Empirical Study,” International Symposium on Software Reliability Engineering, ISSRE. 2011, pp. 100-109.
11. Gaikwad, Anil, “FIREBASE-OVERVIEW AND USAGE,” Journal of Engineering and Technology Management. 2022, pp 2582-5208.
12. L. Dabbish, C. Stuart, J. Tsay, and J. Herbsleb. Social coding in github: transparency and collaboration in an open software repository. In Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work, CSCW '12. 2012, pp 1277–1286.
13. Kalliamvakou, Eirini & Damian, Daniela & Blincoe, Kelly & Singer, Leif & German, Daniel. “Open Source-Style Collaborative Development Practices in Commercial Projects Using GitHub.” 2015.
14. Allen, Grant. Android for absolute beginners. London, UK. APress. 2021, pp. 16.
15. Velasco Raúl y Antón Miriam. “Aplicación Android para el entrenamiento cognitivo de personas bajo tutela jurídica con discapacidad intelectual o del desarrollo,” Universidad de Valladolid. Valladolid, ESP. 2017.
16. M. Banzi, Getting Started with arduino. “ O’Reilly Media, Inc.”, 2009.
17. Abdullahi Badamasi, Yusuf. The working principle of an Arduino. 2014, pp 1-4.
18. P. D. Minns, C Programming For the PC the MAC and the Arduino Microcontroller System. Author House, 2013.
19. J. Candelario, “Implementación de WPS en el firmware NodeMCU para el ESP8266”, Universidad de Sevilla, Sevilla, 2016.
20. Espressif Inc., “Overview ESP8266” [En línea]. Disponible en: <https://espressif.com/en/products/hardware/esp8266ex/overview>.

21. Gay, Warren. *Matering the Rasperry Pi*. London, UK. APRESS. pp. 263.
22. Datasheet.es, "DHT11 Data sheet" [En línea]. Disponible en <http://www.datasheet.es/PDF/792210/DHT11-pdf.html>
23. D. Srivastava, A. Kesarwani, S. Duvey, "Measurement of Temperature and Humidity by using Arduino Tool and DHT11", *International Research Journal of Engineering and Technology*. 2018.
24. SanDoRobotics, "Micro servomotor tower pro SG90", [En línea]. Disponible en: <https://sandorobotics.com/producto/sg90>
25. E. J. De la Hoz Domínguez, T. J. . Fontalvo Herrera, y A. A. Mendoza Mendoza, "Aprendizaje automático y PYMES: Oportunidades para el mejoramiento del proceso de toma de decisiones", *Investigación e Innovación en Ingenierías*, vol. 8, n.º 1, pp. 21–36, 2020. DOI: <https://doi.org/10.17081/invinno.8.1.3506>