

Firmware for graphical block programming of NodeMCU boards with Android devices

Firmware para programar gráficamente por bloques placas NodeMCU con dispositivos Android

Jhonny Dávila Ramírez¹ Darío Fernando Yépez Ponce² Héctor Mauricio Yépez Ponce³

¹Instituto Superior Tecnológico Luis Tello, Esmeraldas Ecuador
E-mail: jhonyjdvla@gmail.com

²Instituto Superior Tecnológico Luis Tello, Esmeraldas, Ecuador
E-mail: dfyp1991@gmail.com

³CEO Arduelectronics, Otavalo, Ecuador
E-mail: mauroyeppez32@gmail.com

RESUMEN

Actualmente, la programación es una de las asignaturas que debe estar presente en todo nivel educativo para innovar los procesos de enseñanza – aprendizaje. La programación de placas microcontroladas es una temática que se estudia generalmente en educación superior, para mitigar este inconveniente se han desarrollado varios softwares que permiten programar ciertas placas microcontroladas gráficamente. Teniendo esto en mente se desarrolló un Firmware en AppInventor2 y el IDE Arduino que permite programar las entradas/salidas digitales y la entrada analógica de las placas NodeMCU ESP8266 por medio de dispositivos inteligentes con sistema operativo Android. Para el desarrollo del Firmware se utilizó el modelo de desarrollo evolutivo (espiral) basado en la metodología STEAM, este modelo permitió tener un Firmware robusto debido a los ciclos repetitivos. El Firmware le permite al usuario programar la placa NodeMCU ESP8266 sin la necesidad de que tenga conocimientos de programación, debido a que utiliza la programación gráfica de bloques. Además, la

placa es programada en tiempo real y de forma inalámbricamente desde el dispositivo inteligente, contribuyendo así al movimiento de los proyectos DIY (hágalo usted mismo) para que pueda ser incorporado en los procesos educativos desde tempranas edades. Conjuntamente con el Firmware se implementó una placa de desarrollo para que los usuarios no necesiten tener conocimientos de electrónica para el armado, sin embargo; la placa de desarrollo cuenta con pines de conexión para que el usuario pueda conectar elementos electrónicos adicionales. El Firmware le permite al usuario ver las líneas de código que se van generando en función de los bloques que va usando para que vaya aprendiendo el lenguaje de programación escrito, para que en el futuro desarrolle proyectos más sofisticados. Finalmente, el Firmware y la placa de desarrollo fueron valorados por tres expertos en el área de docencia y programación.

Palabras clave-- APP para NodeMCU; Firmware para NodeMCU; Programación interactiva; Robótica educativa; Metodología STEAM.

ABSTRACT

Currently, programming is one of the subjects that must be present at all educational levels to innovate the teaching-learning processes. The programming of microcontroller boards is a subject that is generally studied in higher education, to mitigate this drawback have been developed several software that allow programming certain microcontroller boards graphically. With this in mind a Firmware was developed in AppInventor2 and the Arduino IDE that allows programming the digital inputs/outputs and analog input of the NodeMCU ESP8266 boards by means of smart devices with Android operating system. For the development of the Firmware, we used the evolutionary development model (spiral) based on the STEAM methodology, this model allowed to have a robust Firmware due to the repetitive cycles. The Firmware allows the user to program the ESP8266 NodeMCU board without the need for programming knowledge, since it uses graphical block programming. In addition, the board is programmed in real time and wirelessly from the smart device, thus contributing to the DIY (do-it-yourself) project movement so that it can be incorporated into educational processes from an early age. In conjunction with the Firmware, a development board was implemented so that users do not need to have knowledge of electronics for assembly, however; the development board has connection pins so that the user can connect additional electronics. The Firmware allows the user to see the lines of code that are generated based on the blocks that are used to learn the written programming language, so that in the future the user can develop more sophisticated projects. Finally, the Firmware and the development board were evaluated by three experts in the area of teaching

and programming.

Index terms— APP for NodeMCU; Firmware for NodeMCU; Interactive programming; Educational robotics; STEAM Methodology.

1. INTRODUCCIÓN.

Para Granados (2016), hoy en día la sociedad emplea gran parte del tiempo en entornos digitales, donde se facilita el acceso al intercambio de información, productividad y entretenimiento. Las áreas de programación se vuelven cada vez más indispensables para las empresas, solo en los Estados Unidos, se calcula que más de un millón de empleos de programación quedarán vacantes para el año 2020 (El diario de la república, 2018). Pujades (2017), menciona que en Latinoamérica se dio un crecimiento notable en el uso de tecnología, sin embargo; carece de un desarrollo tecnológico propio.

En España, en la Universidad de Valladolid se desarrolló un proyecto que crea una red WiFi estándar que los usuarios sin conocimientos avanzados de programación pueden adaptar a sus propios proyectos de domótica de una forma rápida y sencilla desde una aplicación para dispositivos móviles (Castrillejo, 2020).

En Rusia, (Chursin & Semenov, 2020) desarrollaron un juego de aprendizaje con la placa ESP8266. El juego desarrollado enseña y desarrollar las habilidades en las áreas de matemáticas, física, programación y los fundamentos de la robótica de los estudiantes. En este trabajo realizaron además la comparativa entre la programación por USB y por WiFi, la cual arrojó como resultado que la forma inalámbrica tardaba aproximadamente dos segundos más que la cableada.

En Taiwan, (Hsieh, 2021) desarrolló un robot barato, interactivo y programable para ayudar a los estudiantes a aprender Ciencia, Tecnología, Ingeniería, Arte y Matemáticas (STEAM, por sus siglas en inglés), de forma efectiva y amigable. El robot fue desarrollado en el código abierto Ottobot, la estructura externa fue impresora en

3D para que su forma puede ser personalizada por los usuarios. Para el control empleó una placa Arduino Nano integrada con un módulo bluetooth HC-06 y una placa ESP8266 para que el robot pueda ser controlado de forma remota a través de Bluetooth o WiFi.

En Ecuador, en la Universidad Técnica de Manabí, realizaron un proyecto para la orientación de personas con discapacidad visual mediante una APP que usa los sensores propios (GPS y acelerómetro) del dispositivo móvil (Zambrano et al., 2019).

En Esmeraldas, en el Instituto Superior Tecnológico Eloy Alfaro implementaron un robot móvil basado en la metodología STEAM. El robot utiliza una placa NodeMCU y por medio de una APP amigable e intuitiva le permite al usuario aprender los fundamentos de matemática y orientación espacial siguiendo el eslogan “aprender jugando” (Bastidas, Yépez & Mosquera, 2020).

En la revisión literaria, la mayor parte de aplicaciones en las que se utiliza la placa NodeMCU son para aplicaciones orientadas al IoT, domótica, inmótica, puntos de acceso en redes, otras (Ingeniería e Investigación, 2011).

En los últimos años, se ha popularizado la creación de dispositivos que transformen la enseñanza tradicional, con estos antecedentes nació la idea de desarrollar un Firmware y una APP con comunicación inalámbrica para programar placas NodeMCU ESP8266 de manera sencilla contribuyendo a las filosofías DIY y STEAM.

La creación del Firmware para programar placas NodeMCU con dispositivos Android que se presenta en este artículo consta de una placa de desarrollo y un Firmware que permite programar la las entradas/salidas digitales y la entrada

analógica de la placa. La APP desarrollada en APPInventor2 se vincula con el Firmware desarrollado en el IDE Arduino para que el usuario por medio de una programación gráfica por bloques pueda programar la placa de desarrollo de forma fácil e intuitiva. La placa de desarrollo está constituida de 12 led's, un sensor electromagnético, el módulo relé y un buzzer, con la cual se puede enseñar robótica educativa. Además, en la placa de desarrollo se han dejado unos pines para que el usuario pueda añadir tanto elementos electrónicos como sensores y con ello realizar prácticas y/o proyectos de mayor complejidad. Para el desarrollo la APP, se eligió el sistema operativo Android debido a que es el sistema operativo más utilizado en el mundo, en su lista de fabricantes se encuentran Sony, Samsung, HTC, Motorola, ZTE, Huawei, LG, entre otros (Torres, 2020). Otra de las ventajas que brinda el sistema operativo Android es la libertad que ofrece para el desarrollo de aplicaciones y cuenta con una tienda de APP's disponibles para que cualquier usuario pueda instalar y/o subir sus APP's creadas (Android, s.f.).

El presente artículo se encuentra distribuido de la siguiente forma, en la Sección 2 se dan a conocer las metodología y materiales empelados en el desarrollo del Firmware. Los resultados obtenidos, la funcionalidad y validación del proyecto son presentados en la Sección 4, la discusión es realizada en la Sección 5 y finalmente, en la Sección 6 se plasman las conclusiones de la investigación y el desarrollo realizado.

2. MATERIALES Y METODOLOGÍA.

2.1. Materiales.

2.1.1. Placa NodeMCU ESP8266

En la Tabla 1, se especifican las características más relevantes de las placas NodeMCU ESP8266

LoLin V3. En la placa NodeMCU se encuentra cargado el Firmware desarrollado por medio del cual se pueden comunicar inalámbricamente la placa de desarrollo y la APP. La placa NodeMCU es una placa de desarrollo "open source" de bajo costo que puede ser programada en diferentes lenguajes de programación.

Tabla 1

Características de la placa NodeMCU LoLin V3.

NodeMCU LoLin V3	
Descripción	Valor
Módulo	ESP8266
Convertor serial	CH340G
Módulo Wi-Fi	2.4 GHz
Memoria FLASH	4MB
Pines GPIO	17
Pines Analógicos	1
VIN	Máximo 5V
Regulador integrado	3.3V

Fuente: (Díaz, 2016).

2.1.2. Módulo MP1584

Para aumentar la robustez de la placa de desarrollo, se empleó el módulo "step down" MP1584 (Ver Figura 1) cuyas principales características de este convertidor DC-DC se presenta en la Tabla 2. Se recomienda utilizar este módulo debido a la sensibilidad de la placa al voltaje de entrada.

Tabla 2

Características de Módulo MP1584.

Step-down MP1584	
Descripción	Valor
Voltaje de entrada	4.5V a 28V DC
Voltaje de salida	0.8V a 18V DC
Corriente de Salida	Máximo 3A
Eficiencia de conversión	92%
Frecuencia de conmutación:	100kHz a 1.5MHz

Fuente: <https://protosupplies.com>.



Figura 1: Módulo MP1584.

Fuente: <https://protosupplies.com>

El resto de componentes electrónicos (led's, resistencias, potenciómetros, buzzer, otros) no se los describe debido a que son de uso común en la electrónica.

2.2. Metodología.

2.2.1. Diagrama de flujo del sistema

La lógica con la que fue programada el Firmware y la APP se presenta en la Figura 2. El mismo diagrama sirvió para realizar las pruebas de funcionamiento, detección y corrección de errores.

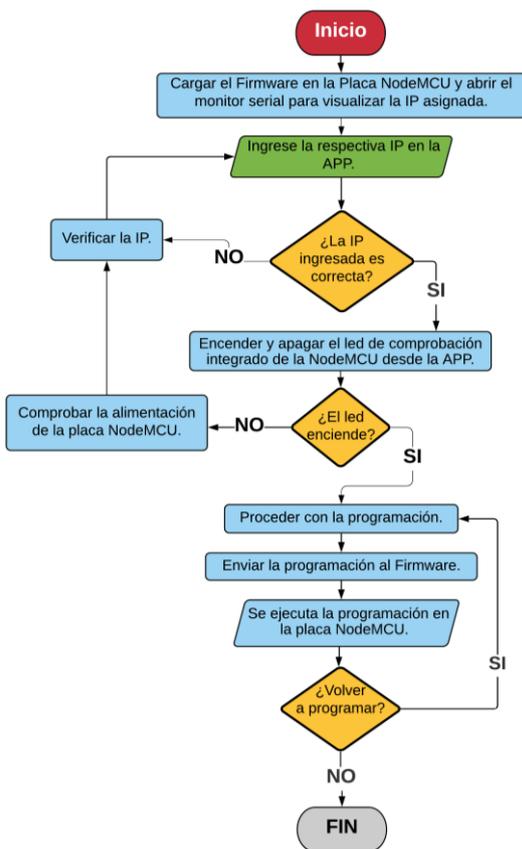


Figura 1: Diagrama de flujo del sistema.
Fuente: Propia, 2021.

2.2.2. Estructura general del sistema

En la Figura 3, se muestra como está estructurado el sistema para el funcionamiento del Firmware para programar placas NodeMCU ESP8266 desde dispositivos Android. El Firmware fue

desarrollado en el IDE Arduino y grabado en la placa desarrollo NodeMCU, la misma crea una red WiFi para comunicarse inalámbricamente con la APP. La APP fue creada en APPInventor2, desde esta aplicación se envían los datos al Firmware, el cual decodifica las instrucciones y las ejecuta las mismas en la placa de desarrollo.

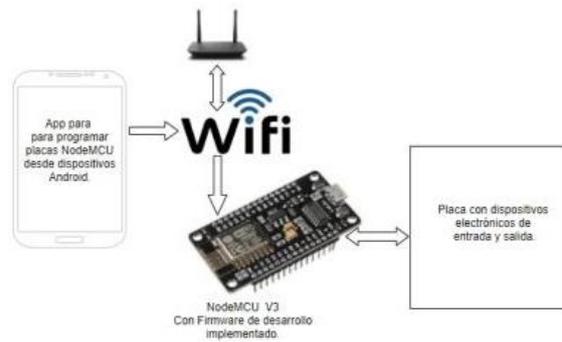


Figura 3: Estructura general del sistema
Fuente: Propia, 2021.

2.2.3. Interconexión de los componentes electrónicos del sistema

La interconexión de los distintos componentes electrónicos que conforman la placa de desarrollo se muestra en la Figura 4. Para las salidas digitales se ocuparon los pines desde el D0 hasta el D7, el pin D8 es usado como entrada digital y el A0 como entrada analógica. Debido a que las placas NodeMCU no toleran tensiones superiores a 5V, al sistema se le agregó un módulo "step-down" para tener una tensión constante de 5V.

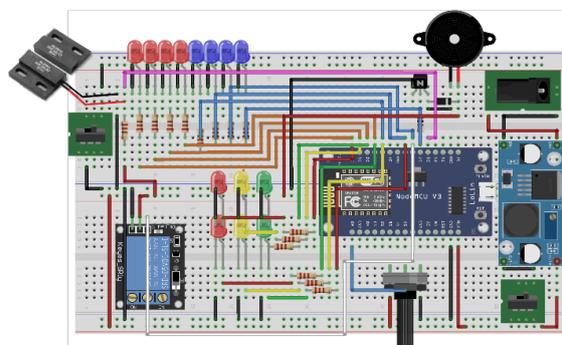


Figura 4: Interconexión de los componentes electrónicos.
Fuente: Propia, 2021.

2.2.4. Modelo de desarrollo evolutivo (espiral)

El modelo fue propuesto por Barry Boehm en 2010. Este modelo conjuga una naturaleza iterativa en la construcción de prototipos con aspectos controlados y sistemáticos del modelo en cascada. Al emplear el modelo en espiral, el software va ganando robustez en cada ciclo repetitivo. (Zumba & León, 2018).

La Figura 5, muestra las actividades del marco de trabajo las mismas que representan un segmento de la ruta de la espiral.

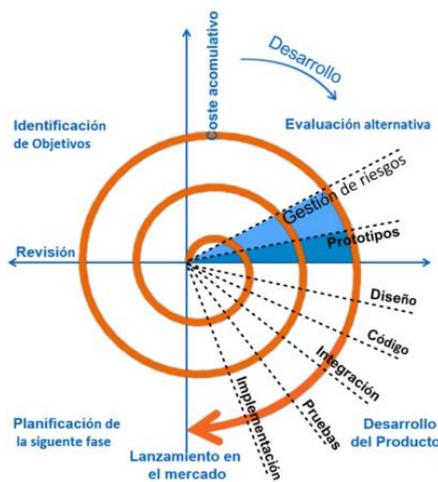


Figura 5: Modelo en espiral.

Fuente: <https://www.tutorialspoint.com>

2.2.5. Prueba de jueces

Se utilizó esta técnica para validar el desarrollo del Firmware, APP y placa de desarrollo.

Para Ramírez (2010), la mejor forma de evaluar un proyecto experimental es aplicando la prueba de jueces. Para tal fin, el instrumento debe tener un formato estructurado que recolecte información relevante del proyecto, los ítems del instrumento no deben surgir por inspiración del investigador sino mediante un proceso previo de la operacionalización de las variables.

Los jueces o expertos evaluaron y/o validaron el proyecto dando una ponderación numérica

según la siguiente escala: 1 = Deficiente, 2 = Regular, 3 = Bueno, 4 = Muy Bueno y 5=Excelente.

La técnica empleada es un método cuantitativo el mismo que valida una de las dos hipótesis de acuerdo al coeficiente de proporción de rango (CPR) obtenido.

- a) Si el $CPR > 0.75$ se acepta como válido el proyecto experimental.
- b) Si el $CPR \leq 0.75$ el proyecto experimental necesita ser modificado.

En la Tabla 3, se puede ver el instrumento que se empleó para la validación del proyecto experimental.

Tabla 3

Instrumento para la valoración de los jueces.

Ítem	Descripción
1	La interfaz de la APP es agradable e interactiva para el usuario.
2	El firmware desarrollado permite programar placas NodeMCU.
3	La APP permite visualizar el código de programación que genera el Firmware según las opciones elegidas.
4	La APP muestra alertas de error y consejos de programación al usuario.
5	El Firmware desarrollado no presenta fallos en su ejecución.

Fuente: Propia, 2021.

3. RESULTADOS.

Los códigos fuente del Firmware y la APP se encuentran subidos en el siguiente repositorio: <https://www.dropbox.com/sh/2tb89efjyv7dk59/AAAQ6Wpoid9oPzeWtDqXcTyGa?dl=0>

3.1. Diseño del esquemático y PCB en 2D y 3D.

El diseño del esquemático de la Tarjeta de Circuito Impreso (PCB, por sus siglas en inglés) de la placa de desarrollo para programar las placas

NodeMCU se lo realizó en el software EAGLE. En la Figura 6, se muestra el esquemático del sistema desarrollado.

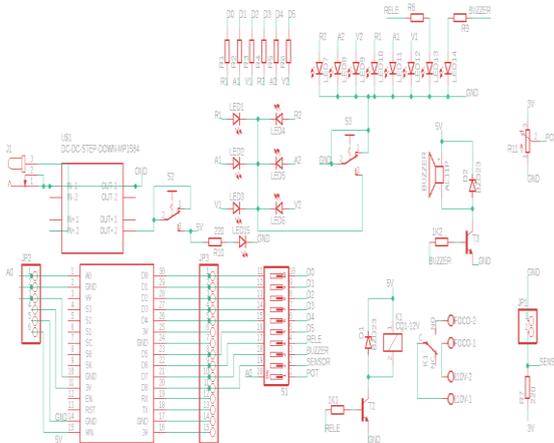


Figura 6: Esquemático del sistema.

Fuente: Propia, 2021.

El diseño de la PCB implementada se presenta en la Figura 7, las líneas azules representan las pistas y los puntos de color verde los orificios en los que se soldarán los elementos electrónicos.

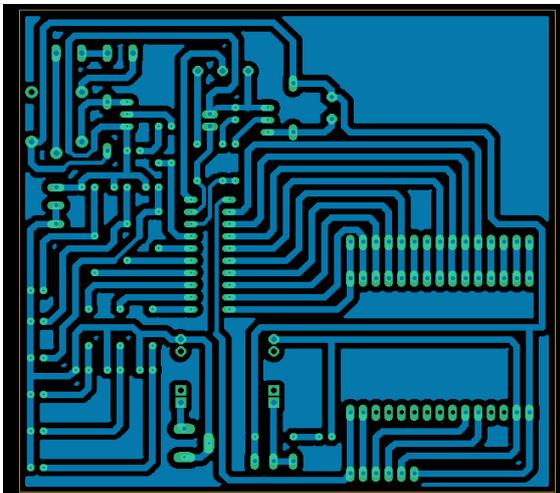


Figura 7: Diseño de la PCB del sistema en 2D.

Fuente: Propia, 2021.

En la Figura 8, se presenta el diseño del sistema en 3D, el cual permite tener una visión del sistema en la vida real sin ser implementado físicamente.

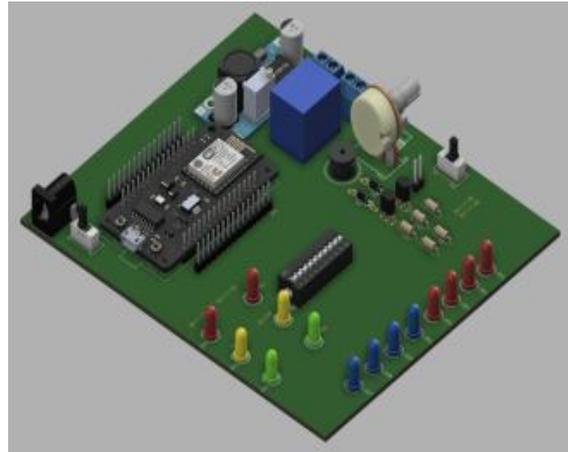


Figura 8: Diseño de la PCB del sistema en 3D.

Fuente: Propia, 2021.

3.2. Pruebas de funcionamiento.

La APP tiene una interfaz ordenada e intuitiva como se muestra en la Figura 9. La interfaz de inicio contiene un campo de texto en el cual se debe ingresar la dirección IP de la placa NodeMCU, una vez ingresada la IP se recomienda revisar que la conexión entre la APP y la placa se realizó satisfactoriamente por medio de los botones ON y OFF que encenderán y apagarán el Led integrado de la placa NodeMCU.



Figura 9: Pantalla principal de la APP.

Fuente: Propia, 2021.

Al presionar el botón de inicio, el usuario puede elegir entre los dos modos de programación

(Básico o Manual). La Figura 10 presenta la ventana descrita en la cual en la parte inferior se describe las funcionalidades de cada modo.



Figura 10: Modos de programación.
Fuente: Propia, 2021.

En la Figura 11, se puede visualizar el entorno de programación en bloques desarrollado en *Modo Básico* en la APP. Para una mejor experiencia el usuario, en cada bloque al usuario le van apareciendo indicaciones y consejos que le facilitan completar los ejercicios de aprendizaje propuestos.



Figura 11: Ventana de programación en Modo Básico.
Fuente: Propia,2021.

En la parte inferior de la pantalla se va generando el código en lenguaje de texto que cambia según las opciones se van eligiendo. Además, el usuario también la opción de limpiar para volver a iniciar, compilar para comprobar la programación realizada y enviar cuyo botón cambia de color a verde en caso de estar correcta la programación para posteriormente ejecutar los códigos en la placa NodeMCU.

La Figura 12, muestra la interfaz en el *Modo Manual* que se diferencia del *Modo Básico* al no limitar las opciones para programar, en este apartado los bloques de selección se encuentran visibles todo el tiempo para mostrar la estructura completa de la programación. Para programar salidas digitales, se utilizaron los pines desde el D0 hasta el D7, en dichos pines se conectaron seis led's, un relé y un buzzer.



Figura 12: Ventana de programación en Modo Manual.
Fuente: Propia, 2021.

En la Figura 13, se puede observar la activación de las salidas digitales en las que se encienden los led's.

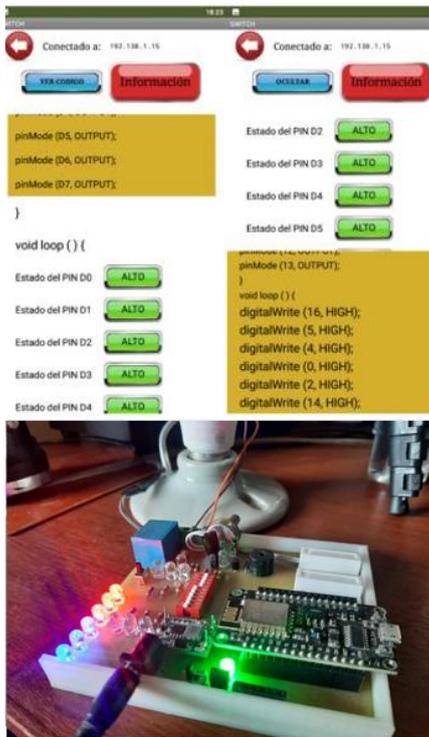


Figura 13: Programación de las salidas digitales.
Fuente: Propia, 2021.

La Figura 14, muestra la activación de un relé de 5V que enciende un bombillo eléctrico de 110V. La bombilla es controlada con el cambio de estado del sensor electromagnético que está conectado al pin D8 que es la entrada digital del sistema.

3.3. Resultados de la Técnica de los tres jueces.

La prueba de la técnica de jueces la realizaron tres ingenieros mecatrónicos con experiencia en el tema de investigación y desarrollo de proyectos, los cuales validaron el trabajo realizado.

En la Tabla 4, se detalla la calificación obtenida por cada uno de los tres jueces en cada ítem y la tabulación de los resultados. Donde r_i representa la sumatoria de los rangos por cada ítem, PR_i el promedio de rango por cada ítem y PpR_i es la proporción de rango por cada ítem.



Figura 14: Programación de una salida digital y una entrada digital.
Fuente: Propia, 2021.

Tabla 4

Resultados de la Técnica de los tres jueces.

Ítems	Validez de contenido por el método del coeficiente de proporción de rango.			r_i	PR_i	PpR_i
	Juez 1	Juez 2	Juez 3			
1	5	5	4	14	4.67	0.93
2	4	5	5	14	4.67	0.93
3	5	4	4	13	4.33	0.87
4	4	4	4	12	4.00	0.80
5	4	4	5	13	4.33	0.87
CPR						0.88

Fuente: Propia.

En vista que el coeficiente de proporción de rango obtenido es mayor a 0.75 se cumple la primera hipótesis, dando como resultado la validación del trabajo experimental.

4. DISCUSIÓN

Al trabajar con dispositivos móviles inteligentes, las mejores formas de establecer una conexión inalámbrica estable es el uso de datos móviles, conexión bluetooth o redes Wi-Fi, en base a la experiencia obtenida por Castrillejo (2020). El Firmware desarrollado usa la conectividad WiFi que incorporan las placas NodeMCU, reduciendo de esta manera los costos al no adquirir un módulo bluetooth o los gastos generados por los datos móviles.

Para crear APP con mayor calidad y agregar funciones avanzadas que aprovechan al máximo los recursos que poseen actualmente los dispositivos móviles inteligentes, existe software dedicado al desarrollo de estas APP's por lo que Zambrano (2019), para lograr un sistema confiable en su APP para personas con discapacidad visual hizo uso del software Android Estudio. Así mismo, para la creación de APP menos exigentes existe software gratuito como MIT APPInventor2, donde fue desarrollada la APP creada en este proyecto la cual tiene funciones menos exigentes.

El Firmware desarrollado para programar placas NodeMCU con dispositivos Android requiere de una APP con interfaz gráfica amigable e intuitiva para el usuario. Es por eso que, en el proyecto desarrollado por Bastidas, Yépez & Mosquera (2020), optaron por el uso de imágenes y botones para diseñar la interfaz gráfica que enseña matemática básica y orientación espacial. Por tal motivo, la interfaz desarrollada usa bloques con el fin de lograr una APP agradable e interactiva para el usuario.

En cuanto al Firmware desarrollado en comparación con los realizados por otros investigadores, permite programar placas NodeMCU de forma inalámbrica con bloques y además, muestra el código que representa a la

instrucción del bloque y si el usuario cometió un error es alertado para que lo corrija y pueda enviarse a las instrucciones de la APP a la placa microcontrolada NodeMCU.

5. CONCLUSIONES

La investigación realizada en los diferentes motores de búsqueda permitió comprender el funcionamiento, características y requerimientos necesarios para desarrollar un Firmware que permita programar placas NodeMCU desde dispositivos móviles Android. Se utilizó el software Fritzing para esquematizar las conexiones de todos los elementos electrónicos que formaron parte de la placa de desarrollo, además; el mismo facilitó el armado del sistema en el protoboard y la rápida detección de malos funcionamientos en el hardware. Se usó la librería *ESP8266WiFi.h* para establecer la comunicación inalámbrica entre la APP con la placa NodeMCU, se estableció el puerto de comunicación 80 como servidor para el envío y recepción de datos.

El Firmware desarrollado permite programar placas NodeMCU desde una APP interactiva desarrollada en la plataforma MIT APPInventor2, la cual presenta una interfaz de programación sencilla e interactiva. La APP junto al Firmware desarrollado facilitan la comprensión de las bases de la programación orientada a objetos. La APP usa bloques de selección para la programación por lo que el usuario no necesita escribir código y en la aplicación se despliega el código generado en cada caso para ir familiarizándose con el lenguaje de programación escrito.

Para conseguir el resultado esperado fue necesario realizar un sin número de pruebas de funcionamiento en protoboard y luego en la PCB, la validación de la funcionalidad de la APP y el Firmware se lo realizó mediante la técnica de los tres jueces expertos. La tabulación de las

calificaciones obtenidas arrojó un valor de CPR de 0.88 con lo cual se afirmó la primera hipótesis de la técnica que indica que el sistema desarrollado es válido.

A pesar de los resultados alcanzados, se sugiere que en base al Firmware y APP desarrollados se implemente el mismo en Android Studio para poder aumentar más funcionalidades y expandir el Firmware para más placas.

6. REFERENCIAS

Adeva, R. (03 de 03 de 2021). *Qué es Android: todo sobre el sistema operativo de Google*. Obtenido de www.adslzone.net: <https://www.adslzone.net/reportajes/software/que-es-android/>

aulaPlaneta . (15 de 02 de 2021). *STEAM: Una metodología educativa para el futuro*. Obtenido de www.aulaplaneta.com: <https://www.aulaplaneta.com/2020/12/01/recursos-tic/steam-una-metodologia-educativa-para-el-futuro/#:~:text=La%20metodolog%C3%ADa%20STEAM%20apuesta%20por,aprendizaje%20aplicado%20de%20los%20mismos.>

El Diario de la República. (25 de 01 de 2018). *La importancia de formar programadores*. Obtenido de www.eldiariodelarepublica.com: <https://www.eldiariodelarepublica.com>

Bastidas, Danilo, Yépez, F., & Mosquera, C. (2020). Implementación de un robot móvil con punto de acceso basado en La metodología STEAM. *Investigación Tecnológica*, 99–108. http://www.investigacionistct.ec/ojs/index.php/investigacion_tecnologica/article/view/57/28

Castrillejo, Ó. G. (2020). *Red inalámbrica entre dispositivos Arduino/ModemMCU*. <http://uvadoc.uva.es/handle/10324/43844>

Chursin, G., & Semenov, M. (2020). Using an ESP8266 Microcontroller to Develop a Learning Game. *Journal of Physics: Conference Series*, 1611(1). <https://doi.org/10.1088/1742-6596/1611/1/012059>

Granados, E. (10 de 11 de 2016). *La importancia de las app*. <https://www.developo.net/la-importancia-de-las-app/#:~:text=Las%20aplicaciones%20son%20importantes%20porque,guardar%20nuestras%20experiencias%20e%20informaci%C3%B3n.>

Hsieh, C. T. (2021). Developing programmable robot for K12 STEAM education. *IOP Conference Series: Materials Science and Engineering*, 1113(1), 012008. <https://doi.org/10.1088/1757-899x/1113/1/012008>

Ingeniería e Investigación, R. (2011). Ingeniería eléctrica. *Ingeniería e Investigación; Núm. 51 (2002); 13-15 2248-8723 0120-5609*. <http://revistas.unal.edu.co/index.php/ingevin/article/view/21432>

PUCE. (25 de 08 de 2015). *¿Qué es MIT APP Inventor* 2? [http://repositorio.puce.edu.ec/bitstream/handle/22000/13113/Anexo%203%20MIT%20App%20Inventor%202.pdf?sequence=4&isAllowed=y#:~:text=App%20Inventor%202%20\(AI2\)%20es,señala%20aplicaciones%20para%20dispositivos%20Android.](http://repositorio.puce.edu.ec/bitstream/handle/22000/13113/Anexo%203%20MIT%20App%20Inventor%202.pdf?sequence=4&isAllowed=y#:~:text=App%20Inventor%202%20(AI2)%20es,señala%20aplicaciones%20para%20dispositivos%20Android.)

Pujades, N. (15 de 10 de 2017). *La importancia de la programación en Latinoamérica*. <https://www.scratch.school/aprender/importancia-programacion-latinoamerica/>

Ramírez, T. (2010). *Cómo hacer un proyecto de Investigación* (E. Panapo (ed.); 2016th ed.).

Torres, A. (15 de 07 de 2020). *Sistema Operativo Android: ventajas y desventajas* (2020).

<https://psicologiaymente.com/miscelanea/sistema-operativo-android-ventajas-desventajas>

Tomala, O. (2016). *Métodos e instrumentos de investigación*.

<https://sites.google.com/site/misitioweboswaldotomala2016/home/recoleccion-de-datos-cuantitativos/metodos-e-instrumentos-de-investigacion>

Zambrano, D. M., Daza Álava, Y. D., Pinargote Zambrano, J. D., & Lituma Ramirez, E. D. (2019). Prototipo para orientación de personas con discapacidad Visual mediante una aplicación para móvil. *Revista Científica*, 2(35), 247–257. <https://doi.org/10.14483/23448350.14523>

Zumba, J., & León, C. (2018). Evolución de las Metodologías y Modelos utilizados en el Desarrollo de Software. *INNOVA Research Journal*, 3(10), 20–33.