



Implementación de una maqueta de una casa domótica controlada mediante celular.

Jami Guaman, Joel Alexis

Departamento de Eléctrica y Electrónica

Carrera de Tecnología Superior en Automatización e Instrumentación

Monografía, previo a la obtención del título de Tecnólogo Superior en Automatización e Instrumentación

Ing. Guerrero Rodríguez, Lucía Eliana

Latacunga, 16 de marzo del 2021



DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA.
CARRERA DE TECNOLOGÍA SUPERIOR EN AUTOMATIZACIÓN E
INSTRUMENTACIÓN

CERTIFICACIÓN

Certifico que la monografía, **“Implementación de una maqueta de una casa Domótica controlada mediante celular”** fue realizado por el señor **Jami Guaman, Joel Alexis** la cual ha sido revisada y analizada en su totalidad por la herramienta de verificación de similitud de contenido; por lo tanto cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, razón por la cual me permito acreditar y autorizar para que lo sustente públicamente.

Latacunga, 16 de marzo de 2021

Firma:



Firmado digitalmente por
**LUCIA ELIANA
GUERRERO
RODRIGUEZ**

Ing. Guerrero Rodríguez, Lucía Eliana

C. C.0501878649



Document Information

Analyzed document	JAMI JOEL_MONOGRAFÍA.docx (D99544233)
Submitted	3/25/2021 3:41:00 AM
Submitted by	Guerrero Rodriguez Lucia Eliana
Submitter email	leguerrero6@espe.edu.ec
Similarity	6%
Analysis address	leguerrero6.espe@analysis.urkund.com

Sources included in the report

SA	PROYECTO TECNOLOGICO_KJaramillo.docx Document PROYECTO TECNOLOGICO_KJaramillo.docx (D42975006)		1
W	URL: https://addi.ehu.es/bitstream/handle/10810/32243/TFG_Pe%C3%B1aUriarte.pdf?sequence ... Fetched: 12/31/2020 7:24:23 AM		1
SA	URKUNTITULACION DANY DELGADO URKUND.docx Document URKUNTITULACION DANY DELGADO URKUND.docx (D41743428)		1
SA	tesis_calderon_morales_miguel_angel.docx Document tesis_calderon_morales_miguel_angel.docx (D14814096)		3
W	URL: https://avelectronics.cc/producto/mini-bomba-de-agua-sumergible-120l-h/#:~:text=De ... Fetched: 3/25/2021 4:02:00 AM		1



Firma digitalizada por
LUCIA ELIANA
GUERRERO
RODRIGUEZ

ING. GUERRERO RODRÍGUEZ, LUCÍA ELIANA
C.C. 0501878649



DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE TECNOLOGÍA SUPERIOR EN AUTOMATIZACIÓN E
INSTRUMENTACIÓN

RESPONSABILIDAD DE AUTORÍA

Yo, **Jami Guaman, Joel Alexis**, con cédula de ciudadanía N° **0504445909**, declaro que el contenido, ideas y criterios de la monografía: **“Implementación de una maqueta de una casa Domótica controlada mediante celular”** es de mi autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Latacunga, 16 de marzo de 2021

Firma

Jami Guaman, Joel Alexis

C.C.: 0504445909



DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE TECNOLOGÍA SUPERIOR EN AUTOMATIZACIÓN E
INSTRUMENTACIÓN

AUTORIZACIÓN DE PUBLICACIÓN

Yo **Jami Guaman, Joel Alexis**, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar la monografía: **“Implementación de una maqueta de una casa Domótica controlada mediante celular”** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

Latacunga, 16 de marzo de 2021

Firma

Jami Guaman, Joel Alexis

C.C.: 0504445909

DEDICATORIA

El presente proyecto está dedicado primeramente a mis Padres, por ser un pilar fundamental en mis estudios y en mi diario vivir, a mi hermano por compartir momentos significativos conmigo y por siempre estar dispuesto a escucharme y ayudarme en cualquier momento, amigos quienes fueron un gran apoyo emocional durante el tiempo académico, los Docentes quienes brindaron motivación en nuestro proyecto mediante sus enseñanzas día a día

JAMI GUAMAN, JOEL ALEXIS

AGRADECIMIENTO

En primer lugar, doy infinitamente gracias a Dios y mis Padres, por haberme dado fuerza y valor para culminar esta tapa tan anhelada de mi vida.

Agradezco también la confianza y el apoyo brindado por parte de mi Padre y Madre que sin duda alguna me han demostrado su amor, corrigiendo mis faltas y celebrando mis triunfos.

A mi hermano, que con sus consejos me han ayudado a superar los retos y problemas que se me han presentado a lo largo de la vida

A todos los Docentes, quienes impartieron sus conocimientos y dedicación hacia el mejoramiento de los estudiantes.

JAMI GUAMAN, JOEL ALEXIS

Tabla de contenidos	
Carátula	1
Certificación	2
Reporte de verificación	3
Responsabilidad de auditoria	4
Autorización de publicación	5
Dedicatoria	6
Agradecimiento	7
Tabla de contenidos	8
Índice de figuras	11
Resumen	13
Abstrac	14
Introducción	15
Generalidades	15
<i>Tema</i>	15
<i>Línea de investigación</i>	15
Antecedentes	15
<i>Antecedentes de la Domótica</i>	15
<i>Antecedentes de Arduino</i>	16
Planteamiento del problema	17
Justificación	17
Objetivos	18
<i>Objetivo General</i>	18

Objetivos Específicos	18
Alcance	19
Investigación teórica	20
Que es Arduino	20
Arduino Mega	20
Comunicación de Bluetooth	22
Que es App Inventor	23
Módulo Bluetooth HC-06	24
Sensor MQ4	26
Servomotor SG90	27
Sensor LM35	28
Mini bomba de agua JT-DC3W	29
Desarrollo	31
Armado e implementación de la maqueta de una casa domótica	31
Simulación en Proteus	31
Código de programación en Arduino	32
Interfaz de la aplicación en App Inventor	47
Programación en diagrama de bloques en App Inventor para conectar el módulo Bluetooth	49
Programación en diagrama de bloques en App Inventor para regular intensidad, encender y apagar luces leds	49
Programación en diagrama de bloques en App Inventor para encender y apagar servomotores	51
Programación en diagrama de bloques en App Inventor para el llenado y vaciado de la piscina	52
Programación en diagrama de bloques en App Inventor para el control de temperatura	53

Programación en diagrama de bloques en App Inventor para cerrar la aplicación en el celular	54
Implementación de la maqueta de una casa domótica controlada mediante celular	54
Implementación de la iluminación en la maqueta de una casa domótica	55
Implementación del llenado y vaciado de una piscina en la maqueta de una casa domótica	56
Implementación de un control de temperatura en la maqueta de una casa domótica	57
Conclusiones y recomendaciones	58
Conclusiones	58
Recomendaciones	59
Bibliografías	60

Índice de figuras

Figura 1 Descripción de entradas y salidas del Arduino Mega 2560	21
Figura 2 Comunicación Bluetooth con Arduino	23
Figura 3 Mit App Inventor	24
Figura 4 Módulo Bluetooth HC-06	25
Figura 5 Sensor de gas MQ-4	26
Figura 6 Servomotor SG90	27
Figura 7 Pines de conexión del LM35	29
Figura 8 Mini bomba de agua JT-DC3W	30
Figura 9 Simulación en Proteus de los leds y servomotores	32
Figura 10 Diseño de la aplicación en App Inventor	48
Figura 11 Programación en diagrama de bloques en App Inventor para conectar con el módulo Bluetooth.....	49
Figura 12 Regular, encender y apagar los leds en el diagrama de bloques de App Inventor	50
Figura 13 Encender y apagar los servomotores en el diagrama de bloques de App Inventor	52
Figura 14 Programación en el diagrama de bloques de App Inventor para llenado y vaciado de la piscina	52
Figura 15 Programación en el diagrama de bloques de App Inventor para control de temperatura.....	53

Figura 16 <i>Programación en el diagrama de bloques para cerrar la aplicación en el celular</i>	54
Figura 17 <i>Proceso de construcción de la maqueta de una casa domótica</i>	55
Figura 18 <i>Encendido y apagado de luces leds</i>	56
Figura 19 <i>Llenado y vaciado de la piscina</i>	56
Figura 20 <i>Control de temperatura</i>	57

Resumen

El presente proyecto tiene como objetivo presentar y explicar el funcionamiento de una casa automatizada mediante la creación de un prototipo de una casa Domótica, dando a conocer las múltiples ventajas y facilidades que ofrece un sistema automatizado, la base principal de automatizar una casa es tener un control total mediante un dispositivo móvil que permite al ser humano visualizar como sería una vida futurista con la nueva tecnología que existe en la actualidad, el sistema de control está diseñado mediante la tecnología Arduino la cual permite realizar las funciones que serán ejecutadas mediante una aplicación en el dispositivo móvil, en la cual ha sido creada y diseñada la aplicación de acuerdo a la necesidad del control y proceso que se realice, los dispositivos en el prototipo tales como luces leds, servomotores, motores, y sensores son controlados remotamente mediante la comunicación Bluetooth, el objetivo general de la maqueta es dar a conocer las mejoras que puede ofrecer en la seguridad, el confort, la flexibilidad, las comunicaciones, el ahorro energético, facilitar el control integral de los sistemas para los usuarios y ofrecer nuevos y novedosos servicios, los resultados obtenidos sobre la maqueta muestran una efectividad obtenida del funcionamiento Domótico, por lo que dejan un aprendizaje exitoso.

Palabras clave:

- **CASA DOMÓTICA**
- **ARDUINO**
- **COMUNICACIÓN BLUETOOTH**
- **DISPOSITIVO MÓVIL**

Abstract

This project aims to present and explain the operation of an automated house through the creation of a prototype of a home automation system, revealing the many advantages and facilities offered by an automated system, the main basis of automating a house is to have full control through a mobile device that allows humans to visualize how it would be a futuristic life with the new technology that exists today, the control system is designed using Arduino technology which allows to perform the functions to be executed by an application on the mobile device, in which the application has been created and designed according to the need for control and process to be performed, the devices in the prototype such as LED lights, servo motors, motors, and sensors are controlled remotely via Bluetooth communication, the overall objective of the model is to publicize the improvements that can offer security, comfort, flexibility, communications, energy savings, facilitate the comprehensive control of systems for users and offer new and innovative services, the results obtained on the model show an effectiveness obtained from the Domotic operation, so they leave a successful learning.

Key word:

- **DOMOTIC HOUSE**
- **ARDUINO**
- **BLUETOOTH COMMUNICATION**
- **MOBILE DEVICE**

CAPÍTULO I

1 Introducción

1.1 Generalidades

1.1.1 Tema

Implementación de una maqueta de una casa domótica controlada mediante celular.

1.1.2 Línea de investigación

Electrónica

1.2 Antecedentes

1.2.1 Antecedentes de la Domótica.

La domótica se inició a comienzo de la década de los 70, cuando aparecieron las primeras pruebas en pisos piloto. Ya en los 80 cuando se empezaron a comercializar los integrados, fue cuando la domótica se empezó a expandir al hogar. Allí es cuando la domótica consigue integrar dos sistemas (el eléctrico y el electrónico) en igualdad de la comunicación integral de los dispositivos del hogar. (Carretero, 2016)

El desarrollo de la tecnología informática permite la expansión del sistema, sobre todo en países de vanguardia como Estados Unidos, Alemania y Japón y en los demás países subdesarrollados, pero con ansias de tener un buen futuro como es nuestro Ecuador. (Carretero, 2016)

El primer programa que utilizó la domótica fue el Save. Creado en Estados Unidos en 1984, permite lograr eficiencia y bajo consumo de energía en los sistemas de control de edificios inteligentes. Estas instalaciones regían bajo el sistema X-10, protocolo de comunicaciones que opera a través del accionar de un control remoto. (Carretero, 2016)

Desarrollado en 1976 por Pico Electronics (Escocia), sigue siendo la tecnología más utilizada dentro de la domótica. Al transmitir datos por líneas de baja tensión, la relación costo-beneficio sigue siendo la mejor opción en el rubro. Implantada desde hace más de treinta años, la domótica ha progresado a gran escala desde que se desarrollaron las redes informáticas de comunicación, ya sea por sistema cableado o vía Wi-Fi. (Carretero, 2016)

1.2.2 Antecedentes de Arduino

Arduino fue inventado en el año 2005 por el entonces estudiante del instituto IVRAE Massimo Banzi, quien, en un principio, pensaba en hacer Arduino por una necesidad de aprendizaje para los estudiantes de computación y electrónica del mismo instituto, ya que en ese entonces, adquirir una placa de micro controladores eran bastante caro y no ofrecían el soporte adecuado; no obstante, nunca se imaginó que esta herramienta se llegaría a convertir en años más adelante en el líder mundial de tecnologías DIY (Do It Yourself). Inicialmente fue un proyecto creado no solo para economizar la creación de proyectos escolares dentro del instituto, sino que, además, Banzi tenía la intención de ayudar a su escuela a evitar la quiebra de la misma con las ganancias que produciría vendiendo sus placas dentro del campus a un precio accesible (1 euro por unidad). El primer prototipo de Arduino fue fabricado en el instituto IVRAE. (Carretero, 2016)

Inicialmente estaba basado en una simple placa de circuitos eléctricos, donde estaban conectados un micro controlador simple junto con resistencias de voltaje, además de que únicamente podían conectarse sensores simples como leds u otras resistencias, y es más, aún no contaba con el soporte de algún lenguaje de programación para manipularla. Años más tarde, se integró al equipo de Arduino Hernando Barragán, un estudiante de la Universidad de Colombia que se encontraba haciendo su tesis, y tras enterarse de este proyecto, contribuyó al desarrollo de un entorno para la programación del procesador de esta placa: Wiring, en colaboración con David Mellis, otro integrante del mismo instituto que Banzi, quien más adelante, mejoraría la interfaz de software. (Carretero, 2016)

1.3 Planteamiento del problema

La domótica contribuye a llevar una calidad de vida cómoda aportando con una mejor gestión del servicio energético, seguridad y bienestar del usuario, así como por ejemplo hacer la vida más fácil a personas mayores de edad, niños o personas con discapacidad, la accesibilidad al mundo de la domótica es realmente sencilla debido a la nueva y reciente tecnología que nos ofrece hoy en día. Con la finalidad de dar a conocer estas y otras prestaciones se implementó una maqueta de una casa domótica

¿Qué sucedería si no implementamos la maqueta de la casa domótica?

El usuario no conocerá las ventajas que puede presentar el modelo de una maqueta a escala de una casa física real. Seguiría con el inconveniente de que el beneficiario continúe realizando las acciones de encendido y apagado de manera manual a los dispositivos que podrían ser controlados remotamente.

En el caso de la iluminación podría tener un alto consumo de energía debido a un alto grado de intensidad que presenta la iluminación actual.

En la seguridad no tendría un alto grado de protección debido a que no contara con las mejoras actuales del sistema de seguridad.

1.4 Justificación

En el desarrollo del presente trabajo se construyó una maqueta a escala de una casa física real sobre la que se van a desarrollar diferentes tipos de actividades prácticas relacionadas con el área de la domótica y electrónica.

El presente proyecto dispone de un sistema de control mediante dispositivo móvil a través de comunicación Bluetooth, que permitirá controlar diferentes instancias de la casa como:

- Control de encendido y apagado de iluminación
- Control de intensidad de iluminación
- Control de apertura y cierre de puertas mediante servomotores.

- Control de temperatura.
- Sistema de seguridad de alarma en el caso de recibir golpes en los cristales y en caso de detectar la presencia de gas.
- Llenado y vaciado de una piscina
- Control de electrodomésticos y equipos electrónicos existentes en la maqueta.

La maqueta a escala de una casa física real permitirá a los usuarios conocer cómo lograr un espacio de confort ya que con un solo toque podrán tener el control total de su hogar, así como el control a disposición de la iluminación ya sea encender o apagar un foco del hogar, además podrá regular la intensidad de luminosidad según la ocasión deseada.

La seguridad estará presente a todo momento debido a que cuenta con sensores y alarmas de alerta.

1.5 Objetivos

1.5.1 Objetivo General

Implementar una maqueta de una casa Domótica controlada mediante celular.

1.5.2 Objetivo Específicos:

- Investigar los elementos que permiten la implementación de una maqueta de una casa domótica
- Construir una maqueta didáctica para conocer el funcionamiento en pequeña escala de una casa domótica.
- Controlar mediante el dispositivo móvil el proceso de domotización de la maqueta.

1.6 Alcance

Este proyecto tiene como propósito dar a conocer las disposiciones que ofrece una maqueta de una casa domótica para implementarlo en una casa física real.

CAPÍTULO II

2. Investigación teórica

La presente monografía requiere de la revisión bibliográfica y conceptos referentes a la programación de Arduino y App Inventor mediante una comunicación Bluetooth. Un resumen de la revisión bibliográfica se presenta a continuación.

2.1 ¿Qué es Arduino?

Arduino es una plataforma de creación de electrónica de código abierto, la cual está basada en hardware y software libre, flexible y fácil de utilizar para los creadores y desarrolladores. Esta plataforma permite crear diferentes tipos de microordenadores de una sola placa a los que la comunidad de creadores puede darles diferentes tipos de uso. (Arduino.cl, 2019)

Para poder entender este concepto, primero debe entender los conceptos de hardware libre y el software libre. El hardware libre son los dispositivos cuyas especificaciones y diagramas son de acceso público, de manera que cualquiera puede replicarlos. Esto quiere decir que Arduino ofrece las bases para que cualquier otra persona o empresa pueda crear sus propias placas, pudiendo ser diferentes entre ellas, pero igualmente funcionales a partir de la misma base. (Arduino.cl, 2019)

2.2 Arduino Mega

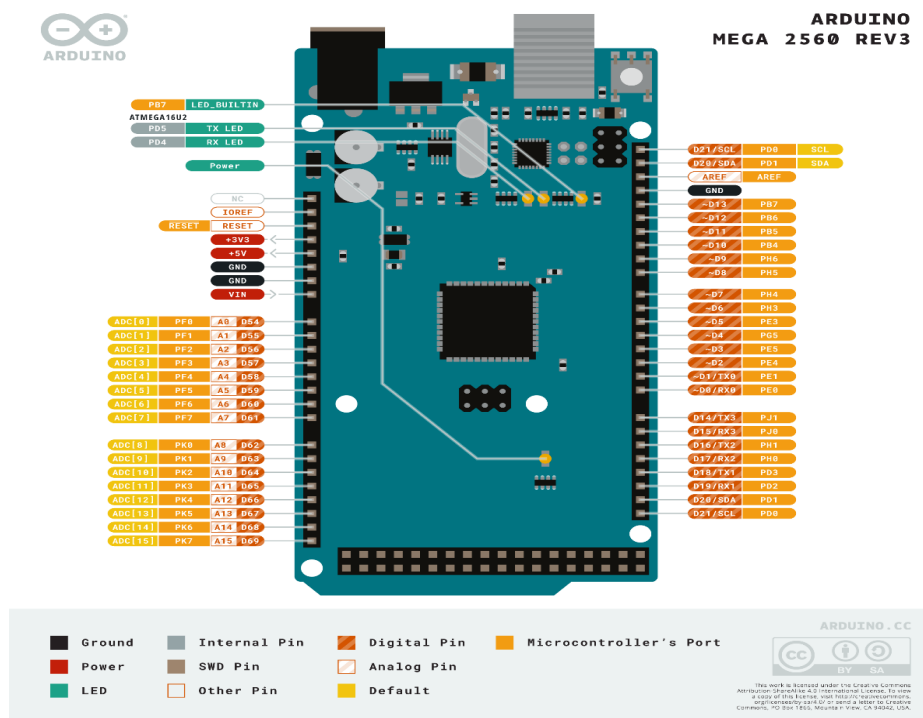
El Arduino Mega 2560 es una placa de microcontrolador basada en el ATmega2560. Tiene 54 pines de entrada / salida digital (de los cuales 15 se pueden usar como salidas PWM), 16 entradas analógicas, 4 UART (puertos serie de hardware), un oscilador de cristal de 16 MHz, una conexión USB, un conector de alimentación, un encabezado ICSP, y un botón de reinicio. Contiene todo lo necesario para soportar el microcontrolador; simplemente conéctelo a una computadora con un cable USB o enciéndalo con un adaptador de CA a CC o una batería para comenzar. La placa Mega 2560 es compatible con la mayoría

de los elementos diseñados para la Uno y las antiguas placas Duemilanove o Diecimila. El Mega 2560 es una actualización del Arduino Mega , al que reemplaza. (Arduino.cl, 2018)

El Arduino Mega 2560 es una placa de microcontrolador basada en el ATmega2560. Tiene 54 pines de entrada / salida digital (de los cuales 15 se pueden usar como salidas PWM), 16 entradas analógicas, 4 UART (puertos serie de hardware), un oscilador de cristal de 16 MHz, una conexión USB, un conector de alimentación, un encabezado ICSP, y un botón de reinicio. Contiene todo lo necesario para soportar el microcontrolador; simplemente conéctelo a una computadora con un cable USB o enciéndalo con un adaptador de CA a CC o una batería para comenzar. (TRONIC, 2018)

Figura 1

Descripción de entradas y salidas del Arduino Mega 2560



Nota. En la imagen se presenta la disposición de pines de entrada y salida de Arduino Mega. Tomado de (TRONIC, 2018)

2.3 Comunicación Bluetooth con Arduino

Bluetooth es una especificación industrial para Redes Inalámbricas de Área Personal (WPAN) que posibilita la transmisión de voz y datos entre diferentes dispositivos mediante un enlace por radiofrecuencia en la banda ISM de los 2.4 GHz. (Aprendiendoarduino, 2018)

Los principales objetivos que se pretenden conseguir con esta norma son:

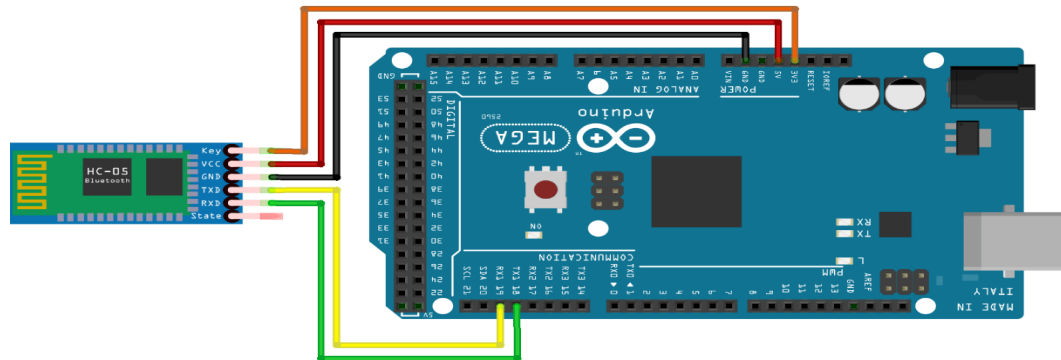
- Facilitar las comunicaciones entre equipos móviles.
- Eliminar los cables y conectores entre estos.
- Ofrecer la posibilidad de crear pequeñas redes inalámbricas y facilitar la sincronización de datos entre equipos personales.

Se denomina Bluetooth al protocolo de comunicaciones diseñado especialmente para dispositivos de bajo consumo, que requieren corto alcance de emisión y basados en transceptores de bajo costo. (Aprendiendoarduino, 2018)

Los dispositivos que incorporan este protocolo pueden comunicarse entre sí cuando se encuentran dentro de su alcance. Las comunicaciones se realizan por radiofrecuencia de forma que los dispositivos no tienen que estar alineados y pueden incluso estar en habitaciones separadas si la potencia de transmisión es suficiente. Estos dispositivos se clasifican como “Clase 1”, “Clase 2” o “Clase 3” en referencia a su potencia de transmisión. (Aprendiendoarduino, 2018)

Figura 2

Comunicación Bluetooth con Arduino



Nota. En la gráfica se representa la conexión física del Arduino Mega y módulo Bluetooth. Tomado de (Aprendiendoarduino, 2018)

2.4 ¿Qué es AppInventor?

AppInventor es un entorno de desarrollo de software creado por Google para la elaboración de aplicaciones destinadas al sistema operativo de Android. El lenguaje es gratuito y se puede acceder fácilmente de la web. Las aplicaciones creadas con AppInventor están limitadas por su simplicidad, aunque permiten cubrir un gran número de necesidades básicas en un dispositivo móvil. (Programo Ergo Sum, 2019)

Con AppInventor, se espera un incremento importante en el número de aplicaciones para Android debido a dos grandes factores: la simplicidad de uso, que facilitará la aparición de un gran número de nuevas aplicaciones; y Google Play, el centro de distribución de aplicaciones para Android donde cualquier usuario puede distribuir sus creaciones libremente. (Programo Ergo Sum, 2019)

Figura 3

Mit App Inventor



Nota. En la gráfica se presenta el logo de Mit App Inventor. Tomado de (Programo Ergo Sum, 2019)

2.5 Módulo Bluetooth HC-06

El módulo Bluetooth HC-06 permite conectar proyectos con Arduino a un smartphone, celular o PC de forma inalámbrica (Bluetooth), con la facilidad de operación de un puerto serial. La transmisión se realiza totalmente en forma transparente al programador, por lo que se conecta en forma directa a los pines seriales de microcontrolador preferido (respetando los niveles de voltaje, ya que el módulo se alimenta con 3.3V). Todos los parámetros del módulo se pueden configurar mediante comandos AT. La placa también incluye un regulador de 3.3V, que permite alimentar el módulo con un voltaje entre 3.6V - 6V. Este módulo es el complemento ideal para proyectos de robótica, domótica y control remoto con Arduino, PIC, Raspberry PI, ESP8266, ESP32, STM32, etc. (Naylamp Mechatronics, 2018)

Figura 4

Módulo Bluetooth HC-06.



Nota. La figura representa la estructura física del módulo Bluetooth HC-06.
Tomado de (Naylamp Mechatronics, 2018)

ESPECIFICACIONES TÉCNICAS

- Voltaje de operación: 3.3V - 5VDC
- Corriente de operación: < 40mA
- Corriente modo sleep: < 1mA
- Chip: BC417143
- Bluetooth: V2.0+EDR
- Frecuencia: Banda ISM de 2,4 GHz
- Modulación: GFSK (Gaussian Frequency Shift Keying)
- Potencia de emisión: 4 dBm, clase 2
- Sensibilidad: -84dBm a 0.1%
- Alcance 10 metros
- Interfaz de comunicación: Serial UART TTL
- Velocidad de transmisión: 1200bps hasta 1.3Mbps

2.6 Sensor MQ-4

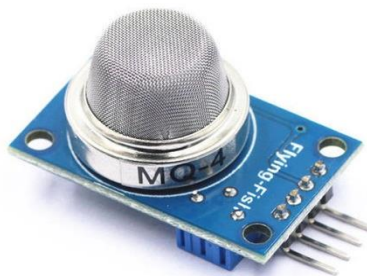
El sensor MQ-4 es un sensor de gas de tipo CNG, principalmente metano (CH₄) como vapores de gasolina, diesel etc. Como casi todos los sensores de éste tipo, es muy sencillo de utilizar ya que para funcionar tan solo necesita una resistencia y alimentarlo con 5V. El sensor devolverá un voltaje proporcional a la medición en una escala de entre 300 y 10000ppm. Por lo tanto, puede utilizarlo muy fácilmente con microcontrolador favorito como Arduino por ejemplo. (Brico Geek, 2020)

Características:

- Tipo de sensor: Semiconductor
- Sensible a gases de tipo CNG
- Rango de medición: 300 a 10000ppm
- Alimentación: 5V
- Consumo: 950mW
- Salida: 2.5 a 4.0V (a 5000ppm con CH₄)

Figura 5

Sensor de gas MQ-4



Nota. La figura representa la estructura física del sensor de gas MQ-4. Tomado de (Brico Geek, 2020)

2.7 Servomotor SG90

El servo SG90 es un servo miniatura de gran calidad y diminutas dimensiones, además es bastante económico. Funciona con la mayoría de tarjetas electrónicas de control con microcontroladores y además con la mayoría de los sistemas de radio control comerciales. (Electronilab, 2019)

El servo SG90 tiene un conector universal tipo “S” que encaja perfectamente en la mayoría de los receptores de radio control incluyendo los Futaba, JR, GWS, Cirrus, Hitec y otros. Los cables en el conector están distribuidos de la siguiente forma: Rojo =Alimentación (+), Café = Alimentación (-) o tierra, Naranja= Señal PWM. (Electronilab, 2019)

Características

- Velocidad: 0.10 sec/60° @ 4.8V
- Torque: 1.8 Kg-cm @ 4.8V
- Voltaje de funcionamiento: 3.0-7.2V
- Temperatura de funcionamiento: -30 °C ~ 60 °C
- Ángulo de rotación: 180°
- Ancho de pulso: 500-2400 μ s

Figura 6

Servomotor SG90



Nota. La figura representa la estructura física del servomotor SG90. Tomado de (Electronilab, 2019)

2.8 Sensor LM35

El LM35 es un sensor que puede medir temperatura. Su salida es analógica, es decir, proporciona un voltaje proporcional a la temperatura. El sensor tiene un rango desde -55°C a 150°C . Su popularidad se debe a la facilidad con la que se puede medir la temperatura. Incluso no es necesario de un microprocesador o microcontrolador para medir la temperatura. Dado que el sensor LM35 es analógico, basta con medir con un multímetro, el voltaje de salida del sensor. (HETPRO, 2017)

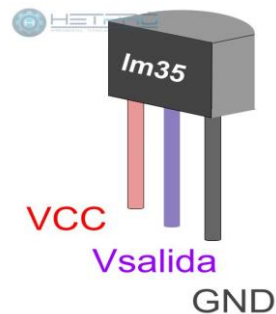
Para convertir el voltaje a la temperatura, el LM35 proporciona 10mV por cada grado centígrado. (HETPRO, 2017)

Características principales

- Resolución: 10mV por cada grado centígrado.
- Voltaje de alimentación. Por ejemplo, este sensor se puede alimentar desde 4Vdc hasta 20Vdc.
- Tipo de medición. Salida analógica.
- Numero de pines: 3 pines, GND, VCC y VSalida.
- No requiere calibración.
- Tiene una precisión de $\pm 1/4^{\circ}\text{C}$.
- Esta calibrado para medir $^{\circ}\text{C}$.
- Consumo de corriente: 60 μA

Figura 7

Pines de conexión del LM35



Nota. La figura representa los pines de conexión del sensor de temperatura LM35. Tomado de (HETPRO, 2017)

2.9 Mini bomba de agua JT-DC3W

Mini bomba de agua sumergible con un voltaje de funcionamiento de 2,5v a 6v DC, es un producto muy práctico y confiable que permitirá tener un flujo de hasta 2 litros de agua por minuto (de 80 -120l/h dependiendo del voltaje de alimentación), sirve para trabajos y proyectos sencillos que requieran de un pequeño caudal. (AV ELECTRONICS, 2020)

Incluye un motor interno de 0,3A y la carcasa está fabricada de termoplástico resistente a la erosión provocada por el fluido. Es totalmente sumergible. Su instalación es sencilla, solo se necesita colocar una pequeña manguera en la salida de la bomba, alimentarla con el voltaje deseado y sumergirla completamente en el líquido que deseamos bombear. (AV ELECTRONICS, 2020)

Características:

- Tensión: 2.5-6V.
- Elevación máxima: 40-110cm / 15.75"-43.4".
- Tasa de flujo: 80-120L / H.

- Diámetro exterior : 7.5mm / 0.3".
- Diámetro interior: 5mm / 0.2".
- Altura: aprox.30mm / 1.2".
- Material: plástico de ingeniería.
- Modo de conducción: Diseño cc, motriz magnética.
- La vida laboral continua durante 500 horas.

Figura 8

Mini bomba de agua JT-DC3W



Nota. La figura representa la estructura física de una mini bomba de agua JT-DC3W. Tomado de (AV ELECTRONICS, 2020)

CAPÍTULO III

3. Desarrollo

3.1 Armado e implementación de la maqueta de una casa domótica

Para la implementación de una casa domótica controlada mediante celular se realizaron los siguientes pasos que se detallan a continuación:

3.2 Simulación en Proteus

Se realizó una simulación mediante el software Proteus para verificar que todos los parámetros a realizarse se ejecuten de manera correcta y así evitar posibles complicaciones al momento de la implementación del proceso de manera física

Como primer punto se realizó la simulación de los leds y los servomotores en Proteus con la ayuda de un Arduino Mega y un Módulo Bluetooth HC-06


```
double tempC = 0;
```

```
double tempF = 0;
```

```
//-----//
```

```
#include <Servo.h>
```

```
Servo myservo;
```

```
Servo myservo1;
```

```
Servo myservo2;
```

```
Servo myservo3;
```

```
Servo myservo4;
```

```
Servo myservo5;
```

```
Servo myservo6;
```

```
int pos = 0;
```

```
//-----//
```

```
int LED1 = 2;
```

```
int LED2 = 3;
```

```
int LED3 = 4;
```

```
int LED4 = 5;
```

```
int LED5 = 6;
```

```
int LED6 = 7;
```

```
String readString;
```

```
//-----//
```

```
int led1 = 22;
```

```
int led2 = 23;
```

```
int estado = 0;
```

```
//-----//
```

```
void setup() {
```

```
  Serial.begin(9600);
```

```
  pinMode (LED1, OUTPUT);
```

```
  pinMode (LED2, OUTPUT);
```

```
  pinMode (LED3, OUTPUT);
```

```
  pinMode (LED4, OUTPUT);
```

```
  pinMode (LED5, OUTPUT);
```

```
  pinMode (LED6, OUTPUT);
```

```
  pinMode(led1, OUTPUT);
```

```
  pinMode(led2, OUTPUT);
```

```
//-----//  
  
Serial.begin(9600);  
  
myservo.attach(7,500,2500);  
  
myservo1.attach(8,500,2500);  
  
myservo2.attach(9,500,2500);  
  
myservo3.attach(10,500,2500);  
  
myservo4.attach(11,500,2500);  
  
myservo5.attach(12,500,2500);  
  
myservo6.attach(13,500,2500);  
  
//-----//  
  
Serial.begin(9600);  
  
pinMode(tempPin, INPUT);  
  
}  
  
//-----//  
  
void loop() {  
  
  PIS();  
  
  LEDS();  
  
  LM35();  
  
  SERVO();  
  
}
```

```
void LEDES (){  
  
char dato =Serial.read();{  
  
switch(dato) {  
  
    delay(1);  
  
    case 'A' :intensidad();  
  
    break;  
  
    case 'B' :intensidad1();  
  
    break;  
  
    case 'C' :intensidad2();  
  
    break;  
  
    case 'D' :intensidad3();  
  
    break;  
  
    case 'E' :intensidad4();  
  
    break;  
  
    case 'F' :intensidad5();  
  
    break;  
  
    }  
  
    }  
  
}
```

```
//-----//  
  
void LM35 (){  
  
  read_ADC =0;{  
  
  for (int i=0; i < 100; i++){  
  
    read_ADC = read_ADC+ analogRead(tempPin);  
  
    delay(1);  
  
  }  
  
  read_ADC = read_ADC/100;  
  
  temp = (read_ADC / 1023.0) * 5000;  
  
  tempC = temp * 0.1;  
  
  Serial.print("G2");  
  
  Serial.print(";");  
  
  Serial.print(tempC,0);  
  
  Serial.println(";");  
  
  }  
  
}  
  
//-----//  
  
void SERVO (){  
  
  if(Serial.available()>0)  
  
  {
```

```
char data = Serial.read();

delay(1);

if (data == 'y'){
  if (digitalRead('y') == LOW)
  {
    myservo.write(180);
    delay(200);
  }
}

if (data == 'z'){
  if (digitalRead('z') == LOW)
  {
    myservo.write(0);
    delay(200);
  }
}

if (data == 'w'){
  if (digitalRead('w') == LOW)
  {
    myservo1.write(180);
    delay(200);
```

```
}  
  
}  
  
if (data == 'x'){  
  if (digitalRead('x') == LOW)  
  {  
    myservo1.write(0);  
    delay(200);  
  }  
}  
  
if (data == 'u'){  
  if (digitalRead('u') == LOW)  
  {  
    myservo2.write(180);  
    delay(200);  
  }  
}  
  
if (data == 'v'){  
  if (digitalRead('v') == LOW)  
  {
```

```
myservo2.write(0);  
  
delay(200);  
  
}  
  
}  
  
if (data == 's'){  
  
if (digitalRead('s') == LOW)  
  
{  
  
myservo3.write(180);  
  
delay(200);  
  
}  
  
}  
  
  
  
if (data == 't'){  
  
if (digitalRead('t') == LOW)  
  
{  
  
myservo3.write(0);  
  
delay(200);  
  
}  
  
}  
  
  
  
if (data == 'q'){
```



```
if (digitalRead('q') == LOW)
{
myservo4.write(180);
delay(200);
}
}
```

```
if (data == 'r'){
if (digitalRead('r') == LOW)
{
myservo4.write(0);
delay(200);
}
}
if (data == 'o'){
if (digitalRead('o') == LOW)
{
myservo5.write(180);
delay(200);
}
}
```

```
if (data == 'p'){  
  
  if (digitalRead('p') == LOW)  
  
  {  
  
    myservo5.write(0);  
  
    delay(200);  
  
  }  
  
}  
  
if (data == 'm'){  
  
  if (digitalRead('m') == LOW)  
  
  {  
  
    myservo6.write(180);  
  
    delay(200);  
  
  }  
  
}  
  
  
  
if (data == 'n'){  
  
  if (digitalRead('n') == LOW)  
  
  {  
  
    myservo6.write(0);  
  
    delay(200);
```

```
    }  
  }  
}  
}  
  
//-----//  
  
void PIS(){  
  if( Serial.available(>0)  
  {  
    estado = Serial.read();  
  
  }  
  
  switch( estado){  
    delay(1);  
    case '1':  
      digitalWrite(led1, HIGH);  
      break;  
    case '2':  
      digitalWrite(led1, LOW);  
      break;  
  
    case '3':
```

```
    digitalWrite(led2, HIGH);

    break;

    case '4':

    digitalWrite(led2, LOW);

    break;

}

}

//-----//

void intensidad(){

    delay(15);

    while (Serial.available()){

    char C = Serial.read();

    readString += C;

    }

    if (readString.length()>0){

        analogWrite (LED1,(readString.toInt()));

        readString = "";

    }

}

//-----//

void intensidad1(){
```

```
    delay(15);

    while (Serial.available()){

char C = Serial.read();

    readString += C;

    }

if (readString.length(>0){

    analogWrite (LED2,(readString.toInt()));

    readString = "";

}

}

//-----//

void intensidad2(){

    delay(15);

    while (Serial.available()){

char C = Serial.read();

    readString += C;

    }

if (readString.length(>0){

    analogWrite (LED3,(readString.toInt()));

    readString = "";

}

}
```

```
}  
  
//-----//  
  
void intensidad3(){  
  
    delay(15);  
  
    while (Serial.available()){  
  
        char C = Serial.read();  
  
        readString += C;  
  
    }  
  
    if (readString.length()>0){  
  
        analogWrite (LED4,(readString.toInt()));  
  
        readString = "";  
  
    }  
  
}  
  
//-----//  
  
void intensidad4(){  
  
    delay(15);  
  
    while (Serial.available()){  
  
        char C = Serial.read();  
  
        readString += C;  
  
    }  
  
    if (readString.length()>0){
```

```

    analogWrite (LED5,(readString.toInt()));

    readString = "";

}

}

//-----//

void intensidad5(){

    delay(15);

    while (Serial.available()){

    char C = Serial.read();

    readString += C;

    }

    if (readString.length()>0){

        analogWrite (LED6,(readString.toInt()));

        readString = "";

    }

}

//-----//

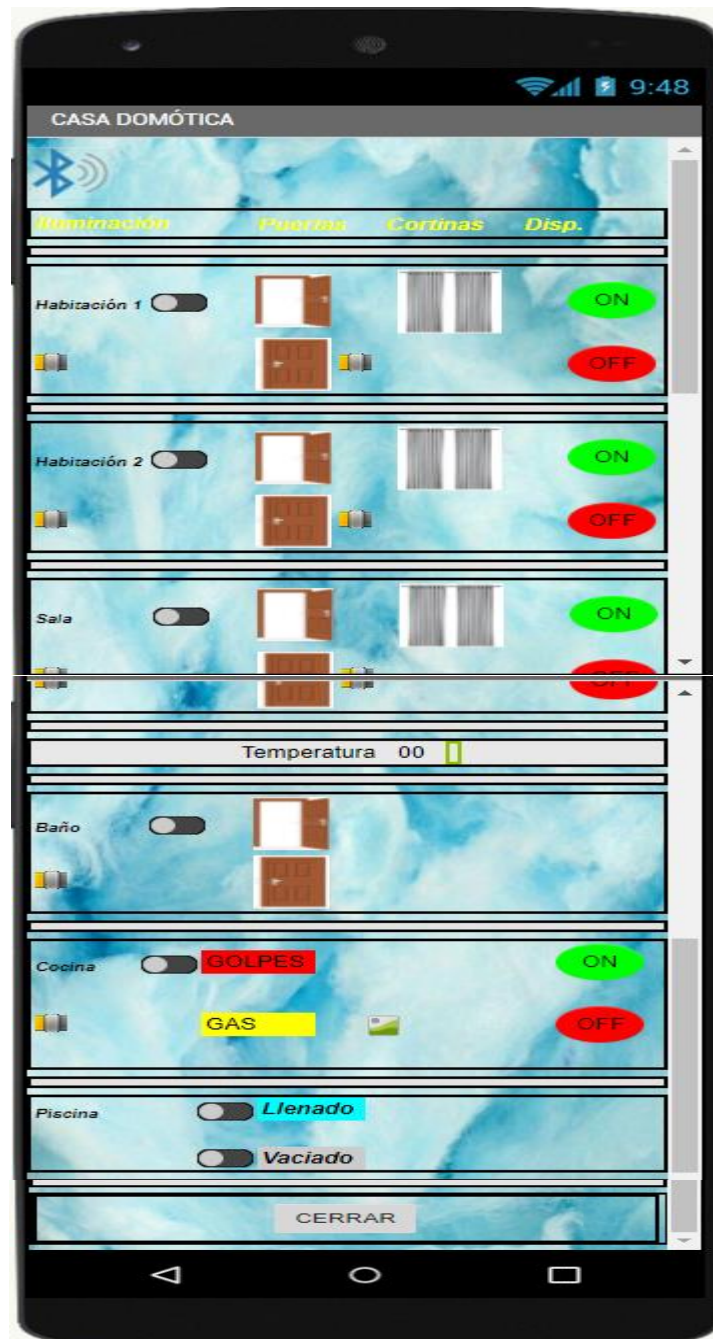
```

3.4 Interfaz de la aplicación en App Inventor

Para el control del proceso, se diseñó una aplicación en App Inventor la cual permitirá controlar el encendido y apagado de luces leds, servomotores, control de temperatura y el llenado y vaciado de una piscina en la simulación de la maqueta de una casa domótica.

Figura 10

Diseño de la aplicación en App Inventor



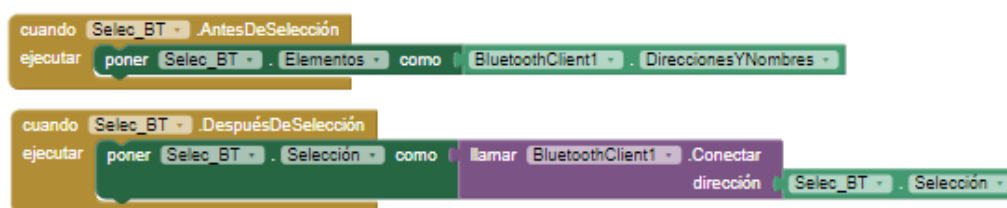
Nota. La imagen muestra el diseño de la aplicación mediante App Inventor

3.5 Programación en diagrama de bloques en App Inventor para conectar el Modulo Bluetooth

Para el control mediante la aplicación móvil se realizó la siguiente programación en diagrama de bloques, de acuerdo al código de Arduino para que siga la línea de programación.

Figura 11

Programación en diagrama de bloques en App Inventor para conectar con el módulo Bluetooth.



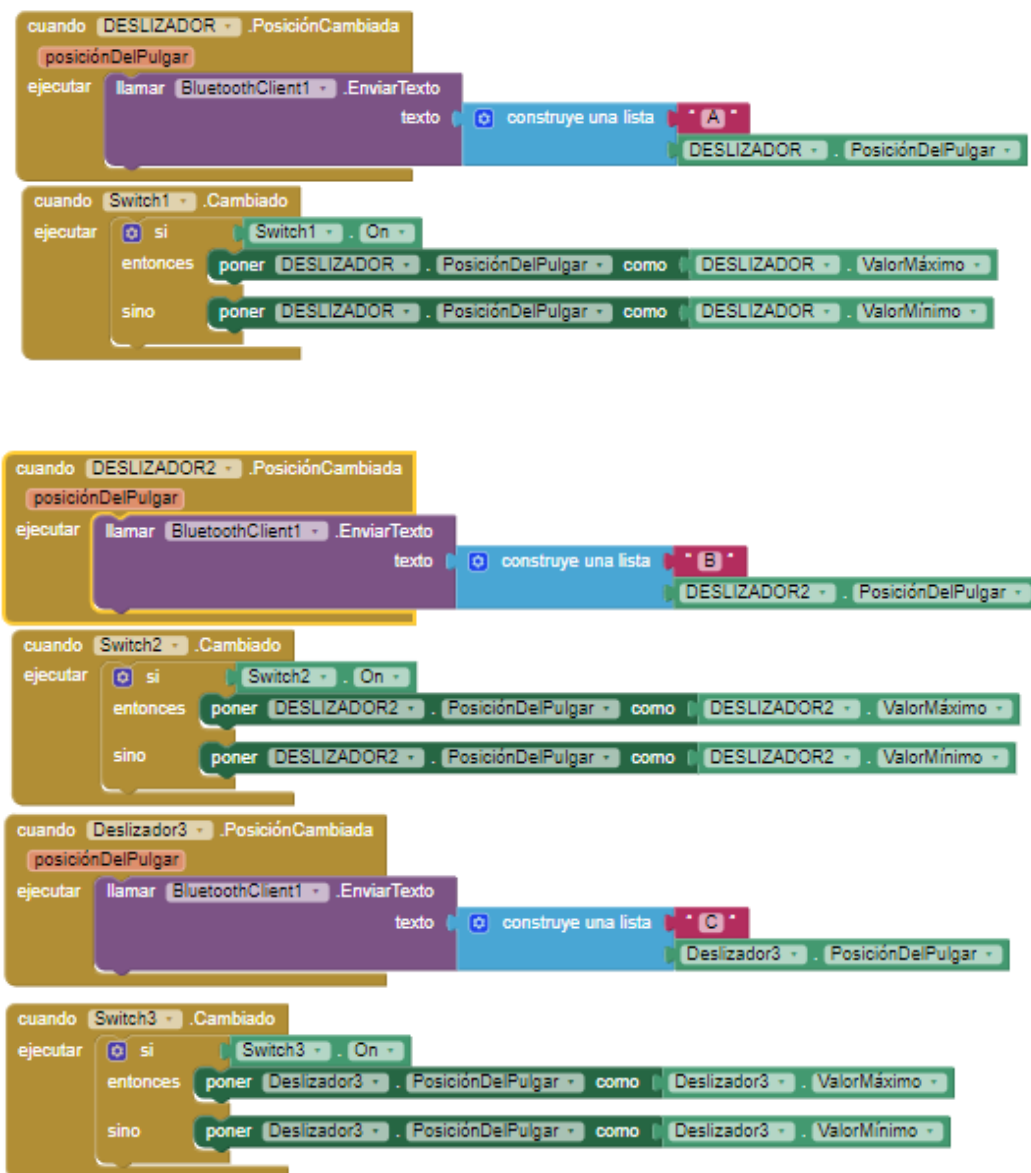
Nota. La imagen muestra la programación en diagrama de bloques en App Inventor para conectar realizar la conexión con el módulo Bluetooth.

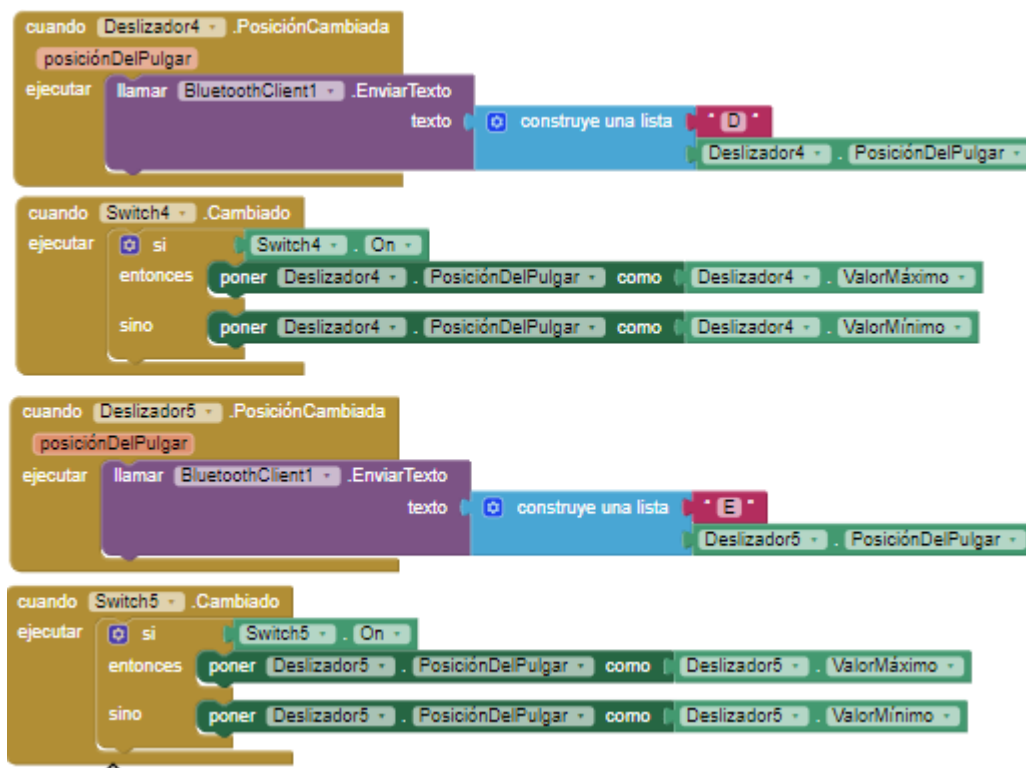
3.6 Programación en diagrama de bloques en App Inventor para regular intensidad, encender y apagar luces leds

Se realizó la programación en diagrama de bloques para que ejecuten los comandos para regular la intensidad luminosa de los leds, además el encendido y apagado de los mismos.

Figura 12

Regular, encender y apagar los leds en el diagrama de bloques de App Inventor





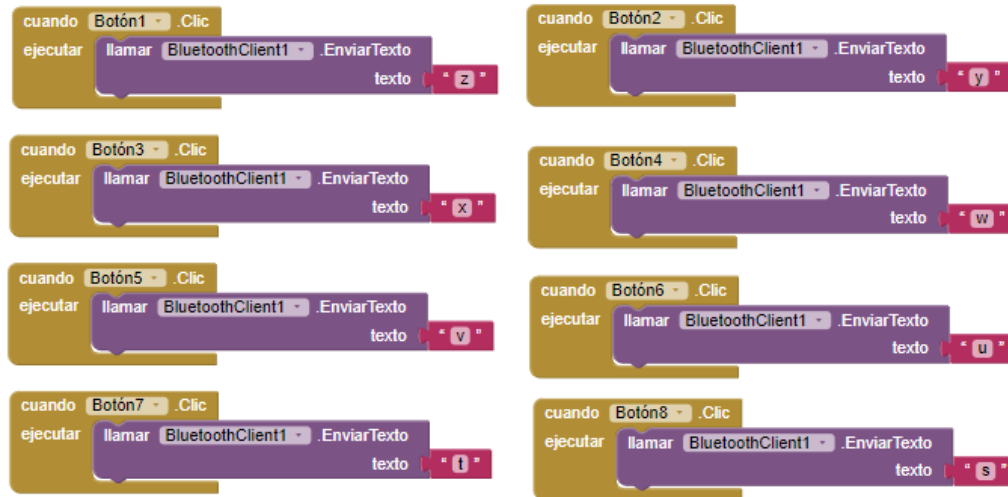
Nota. La imagen muestra la programación en diagrama de bloques en App Inventor para regular, encender y apagar leds.

3.7 Programación en diagrama de bloques en App Inventor para encender y apagar servomotores

Se realizó la programación en diagrama de bloques en App Inventor como se muestra en la figura 13, para que ejecuten los comandos de encendido y apagado de los servomotores.

Figura 13

Encender y apagar los servomotores en el diagrama de bloques de App Inventor



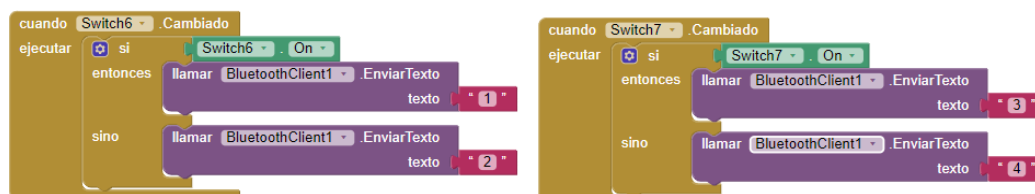
Nota. La imagen muestra la programación para encender y apagar servomotores en el diagrama de bloques de App Inventor

3.8 Programación en diagrama de bloques en App Inventor para el llenado y vaciado de la piscina

Se realizó la programación en diagrama de bloques para que ejecute los comandos de control de llenado y vaciado de una piscina.

Figura 14

Programación en el diagrama de bloques de App Inventor para llenado y vaciado de la piscina



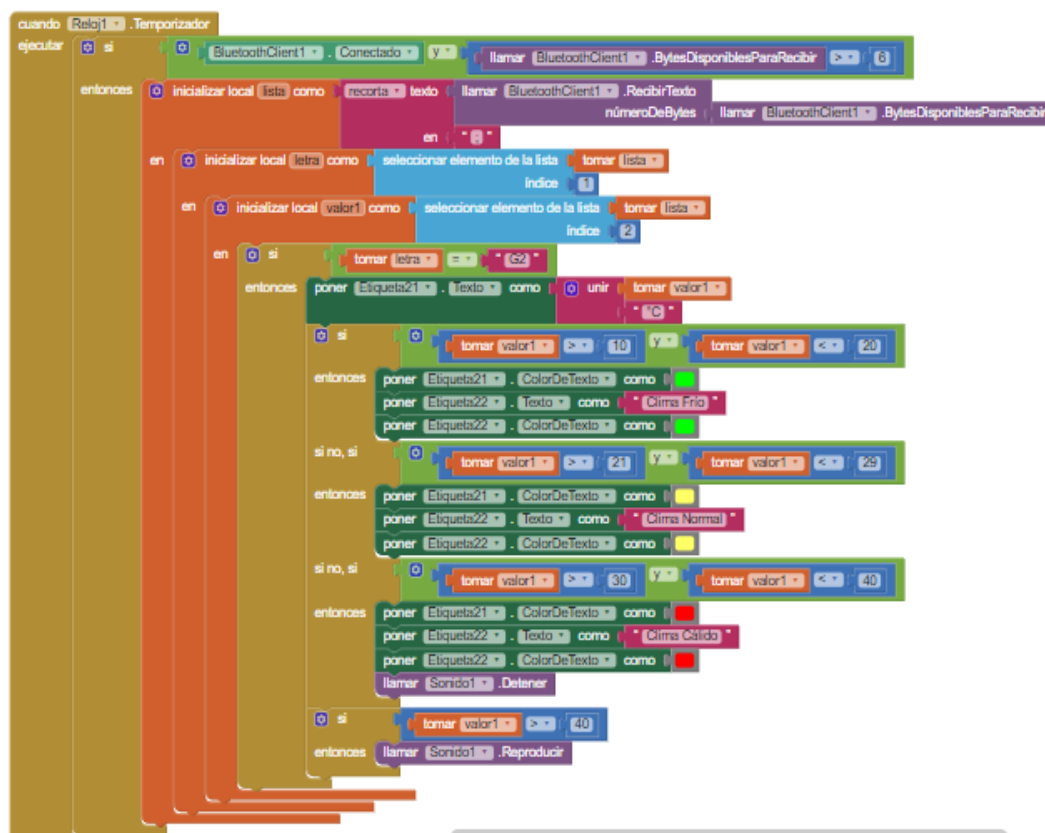
Nota. La imagen muestra la programación para el llenado y vaciado de una piscina en el diagrama de bloques de App Inventor

3.9 Programación en diagrama de bloques en App Inventor para el control de temperatura

Se realizó la programación en diagrama de bloques de App Inventor para que ejecute los comandos control de temperatura con el sensor Lm35.

Figura 15

Programación en el diagrama de bloques de App Inventor para control de temperatura



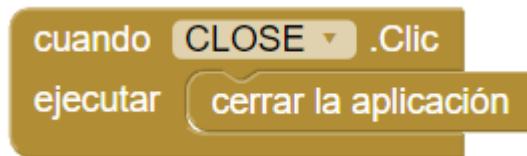
Nota. La imagen muestra la programación para el control de temperatura con el sensor LM35 en el diagrama de bloques de App Inventor

3.10 Programación en diagrama de bloques en App Inventor para cerrar la aplicación en el celular

Se realizó la programación en diagrama de bloques de App Inventor para que ejecute el comando de cierre de aplicación

Figura 16

Programación en el diagrama de bloques para cerrar la aplicación en el celular



Nota. La imagen muestra la programación de diagrama de bloques en App Inventor para cerrar la aplicación en el celular

3.11 Implementación de la maqueta de una casa domótica controlada mediante celular

Se realizó la construcción de una maqueta para una casa domótica mediante el control con el celular, a continuación, se muestra la maqueta

Figura 17

Proceso de construcción de la maqueta de una casa domótica



Nota. La imagen representa el proceso de construcción de la maqueta de una casa domótica controlada mediante celular

3.12 Implementación de la iluminación en la maqueta de una casa domótica

Se implementó la iluminación de luces leds en la maqueta de la casa domótica controlada mediante celular, además se comprobó el funcionamiento de la misma.

Figura 18

Encendido y apagado de luces leds



Nota. La imagen muestra el encendido y apagado de luces leds efectuadas mediante la aplicación en el celular

3.13 Implementación del llenado y vaciado de una piscina en la maqueta de una casa domótica

Se implementó el llenado y vaciado de una piscina en la maqueta de la casa domótica controlada mediante celular, además se comprobó el funcionamiento de la misma.

Figura 19

Llenado y vaciado de la piscina



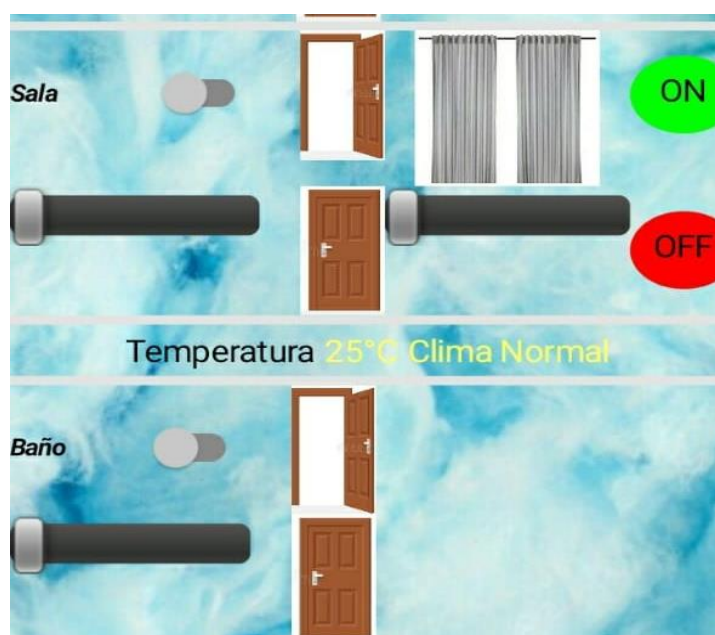
Nota. La imagen muestra el llenado y vaciado de una piscina efectuada mediante la aplicación en el celular

3.14 Implementación del control de temperatura en la maqueta de una casa domótica

Se implementó el control de temperatura con el sensor LM35 en la maqueta de la casa domótica controlada mediante celular, además se comprobó el funcionamiento de la misma.

Figura 20

Control de temperatura



Nota. La imagen muestra el control de temperatura efectuado mediante la aplicación en el celular.

Culminado el procedimiento expuesto, se ha logrado la implementación de una casa domótica controlada por celular, que permitirá exponer los beneficios, prestaciones y utilidades que puede brindar el desarrollo de este tipo de tecnología en los domicilios de usuarios que deseen acceder a un nivel de confort y seguridad.

CAPÍTULO IV

4. Conclusiones y recomendaciones

4.1 Conclusiones

- Se implementó la maqueta de una casa domótica controlada mediante celular, donde se exponen las prestaciones que brinda actualmente este tipo de tecnología.
- Se conoció el funcionamiento de los distintos elementos que componen una maqueta para la simulación de una casa domótica que podrían ser considerados para la instalación en una casa real.
- Para la comunicación entre dispositivos de la maqueta se utilizó la comunicación Bluetooth, permitiendo realizar el control de los diferentes sistemas a través del celular.
- El diseño de una aplicación en App Inventor puede variar de acuerdo a la necesidad y circunstancia que lo desee.
- Se verificó el funcionamiento óptimo que ofrece una maqueta para conocer los beneficios y ventajas que brinda una casa domótica a sus usuarios.

4.2 Recomendaciones

- Es importante tener un conocimiento básico de electrónica y robótica para realizar los procesos de domotización ya sea en procesos físicos o procesos reales.
- Para el diseño de la aplicación es recomendable tener concordancia con el código realizado en Arduino.
- La interfaz de la aplicación se puede modificar en su diseño dependiendo del uso que se considere, ya sea para personas mayores de edad, niños o personas con discapacidad.

Bibliografías

- Aprendiendoarduino. (2018). Recuperado el 25 de 01 de 2021, de <https://aprendiendoarduino.wordpress.com/2016/11/13/bluetooth-en-arduino/>
- Arduino.cl. (2018). Recuperado el 27 de 01 de 2021, de <https://store.arduino.cc/usa/mega-2560-r3>
- Arduino.cl. (2019). Recuperado el 26 de 01 de 2021, de <https://arduino.cl/que-es-arduino/>
- AV ELECTRONICS. (2020). Recuperado el 04 de 02 de 2021, de <https://avelectronics.cc/producto/mini-bomba-de-agua-sumergible-120l-h/#:~:text=Descripci%C3%B3n-,Mini%20bomba%20de%20agua%20sumergible%20con%20un%20voltaje%20de%20funcionamiento,que%20requieran%20de%20un%20peque%C3%B1o>
- Brico Geek. (2020). Recuperado el 08 de 02 de 2021, de <https://tienda.bricogeek.com/descatalogado/916-sensor-de-gas-metano-mq-4.html>
- Carretero, O. S. (2016). Casa Domótica con Arduino. Obtenido de <https://riunet.upv.es/bitstream/handle/10251/75797/SANCLEMENTE%20-%20DISE%C3%91O%20DE%20CASA%20DOM%C3%93TICA%20CONTROLADA%20POR%20ARDUINO.pdf?sequence=5>
- Electronilab. (2019). Recuperado el 19 de 02 de 2021, de <https://electronilab.co/tienda/micro-servo-9g-towerpro/>
- HETPRO. (2017). Recuperado el 15 de 02 de 2021, de <https://hetpro-store.com/TUTORIALES/lm35/>
- Naylamp Mechatronics. (2018). Recuperado el 10 de 02 de 2021, de <https://naylampmechatronics.com/inalambrico/24-modulo-bluetooth-hc06.html>
- Programo Ergo Sum. (2019). Recuperado el 02 de 2021, de <https://www.programoergosum.com/cursos-online/appinventor/27-curso-de-programacion-con-app-inventor/primeros-pasos>

TRONIC, T. (2018). Recuperado el 27 de 02 de 2021, de
<https://tostatronic.com/store/es/placas/1925-arduino-mega-2560-rev3-7630049200067.html>