



Implementación de un prototipo didáctico domótico para el control automático y seguro, mediante comandos de voz con Sistema Operativo Android

Acuña Espin, Carlos Andres

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. Cajas Buenaño, Mildred Lisseth

Latacunga, 17 de agosto del 2021



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

CERTIFICACIÓN

Certifico que la monografía, **“Implementación de un prototipo didáctico domótico para el control automático y seguro, mediante comandos de voz con Sistemas Operativo Android”** fue realizado por el señor **Acuña Espin, Carlos Andres** 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 la Fuerzas Armadas ESPE, razón por la cual me permito acreditar y autorizar para que lo sustente públicamente.

Latacunga, 17 de agosto del 2021

Firma:



Firmado electrónicamente por:
MILDRED
LISSETH CAJAS
BUENANO

Ing. Cajas Buenaño, Mildred Lisseth

C.C.: 0503497604

Urkund Analysis Result

Analysed Document: ACUÑA CARLOS_MONOGRAFÍA TUIJ.pdf
(D111286736)Submitted: 8/15/2021 12:11:00 AM
Submitted By: caacuna@espe.edu.ec
Significance: 8 %

Sources included in the report:

JAMI JOEL_MONOGRAFÍA.docx (D99544233)
RIVERA KEVIN_MONOGRAFÍA TUIJ.pdf (D111190967)

Instances where selected sources appear:

13



Firmado electrónicamente por:
**MILDRED
LISSETH CAJAS
BUENANO**

Ing. Cajas Buenaño, Mildred Lisseth

C.C.: 0503497604



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

RESPONSABILIDAD DE AUTORÍA

Yo, **Acuña Espin, Carlos Andres**, con cédula de ciudadanía N° **0503689028**, declaro que el contenido, ideas y criterios de la monografía: **“Implementación de un prototipo didáctico domótico para el control automático y seguro, mediante comandos de voz con Sistemas Operativo Android”** 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, 17 de agosto del 2021

Firma



Acuña Espin, Carlos Andres

C.C.: 0503689028



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

AUTORIZACIÓN DE PUBLICACIÓN

Yo, **Acuña Espin, Carlos Andres**, con cédula de ciudadanía 0503689028 autorizo a la Universidad de las Fuerzas Armadas ESPE publicar la monografía: **Implementación de un prototipo didáctico domótico para el control automático y seguro, mediante comandos de voz con Sistemas Operativo Android** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

Latacunga, 17 de agosto del 2021

Firma

Acuña Espin, Carlos Andres

C.C.: 0503689028

DEDICATORIA

El presente proyecto está dedicado a mis padres, por brindarme su amor, apoyo, comprensión y por ser un pilar fundamental en mis estudios a en mi diario vivir, a mi hermana por compartir momentos significativos conmigo y por siempre estar dispuesto a escucharme y ayudarme en cualquier momento, los Docentes quienes brindaron motivación en el proyecto mediante sus enseñanzas día a día.

ACUÑA ESPIN, CARLOS ANDRES

AGRADECIMIENTO

Quiero agradecer a mi asesora, la ing. Mildred Lisseth Cajas Buenaño por sus invaluable aportaciones académicas, porque a pesar de las horas de trabajo agotador, siempre tuvo una actitud amable e inspiradora, acompañándome en todo el proceso de investigación.

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

A Paul Mena quien ha estado conmigo desde mis primeros estudios apoyándome en las buenas y en las malas a lo largo de la vida.

A mi hermana, que con sus locuras y sonrisa siempre ha estado ahí para mi siendo una segunda madre, que con sus consejos me han ayudado a superar los retos y problemas que se han presentado.

A todos mis Docentes, quienes impartieron sus conocimientos y dedicación por enseñarme todo lo sé y más que eso, guiarme para ser una mejor persona y profesional.

ACUÑA ESPIN, CARLOS ANDRES

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
Abstract	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
<i>Objetivos Específicos</i>	18
Alcance	19

Investigación teórica	20
Arduino	20
Arduino Uno	20
Arduino Mega	22
Comunicación Bluetooth con Arduino.....	23
App Inventor	24
Módulo Bluetooth HC-06	25
Servomotor SG90	26
Mini bomba de agua JT-DC 3 W	27
Diodos emisores de luz de 1W	28
Teclado matricial 4x4 – Tipo membran	29
LCD 16x2	30
Desarrollo	32
Armado e implementación de un prototipo domótico para el control automático y seguro.....	34
<i>Simulación en Proteus del encendido y apagado de leds, control de los servomotores y de los motores.....</i>	34
<i>Código de programación en Arduino I</i>	35
<i>Simulación en Tinkercad de la puerta segura.....</i>	42
<i>Código de programación en Arduino II</i>	42
Interfaz de la aplicación en App Inventor	47
Programación en diagrama de bloques en App Inventor para conectar el módulo Bluetooth	48
Programación en diagrama de bloques en App Inventor para enviar comandos de voz	49
Implementación del prototipo de una casa domótica controlada mediante comandos de voz	50
Implementación de la iluminación en la maqueta de una casa domótica.....	51

Implementación de los servomotores en la maqueta de una casa domótica.....	52
Implementación del llenado y vaciado de una piscina en la maqueta de una casa domótica	53
Implementación de electrodomésticos en la maqueta de una casa domótica	54
Implementación de acceso seguro de la puerta principal en la maqueta de una casa domótica	55
Conclusiones y recomendaciones	57
Conclusiones	57
Recomendaciones	58
Bibliografías	59

Índice de figuras

Figura 1 Descripción de entradas y salidas del Arduino Uno	21
Figura 2 Descripción de entradas y salidas del Arduino Mega 2560	22
Figura 3 Comunicación Bluetooth con Arduino	23
Figura 4 Mit App Inventor	24
Figura 5 Módulo Bluetooth HC-06	25
Figura 6 Servomotor SG90	27
Figura 7 Mini bomba de agua JT-DC 3 W	28
Figura 8 LEDs de alta potencia 1W	29
Figura 9 Teclado Matricial 4x4 – Tipo membrana	30
Figura 10 LCD 16x2	31
Figura 11 Esquema de la instalación eléctrica del prototipo domótico... ..	33
Figura 12 Esquema de la instalación eléctrica del acceso seguro de la puerta principal... ..	33
Figura 13 Simulación en Proteus de los leds, servomotores y motores	35
Figura 14 Simulación en Tinkercad del acceso de una puerta segura.....	42
Figura 15 Diseño de la aplicación en App Inventor	48

Figura 16 <i>Programación en diagrama de bloques en App Inventor para conectar con el módulo Bluetooth.....</i>	49
Figura 17 <i>Programación en el diagrama de bloques de App Inventor para generar comandos de voz</i>	50
Figura 18 <i>Proceso de construcción de la maqueta de una casa domótica</i>	51
Figura 19 <i>Encendido y apagado de luces leds</i>	52
Figura 20 <i>Control de movimiento de los servomotores</i>	53
Figura 21 <i>Llenado y vaciado de la piscina</i>	54
Figura 22 <i>Encendido y apagado de electrodomésticos.....</i>	55
Figura 23 <i>Acceso seguro de la puerta principal</i>	56

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 a través de comandos de voz que permite al ser humano visualizar cómo 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 comandos de voz del usuario, en la cual ha sido creada y diseñada la aplicación de acuerdo con la necesidad del control y proceso que se realice, los dispositivos en el prototipo tales como luces leds, servomotores, motores, son controlados remotamente mediante la comunicación Bluetooth. También para la ejecución de puerta segura se encuentra un teclado matricial, servomotor y un LCD, 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**
- **COMUNICACIÓN BLUETOOTH**
- **COMANDOS DE VOZ**

Abstract

This project aims to present and explain the operation of an automated house through the creation of a prototype of a Domotic house, 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 through voice commands 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 voice commands from the user, which has been created and designed the application according to the need for control and process to be performed, the devices in the prototype such as LED lights, servo motors, motors, are controlled remotely via Bluetooth communication. Also for the execution of safe door is a matrix keyboard, servomotor and LCD, 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**
- **BLUETOOTH COMMUNICATION**
- **VOICE COMMANDS**

CAPÍTULO I

1. Introducción

1.1 Generalidades

1.1.1 Tema

Implementación de un prototipo didáctico domótico para el control automático y seguro, mediante comandos de voz con Sistemas Operativo Android.

1.1.2 Línea de investigación

Electrónica

1.2 Antecedentes

1.2.1 Antecedentes de la Domótica.

La domótica se inició a comienzos 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 microcontrolador 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, 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 contaría 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 seguridad y control mediante dispositivo móvil con comandos de voz 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.
- 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 solo comandos de voz 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 en todo momento ya que cuenta con un acceso seguro en la puerta principal diseñada para generar alertas de intrusos.

1.5 Objetivos

1.5.1 Objetivo General

Implementar un prototipo didáctico domótico para el control automático y seguro, mediante comandos de voz con Sistemas Operativo Android.

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 comandos de voz el proceso de domotización de la maqueta.
- Generar un acceso seguro a escala para controlar y salvaguardar la seguridad de quien acceda a la casa.

1.6 Alcance

Implementar un sistema domótico capaz de recibir una instrucción, por parte del usuario, desde un dispositivo móvil con una aplicación diseñada a medida, la cual será transmitida vía comunicación Bluetooth.

El desarrollo del trabajo se divide en dos grandes bloques: por un lado, el diseño y selección de la parte hardware, en la que se incluyen, motores, leds, LCD, servomotores y, por otro lado, la parte software que aglutina el código de programación. En ambos casos, se trata de emplear dispositivos de coste bajo como Arduino que además utilizan software libre, algo que sin duda marca una tendencia hoy en día.

CAPÍTULO II

2. Investigación teórica

El presente proyecto requiere de la revisión de citas bibliográficas y conceptos referentes a la programación APP inventor y Arduino mediante un módulo de comunicación Bluetooth. Un resumen de la revisión bibliográfica se presenta a continuación.

2.1 Arduino

Arduino es una plataforma de código abierto basada en hardware y software libre, fácil de usar para los usuarios, para la creación rápida de prototipos de electrónica, con lo que programar en Arduino resulta muy sencillo hasta para los más principiantes. La placa contiene un microcontrolador capaz de ejecutar instrucciones que se le envían a través del lenguaje de programación de Arduino y una plataforma de software llamada IDE (Ingeniería Mecafenix, 2017).

El principal lenguaje de programación utilizado para codificar un Arduino es C o C ++. Arduino tiene un entorno de desarrollo integrado (IDE) oficial para facilitar la codificación, especialmente para los principiantes. El IDE de Arduino tiene una interfaz minimalista, lo que facilita escribir, compilar y cargar el código en la placa Arduino. Es un software versátil compatible con MAC, Windows y Linux. (Ingeniería Mecafenix, 2017)

2.2 Arduino Uno

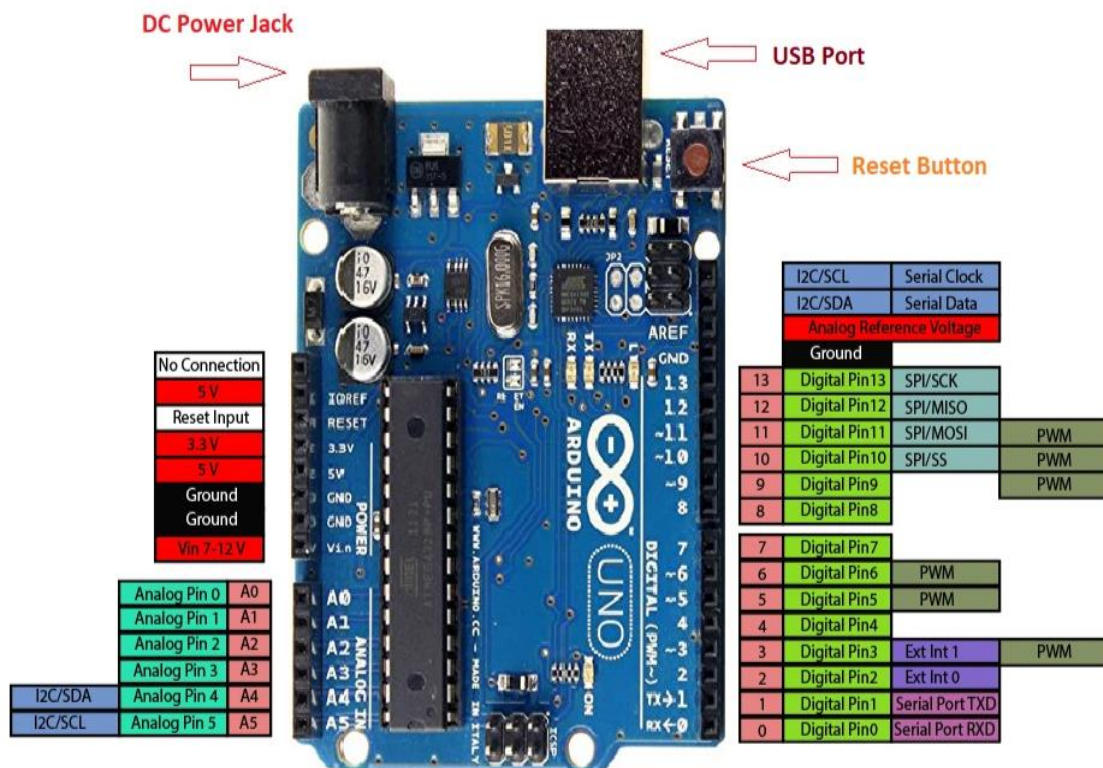
Arduino UNO es una placa de microcontrolador de código abierto programable, flexible y fácil de usar que se puede integrar en una variedad de proyectos electrónicos. Esta placa se puede conectar con otras placas Arduino, protectores Arduino, placas Raspberry Pi y puede controlar relés, LED, servos y motores como salida. Arduino UNO cuenta con un microcontrolador AVR Atmega328, 6 pines de entrada analógica y 14 pines de E / S digitales, de los

cuales 6 se utilizan como salida PWM, cómo se observa en la figura 1.
(Arduino.cl, 2017)

Esta placa contiene una interfaz USB, es decir, se utiliza un cable USB para conectar la placa con la computadora y el software Arduino IDE se utiliza para programar la placa. La unidad viene con una memoria flash de 32 KB que se utiliza para almacenar el número de instrucciones, mientras que la SRAM es de 2 KB y la EEPROM es de 1 KB. (TECmikro, 2018)

Figura 1

Descripción de entradas y salidas del Arduino Uno



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

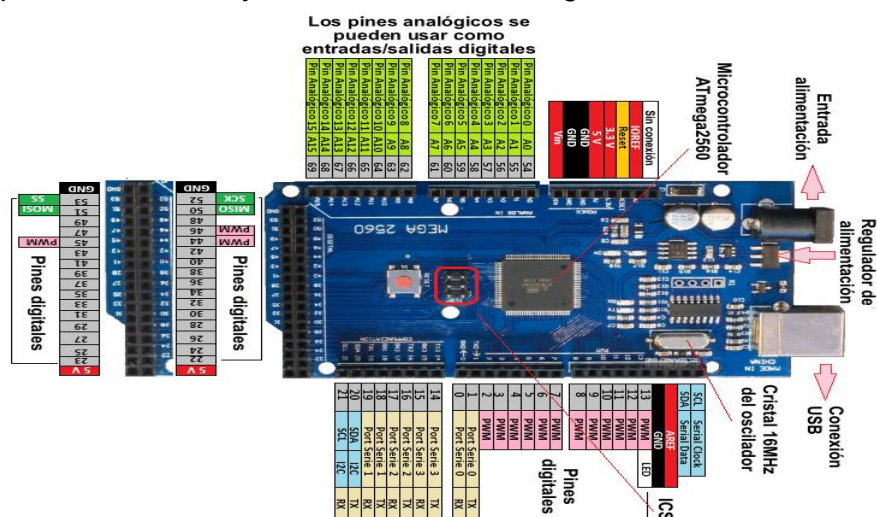
2.3 Arduino Mega

Arduino Mega 2560 es una placa electrónica de desarrollo basada en el microcontrolador Atmega2560. Esta placa es una buena combinación para proyectos que requieren más pines GPIO y espacio de memoria porque lleva 16 pines analógicos y 54 pines de E / S digitales de los cuales 15 pines se utilizan para salida PWM, como se muestra en la figura 2. La placa viene con un conector de alimentación de CC para encender esta unidad y también puede encender la placa usando el pin VIN en la placa. La unidad también admite una interfaz USB donde se utiliza un cable USB para conectar la placa con la computadora. (Arduino, 2018)

La unidad también admite el encabezado ICSP que se utiliza para programar la placa sin desconectarla del circuito principal. En la placa se incluyen dos reguladores de voltaje a través de los cuales puedes regular el voltaje como más te guste. Arduino Mega 2560 se programa utilizando el software Arduino IDE, que es el software oficial introducido por Arduino.cc. (PrestaShop, 2018)

Figura 2

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 (PrestaShop, 2018)

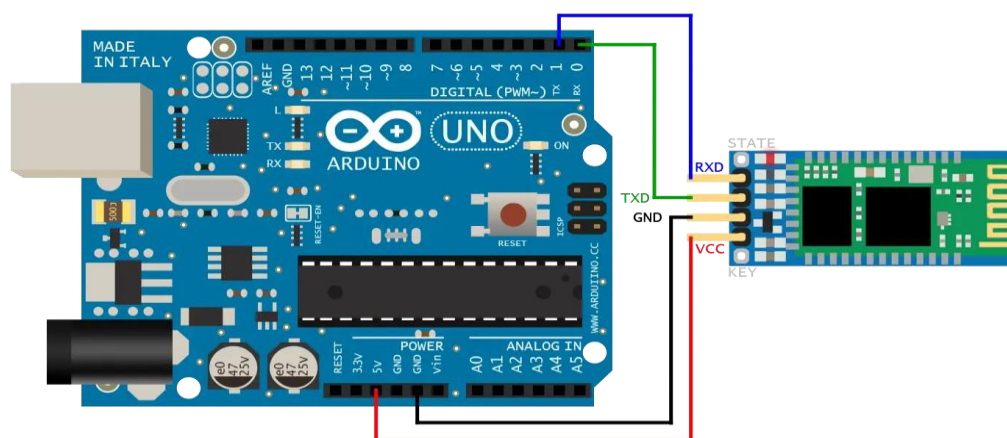
2.4 Comunicación Bluetooth con Arduino

Los módulos bluetooth funcionan con una comunicación en serie. La aplicación de Android está diseñada para enviar datos en serie al módulo Bluetooth Arduino cuando se presiona un botón en la aplicación. El módulo Bluetooth Arduino en el otro extremo recibe los datos y los envía al Arduino a través del pin TX del módulo Bluetooth, como se observa en la figura 3. (WordPress.com, 2018)

Los módulos bluetooth tienen dos modos: modo de datos y comando. Cuando está en modo de datos, el módulo actúa como un puente inalámbrico entre dos dispositivos. El módulo está en modo de datos por defecto. El pin KEY, cuando se tira hacia arriba, se usa para permitir que dichos módulos ingresen al modo AT Command. Los comandos AT se utilizan para configurar el módulo. (WordPress.com, 2018)

Figura 3

Comunicación Bluetooth con Arduino



Nota. En la gráfica se representa la conexión física del Arduino Mega y módulo Bluetooth en donde se encuentran los pines RXD y TXD del módulo bluetooth HC-06 conectados a los pines serial port del Arduino. Tomado de (WordPress.com, 2018)

2.5 App Inventor

MIT App Inventor es una interfaz de programación visual e intuitiva que permite a todos los usuarios la creación de aplicaciones completamente funcionales para teléfonos inteligentes y tablets. cuenta con un sistema de programación en bloques que inspiran el empoderamiento intelectual y creativo el, logo de la página se puede observar en la figura 4. (Asociación Programo Ergo Sum, 2019)

La interfaz de usuario de MIT App Inventor incluye dos editores principales: el editor de diseño y el editor de bloques. El editor de diseño, o diseñador, es una interfaz de arrastrar y soltar para diseñar los elementos de la interfaz del usuario de la aplicación. El editor de bloques es un entorno en el que los inventores de aplicaciones pueden diseñar visualmente la lógica de sus aplicaciones utilizando bloques codificados por colores que se conectan a través de piezas de rompecabezas para describir el programa. (Asociación Programo Ergo Sum, 2019)

Figura 4

Mit APP Inventor



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

2.6 Módulo Bluetooth HC-06

HC-06 es un módulo Bluetooth que permite la comunicación de datos inalámbrica entre dos microcontroladores o sistemas. El módulo funciona con el protocolo de comunicación Bluetooth 2.0 y solo puede actuar como dispositivo esclavo. Resulta ser un método muy eficaz para la transmisión inalámbrica de datos y también flexible en comparación con otros métodos e incluso puede transmitir archivos a una velocidad de hasta 2,1 Mb / s. (Naylamp Mechatronics SAC , 2018)

La comunicación con este módulo HC-06 se realiza a través de la interfaz UART. Los datos se envían al módulo o se reciben desde el módulo a través de esta interfaz. Entonces podemos conectar el módulo a cualquier microcontrolador o directamente a una PC que tenga puerto. A continuación, se muestra un circuito de interfaz típico del módulo a un Arduino. cómo se observa en la figura 5. (Naylamp Mechatronics SAC , 2018)

Figura 5

Módulo Bluetooth HC-06.



Nota. La figura representa la estructura física del módulo Bluetooth HC-06.

Tomado de (Naylamp Mechatronics SAC , 2018)

Especificaciones Técnicas

- Voltaje de operación: 3.3V - 5VDC
- Corriente de operación: < 40 mA
- Corriente modo sleep: < 1mA

- Frecuencia: Banda ISM de 2,4 GHz
- Potencia de emisión: 4dBm, clase 2
- Alcance 10 metros
- Velocidad de transmisión: 1200 bps hasta 1.3Mbps

2.7 Servomotor SG90

El Servomotor SG90 opera de 3.0V a 7.2 V, cuanto mayor es el voltaje, mayor es el par que podemos lograr, pero lo más común es que operan a + 5V. Casi todos los servomotores de hobby pueden girar solo de 0 ° a 180 ° debido a su disposición de engranajes, así que asegúrese de que el proyecto pueda vivir con el semicírculo si no, puede preferir un motor de 0 ° a 360 ° o modificar el motor para hacer un círculo completo, como se observa en la figura 6. Los engranajes de los motores se desgastan fácilmente, por lo que, si su aplicación requiere motores más fuertes y de larga duración, puede usar engranajes metálicos o simplemente usar engranajes de plástico normales. (Electronilab, 2019)

Características

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

Figura 6*Servomotor SG90*

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

2.8 Mini bomba de agua JT-DC 3W

La Mini bomba de agua sumergible tiene una carcasa ambiental que resiste a altas temperaturas. El rotor de la mini bomba no se desgasta por lo que es muy eficiente y alarga su vida útil. La ventosa de amortiguación de alta calidad puede reducir por completo el ruido que genera otras mini bombas de agua, se puede usar para el enfriamiento de agua de computadora, fuentes, artesanías, según las necesidades del usuario. (AV Electronics, 2021)

Características:

- Elevación máxima: 40-110 cm / 15.75"-43.4".
- Tasa de flujo: 80-120L / H.
- Altura: aprox.30mm / 1.2".
- Material: plástico de ingeniería.
- Modo de conducción: Diseño cc, motriz magnética.
- La vida laboral continua durante 500 horas.

Figura 7

Mini bomba de agua JT-DC 3W



Nota. La figura representa la estructura física de una mini bomba de agua JT-DC 3W. Tomado de (AV Electronics, 2021)

2.9 Diodos emisores de luz de 1W

Un LED es un tipo de diodo que convierte la energía eléctrica en luz. Para aquellos que no lo sepan, un diodo es un componente eléctrico que solo funciona en una dirección. Básicamente, un LED es un componente eléctrico que emite luz cuando la electricidad fluye en una dirección, desde el ánodo al cátodo, como se muestra en la figura 8. (Aguilar, 2018)

LED es un acrónimo que significa Light Emitting Diode. Básicamente, los LED son como pequeñas bombillas, solo requieren mucha menos energía para encenderse y son mucho más eficientes para producir altas salidas de luz. (Aguilar, 2018)

Características:

- Voltaje de operación de 3-4 Volts de corriente directa.
- Corriente de operación de 350 mA.
- Ángulo de apertura del haz luminoso de 120° sin óptica secundaria
- Control preciso y direccional del flujo luminoso emitido.
- Bajas pérdidas por disipación de calor.
- Mínima emisión de radiaciones infrarrojas y ultravioletas.

Figura 8

LEDs de alta potencia de 1W



Nota. La figura representa la estructura física de un LED de alta potencia de 1 W. Tomado de (Aguilar, 2018)

2.10 Teclado matricial 4x4 – Tipo membrana

El teclado Matricial contiene 16 botones que proporciona un componente de interfaz humana útil para proyectos de microcontroladores. El respaldo adhesivo proporciona una manera simple de montar el teclado en una variedad de aplicaciones. El teclado 4x4 cuenta con un total de 16 botones en forma de matriz. Este es un teclado de membrana sin partes móviles. Tiene una bonita superposición que representa un teclado tipo teléfono con cuatro botones funcionales adicionales. Se proporciona un conector berg hembra de 8 pines para interconectarse con los circuitos de su microcontrolador, como se observa en la figura 9. (UNIT Electronics, 2019)

Especificaciones Técnicas:

- 16 botones con organización matricial (4 filas x 4 columnas)
- Mayor resistencia al agua y al polvo
- Tiempo de rebote (Bounce time): ≤ 5 ms
- Máximo voltaje operativo: 24 V DC
- Máxima corriente operativa: 30 mA
- Expectativa de vida: 1.000.000 de operaciones
- Temperatura de operación: 0 a 50 °C

Figura 9

Teclado matricial 4x4 – Tipo membrana



Nota. La figura representa la estructura física de un teclado matricial 4x4 – tipo membrana. Tomado de (UNIT Electronics, 2019)

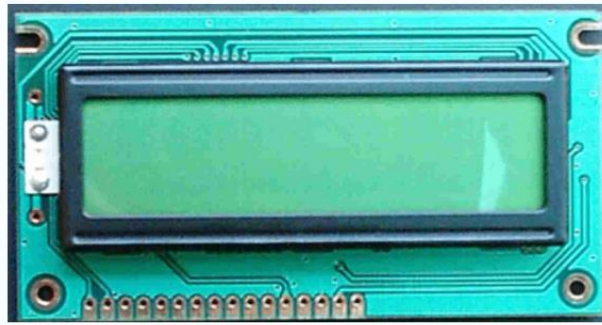
2.11 LCD 16X2

Un LCD es un módulo de visualización electrónico que utiliza cristal líquido para producir una imagen visible. La pantalla LCD de 16 x 2 es un módulo muy básico que se usa principalmente en circuitos eléctricos. El módulo LCD se traduce en una pantalla de 16 caracteres por línea. En esta pantalla LCD, cada carácter se muestra en una matriz de 5 x 7 píxeles, como se puede observar en la figura 10. (Barragan, 2017)

Las principales ventajas de este dispositivo LCD, es que tiene un bajo consumo de energía y de bajo costo. Las principales desventajas de este dispositivo LCD es que ocupa un área grande, son dispositivos lentos y también la vida útil se reducirá debido a la corriente continua. Por tanto, estas pantallas LCD utilizan una fuente de alimentación de CA con una frecuencia inferior a 500 Hz. (Barragan, 2017)

Figura 10

LCD 16X2



Nota. La figura representa la estructura física de un LCD 16X2. Tomado de (Barragan, 2017)

CAPÍTULO III

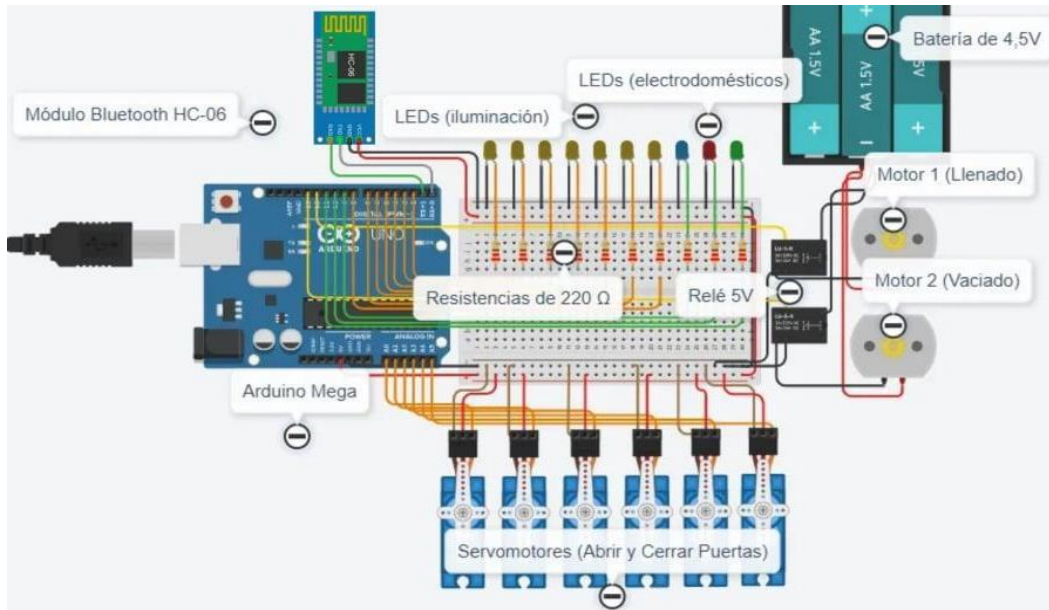
3. Desarrollo del proyecto

En el siguiente capítulo se desarrolla la implementación del prototipo domótico en donde previamente se realiza la conexión de los leds que permitirá controlar el encendido y apagado, la intensidad en 25%, 50% y 75%. También se realiza la instalación de los servomotores que permite simular el abierto y cerrado de las diferentes puertas de la casa, Por otro lado, el prototipo cuenta con dos mini bombas de agua conectados con relés de 5v para la comunicación con el Arduino mega utilizado, y así poder controlar el llenado y vaciado de la piscina. También se realiza la conexión de encendido y apagado de los electrodomésticos existentes en el prototipo domótico, toda la instalación se muestra en la figura 11.

Para finalizar se realiza la instalación del acceso segura de la puerta principal de la casa, en donde contamos un teclado matricial 4x4 que nos permitirá ingresar nuestra contraseña o código y así poder controlar el movimiento de nuestro servomotor, también cuenta con un LCD 16x2 que muestra los mensajes propuestos, por otro lado, se tiene dos leds indicadores que muestra cuando la puerta está abierta o cerrada. Por último, se encuentra un botón que nos servirá para resetear la programación cuando se encuentre en un bucle infinito, dicha instalación se puede observar en la figura 12.

Figura 11

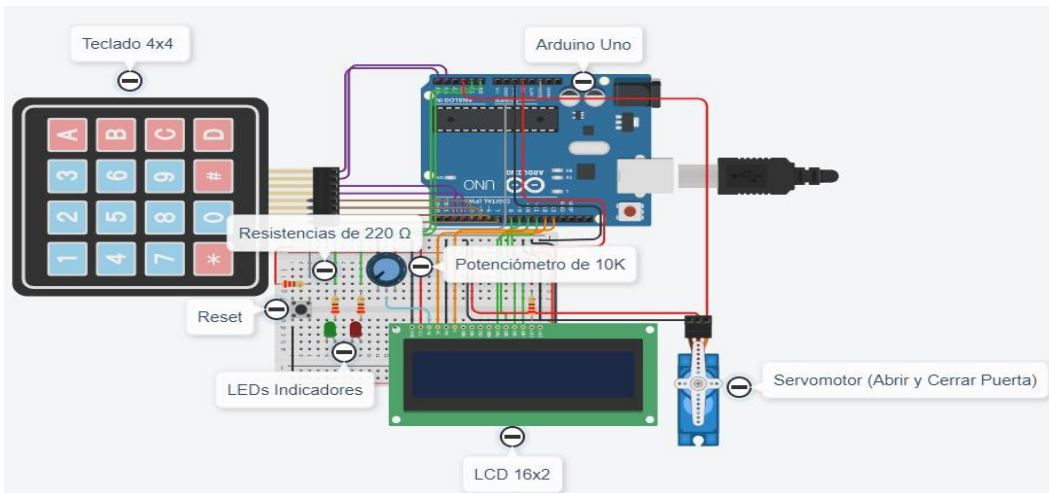
Esquema de la instalación eléctrica del prototipo domótico



Nota. El esquema muestra toda la instalación eléctrica implementado en el prototipo domótico

Figura 12

Esquema de la instalación eléctrica del acceso seguro de la puerta principal



Nota. El esquema muestra la instalación eléctrica del acceso segura de la puerta principal del prototipo domótico

3.1 Armado e implementación de un prototipo domótico para el control automático y seguro

Para la implementación de un prototipo domótico para el control automático y seguro, mediante comandos de voz en sistemas operativos Android se realizaron los siguientes pasos:

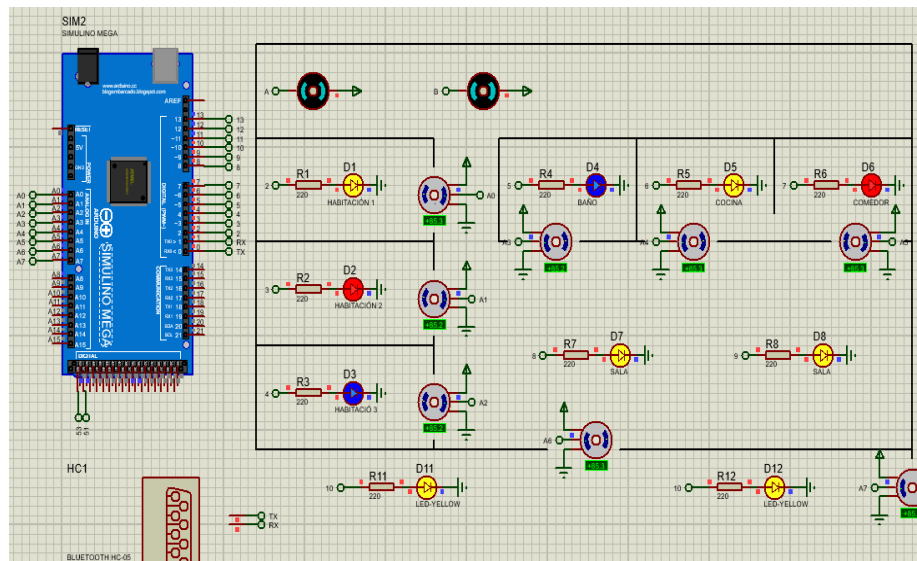
- 3.1.1 Simulación en Proteus del encendido, apagado e intensidad de leds, control de los servomotores y de los motores
- 3.1.2 Código de programación en Arduino I
- 3.1.3 Simulación en Tinkercad de la puerta Segura
- 3.1.4 Código de programación en Arduino II

3.1.1 Simulación en Proteus del encendido y apagado de leds, control de los servomotores y de los motores

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, por un lado tenemos leds que servirán para el encendido y apagado de las luces y con eso controlar su intensidad, también se obtiene unos servomotores que simularan el abierto y cerrado de las puertas dentro del prototipo domótico, se cuenta también con dos motores para ejercer el llenado y vaciado de la piscina, como se muestra en la figura 13.

Figura 13

Simulación en Proteus de los leds, servomotores y motores



Nota. La imagen representa la simulación en Proteus y la aplicación mediante la comunicación Bluetooth

3.1.2 Código de programación en Arduino I

Para realizar el control de encendido y apagado de leds, servomotores y llenado y vaciado de una piscina se realizó la programación en Arduino, la cual se muestra a continuación:

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

```
Servo servo1;
```

```
Servo servo2;
```

```
Servo servo3;
```

```
Servo servo4;
```

```
Servo servo5;
```

```
Servo servo6;
```

```
Servo servo7;

Servo servo8;

Servo servo9;

String mensaje;

void setup(){

  Serial.begin(9600);

  pinMode(2,OUTPUT);

  pinMode(3,OUTPUT);

  pinMode(4,OUTPUT);

  pinMode(5,OUTPUT);

  pinMode(6,OUTPUT);

  pinMode(7,OUTPUT);

  pinMode(8,OUTPUT);

  pinMode(9,OUTPUT);

  pinMode(10,OUTPUT);

  pinMode(11,OUTPUT);

  pinMode(12,OUTPUT);

  pinMode(13,OUTPUT);

  pinMode(35,OUTPUT);

  pinMode(37,OUTPUT);

  servo1.attach(A0);

  servo2.attach(A1);

  servo3.attach(A2);
```

```
servo4.attach(A3);
servo5.attach(A4);
servo6.attach(A5);
servo7.attach(A6);
servo8.attach(A7);
servo9.attach(A8);
}
void loop(){
  while(Serial.available()){
    delay(10);
    char c = Serial.read();
    mensaje += c;
  }
  if(mensaje.length(>0){
//_____ CODIGO DE LEDS _____//
    if(mensaje=="*Encender luz de la habitación 1"){digitalWrite(2,HIGH);}
    if(mensaje=="*Apagar luz de la habitación 1"){digitalWrite(2,LOW);}
    if(mensaje=="*Encender luz de la habitación 2"){digitalWrite(3,HIGH);}
    if(mensaje=="*Apagar luz de la habitación 2"){digitalWrite(3,LOW);}
    if(mensaje=="*Encender luz de la habitación 3"){digitalWrite(4,HIGH);}
    if(mensaje=="*Apagar luz de la habitación 3"){digitalWrite(4,LOW);}
    if(mensaje=="*Encender luz del baño"){digitalWrite(5,HIGH);}
    if(mensaje=="*Apagar luz del baño"){digitalWrite(5,LOW);}
```

```

if(mensaje=="*Encender luz de la cocina"){digitalWrite(6,HIGH);}

if(mensaje=="*Apagar luz de la cocina"){digitalWrite(6,LOW);}

if(mensaje=="*Encender luz del comedor"){digitalWrite(7,HIGH);}

if(mensaje=="*Apagar luz del comedor"){digitalWrite(7,LOW);}

if(mensaje=="*Encender luz de la
sala"){digitalWrite(8,HIGH);digitalWrite(9,HIGH);}

if(mensaje=="*Apagar luz de
lasala"){digitalWrite(8,LOW);digitalWrite(9,LOW);}

if(mensaje=="*Encender luz de afuera"){digitalWrite(10,HIGH);}

if(mensaje=="*Apagar luz de afuera"){digitalWrite(10,LOW);}

if(mensaje=="*encender todas las
luces"){digitalWrite(2,HIGH);digitalWrite(3,HIGH);digitalWrite(4,HIGH);digitalWrite
(5,HIGH);digitalWrite(6,HIGH);digitalWrite(7,HIGH);digitalWrite(8,HIGH);digitalWr
ite(9,HIGH);digitalWrite(10,HIGH);}

if(mensaje=="*Apagar todas las
luces"){digitalWrite(2,LOW);digitalWrite(3,LOW);digitalWrite(4,LOW);digitalWrite(
5,LOW);digitalWrite(6,LOW);digitalWrite(7,LOW);digitalWrite(8,LOW);digitalWrite(
9,LOW);digitalWrite(10,LOW);}

// _____ INTENCIDAD DE LED EN 25% 50% 70% _____ //

if(mensaje=="*foco 1 en 25"){analogWrite(2,63);}

if(mensaje=="*foco 1 en 50"){analogWrite(2,127);}

if(mensaje=="*foco 1 en 75"){analogWrite(2,190);}

if(mensaje=="*foco 2 en 25"){analogWrite(3,63);}

if(mensaje=="*foco 2 en 50"){analogWrite(3,127);}

if(mensaje=="*foco 2 en 75"){analogWrite(3,190);}

```

```
if(mensaje=="*foco 3 en 25"){analogWrite(4,63);}
if(mensaje=="*foco 3 en 50"){analogWrite(4,127);}
if(mensaje=="*foco 3 en 75"){analogWrite(4,190);}
if(mensaje=="*foco 4 en 25"){analogWrite(5,63);}
if(mensaje=="*foco 4 en 50"){analogWrite(5,127);}
if(mensaje=="*foco 4 en 75"){analogWrite(5,190);}
if(mensaje=="*foco 5 en 25"){analogWrite(6,63);}
if(mensaje=="*foco 5 en 50"){analogWrite(6,127);}
if(mensaje=="*foco 5 en 75"){analogWrite(6,190);}
if(mensaje=="*foco 6 en 25"){analogWrite(7,63);}
if(mensaje=="*foco 6 en 50"){analogWrite(7,127);}
if(mensaje=="*foco 6 en 75"){analogWrite(7,190);}
if(mensaje=="*foco 7 en 25"){analogWrite(8,63);}
if(mensaje=="*foco 7 en 50"){analogWrite(8,127);}
if(mensaje=="*foco 7 en 75"){analogWrite(8,190);}
if(mensaje=="*foco 7 en 25"){analogWrite(9,63);}
if(mensaje=="*foco 7 en 50"){analogWrite(9,127);}
if(mensaje=="*foco 7 en 75"){analogWrite(9,190);}
if(mensaje=="*foco 8 en 25"){analogWrite(10,63);}
if(mensaje=="*foco 8 en 50"){analogWrite(10,127);}
if(mensaje=="*foco 8 en 75"){analogWrite(10,190);}
//_____PISCINA_____//
if(mensaje=="*llenar piscina"){digitalWrite(37,HIGH);}
```

```
if(mensaje=="*vaciar piscina"){digitalWrite(35,HIGH);}

if(mensaje=="*Apagar motores"){digitalWrite(35,LOW);digitalWrite(37,LOW);}

//_____ CODIGO DE SERVOMOTORES _____//

if(mensaje=="*cerrar puerta 1"){servo1.write(0);}

if(mensaje=="*abrir puerta 1"){servo1.write(90);}

if(mensaje=="*cerrar puerta 2"){servo2.write(0);}

if(mensaje=="*abrir puerta 2"){servo2.write(90);}

if(mensaje=="*cerrar puerta 3"){servo3.write(0);}

if(mensaje=="*abrir puerta 3"){servo3.write(90);}

if(mensaje=="*cerrar puerta 4"){servo4.write(0);}

if(mensaje=="*abrir puerta cuatro"){servo4.write(90);}

if(mensaje=="*abrir puerta 5"){servo5.write(90);}

if(mensaje=="*cerrar puerta 5"){servo5.write(0);}

if(mensaje=="*abrir puerta 6"){servo6.write(90);}

if(mensaje=="*cerrar puerta 6"){servo6.write(180);}

if(mensaje=="*abrir puerta 7"){servo7.write(90);}

if(mensaje=="*cerrar puerta 7"){servo7.write(0);}

if(mensaje=="*abrir puerta 8"){servo8.write(90);}

if(mensaje=="*cerrar puerta 8"){servo8.write(0);}

if(mensaje=="*cerrar puerta 9"){servo9.write(0);}

if(mensaje=="*abrir puerta 9"){servo9.write(90);}

if(mensaje=="*cerrar puertas de las
habitaciones"){servo1.write(0);servo2.write(0);servo3.write(0);}
```



```

    if(mensaje=="*abrir puertas de las
habitaciones"){servo1.write(90);servo2.write(90);servo3.write(90);}

    if(mensaje=="*cerrar puerta de las
habitaciones"){servo1.write(0);servo2.write(0);servo3.write(0);}

    if(mensaje=="*abrir puerta de las
habitaciones"){servo1.write(90);servo2.write(90);servo3.write(90);}

    if(mensaje=="*Cerrar todas las
puertas"){servo4.write(0);servo5.write(0);servo6.write(180);servo7.write(0);servo
8.write(0);servo9.write(0);}

    if(mensaje=="*abrir todas las
puertas"){servo4.write(90);servo5.write(90);servo6.write(90);servo7.write(90);ser
vo8.write(90);servo9.write(90);}

//_____CODIGO ELECTRODOMESTICOS_____//

    if(mensaje=="*encender microondas"){digitalWrite(11,HIGH);}

    if(mensaje=="*Apagar microondas"){digitalWrite(11,LOW);}

    if(mensaje=="*encender televisión"){digitalWrite(12,HIGH);}

    if(mensaje=="*Apagar televisión"){digitalWrite(12,LOW);}

    if(mensaje=="*encender calefactor"){digitalWrite(13,HIGH);}

    if(mensaje=="*Apagar calefactor"){digitalWrite(13,LOW);}

}

mensaje="";

}

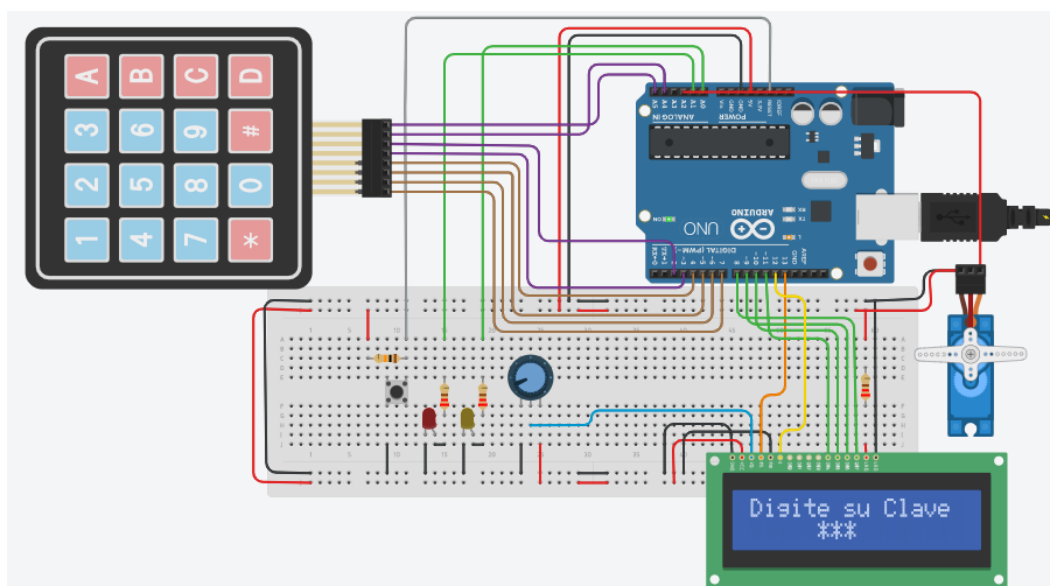
```

3.1.3 Simulación en Tinkercad de la puerta Segura

Se obtiene la simulación en Tinkercad de la puerta segura utilizando un teclado matricial 4x4 junto con un LCD que muestra los mensajes, como se muestra en la figura 14. También existen dos Leds que muestra el estado de abierto y cerrado de la puerta y con ello la ejecución del servomotor con la ayuda de un Arduino uno.

Figura 14

Simulación en Tinkercad del acceso de una puerta segura



Nota. La imagen representa la simulación en Tinkercad del acceso a una puerta segura.

3.1.4 Código de programación en Arduino II

Para realizar el control del acceso seguro de la puerta principal se realizó la programación en Arduino en donde incluimos diferentes librerías para los servomotores, para el teclado matricial 4x4 y para el LCD 16x2. Con ello se puede verificar que el código de programación en Arduino mega complida de mejor manera sin ningún fallo existente, la cual se muestra a continuación:

```
#include <Servo.h>

Servo servo1;

#include <LiquidCrystal.h>

#include <Key.h>

#include <Keypad.h>

int a=0, b=0, c=0, d=0;

int var=0;

int C1=1,C2=2,C3=3,C4=4;

char f='*';

int veces=0,incorrecto=0;

int aviso=3;

const byte filas = 4;

const byte columnas = 4;

char tecla[filas][columnas] = {

    {'1','2','3','A'},

    {'4','5','6','B'},

    {'7','8','9','C'},

    {'*','0','#','D'}};

byte pinFilas[filas] = {7, 6, 5, 4};

byte pinColumnas[columnas] = {3, 2, A4, A5};

Keypad keypad = Keypad( makeKeymap(tecla), pinFilas, pinColumnas,
filas, columnas );
```

```
LiquidCrystal lcd(13,12,11,10,9,8);

void setup(){

  servo1.attach(A2);

  lcd.begin(16,2);

  pinMode(A0,OUTPUT);

  pinMode(A1,OUTPUT);}

void loop(){

  char key = keypad.getKey();

  if (key){

    lcd.setCursor(5+var,1);

    lcd.print(key),lcd.setCursor(5+var,1),lcd.print(f);

    key=key-48;

    var++;

    switch(var){

      case 1:

        a=key;

        break;

      case 2:

        b=key;

        break;

      case 3:

        c=key;
```

```
break;

case 4:

d=key;

delay(100);

if(a==C 1 && b==C2 && c==C3 && d==C4){

    lcd.clear();

    lcd.setCursor(3,0);

    lcd.print("CLAVE");

    lcd.setCursor(3,1);

    lcd.print("CORRECTA");

    digitalWrite(A0,HIGH);

    servo1.write(180);

    delay(5000);

    lcd.clear();

    digitalWrite(A0,LOW);

    servo1.write(90); }

else{

    lcd.clear();

    lcd.setCursor(3,0);

    lcd.print("CLAVE");

    lcd.setCursor(2,1);

    lcd.print("INCORRECTA");

    digitalWrite(A1,HIGH);
```

```
delay(2000);

lcd.clear();

digitalWrite(A1,LOW); }

//-----Seguridad para la contraseña y sus restricciones-----//

if(a==C 1 && b==C2 && c==C3 && d==C4){

    veces=0;

    aviso=3;

}

else{

    veces ++;

    aviso --;

    lcd.setCursor(2,0);

    lcd.print("LE QUEDA: ");

    lcd.setCursor(13,0);

    lcd.print(aviso);

    lcd.setCursor(2,1);

    lcd.print("OPORTUNIDAD");

    if(avisó==0){

        lcd.clear();

        lcd.setCursor(5,0);

        lcd.print("ALARMA");

        lcd.setCursor(4,1);

        lcd.print("ACTIVADO"); }
```

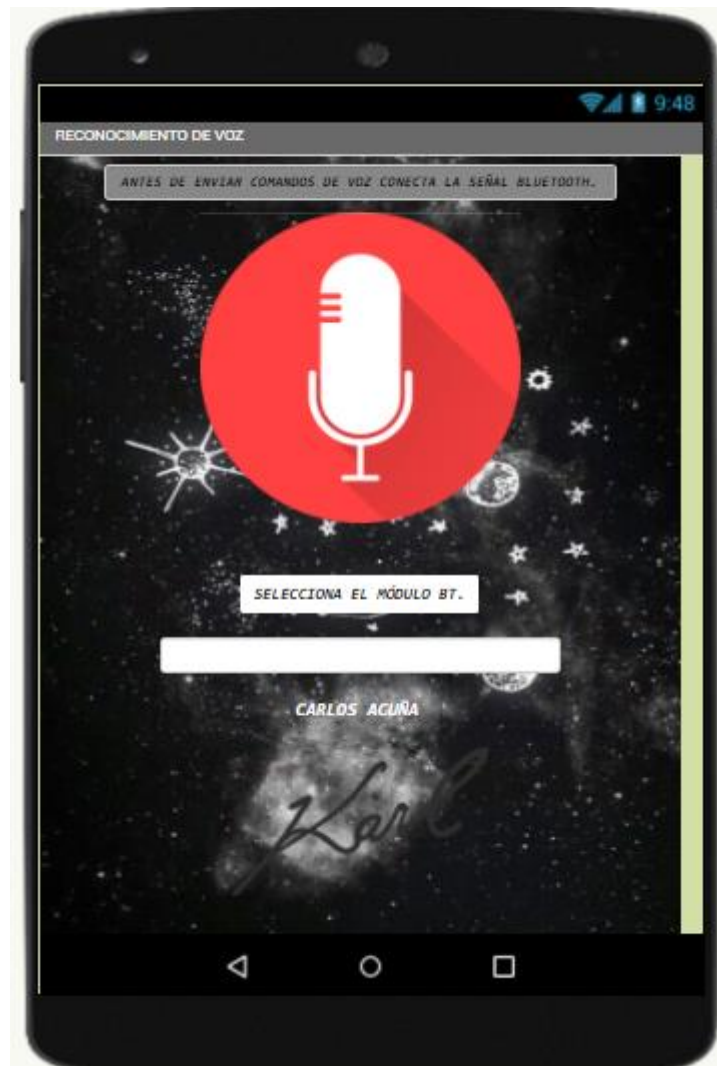
```
        delay(2000);lcd.clear();
    }
    while(veces>=3){
        lcd.setCursor(1,0),lcd.print("Alerta Policía");
        lcd.setCursor(4,1),lcd.print("Intrusos");
        digitalWrite(A1,HIGH);
        delay(100);
        lcd.clear();
        digitalWrite(A1,LOW);delay(50);
    }
    var=0;
    lcd.clear();
    break; }
}
if(!key){lcd.setCursor(0,0),lcd.print("Digite su Clave");}
delay(2);
}
```

3.2 Interfaz de la aplicación en APP Inventor

Para el control del proceso, se diseñó una aplicación en APP Inventor, como se muestra en la figura 15, la cual permitirá controlar el encendido y apagado de luces leds, la intensidad de los leds en 25% 50% y 70%, se controlará también los servomotores y el llenado y vaciado de una piscina en la simulación de la maqueta de una casa domótica.

Figura 15

Diseño de la aplicación en APP Inventor



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

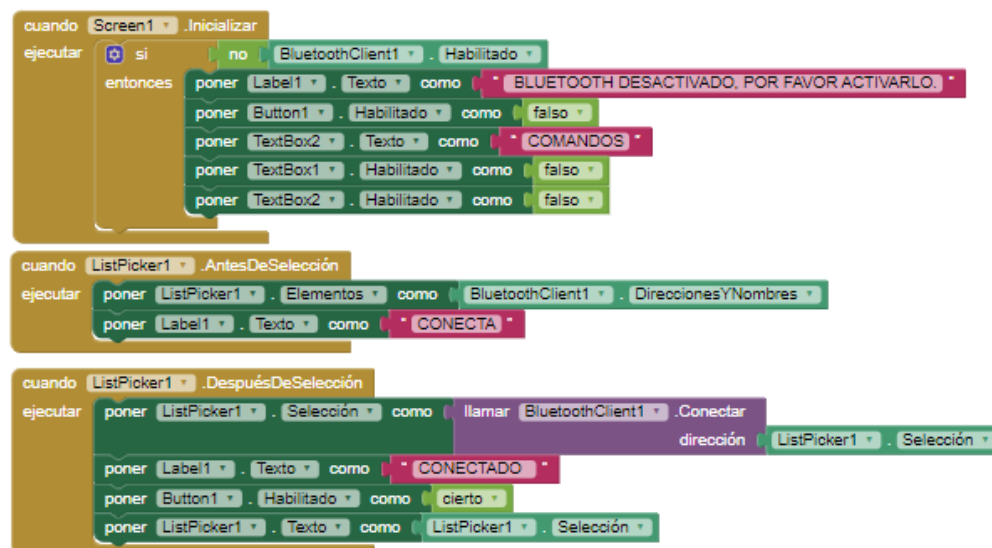
3.3 Programación en diagrama de bloques en APP Inventor para conectar el Módulo Bluetooth

Para el control mediante la aplicación móvil se realizó la siguiente programación en diagrama de bloques, como se observa en la figura 16. Se

realiza para la configuración de la aplicación con el módulo bluetooth HC-06, con ello se muestra si está previamente conectado y esto permite habilitar o deshabilitar la aplicación según su estado (True o False).

Figura 16

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



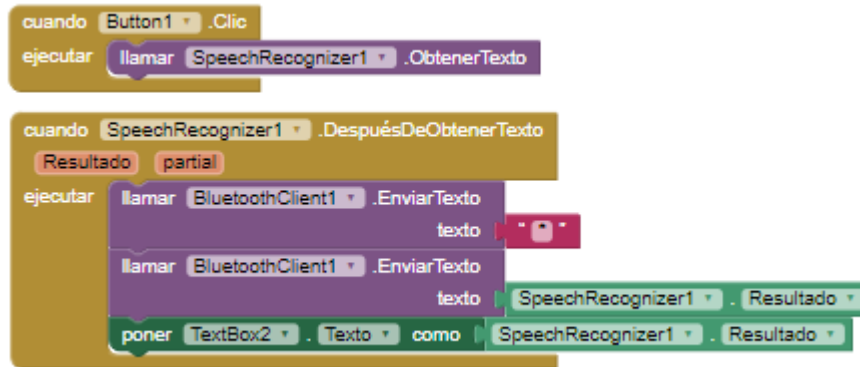
Nota. La imagen muestra la programación en diagrama de bloques en App Inventor para conectar y realizar la conexión con el módulo Bluetooth.

3.4 Programación en diagrama de bloques en APP Inventor para enviar comandos de voz

Se realizó la programación en diagrama de bloques para poder generar los comandos de voz en las lecturas de Arduino, como se muestra en la figura 17. La aplicación controla la intensidad luminosa de los leds, además el encendido y apagado de los mismos, también se controlará el movimiento de los servomotores y del llenado y vaciado de la piscina.

Figura 17

Programación en el diagrama de bloques de APP Inventor para generar comandos de voz



Nota. La imagen muestra la programación en diagrama de bloques en APP Inventor para que lea y genere comandos.

3.5 Implementación del prototipo de una casa domótica controlada mediante comandos de voz

Se realizó la construcción de un prototipo para una casa domótica en donde se utilizó diferentes materiales para la construcción, se tomó como referencia las medidas realizadas en AutoCAD, también se implementó pequeños detalles de decoración para cada habitación, a continuación, se muestra la maqueta en construcción en la figura 18.

Figura 18

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 por comandos de voz

3.6 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 comandos de voz, además se comprobó el funcionamiento de las mismas, con ello se pudo controlar el encendido y el apagado de iluminaria y lo que es la intensidad en 25%, 50% y 75%, como se puede observar en la figura 19.

Figura 19

Encendido y apagado de luces leds



Nota. La imagen muestra el encendido de luces leds efectuadas mediante comandos de voz en la aplicación de App inventor.

3.7 Implementación de los servomotores en la maqueta de una casa domótica

Se implementó los servomotores en la maqueta de la casa domótica controlada mediante comandos de voz, en donde se controló el movimiento de cada uno, dando como objetivo la interacción de las puertas (abiertas y cerradas). Además, se comprobó el funcionamiento de la misma. cómo se observa en la figura 20.

Figura 20

Control de movimiento de los servomotores



Nota. La imagen muestra el movimiento de los servomotores efectuados mediante el comando de voz respectivo.

3.8 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 por comandos de voz, se utilizó dos mini bombas conectados con relés de 5V para la comunicación con el Arduino mega. Además, se comprobó el funcionamiento de la misma. como se muestra en la figura 21.

Figura 21

Llenado y vaciado de la piscina



Nota. La imagen muestra el llenado y vaciado de una piscina efectuada mediante comandos de voz mencionados.

3.9 Implementación de electrodomésticos en la maqueta de una casa domótica

Se implementó el control de encendido y apagado de los diferentes electrodomésticos existentes en el prototipo domótico, simulando a través de leds que son controladas mediante comandos de voz, además en la figura 22 se puede observar el funcionamiento de la misma.

Figura 22

Encendido y apagado de electrodomésticos



Nota. La imagen muestra el encendido y apagado de los electrodomésticos implementados en el prototipo domótico a través de los respectivos comandos de voz

3.10 Implementación del acceso seguro de la puerta principal en la maqueta de una casa domótica

Se implementó el acceso seguro en la puerta principal donde se utiliza un teclado matricial 4x4 para digitar la clave o contraseña y un LCD donde muestra los mensajes propuestos dentro del prototipo domótico, además se comprobó el funcionamiento de la misma. Se puede observar el funcionamiento en la figura 21.

Figura 23

Acceso seguro de la puerta principal



Nota. La imagen muestra el acceso seguro de la puerta principal efectuada a través de una contraseña.

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 desarrolló un prototipo domótico para controlar mediante comandos de voz el encendido/apagado de luces, la intensidad de las mismas en 25%, 50% y 75%, controlar el llenado y vaciado de la piscina, controlar el movimiento de los servomotores y también el encendido y apagado de los diferentes electrodomésticos existe, mediante una comunicación bluetooth entre un dispositivo móvil y el módulo HC-06 de la tarjeta Arduino MEGA.
- El diseño de una aplicación en App Inventor puede variar de acuerdo a la necesidad y circunstancia que lo desee, ya que la página cuenta con diferentes funciones de personalización, en donde no solo podemos contar con comandos de voz, sino que también controlar a través de botones y deslizadores las diferentes funciones que propone el prototipo domótico.
- Dentro de la comunicación bluetooth nos permite acceder a diferentes funciones dentro de la electrónica, ya que es un método muy efectivo para la transferencia de datos. Es un tipo de conexión en el que puede llegar a unas velocidades muy altas de transmisión, bajo consumo de energía y, sobre todo, la mayoría de los dispositivos de control lo admiten.
- La comunicación inalámbrica y la interfaz de usuario en móviles influye directamente en su facilidad de uso y contribuye enormemente a crear aplicaciones mejores que ayuden a la sociedad a brindarle una mejor experiencia. La plataforma de App inventor y la plataforma de hardware Arduino es una de las herramientas más potentes para crear aplicaciones para Android permitiendo el ahorro de tiempo y diseño, permitiendo crear proyectos complejos que utilicen circuitos electrónicos.

4.2 Recomendaciones

- Es importante tener un conocimiento básico de electrónica para poder realizar la instalación de un prototipo domótico, también poder mejorar las aplicaciones de la domótica realizando estudios futuros en cuanto al control inalámbrico y otras opciones de comunicación remota, para no depender tanto del uso con conexiones a internet.
- 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.
- Es imprescindible seguir desarrollando nuevas tecnologías en base a sistemas móviles que sean creativas, útiles, eficientes y prácticas, ya que el uso de dispositivos con sistemas operativos Android aumenta día a día permitiendo la solución a mayor cantidad de problemas tecnológicos simplificando el uso de dispositivos con elementos de la vida cotidiana.

Bibliografía

- Aguilar, I. G. (31 de enero de 2018). *Iluminet*. Recuperado el 22 de 06 de 2021, de <https://www.iluminet.com/diodos-emisores-de-luz-de-alta-potencia-de-1w/>
- Arduino. (julio de 23 de 2018). *Arduino.cl*. Recuperado el 10 de 08 de 2021, de <https://store.arduino.cc/usa/mega-2560-r3>
- Arduino.cl*. (05 de mayo de 2017). Recuperado el 14 de 07 de 2021, de <https://arduino.cl/arduino-uno/>
- Asociación Programa Ergo Sum. (04 de septiembre de 2019). *Programo Ergo Sum*. Recuperado el 02 de 2021, de <https://www.programoergosum.com/cursos-online/appinventor/27-curso-de-programacion-con-app-inventor/primeros-pasos>
- AV Electronics. (11 de enero de 2021). 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>
- Barragan, B. M. (21 de mayo de 2017). *HeTPro*. Recuperado el 13 de 07 de 2021, de <https://hetpro-store.com/lcd-16x2-blog/>
- Carretero, O. S. (2016). *Casa Domótica con Arduino*. Recuperado el 21 de 06 de 2021, de <https://riunet.upv.es/bitstream/handle/10251/75797/SANCLEMENTE%20-%20DISE%C3%91O%20DE%20CASA%20DOM%C3%93TICA%20CONTROLADA%20OPOR%20ARDUINO.pdf?sequence=5>
- Electronilab*. (19 de noviembre de 2019). Recuperado el 19 de 02 de 2021, de <https://electronilab.co/tienda/micro-servo-9g-towerpro/>
- Ingeniería Mecafenix*. (25 de abril de 2017). Recuperado el 22 de 06 de 2021, de <https://www.ingmecafenix.com/electronica/arduino/>
- Naylamp Mechatronics SAC . (22 de febrero de 2018). *Naylamp Mechatronics*. Recuperado el 10 de 02 de 2021, de <https://naylampmechatronics.com/inalambrico/24-modulo-bluetooth-hc06.html>
- PrestaShop. (17 de enero de 2018). *Tosta Tronic*. Recuperado el 27 de 02 de 2021, de <https://tostatronic.com/store/placas/1925-arduino-mega-2560-rev3-7630049200067.html>

TECmikro. (14 de agosto de 2018). Recuperado el 23 de 06 de 2021, de <https://tecmikro.com/content/17-arduino-uno-r3-caracteristicas>

UNIT Electronics. (14 de diciembre de 2019). Recuperado el 14 de 07 de 2021, de <https://uelectronics.com/producto/teclado-matriz-4x4-membrana/>

WordPress.com. (27 de julio de 2018). *Aprendiendoarduino*. Recuperado el 25 de 01 de 2021, de <https://aprendiendoarduino.wordpress.com/2016/11/13/bluetooth-en-arduino/>