

ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

IMPLEMENTACIÓN DE UN SISTEMA DOMÓTICO CONTROLADO MEDIANTE UN SERVIDOR IP

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE
TECNÓLOGO EN ELECTRÓNICA Y TELECOMUNICACIONES**

Ronal Vinicio Cordonez Arguello

ronal.cordonez@epn.edu.ec

Angel Armando Vela Tituaña

angel.vela@epn.edu.ec

DIRECTOR: ING. ALAN DANIEL CUENCA SÁNCHEZ.

alan.cuenca@epn.edu.ec

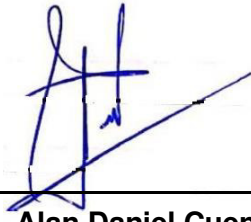
CODIRECTOR: ING. FABIO MATIAS GONZÁLEZ GONZÁLEZ.

fabio.gonzalez@epn.edu.ec

Quito, octubre 2021

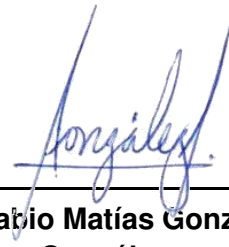
CERTIFICACIÓN

Certificamos que el presente trabajo fue desarrollado por los Sres. Cordonez Arguello Ronal Vinicio y Vela Tituaña Angel Armando como requerimiento parcial a la obtención del título de TECNÓLOGO EN ELECTRÓNICA Y TELECOMUNICACIONES, bajo nuestra supervisión:



**Ing. Alan Daniel Cuenca
Sánchez**

DIRECTOR DEL PROYECTO



**Ing. Fabio Matías González
González**

CODIRECTOR DEL PROYECTO

DECLARACIÓN

Nosotros, Cordonez Arguello Ronal Vinicio con CI: 1725176232 y Vela Tituaña Angel Armando con CI: 1723476618 declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y que hemos consultado las referencias bibliográficas que se incluyen en este documento.

Sin perjuicio de los derechos reconocidos en el primer párrafo del artículo 144 del Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación – COESC-, somos titulares de la obra en mención y otorgamos una licencia gratuita, intransferible y no exclusiva de uso con fines académicos a la Escuela Politécnica Nacional.

Entregamos toda la información técnica pertinente, en caso de que hubiese una explotación comercial de la obra por parte de la EPN, se negociará los porcentajes de los beneficios conforme lo establece la normativa nacional vigente.



**Cordonez Arguello Ronal
Vinicio**



**Vela Tituaña Angel
Armando**

DEDICATORIA

Dedico el presente trabajo a mis padres Vinicio y Norma, que con su apoyo incondicional me ayudaron a seguir adelante, también a todas las personas que me apoyaron y me animaron a lo largo de toda mi formación profesional.

También a los docentes de la Escuela de Formación de Tecnólogos, que se esforzaron por impartirme sus conocimientos y velaron por mi educación y que en todo el tiempo que fui su estudiante creyeron en mí.

Ronal Cordonez

AGRADECIMIENTO

Agradezco a Dios, por darme la suficiente fuerza para afrontar los obstáculos que se presentaron en mi camino y alcanzar mis objetivos.

A mi madre por siempre estar para mí cuando más la necesite, gracias a sus consejos y sabiduría.

A mi padre por darme siempre su apoyo y fuerza para afrontar mis problemas, gracias a su preocupación y esfuerzo por ayudarme a conseguir mis metas

A Belén que, con su amor incondicional, confianza y paciencia me ayudó a levantarme cuando más derrotado estaba, gracias porque siempre se preocupó en que yo sea la mejor versión de mí.

A mi familia que siempre vieron por mi bienestar y se preocuparon por verme alcanzar mis sueños.

A los docentes de la Escuela de Formación de Tecnólogos por su esmero y dedicación al momento de impartir sus clases, por la paciencia que me brindaron día a día y por las enseñanzas que me dejaron.

Al Ingeniero Alan Cuenca, mi director de tesis, por siempre creer en mí y ayudarme a esforzarme cada día más, gracias por su tiempo y apoyo para la realización de este proyecto.

Ronal Cordonez

DEDICATORIA

Este trabajo está dedicadísimo a mi madre porque quiero agradecerle todo el esfuerzo que hace por mí y porque estoy orgulloso de ella.

Gracias por ser parte de mis sueños, por enseñarme que lo imposible solo cuesta un poco más, y que las mejores metas son las que cuestan alcanzar. Gracias por enseñarme que mientras haya voluntad nada es complicado y aunque te tropieces, hay que saberse levantar con más fuerza y volver a empezar.

VA POR TI ESPERANCITA.

Angel Vela

AGRADECIMIENTO

Quiero agradecer a mi familia, que, aunque a veces no los soporte y me maten de iras, también les debo mucho. Me enseñaron el valor del trabajo duro, a no rendirme y más cuando se desea llegar a la meta, a ver el lado divertido de las pequeñas cosas y que tener el mundo de cabeza no es tan malo como parece.

Gracias ñaña por nunca dejar de creer en mí, aunque yo ya lo había hecho y, sobre todo, escuchar de vez en cuando un "tú puedes guambra" es lo mejor que te pueden decir. Gracias por ser cómplice de todas las tonteras que me pasan y por acompañarme en todo momento.

También agradecer a mis amigos por ser un apoyo durante este largo camino, a la segunda familia que sin querer con culpa encontré. Gracias por enseñarme que ante todo la responsabilidad, sin embargo, siempre se tiene tiempo para unas bielititas para el estrés no más.

Gracias a la Escuela Politécnica Nacional por las noches de tortura que han acompañado a mi aventura, por los días felices y grises, por todo lo aprendido y los gratos recuerdos que llevo conmigo.

Angel Vela

ÍNDICE DE CONTENIDOS

1	Introducción	1
1.1	Objetivo general	1
1.2	Objetivos específicos.....	1
1.3	Fundamentos.....	2
	Domótica	2
	000WebHost.....	2
	NodeMCU.....	3
	“Thingspeak”.....	4
2	Metodología.....	5
2.1	Descripción de la metodología usada	5
3	Resultados y Discusión.....	7
3.1	Infraestructura del sitio de instalación	7
3.2	Servidor IP.....	10
3.3	Sistema de control.....	13
	Etapa 1: Módulo Central	13
	Etapa 2: Control de Acceso	16
	Etapa 3: Accionamiento de Cortina.....	20
	Sistema de Seguridad.....	22
	Placas PCB	23
3.4	Interfaz de control y supervisión	27
	Medio de identificación	28
3.5	Pruebas y Análisis de Resultados.....	29
	Prueba de Etapa 1: Módulo Central.....	30
	Prueba de Etapa 2: Control de Acceso	32
	Prueba de Etapa 3: Accionamiento de cortina	34
	Prueba de la Cámara de Seguridad.....	34
	Prueba final de todo el sistema.....	35
	Análisis antes y después de la instalación del sistema	36

Comparación del sistema implementado frente a otros existentes en el mercado	37
3.6 Manual de Uso y Mantenimiento.....	38
4 Conclusiones y Recomendaciones	39
4.1 Conclusiones	39
4.2 Recomendaciones	40
5 Referencias Bibliográficas	42
ANEXOS.....	46
ANEXO 1: CERTIFICADO DE FUNCIONAMIENTO.....	47
ANEXO 2: PLANO ESQUEMÁTICO	49
ANEXO 3: DIAGRAMAS DE CIRCUITOS	51
ANEXO 4: DIAGRAMAS DE FLUJO	55
ANEXO 5: COSTO TOTAL DEL PROYECTO	59

ÍNDICE DE FIGURAS

Figura 1.1	Muestra gráfica de domótica	2
Figura 1.2	000WebHost	3
Figura 1.3	Pines del "nodeMCU ESP 8266"	3
Figura 1.4	"Thingspeak"	4
Figura 3.5	Ubicación del Módulo central y Módulo de Cortina.....	7
Figura 3.6	Ubicación de Circuito de Iluminación de Sala.....	8
Figura 3.7	Ubicación de Circuito de Iluminación de Comedor	8
Figura 3.8	Ubicación de Circuito de Iluminación de Pasillo	8
Figura 3.9	Ubicación de Cerradura de la Ventana.....	9
Figura 3.10	Ubicación del Módulo de la Puerta	9
Figura 3.11	Ubicación del Módulo RFID.....	9
Figura 3.12	Ubicación de motor para la cortina	10
Figura 3.13	Ubicación de la Cámara de Seguridad	10
Figura 3.14	Documentos para la página web	11
Figura 3.15	Datos enviados por "Local Storage"	11
Figura 3.16	Código "Local Storage"	12
Figura 3.17	Función "Ajax"	12
Figura 3.18	Código para el Field y API.....	12
Figura 3.19	Configuración inicial del código para módulo central.....	13
Figura 3.20	Visualización del entorno de "Thingspeak"	13
Figura 3.21	Intento de conexión a red WiFi.....	14
Figura 3.22	Conexión con "Thingspeak"	15
Figura 3.23	Cerradura Electromagnética.....	15
Figura 3.24	Pines de conexión del RFID	16
Figura 3.25	"Tags" de RFID en forma de tarjeta y llavero.....	16
Figura 3.26	Pantalla LCD I2C	17
Figura 3.27	Esquema de conexión de RFID con Arduino	18
Figura 3.28	Prevención de casos especiales para el RFID	18
Figura 3.29	Obtención y almacenamiento del UID del "tag"	19
Figura 3.30	Comparación de códigos obtenidos con códigos registrados	19
Figura 3.31	Circuito integrado L293D.....	20
Figura 3.32	Caja Protectora de Motor	22
Figura 3.33	Zona de vigilancia	22
Figura 3.34	Cámara de seguridad HikVision	23

Figura 3.35	Protección de Cámara de Seguridad.....	23
Figura 3.36	Diseño PCB del circuito del módulo central.....	24
Figura 3.37	Diseño PCB del circuito para el control de la cortina	24
Figura 3.38	Circuito del módulo central.....	24
Figura 3.39	Circuito para el control de la cortina	25
Figura 3.40	Caja del módulo central.....	25
Figura 3.41	Interior de la caja con el circuito del módulo central	25
Figura 3.42	Caja del circuito de control de acceso	26
Figura 3.43	Interior de la caja con el circuito del control de acceso.....	26
Figura 3.44	Caja del módulo RFID y pantalla LCD.....	26
Figura 3.45	Documento CSS	27
Figura 3.46	Interfaz de usuario	28
Figura 3.47	Código para datos de acceso.....	28
Figura 3.48	Interfaz para identificación.....	29
Figura 3.49	Conexión entre "Thingspeak" y la página web.....	29
Figura 3.50	Monitor Serial del módulo nodeMCU.....	30
Figura 3.51	Módulo central y módulos relés.....	30
Figura 3.52	Circuito de iluminación del comedor.....	31
Figura 3.53	Circuito de iluminación de la sala	31
Figura 3.54	Circuito de iluminación del pasillo.....	31
Figura 3.55	Cerradura de la ventana.....	32
Figura 3.56	Módulo de control de acceso.....	32
Figura 3.57	Cerradura de la puerta	33
Figura 3.58	Sistema de Identificación RFID	33
Figura 3.59	Motor de la cortina	34
Figura 3.60	Cámara de Seguridad	35
Figura 3.61	Prueba de Funcionamiento de Cámara de Seguridad.....	35
Figura 3.62	Código QR del video de pruebas de funcionamiento.....	36
Figura 3.63	Gráfica de valores de planillas eléctricas.....	37
Figura 3.64	Información de consumo EEQ.....	37
Figura 3.65	Código QR del video de funcionamiento y mantenimiento	39

ÍNDICE DE TABLAS

Tabla 3.1 Tabla de verdad de estados posibles con el integrado L293D	21
Tabla 3.2 Pruebas de funcionamiento de la etapa 1: Módulo Central	32
Tabla 3.3 Pruebas de funcionamiento de la etapa 2: Control de acceso.....	33
Tabla 3.4 Pruebas de funcionamiento de la etapa 3: Accionamiento de cortina	34
Tabla 3.5 Historial de costos mensuales de consumo eléctrico	36
Tabla 3.6 Comparación de precios	38

RESUMEN

Mediante el presente proyecto se desarrolló un sistema domótico, cuya finalidad es brindar seguridad, confort y permitir un ahorro energético dentro de un hogar. El sistema permite controlar el encendido y apagado de luces, el bloqueo y desbloqueo de la puerta principal, así como también de una ventana, la apertura y cierre de una cortina y el monitoreo del hogar mediante una cámara de seguridad.

El presente documento detalla el proceso de implementación y el desarrollo de un servidor IP basado en una interfaz humano-máquina (*HMI*) que permite controlar todo el sistema. El usuario tendrá la posibilidad de ingresar a este servidor IP de manera remota sin importar la forma cómo se encuentre conectado a Internet, ya sea desde una computadora de escritorio, laptop o incluso desde un dispositivo móvil (celular). Los capítulos se detallan a continuación:

En el capítulo de introducción se puntualizan las razones por las cuales es necesario la implementación de este proyecto, también se exponen los objetivos que fueron alcanzados a lo largo de todo el proceso de implementación.

En el capítulo de metodología se explica de manera sencilla la metodología empleada para cumplir con los objetivos establecidos.

En el capítulo de resultados y discusión, se muestra el proceso de instalación del sistema domótico, además se explica el cumplimiento de los objetivos propuestos con sus respectivos resultados.

En el capítulo de conclusiones y recomendaciones se expone, en base a los resultados obtenidos, un análisis del sistema domótico con énfasis en sugerencias que ayudarían a mejorar el proyecto y sacar el máximo provecho de sus utilidades.

En la sección de bibliografía se listan las fuentes utilizadas para llevar a cabo el proyecto. Finalmente, en la sección de anexos se expone información complementaria sobre el sistema.

PALABRAS CLAVE: *HMI*, domótica, servidor IP

ABSTRACT

The purpose of this project is to develop a domotic system to provide security, comfort and energy savings in a home. The system allows controlling the turning on and off of lights, locking and unlocking of the main door, as well as a window, the opening and closing of a curtain and the monitoring of the home through a security camera.

This document details the implementation process and the development of an IP server based on a human-machine interface (HMI) to control the entire system. The user will be able to access this IP server remotely regardless of the Internet network to which he is connected, either from a desktop computer, laptop or even from a mobile device (cell phone). The chapters are detailed below:

In the introduction chapter the reasons why the implementation of this project is necessary are pointed out, also the objectives that were achieved throughout the implementation process are exposed.

In the methodology chapter, the methodology used to meet the established objectives is explained in a simple manner.

In the results and discussion chapter, the installation process of the domotic system is shown, as well as the fulfillment of the proposed objectives and their respective results.

In the conclusions and recommendations chapter, based on the results obtained, an analysis of the domotic system is presented with emphasis on suggestions that would help to improve the project and make the most of its utilities.

The bibliography section lists the sources used to carry out the project. Finally, in the annexes section, complementary information about the system is presented.

KEYWORDS: HMI, domotic, IP server

1 INTRODUCCIÓN

En los últimos meses con el inicio de la pandemia a nivel mundial, todas las personas se vieron obligadas a permanecer en sus hogares. La cuarentena provocó que los planteles educativos y negocios que no proporcionan servicios de primera necesidad y movilidad sean suspendidos [1].

Esto ha provocado un incremento considerable en el consumo de energía eléctrica, lo cual se puede ver reflejado en el valor a pagar en las planillas de la empresa de energía eléctrica en cada hogar. Otro punto importante, es que en la mayoría de los hogares no se cuenta con sistemas o dispositivos que permitan un ahorro energético, debido a su costo. Estos sistemas en la actualidad se han convertido en una necesidad antes que un privilegio, debido a las diversas tareas que se pueden realizar con los mismos. La domótica, aparte de ofrecer un ahorro de energía eléctrica también brinda seguridad y confort para el núcleo familiar, facilitando la realización de varias actividades. [2] Con respecto a la parte de seguridad, es importante resaltar que una cámara de video vigilancia será siempre una de las mejores opciones para poder incrementar dicha seguridad.

Para dar solución al problema planteado se propuso implementar un sistema domótico utilizando dispositivos económicos presentes en el mercado que permitan establecer un consumo reducido de energía eléctrica, además de brindar seguridad y confort en el hogar.

Cabe mencionar que el proyecto se encuentra instalado en una residencia seleccionada con anterioridad y analizada para instalar el sistema en partes estratégicas con el fin de lograr niveles adecuados de confort, seguridad y bienestar a los miembros del hogar. En el Anexo 1 se encuentra el certificado que valida el funcionamiento correcto del sistema domótico.

1.1 Objetivo general

Implementar un sistema domótico controlado mediante un servidor IP.

1.2 Objetivos específicos

- Analizar la infraestructura del lugar en donde se realizará la instalación del sistema domótico.
- Diseñar un servidor IP que controle todo el sistema domótico.

- Elaborar un sistema de control basado en *Arduino*.
- Desarrollar una interfaz para control y monitoreo del sistema.
- Verificar el funcionamiento óptimo de todo el sistema.

1.3 Fundamentos

Domótica

Un sistema domótico tiene la capacidad de recolectar información que proviene de entradas, procesarla y dar órdenes a unos actuadores o salidas [3], como se puede observar en la Figura 1.1. La domótica mejora en gran medida el confort y la seguridad del usuario haciendo más fácil su vida cotidiana.

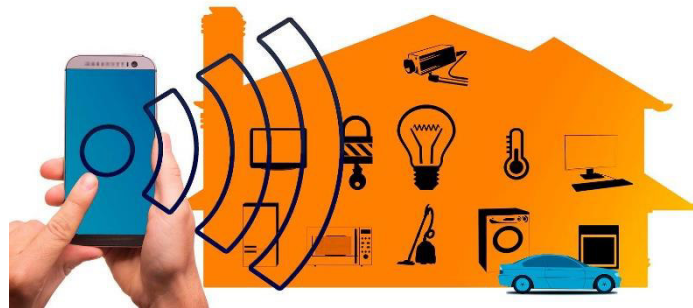


Figura 1.1 Muestra gráfica de domótica [3]

000WebHost

Es una plataforma de alojamiento web gratuito o pagado, que permite publicar un sitio web en internet (Ver Figura 1.2), permitiendo el acceso desde cualquier lugar y cualquier red a la que se esté conectado [4]. La plataforma ofrece varios beneficios dependiendo qué tipo de cuenta tenga. Los beneficios que ofrece la página al contar con una cuenta gratuita son los siguientes [5]:

- Memoria: 300 MB
- Ancho de banda: 3 GB
- Disponibilidad: 99%

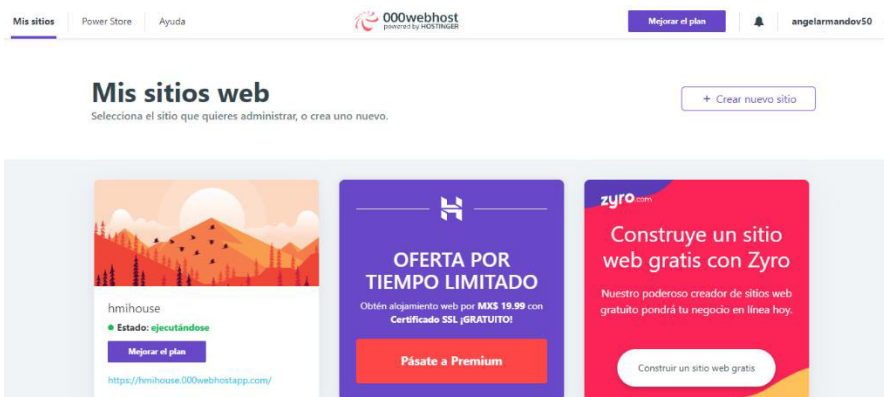


Figura 1.2 000WebHost [5]

NodeMCU

Es una placa de desarrollo bastante utilizada para proyectos *IoT* (*Internet of Things*) y domótica. Esta placa posee un conversor *USB-Serial TTL*, que posibilita programarla directamente desde el *software Arduino IDE*; la placa con la distribución de pines se observa en la Figura 1.3. Algunas de las especificaciones técnicas más importantes son [6] [7]:

- Voltaje de alimentación: 3,3 (V_{DC})
- Frecuencia de reloj: 80 (MHz)
- Pines digitales *GPIO*: 17 (4 pines pueden configurarse como *PWM* a 3,3 (V))
- Pin analógico *ADC*: 1
- Características de conectividad *WiFi*:
 - Protocolos de *WiFi*: 802.11 b/g/n
 - Frecuencia: 2,4 (GHz)

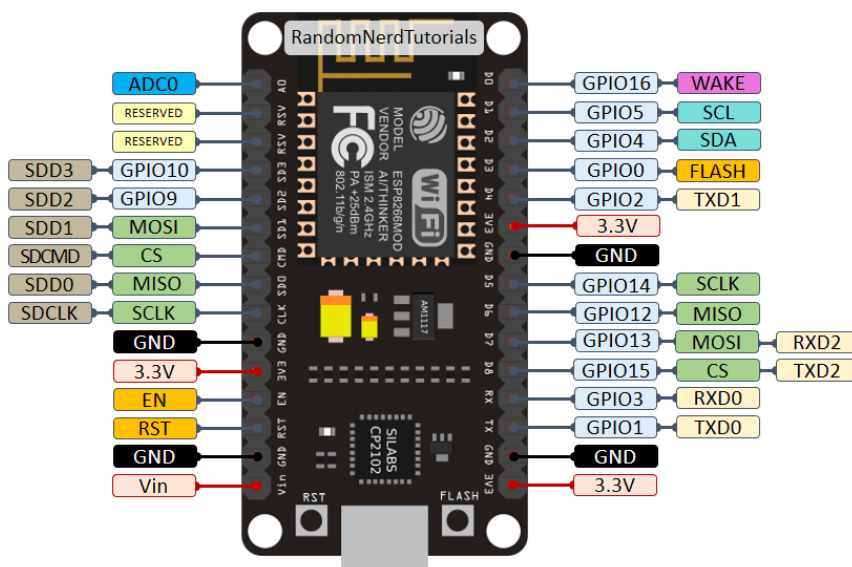


Figura 1.3 Pines del "nodeMCU ESP 8266" [27]

“Thingspeak”

“Thingspeak” es el conjunto de una base de datos, servidor web y *API*, capaz de ofrecer un servicio que se usa especialmente para recopilar y almacenar datos permitiendo crear sistemas domóticos y sirviendo como puente para integrar dispositivos capaces de comunicarse con Internet como se muestra en la Figura 1.4 [8] [9], mediante el uso del protocolo *HTTP* (*Hyper Text Transfer Protocol*).



Figura 1.4 “Thingspeak” [28]

2 METODOLOGÍA

2.1 Descripción de la metodología usada

En la etapa inicial del proyecto se realizó una serie de visitas al hogar donde se instaló el sistema de control, esto con el fin de identificar posibles problemas y qué elementos se pueden automatizar y cuáles no. En base al análisis preliminar, se realizaron los respectivos planos esquemáticos utilizando el *software* de diseño “AutoCad” [10], los planos generados fueron pieza clave para identificar puntos estratégicos para la instalación de los diferentes circuitos de control. Dentro de los planos se exponen las conexiones de los circuitos de iluminación, circuito de seguridad y todos los circuitos que conforman el sistema domótico, esto con el objetivo de facilitar futuros mantenimientos o mejoramientos que se puedan realizar.

Posteriormente se creó una cuenta en la plataforma “Thingspeak”, con la cual se puede hacer uso de un servidor para enviar información y comunicarse con la interfaz que ayuda a comandar todo el sistema [11]. Para la creación de la página web se hizo uso de la plataforma 000WebHost, esta última sirve para generar páginas web con diferentes aplicaciones útiles para el propósito del proyecto [12], donde se registró una cuenta con la cual se puede alojar una página web. Una vez creada la página web, también se obtiene un nombre de dominio único para poder ingresar a la interfaz creada.

Cumpliendo con la etapa de diseño, se utilizó la plataforma *Arduino*, no solo por el *software* que ofrece, sino por el *hardware* disponible en el mercado. Se usó una “shield” de *Arduino* denominada “nodeMCU ESP 8266” [13], cuya función es conectar, vía “WiFi”, la placa con Internet, de esta manera es posible enviar órdenes desde el servidor creado, para esto, se utilizó “Thingspeak” como intermediario, ya que brinda un servicio de base de datos. Otra razón por la que se decidió utilizar este intermediario, es por el fácil uso de la librería “ThingSpeak.h” en el código y así ordenar a la placa la comunicación directa con el servidor y este a su vez, con la página web.

Se procedió a desarrollar la interfaz que controla todo el sistema instalado, como ya se mencionó, una de las herramientas planteadas para este proyecto es una página web, misma que, al ingresar en el buscador de preferencia, despliega un entorno gráfico que contiene botones que están orientados a controlar partes específicas del sistema domótico previamente programadas. Cabe recalcar que este entorno fue desarrollado para ser lo más amigable e intuitivo para que el usuario pueda manejar, monitorear y supervisar el sistema de control.

Una vez instalado el sistema, se comprobó si cada componente funciona correctamente de acuerdo con las especificaciones y requerimientos establecidos. Para la comprobación del correcto funcionamiento del sistema se realizaron pruebas de comunicación entre cada *Arduino*, pruebas de control, monitoreo y supervisión de forma física y remota. Estas pruebas se desarrollaron a lo largo del proceso de instalación.

Finalmente, se realizaron dos videos en los cuales se incluyen el manual de usuario y mantenimiento respectivamente. Dichos manuales indican las características principales del sistema para su correcta utilización, además se mencionan las principales recomendaciones para un mantenimiento adecuado.

3 RESULTADOS Y DISCUSIÓN

Se implementó un sistema domótico capaz de controlar ciertos dispositivos que han permitido automatizar el control y el funcionamiento de varios elementos del hogar, con el fin de mejorar la gestión energética, la seguridad, la comunicación y el bienestar en general.

Con este sistema es posible realizar acciones como: encender y apagar luces, bloquear y desbloquear la puerta y ventana principal, abrir y cerrar una cortina y visualizar la imagen que entrega una cámara de video vigilancia. Todo el sistema fue realizado en base al *software* y *hardware* de *Arduino* con la ayuda de páginas intermediarias para posibilitar la comunicación con Internet, así como la creación del servidor.

3.1 Infraestructura del sitio de instalación

Para iniciar con el proyecto, se visitó la residencia donde se instaló el sistema para analizar y determinar el estado de los lugares en donde se deben colocar los controladores y los elementos a controlar. En base a este análisis se establecieron aspectos importantes a tomar en cuenta como la alimentación eléctrica de los componentes, acceso, configuración, ruta de los cables, y principalmente colocar los elementos y controladores del sistema (Ver Figura 3.5 hasta Figura 3.13).



Figura 3.5 Ubicación del Módulo central y Módulo de Cortina

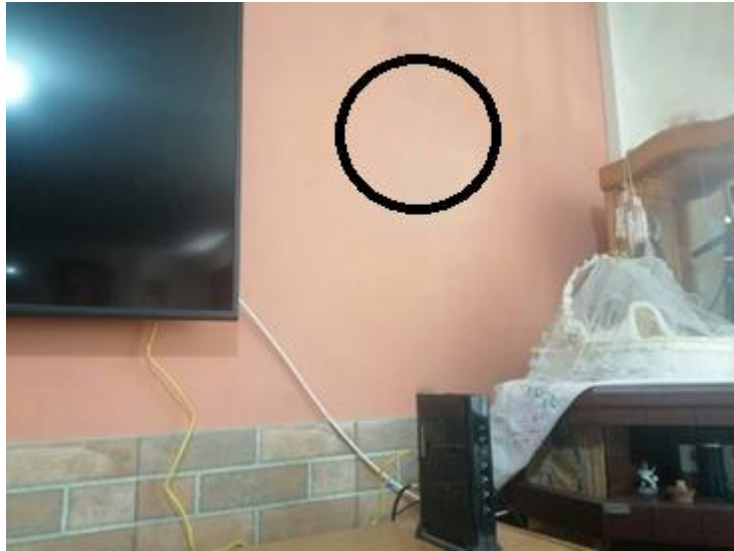


Figura 3.6 Ubicación de Circuito de Iluminación de Sala



Figura 3.7 Ubicación de Circuito de Iluminación de Comedor

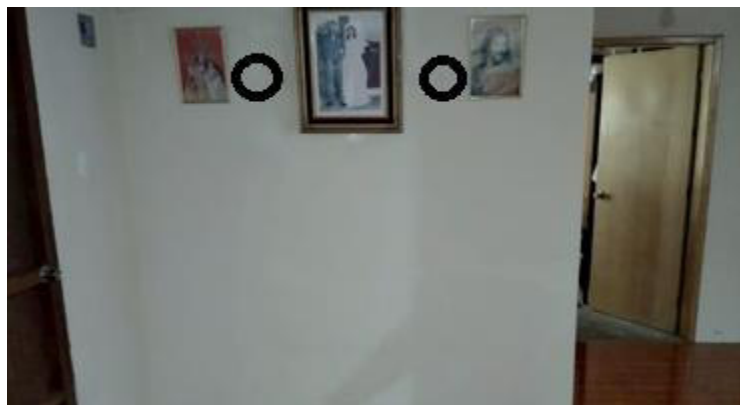


Figura 3.8 Ubicación de Circuito de Iluminación de Pasillo



Figura 3.9 Ubicación de Cerradura de la Ventana



Figura 3.10 Ubicación del Módulo de la Puerta



Figura 3.11 Ubicación del Módulo *RFID*



Figura 3.12 Ubicación de motor para la cortina



Figura 3.13 Ubicación de la Cámara de Seguridad

A través del software “*AutoCad*”, se diseñó el plano esquemático de la vivienda con la distribución y ubicación de los componentes y conexiones del sistema domótico, este plano se encuentra en el Anexo 2.

Con la ayuda de este plano se pudieron determinar los puntos estratégicos para cada etapa del sistema ya que este tiene que satisfacer las necesidades del usuario en cuanto a iluminación, seguridad y control de acceso.

3.2 Servidor IP

Se creó una página web mediante el servicio de hosting web que ofrece la página *000Webhost*. Para esto se subieron los archivos necesarios para crearla, como el archivo

HTML (“HyperText Markup Language”) y el archivo “JavaScript”, los cuales se muestra en la Figura 3.14.

En la URL <https://bit.ly/3AZNDKv> se encuentra el código HTML de la página web y en la URL <https://bit.ly/3oisHep> se muestran los códigos en “JavaScript” que se usó.

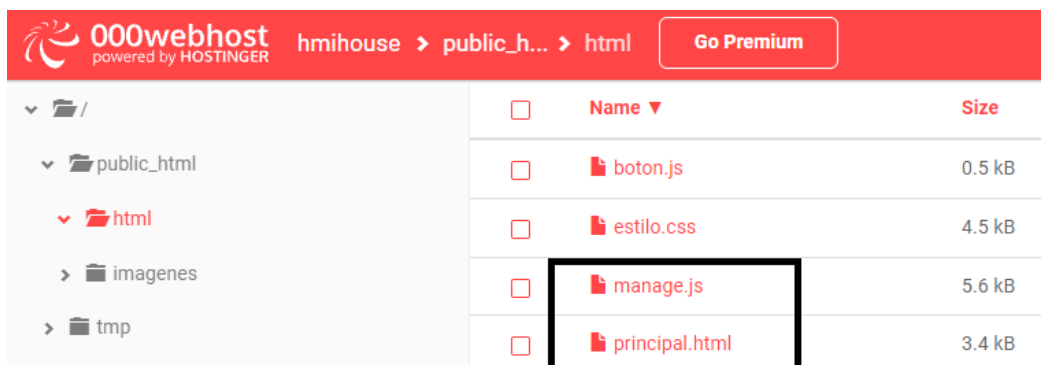


Figura 3.14 Documentos para la página web

HTML es la estructura básica a la hora de la creación de las páginas web en donde se precisa los elementos que va a tener como imágenes, texto, videos, entre otros [14]; además, con esta estructura inicial se pueden crear páginas webs más dinámicas aplicando estos cambios a los documentos HTML y también se pueden controlar elementos de la web como el “Local Storage” mediante el lenguaje de programación “JavaScript” [15]. Una vez creada la página web, esta envía datos dependiendo el caso, los cuales se van a alojar en el “Local Storage” del navegador que se use, como se muestra en la Figura 3.15.

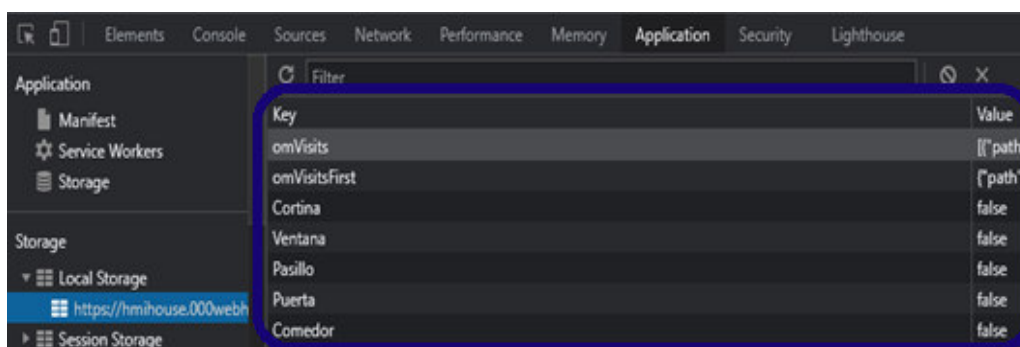


Figura 3.15 Datos enviados por “Local Storage”

Mediante el código de la Figura 3.16, se envía una palabra clave y un valor, los cuales se almacenan en el “Local Storage” dependiendo qué botón se seleccione, y se actualizará cada vez que el valor cambie [14].

```

$("#OnComedor").click(function(){
    localStorage.setItem("Comedor",true);
    console.log("Boton2 Precionado");
    ajax("field2=1");
});
$("#OffComedor").click(function(){
    localStorage.setItem("Comedor",false);
    console.log("Boton2 Precionado");
    ajax("field2=0");
});

```

Figura 3.16 Código “Local Storage”

El “Local Storage” forma parte de los mecanismos proporcionados por las API de almacenamiento web (Interfaz de Programación de Aplicaciones) y permite almacenar información dentro de cualquier navegador web durante un tiempo indefinido sin importar que el navegador se cierre, permitiendo reducir la pérdida de información y reduciendo el tráfico de red [5]. Estas API son el conjunto de protocolos y definiciones para integrar el software de diversas aplicaciones, estableciendo comunicación entre los sistemas para realizar diversas aplicaciones [16]. El API que se maneja en la plataforma “Thingspeak” permite recolectar, almacenar y visualizar la información recibida [17]. También se utilizó la función Ajax (ver Figura 3.17) para realizar el intercambio de datos que se almacenaron en el “Local Storage” con la plataforma “Thingspeak”, con lo que se consigue que los datos se actualicen de forma asíncrona, es decir que solo se van a actualizar partes específicas de la página web sin tener la necesidad de recargar todo el sitio [18].

```

function ajax(value){
    console.log("Entro a funcion Ajax");
}

```

Figura 3.17 Función “Ajax”

HTTP es el protocolo que se encarga de comunicarse con el servidor web y por este medio acceder a la página web [19]. Cuando se crea “Thingspeak” se generan “APIs” para lectura y escritura de los distintos canales que se tenga [9]; la API que se va a utilizar es la API de escritura como se muestra en la Figura 3.18, la cual permite escribir los datos alojados en el “Local Storage”.

```

$.ajax({
    type: 'POST', //Recepcion GET de HTML
    //URL del servidor thingspeak para recoger datos
    url: 'https://api.thingspeak.com/update?api_key=WG1JKPSCRJA9HJA'+value,

```

Figura 3.18 Código para el Field y API

3.3 Sistema de control

Para desarrollar el sistema de control con placas *Arduino* y “*shields*” se decidió dividir este sistema en tres etapas diferentes, tomando en cuenta que posteriormente se comunicarán entre sí para cumplir con los objetivos de control planteados y un sistema de seguridad conformado por una cámara de video vigilancia. Estas tres etapas son: módulo central, control de acceso y accionamiento de cortina.

Etapa 1: Módulo Central

Para esta etapa se usó el “*nodeMCU v3 ESP8266 WiFi*”, el cual se programa mediante la librería “*ESP8266WiFi.h*”, misma que no viene incluida por defecto en *Arduino IDE*. Otra librería usada fue “*ThingSpeak.h*”, misma que ayuda a la conectividad directa con la plataforma del mismo nombre, cuyo objetivo es analizar el flujo de datos en la nube [9]. En la Figura 3.19 se aprecia un ejemplo del entorno gráfico de “*Thingspeak*”.

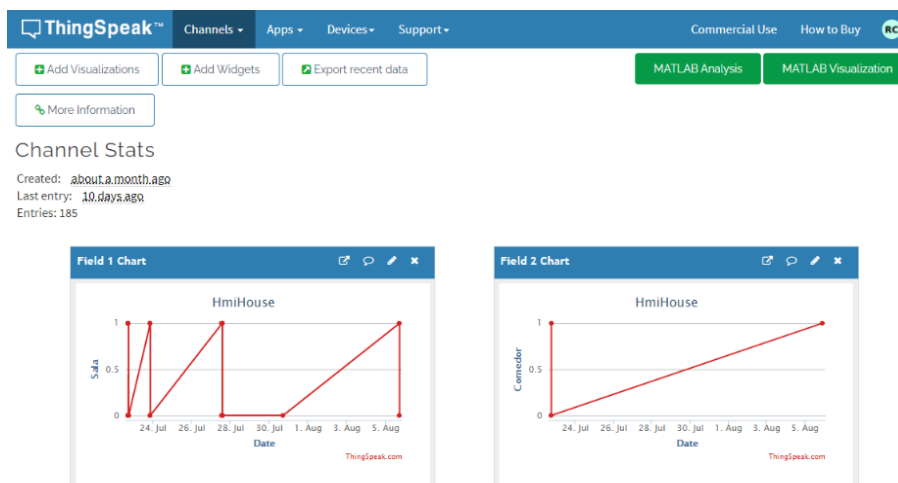


Figura 3.19 Configuración inicial del código para módulo central

Estructura del código de programación

Como ya se ha mencionado, para la estructura del código de la etapa central se utilizó las dos librerías ya descritas. Después de incluir las librerías al código se inicia un cliente *WiFi* para poder empezar con la configuración de las credenciales de la red *WiFi* a la que se desee conectar, como se muestra en la Figura 3.20.

```
20 #include <ESP8266WiFi.h>
21 #include "ThingSpeak.h"
22 char ssid[] = "Nombre de la red (SSID)";
23 char pass[] = "CONTRASEÑA";
24 WiFiClient cliente;
```

Figura 3.20 Visualización del entorno de “*Thingspeak*”

Después de realizar las configuraciones iniciales e identificando cuáles puertos se utilizarán como salidas, se procede a conectar el módulo a Internet. Para este propósito se utilizan las credenciales ya establecidas previamente, además se utilizaron los comandos que incluye la librería del módulo *WiFi*. Para comprobar si se realizó la conexión a Internet se imprime en el monitor serial la confirmación de este y además se coloca un ciclo de reconexión para el módulo *WiFi*, en caso de que no pueda conectarse al primer intento, este proceso se puede observar en la Figura 3.21.

```
if(WiFi.status() != WL_CONNECTED){
  Serial.print("INTENTANDO CONECTANDO A LA RED:");
  Serial.println(ssid);
  //ESTABLECEMOS UN CICLO DE ESPERA
  while(WiFi.status() != WL_CONNECTED){
    WiFi.begin(ssid, pass);
    Serial.print(".");
    delay(5000);
  }
  Serial.println("\nCONECTADO A:");
  Serial.println(ssid);
}
```

Figura 3.21 Intento de conexión a red *WiFi*

Una vez que se haya logrado conectar a una red *WiFi*, se procede con la conexión a la plataforma “*Thingspeak*”, para esto, se debe ingresar a dicha plataforma y registrar algunos datos que ayudarán a la conexión.

En primer lugar, se debe localizar el código único del canal, nombre compuesto por números que identifica un canal, y a qué “*field*” se desea apuntar, esta última es una subsección del canal que se haya creado, se puede tener tantas como sean necesarias, dependiendo del plan y beneficios que brinda la plataforma de manera gratuita.

Recolectados estos datos, se procede a colocarlos en el programa para iniciar la comunicación con la plataforma, y el dato obtenido se almacena en una variable, para la comprobación de la correcta llegada del dato se usa la sentencia “*if*”. Si la respuesta del comprobador es 200, se confirma que el dato llegó satisfactoriamente, caso contrario se vuelve a solicitar el dato, este proceso se observa en la Figura 3.22. La respuesta de comprobación 200 es un código de estado de respuesta *HTTP* que indica que la solicitud de conexión entre la placa “*nodeMCU*” y el servidor “*Thingspeak*” fue exitosa [20].

```

56 //SE INTENTA LEER LOS DATOS DEL FIELD 1 DEL CANAL DE THINGSPEAK
57 //SE COLOCA EL ID DEL CANAL Y EL FIELD AL QUE SE DESEA APUNTAR
58 FIELD1 = ThingSpeak.readFloatField(1442425, 1);
59 //SE REVISAS QUE LA COMUNICACION DE LECTURA SEA SATISFACTORIA
60 COOMPROBADOR = ThingSpeak.getLastReadStatus(); //SE SOLICITA EL ESTATUS DE LA CONEXION
61 if(COOMPROBADOR == 200){ //SI EL ESTATUS ES IGUAL A 200 TODO ESRA BIEN
62     Serial.println("DATO DE FIELD 1: " + String(FIELD1));
63     if (FIELD1==1){ //SI EL FIELD1 ES UNO SE ENVIA UNA SEÑAL PARA QUE ENCENDER
64         digitalWrite (D0,HIGH);
65     }
66     if(FIELD1==0){ // SI EL FIELD1 ES CERO SE ENVIA UNA SEÑAL PARA APAGAR
67         digitalWrite (D0,LOW);
68     }
69 }
70 else{
71     Serial.println("OCURRIO UN PROBLEMA AL INTENTAR LEER EL CANAL. HTTP error code " + String(COOMPROBADOR));
72 }
73 delay(500); //SE COLOCA UN TIEPO DE ESPERA, ES NECESARIO QUE SIEMPRE ESTE LEYENDO LOS DATOS

```

Figura 3.22 Conexión con "Thingspeak"

Una vez que se obtiene el dato de la nube solo queda realizar cualquier acción con el mismo, eso depende del uso que se le quiera dar, en este caso únicamente se transmite en alto o "HIGH" si el dato es 1 y en bajo o "LOW" si el dato es 0.

Después de completar el código se realizó el esquema de conexiones, mismo que se aprecia en la Lámina 1 Anexo 3. El circuito consta de los siguientes dispositivos.

- Modulo *WiFi* "nodeMCU ESP8266" como cerebro central
- Cuatro módulos relés para *Arduino* que funcionan como activadores
- Cinco lámparas que son las que se desea controlar
- Tres conmutadores de pared
- Una cerradura electromagnética para el bloqueo y desbloqueo de la ventana

En el Anexo 4 se encuentra el diagrama de flujo del programa, además en la URL <https://bit.ly/3EdoCh7> se encuentra el código completo y comentado del módulo central.

Control de cerradura de ventana

Para dar seguridad a la ventana se usó una cerradura electromagnética (Ver Figura 3.23), la cual se activa si se induce corriente eléctrica en su bobina, es decir, se cierra la cerradura [21]. La cerradura de la ventana funciona mediante un relé que manipula el accionamiento de la cerradura, enviando 12 (V_{DC}), dependiendo la señal que envíe el "nodeMCU", el cual recibe la orden desde la página web.



Figura 3.23 Cerradura Electromagnética [33]

Etapa 2: Control de Acceso

Para la etapa de control de acceso de la puerta se utilizó el sistema de identificación por radio frecuencia (*RFID*) que hace referencia a una clase de tecnología de intercambio de datos de manera inalámbrica. Se usó este sistema de identificación se utilizó esta tecnología por la fácil instalación y codificación, además por su bajo precio y versatilidad para el usuario.

Para este caso se utilizó el módulo sensor *RFID* RC522, que es un elemento pasivo que no dispone de fuente propia y su alcance de lectura es dependiente de la antena receptora. Dicho módulo se aprecia en la Figura 3.24, con la distribución de sus pines de conexión.

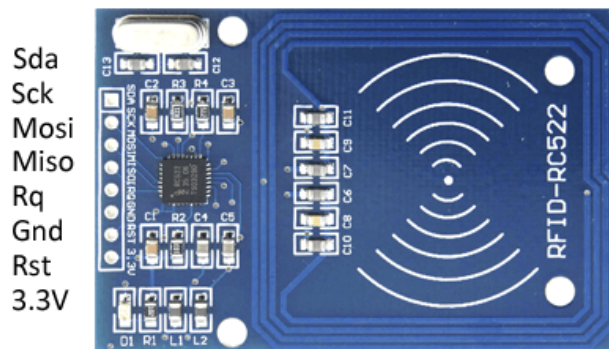


Figura 3.24 Pines de conexión del *RFID* [29]

La manera de uso de este dispositivo es mediante “*tags*”. Estos “*tags*” poseen un código único, el módulo *RFID* reconoce estos códigos los transmite de manera serial a la tarjeta programable, en este caso *Arduino*, en esta última se realizan las configuraciones necesarias para que cuando se reconozca algún “*tag*” se tomen decisiones como abrir o cerrar la cerradura de una puerta. Los “*tags*” se pueden encontrar en varias presentaciones, para este proyecto en particular se utilizaron “*tags*” en forma de tarjeta y llavero como se observa en la Figura 3.25.



Figura 3.25 “*Tags*” de *RFID* en forma de tarjeta y llavero [30]

Para esta “*shield*” se utilizó la librería de *RFID* “*MFRC522.h*” y adicional a esta se usó la librería “*SPI.h*”, que sirve para la utilización de la comunicación del mismo nombre que significa interfaz de periféricos en serie (*SPI*). Es un protocolo de datos en serie síncrono, mayormente utilizado por microcontroladores para la comunicación con uno o varios dispositivos periféricos en distancias relativamente cortas y a una gran velocidad. Por lo general con este tipo de comunicación siempre existe un maestro, normalmente un microcontrolador, que controla a otros dispositivos denominados esclavos. Existen tres líneas que son comunes en todos los dispositivos que utilizan la comunicación *SPI* [22].

- **MISO** (“*Master in Slave Out*” / entrada en el maestro; salida en el esclavo): Esta es la línea esclava que se utiliza para enviar datos al maestro.
- **MOSI** (“*Master Out Slave In*” / salida en el maestro; entrada en el esclavo): Esta es la línea maestra que se usa para enviar datos a los periféricos esclavos.
- **SCK** (reloj en serie): Por aquí se envían los pulsos de reloj generados por el maestro que sincronizan la comunicación.

Otra librería muy importante, que se utilizó para esta etapa, es la llamada “*LiquidCrystal_I2C.h*”; que sirve para poder controlar una pantalla *LCD* con tan solo cuatro pines, dos de comunicación y dos de alimentación.

Con la comunicación serial *I2C* (“*Inter Integrated Circuits*” / Entre Circuitos Integrados). En la Figura 3.26, se puede apreciar la pantalla *LCD* [23].



Figura 3.26 Pantalla *LCD I2C* [31]

Las líneas principales son:

- **SCL**: Es la línea de pulsos de reloj que sirve para sincronizar el sistema.
- **SDA**: Es la línea por donde se transmiten los datos en los dispositivos.

- **GND:** Es el común de la conexión entre todos los dispositivos que están comunicados al bus de datos.
- **VCC:** Alimentación para la pantalla *LCD*.

Para el código que maneja el sistema de la cerradura, se determinaron cuáles eran los códigos o *ID* de los “tags”, para esto, es de gran ayuda uno de los ejemplos que viene por defecto en *Arduino IDE*, tras la descarga de la librería para el módulo *RFID*, este ejemplo se llama “*Dumpinfo*”. Se utilizó este ejemplo para realizar un barrido de toda la información presente dentro de la tarjeta o llavero.

Al momento de acercar, ya sea la tarjeta o el llavero al módulo *RFID*, se podrá apreciar el dato llamado “*Card UID*”, este número está colocado en hexadecimal y es el código único del “tag”. Una vez se haya determinado la *UID* de la tarjeta o llavero es posible comenzar a generar el código final basado en el ejemplo antes mencionado. Mediante la “*Card UID*” se puede hacer valido un “tag”, almacenando el valor del mismo y guardándola en una variable, y solamente el tag que tenga la “*Card UID*” almacenada podrá ser aceptada como valida. En la Figura 3.27 se muestra la conexión que se usó entre el módulo *RFID* con el *Arduino nano*.

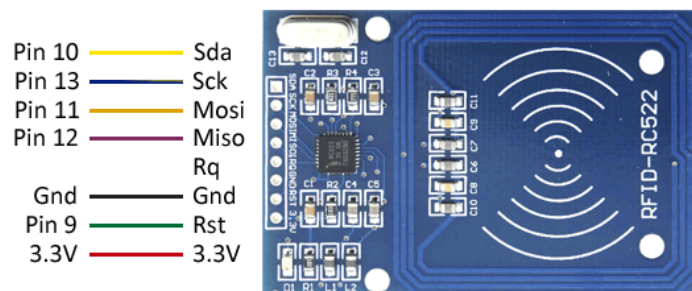


Figura 3.27 Esquema de conexión de *RFID* con *Arduino* [29]

Una vez agregadas las librerías correspondientes y definidos los pines a utilizar como salidas y entradas, se coloca una función para prevenir casos especiales como son la inexistencia de una tarjeta o que no se pueda leer el código dentro de una tarjeta o llavero cualquiera, en los dos casos se ordena que el programa vuelva al inicio y que se quede en espera de una tarjeta válida, este proceso se puede observar en las líneas de programación de la Figura 3.28.

```

if ( ! mfrc522.PICC_IsNewCardPresent()
return;

if ( ! mfrc522.PICC_ReadCardSerial()
return;

```

Figura 3.28 Prevención de casos especiales para el *RFID*

En caso de que exista un “tag” válido, se procede a obtener el código, grabando “Byte” por “Byte” en una variable tipo “char” para posteriormente comparar lo obtenido con los códigos registrados. Si el código coincide con los almacenados, el programa permitirá la apertura de la cerradura colocada en este circuito, pero, si los códigos no coinciden se enviará una señal de alerta, indicando que el “tag” es erróneo o que no existe en los registros. Los códigos que permiten cumplir con las funciones descritas se observan en la Figura 3.29 y en la Figura 3.30.

```
Serial.print("UID:");
for (byte i = 0; i < mfrc522.uid.size; i++) {
  if (mfrc522.uid.uidByte[i] < 0x10){
    Serial.print(" 0");
  }
  else{
    Serial.print(" ");
  }
  Serial.print(mfrc522.uid.uidByte[i], HEX);
  LecturaUID[i]=mfrc522.uid.uidByte[i];
}
```

Figura 3.29 Obtención y almacenamiento del UID del "tag"

```
boolean comparaUID(byte lectura[],byte usuario[])
{
  for (byte i=0; i < mfrc522.uid.size; i++){
    if(lectura[i] != usuario[i])
      return(false);
  }
  return(true);
}
```

Figura 3.30 Comparación de códigos obtenidos con códigos registrados

Después de confirmar que el programa esté correcto, se procede a la respectiva conexión del circuito de esta etapa, mismo que se muestra en la Lámina 2 Anexo 3, los dispositivos que componen el circuito son:

- *Arduino nano*
- *Pantalla LCD I2C*
- *Módulo RFID RC522*
- *Módulo relé*
- Una cerradura electromagnética para el bloqueo y desbloqueo de la puerta
- Un “*buzzer*” para dar una señal de alerta
- Dos *leds* que sirven como indicadores
- Dos resistencias de 330 (Ω) para protección de los *leds*

En el [Anexo 4](#) se encuentra el diagrama de flujo del programa, además en la [URL https://bit.ly/3z1HHiw](https://bit.ly/3z1HHiw) se encuentra el código completo y comentado del control de acceso de la puerta.

Etapa 3: Accionamiento de Cortina

Para esta etapa se utilizó el esquema básico para el control de motores *DC*, con el circuito integrado *L293D*, este integrado posee cuatro circuitos internos capaces de controlar el encendido y apagado de un motor, sin embargo, si lo que se requiere es controlar también el frenado y cambio de dirección del motor, con el integrado *L293D* solo se pueden controlar dos motores. En la Figura 3.31 se muestra la distribución de pines del circuito integrado [24].

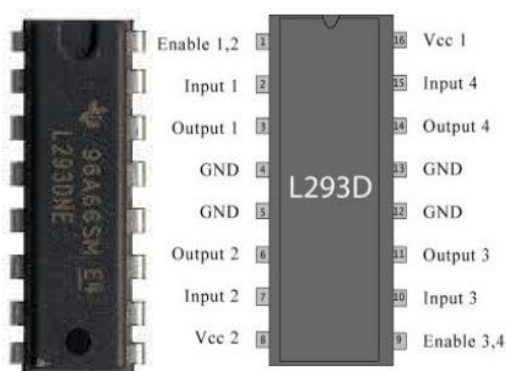


Figura 3.31 Circuito integrado *L293D* [32]

Descripción general de pines:

- Pin 1: Es el pin de “*Enable*” lo que quiere decir es que si recibe 1 lógico o 5 (V_{DC}), aproximadamente, habilita las salidas 1 y 2, caso contrario las bloquea.
- Pin 2: Es el pin de entrada 1, lo que quiere decir es que, si recibe 1 lógico, el pin de salida 1 se comporta como un suministrador de alimentación para motores y en caso de recibir 0 lógico se comporta como un punto de tierra o *GND*.
- Pin 3: Es el pin de salida 1 cuyo funcionamiento depende directamente del pin 2.
- Pin 4,5,12 y 13: Pines que se deben conectar a tierra o *GND*.
- Pin 8 y 16: Pines que se deben conectar a alimentación, cabe mencionar que el pin 16 es la alimentación para el circuito integrado y el pin 8 es la alimentación de los motores.
- Pin 9: Es el pin de “*Enable*” lo que quiere decir es que si recibe 1 lógico o 5 (V_{DC}), aproximadamente, habilita las salidas 3 y 4, caso contrario las bloquea.
- Pin 6: Es el pin de salida 2 cuyo funcionamiento depende del pin 7.

- Pin 7: Es el pin de entrada 2, lo que quiere decir es que, si recibe 1 lógico, el pin de salida 2 se comporta como un suministrador de alimentación para motores y en caso de recibir 0 lógico se comporta como un punto de tierra o *GND*.
- Pin 10: Es el pin de entrada 3, lo que quiere decir es que, si recibe 1 lógico, el pin de salida 3 se comporta como un suministrador de alimentación para motores y en caso de recibir 0 lógico se comporta como un punto de tierra o *GND*.
- Pin 11: Es el pin de salida 3 cuyo funcionamiento depende del pin 10.
- Pin 14: Es el pin de salida 4 cuyo funcionamiento depende del pin 15.
- Pin 15: Es el pin de entrada 4, lo que quiere decir es que, si recibe 1 lógico, el pin de salida 4 se comporta como un suministrador de alimentación para motores y en caso de recibir 0 lógico se comporta como un punto de tierra o *GND*.

Al recibir cero lógico a la placa de *Arduino* nano, se ordena que el motor gire en sentido horario o a su vez que abra la cortina, y cuando se recibe uno lógico se ordena que el motor gire en sentido antihorario o que cierre la cortina. Cabe mencionar que para esta etapa fue necesario programar un bucle que ayude a apagar el motor hasta que el estado del dato de entrada cambie, de esta manera, en el instante que exista un cambio de estado del dato que llega, el motor se accionará por un cierto tiempo, mismo que se determinó con pruebas de calibración, y se apagará hasta detectar otro cambio de estado. Para la parte del programa que controla este sistema, basta con enviar 1 lógico o 0 lógico a los pines 2 y 7 para controlar el giro del motor, así como el frenado. En la Lámina 3 Anexo 3 se puede apreciar el circuito esquemático de este sistema. Cuando el *Arduino* nano recibe 1 lógico, el motor gira en sentido horario, haciendo que las cortinas se abran, y si el *Arduino* recibe 0 lógico, el motor gira hacia el lado contrario para cerrar las cortinas. En la Tabla 3.1 se observan los estados posibles para los pines del integrado *L293D* los cuales permiten controlar un motor.

Tabla 3.1 Tabla de verdad de estados posibles con el integrado *L293D*

Entrada 1 Pin 2	Entrada 2 Pin 7	Acción
0	0	Apagado de motor
0	1	Giro en sentido Horario
1	0	Giro en sentido Antihorario
1	1	Situación prohibida

En el Anexo 4 se encuentra el diagrama de flujo del programa, además en la URL <https://bit.ly/3npqJJJ> se encuentra el código completo y comentado del accionamiento de la cortina.

Para dar protección al motor con el cual se abre la cortina se realizó una caja protectora como se muestra en la Figura 3.32.



Figura 3.32 Caja Protectora de Motor

Sistema de Seguridad

Para dar cumplimiento con esta etapa se instaló una cámara de video vigilancia en una zona estratégica de la casa previamente evaluada en las correspondientes visitas que se realizaron. Se concluyó que la zona más vulnerable que tenía la casa es la salida del garaje, misma que se puede apreciar en la Figura 3.33.



Figura 3.33 Zona de vigilancia

Para lograr obtener, en lo posible, una buena calidad de imagen se optó por instalar una cámara de la marca *HikVision* con 1080 pixeles de resolución y con conectividad IP. Se utilizó esta cámara por el prestigio y calidad de sus productos. La cámara instalada se la puede apreciar en la Figura 3.34.



Figura 3.34 Cámara de seguridad *HikVision*

Al momento de realizar la instalación se verificó el ángulo correcto donde el dispositivo tenga una gran cobertura del lugar, y además se detectó un posible riesgo para el dispositivo, ya que la zona donde se instaló carece de una debida protección por lo cual se colocó un pequeño techo que proteja a la cámara (ver Figura 3.35), de esta manera se asegura que la vida útil del dispositivo sea duradera. En cuanto al medio de transmisión, se realizó de forma cableada, utilizando un cable FTP categoría 5 para exteriores con el cual se conecta la cámara directamente al *router* de la vivienda.



Figura 3.35 Protección de Cámara de Seguridad

Placas PCB

El diseño y dimensionamiento de las placas PCB para el sistema se realizaron en el *software Proteus*, además, se organizaron de forma correcta los diferentes dispositivos que componen los circuitos de control. El circuito diseñado para el módulo central se muestra en la Figura 3.36 y el circuito para el control de la cortina se encuentra en la Figura 3.37. Cabe mencionar que el circuito del control de acceso de la puerta no cuenta con una placa PCB, debido a que las conexiones se realizaron directamente desde los dispositivos hasta el *Arduino* nano que se encuentra en el circuito de control de la puerta. Después de verificar que las pistas se encuentren correctas, se procedió a soldar los

dispositivos a las placas PCB. En las Figuras 3.38 y 3.39 se pueden apreciar los resultados.

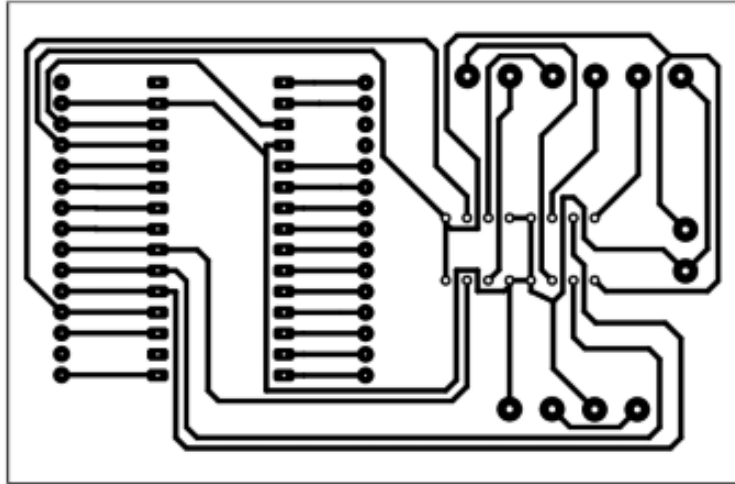


Figura 3.36 Diseño PCB del circuito del módulo central

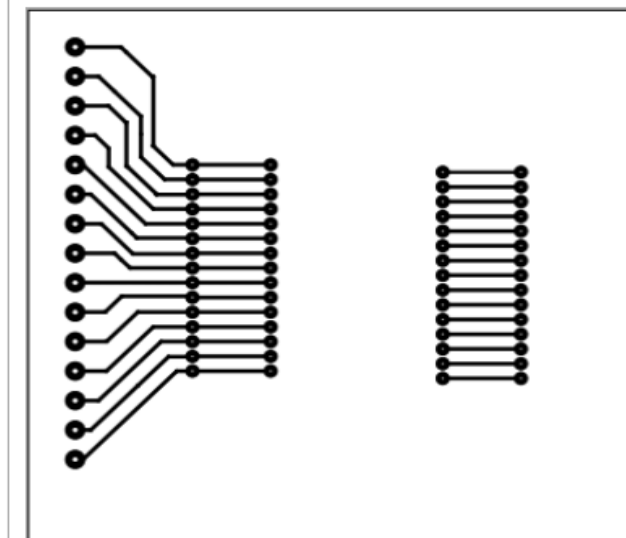


Figura 3.37 Diseño PCB del circuito para el control de la cortina

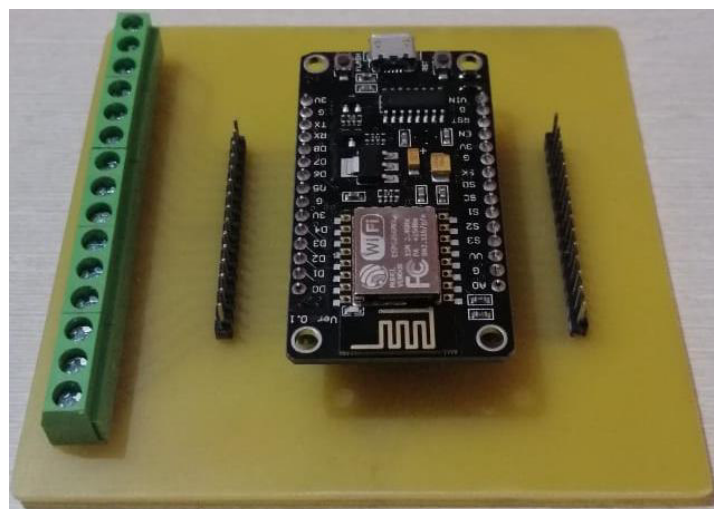


Figura 3.38 Circuito del módulo central

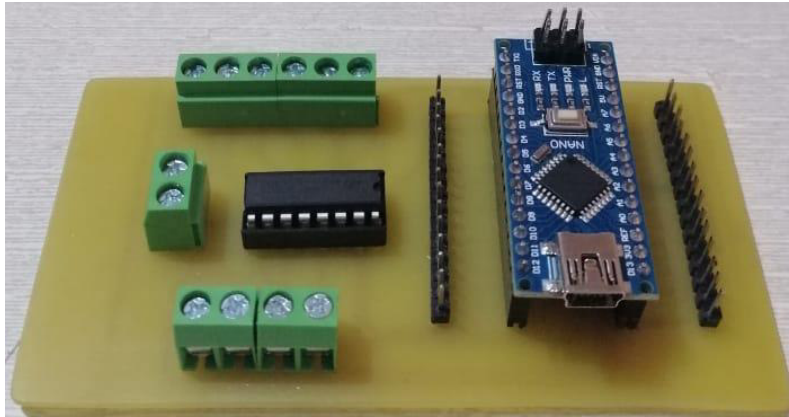


Figura 3.39 Circuito para el control de la cortina

Para proteger a los circuitos del medio externo se fabricaron recubrimientos de madera (cajas) para su respectivo almacenamiento.

La caja contenedora del módulo central se la observa en la Figura 3.40 y en la Figura 3.41 se observa el interior de dicha caja con los circuitos del módulo central y del control de la cortina.



Figura 3.40 Caja del módulo central

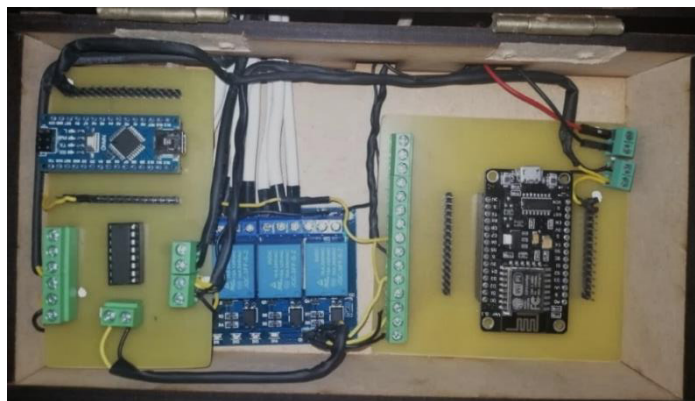


Figura 3.41 Interior de la caja con el circuito del módulo central

La caja contenedora del circuito de control de acceso de la puerta se aprecia en la Figura 3.42 y en la Figura 3.43 se observa el interior de dicha caja con los circuitos. Por último, la caja que contiene al circuito del *RFID* y pantalla *LCD* se observa en la Figura 3.44.



Figura 3.42 Caja del circuito de control de acceso

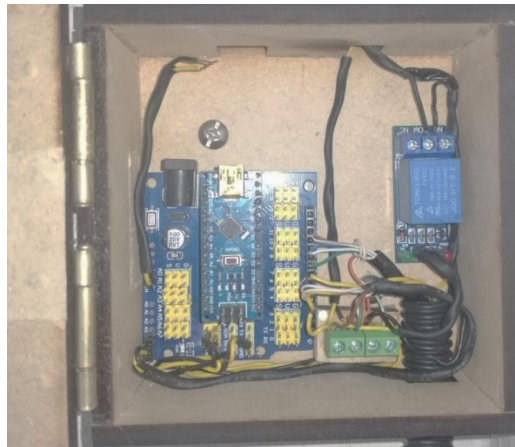


Figura 3.43 Interior de la caja con el circuito del control de acceso



Figura 3.44 Caja del módulo *RFID* y pantalla *LCD*

3.4 Interfaz de control y supervisión

Para que el usuario tenga mayor facilidad a la hora de utilizar las distintas funciones que ofrece el sistema mediante el uso de la página web, se usó el lenguaje de hojas de estilo “*Cascading Style Sheets*” o más conocido como *CSS*. Este lenguaje es utilizado conjuntamente con los archivos *HTML* para dar una mejor presentación a los elementos que se tenga como por ejemplo mover imágenes y texto, agregar colores a las letras o fondo de la pantalla, entre otros [25]. Este archivo se debe subir al servicio de *000WebHost* como se muestra en la Figura 3.45, además en la URL <https://bit.ly/3zUEgdJ> se muestra el código utilizado para esta página.

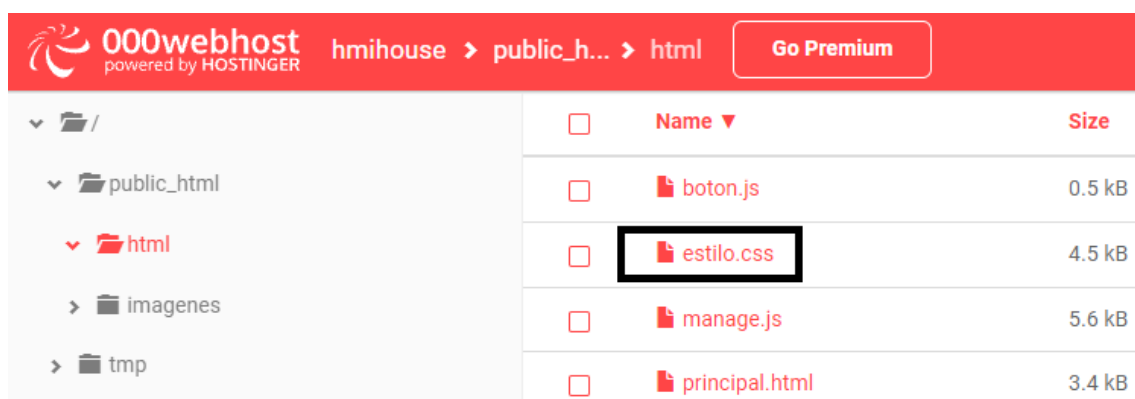


Figura 3.45 Documento CSS

Con los archivos *HTML*, *CSS* y “*JavaScript*” cargados correctamente a la página web, se tiene finalizado el centro de comandos para que el usuario pueda comandar los distintos dispositivos del sistema implementado. Al ingresar a la página con el dominio, lo primero que se va a presentar es una portada para dar estilo a la misma. La interfaz diseñada cuenta con un botón en la parte superior derecha que permite cerrar la sesión, es decir, salir de la página web; sin embargo, la parte principal de la página web son los botones de acción. Estos botones se encuentran en la parte inferior los cuales permiten encender las luces ubicadas en el pasillo, sala y comedor, abrir y cerrar la cerradura de la puerta y ventana principal y el accionamiento de la cortina. Cada vez que se accione un botón de la página, va a marcarse de color verde para mostrar que el dispositivo está activado, y cuando el botón se pinte de color rojo, indicará que el dispositivo se encuentra desactivado. Para el acceso a la cámara de seguridad, el botón de acción de la página web lo que hace es redirigir a la página oficial de *Hikvision* para ingresar con las credenciales asegurando la protección del sistema y poder ver el video en vivo que capta la cámara. La interfaz creada es lo más intuitiva posible para que el usuario pueda manejar el sistema sin problema alguno (ver Figura 3.46).

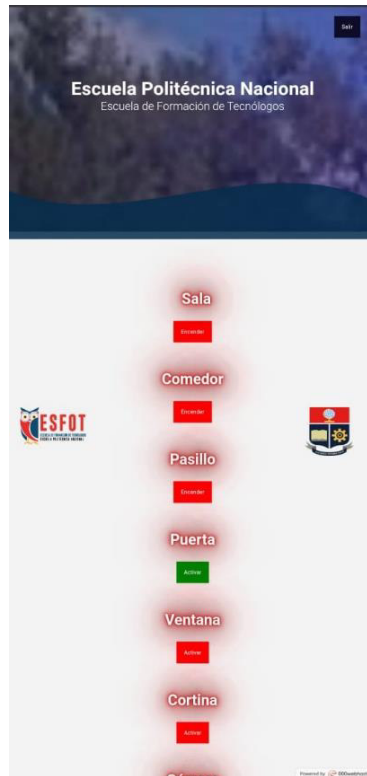


Figura 3.46 Interfaz de usuario

Medio de identificación

Para proteger la página web y evitar que personas ajenas hagan mal uso del sistema, se creó un medio de identificación en el cual se debe ingresar un usuario y contraseña para acceder a la página principal. Este medio de identificación está creado con una estructura de programación en *HTML* y para verificar que los datos ingresados sean correctos, se usó *JavaScript*. Para ingresar a la página web primero se va a ingresar los datos solicitados; luego lo que realiza el programa es que los datos ingresados se almacenan en el *Local Storage* y se los compara con la contraseña y usuario que están permitidos, si son iguales se permite ingresar, pero si no son iguales se niega el acceso.

En la Figura 3.47 se muestran las líneas de código que permiten la verificación de los datos ingresados para validarlos permitiendo o denegando el acceso.

```

if(window.location.href==urlDate){

    var name = localStorage.getItem("username");
    var pass = localStorage.getItem("password");
    if(name!=" " || pass != " "){
        window.location = window.location.origin
    }
}

```

Figura 3.47 Código para datos de acceso

También se utilizó el lenguaje de programación CSS para diseñar la ventana de ingreso de datos, de este modo resulta una interfaz fácil de manejar. Esta interfaz se muestra en la Figura 3.48.

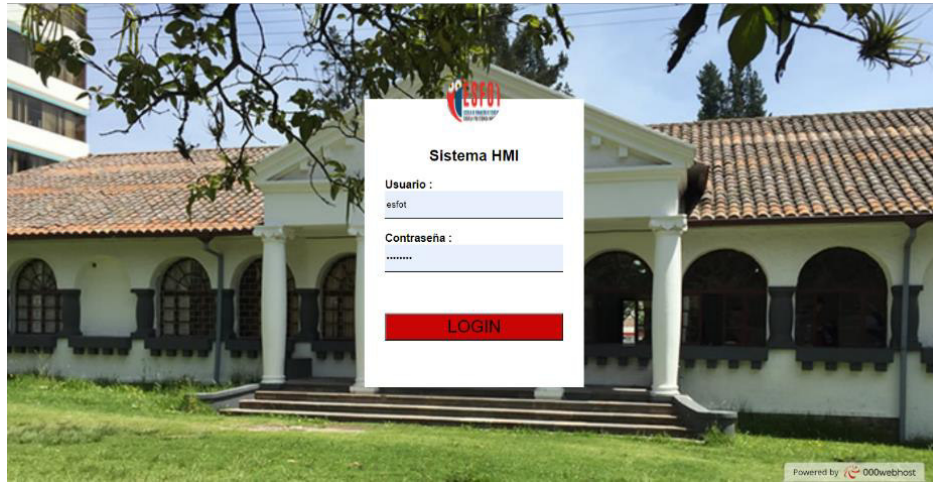


Figura 3.48 Interfaz para identificación

Los códigos utilizados para la creación del medio de identificación se muestran en la siguiente URL <https://bit.ly/3kWYn6Z>.

3.5 Pruebas y Análisis de Resultados

Como ya se mencionó, el proyecto fue dividido en tres etapas, mismas que fueron probadas por separado y posteriormente, el sistema en conjunto. Para comenzar con las pruebas primero se verificó la conexión de la página web con la plataforma "Thingspeak", es decir, si los datos enviados son receptados y procesados adecuadamente para que se puedan realizar las actividades que se solicitan (ver Figura 3.49).

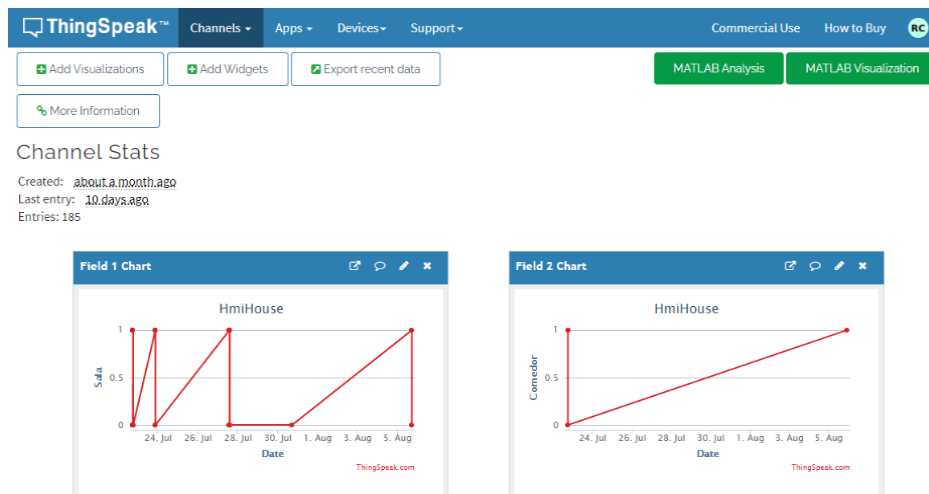


Figura 3.49 Conexión entre "Thingspeak" y la página web

Prueba de Etapa 1: Módulo Central

Para esta prueba se verificó si el módulo *WiFi "nodeMCU"* recibe los datos enviados por la página web, con la ayuda de *Arduino IDE* y su pantalla serial. Se constató la llegada de las órdenes de la página web traducidas a datos digitales, como se puede apreciar en la Figura 3.50.

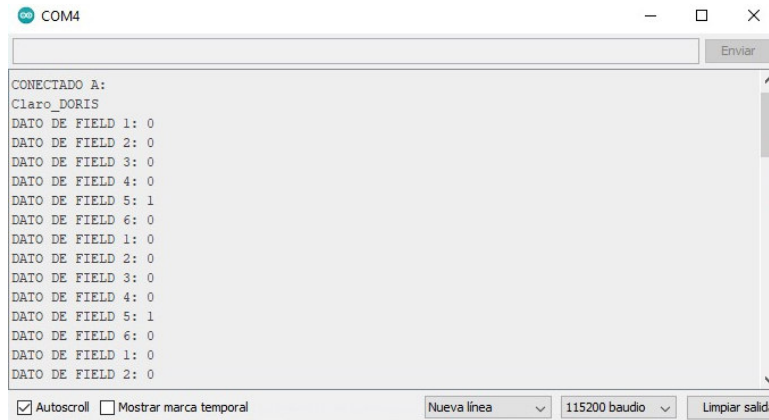


Figura 3.50 Monitor Serial del módulo *nodeMCU*

Una vez confirmada la comunicación entre el módulo central y la página web, se procedió a tratar los datos de llegada. Una vez colocado los módulos en su lugar (Ver Figura 3.51); con ayuda del módulo central y módulos relés se consiguió encender y apagar luces, y también la activación de la cerradura que se encuentra en la ventana principal (Ver Figura 3.52 hasta Figura 3.55). Los resultados de las pruebas de esta etapa se muestran en la Tabla 3.2.



Figura 3.51 Módulo central y módulos relés



Figura 3.52 Circuito de iluminación del comedor



Figura 3.53 Circuito de iluminación de la sala



Figura 3.54 Circuito de iluminación del pasillo

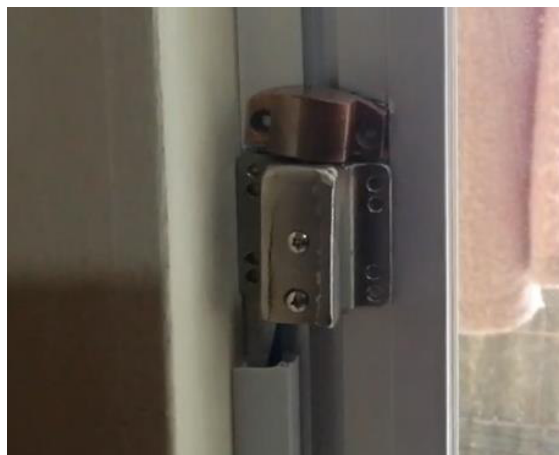


Figura 3.55 Cerradura de la ventana

Tabla 3.2 Pruebas de funcionamiento de la etapa 1: Módulo Central

Módulo	Comunicación con la página web	
"nodeMCU"	SI	
Dispositivos	Conexión con el módulo	Ejecución de la tarea
Luces	SI	SI
Cerradura	SI	SI

Prueba de Etapa 2: Control de Acceso

En esta prueba se verificó principalmente el funcionamiento del relé que se encuentra en la caja contenedora del control de acceso (Ver Figura 3.56), que a su vez activa o desactiva la cerradura electromagnética colocada en la puerta principal (Ver Figura 3.57), por la acción del "shield" RFID o por el uso de la página web. Para constatar la activación de la cerradura se incluyó señales de aviso como *leds* indicadores, un "Buzzer", y para dar una guía visual al usuario, se colocó una pantalla LCD que muestra indicaciones como "Bienvenido Angel" (Ver Figura 3.58).



Figura 3.56 Módulo de control de acceso



Figura 3.57 Cerradura de la puerta



Figura 3.58 Sistema de Identificación *RFID*

Cabe mencionar que para esta etapa se tuvo que ajustar el tiempo de activación del relé para que se acople a la comunicación planeada del módulo de control de acceso con el módulo central. Los resultados de las pruebas de esta etapa se muestran en la Tabla 3.3.

Tabla 3.3 Pruebas de funcionamiento de la etapa 2: Control de acceso

Dispositivos	Conexión con el controlador	Ejecución de la tarea
<i>RFID</i>	SI	SI
<i>Leds</i>	SI	SI
Cerradura	SI	SI
Relé	SI	SI
<i>Buzzer</i>	SI	SI
Pantalla <i>LCD</i>	SI	SI

Prueba de Etapa 3: Accionamiento de cortina

Para confirmar el correcto funcionamiento de esta etapa, se controla un motor para abrir o cerrar una cortina (Ver Figura 3.59). El punto clave para asegurar el funcionamiento de esta etapa fueron los tiempos que permanece activo el motor. Al abrir la cortina el motor debe tener más fuerza y esto genera que el tiempo de activación sea más que al cerrar la cortina, debido a que el mismo peso de la cortina cerrará la misma, reduciendo el tiempo de activación del motor. Además, la estructura en donde se coloca el motor debe ser estable para que al girar el motor no se enrede el hilo con el cual se manipula la cortina. Otro aspecto importante es la fuerza con la que el motor recoge la cortina, funcionando correctamente a 12 (V_{DC}) mejorando la activación de la cortina y tiempos de activación. Los resultados de las pruebas de esta etapa se muestran en la Tabla 3.4.



Figura 3.59 Motor de la cortina

Tabla 3.4 Pruebas de funcionamiento de la etapa 3: Accionamiento de cortina

Dispositivos	Conexión con el controlador	Ejecución de la tarea
Motor	SI	SI
Tarea	Tiempo de Activación (segundos)	
Activar motor	1	
Desactivar motor	0,9	

Prueba de la Cámara de Seguridad

La principal función de la cámara de seguridad es la visualización en vivo de la entrada de la vivienda, mediante un celular o un ordenador. Entonces para ello se colocó la cámara (Ver Figura 3.60) en el lugar deseado obteniendo un gran campo de visualización

del lugar, funcionando perfectamente, además la cámara capta video perfectamente en la noche. Prueba de ello, en la Figura 3.61 se muestra la imagen en vivo captada por la cámara de seguridad.



Figura 3.60 Cámara de Seguridad



Figura 3.61 Prueba de Funcionamiento de Cámara de Seguridad

Prueba final de todo el sistema

Una vez que se probó cada etapa por separado, se procedió a verificar la correcta conexión, comunicación y funcionamiento de todos los circuitos. Entonces, al enviar una orden desde la página web, se encendieron las luces, se desbloqueó y bloqueó la cerradura de la puerta y de la ventana y también el accionamiento de la cortina. La forma de hacer esta prueba fue mediante la página web, ya que con esto se confirma la comunicación desde la página web hasta el último dispositivo de cada etapa. Como constancia del funcionamiento del sistema, se realizó un video en el que se muestran las pruebas realizadas en las tres etapas de forma individual y en conjunto. Se puede acceder al video mediante el código QR de la Figura 3.62.



Figura 3.62 Código QR del video de pruebas de funcionamiento

Análisis antes y después de la instalación del sistema

Después de instalar el sistema, se usaron las planillas de la empresa eléctrica desde el mes de abril hasta el mes de agosto para poder verificar los niveles de ahorro energético del hogar, en cuanto a consumo de energía. Al elaborar el análisis y comparación de los meses antes y después de la instalación y puesta en marcha del sistema, se obtuvieron los valores que se muestran en la Tabla 3.5. A pesar que los datos muestren una leve diferencia, estos podrían mejorar automatizando más lugares del hogar con dispositivos que brinden mejores beneficios.

Tabla 3.5 Historial de costos mensuales de consumo eléctrico

Planillas	
Meses	Valor (\$)
Abril	27,33
Mayo	21,38
Junio	20,74
Julio	15,63
Agosto	18,13

Como se mostró en la Tabla 3.5, los valores del consumo eléctrico se reducen, siendo en el mes de julio, el mes que se registra un valor menor a comparación del resto. En la Figura 3.63 se encuentran los valores representados gráficamente

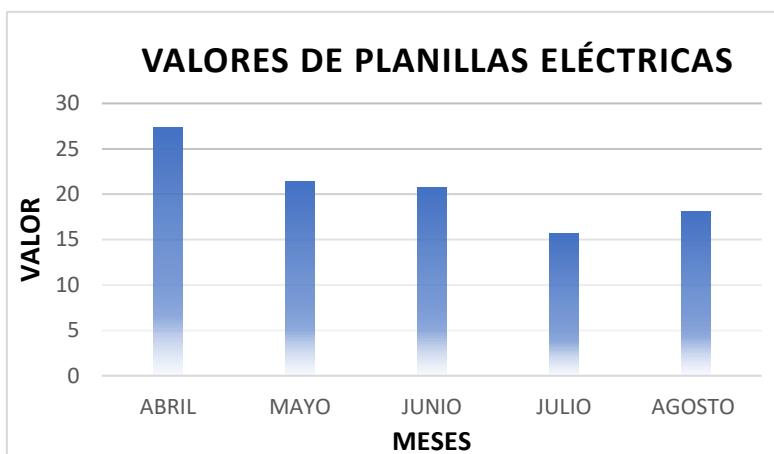


Figura 3.63 Gráfica de valores de planillas eléctricas

También se constató que esta reducción podría ser mucho más evidente a futuro, ya que el sistema lleva instalado desde junio, esto quiere decir que relativamente tiene poco tiempo de uso. Además, se podría mejorar este ahorro energético utilizando dispositivos más sofisticados que ayuden con este propósito y potencialicen al proyecto. En la Figura 3.64 se puede observar los valores a pagar de los meses ya mencionados de donde se realizó el análisis, esta captura se la hizo de una página de la Empresa Eléctrica Quito (EEQ) [26].

SEL	PDF	SECUENCIA	CLIENTE (CUEN)	CONTRATO	FECHA DE EMISIÓN	NÚMERO DE DOCUMENTO	TOTAL CONSUMO	FECHA DE VENCIMIENTO
<input checked="" type="checkbox"/>		1	1401473865	200011506819	06/09/2021	002301185992	22,27	21/09/2021
<input type="checkbox"/>		2	1401473865	200011506819	05/08/2021	002661040932	18,13	20/08/2021
<input type="checkbox"/>		3	1401473865	200011506819	08/07/2021	002451073849	15,63	23/07/2021
<input type="checkbox"/>		4	1401473865	200011506819	04/06/2021	002531026635	20,74	19/06/2021
<input type="checkbox"/>		5	1401473865	200011506819	06/05/2021	003080873208	21,38	21/05/2021
<input type="checkbox"/>		6	1401473865	200011506819	06/04/2021	002112185583	27,33	21/04/2021

Figura 3.64 Información de consumo EEQ

Comparación del sistema implementado frente a otros existentes en el mercado

El valor económico de la inversión en un sistema uno de los principales aspectos a considerar. En el mercado se cuenta con varias opciones en cuanto a controladores inteligentes como *Alexa de Amazon*, *Google Assistant*, entre otros. Estos controladores comandan dispositivos los cuales cumplen distintas funciones como abrir puertas,

cortinas, ventanas, activar luces, en fin, aplicaciones que mejoran y dan mayor confort al usuario.

El presente proyecto está enfocado en obtener un sistema que cumpla con características similares, pero reduciendo costos en cuanto a los dispositivos que se necesitan. En la Tabla 3.6 se muestra una comparación de precios de los dispositivos disponibles en el mercado con los dispositivos utilizados en el proyecto, recalcando que los dispositivos del mercado cuentan con mejores características, sin embargo, el funcionamiento en general, es el mismo.

En el Anexo 5, se muestra la tabla en donde se desglosa los materiales utilizados con su respectivo precio y, además, el costo total del proyecto.

Tabla 3.6 Comparación de precios

	DISPOSITIVOS	VALOR (\$)	UNIDADES	TOTAL (\$)
MERCADO	MOTOR CORTINA	80,00	1	80,00
	ALEXA	40,00	1	40,00
	CERRADURA	70,00	2	140,00
	INTERRUPTOR	17,00	3	51,00
	TOTAL			311,00
PROYECTO	MOTOR CORTINA	3,00	1	3,00
	NODE MCU	7,00	1	7,00
	CERRADURA	10,00	2	20,00
	CONMUTADOR	1,80	3	5,40
	TOTAL			35,40

En base a los datos anteriores se puede establecer que el sistema instalado es mucho más económico que un sistema comercial, resaltando que los dos cumplen con los propósitos establecidos en el presente proyecto: control de acceso, iluminación y seguridad.

3.6 Manual de Uso y Mantenimiento

Para mostrar el manejo y mantenimiento del sistema domótico se realizó un video explicativo donde se detallan todas las acciones a tomar en cuenta para una correcta utilización, además se resaltan las principales recomendaciones para dar mantenimiento a los dispositivos para de esta manera cumplir con su tiempo de vida útil. Se puede acceder al video mediante el código QR de la Figura 3.65.



Figura 3.65 Código QR del video de funcionamiento y mantenimiento

4 CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

- Se instaló un sistema domótico obteniendo los resultados deseados, ya que al cabo de tres meses de pruebas se puede notar una reducción en el consumo eléctrico, al contar con un presupuesto más alto, el ahorro energético sería mayor consiguiendo mejores prestaciones.
- Este proyecto genera varios beneficios como confort y seguridad, además la domótica es un campo en el cual se debe invertir gran cantidad de dinero convirtiéndolo en un problema actualmente. El sistema implementado demostró estar al nivel de los sistemas comerciales existentes a un mejor precio, siendo una gran idea para un posible emprendimiento, pudiéndose repotenciar de distintas formas.
- Las pruebas realizadas en el sistema, indistintamente del horario, fueron correctas, lo que demuestra que usar el *hardware* y *software* de *Arduino*, tanto en interiores como en exteriores, es confiable y no genera fallas en el funcionamiento.
- Aunque la parte física funcione correctamente la responsabilidad del sistema siempre depende del correcto envío de las instrucciones; y el uso de “*Thingspeak*” permitió un funcionamiento correcto, sin retardos excesivos y sin pérdida de información.
- Se elaboró la página web con el uso de los distintos lenguajes los cuales son *HTML*, *CSS* y *JS*; permitiendo un diseño adecuado y sencillo a través de la

utilización de las herramientas web disponibles, además de disponer de un medio de identificación seguro que permite una plena confianza del sistema.

- Para el diseño de las placas PCB se confirmó que ninguna pista posea una curva de 90°, esto con el objetivo de evitar problemas, adicionalmente a esto se verificó que la puesta a tierra fuera común para todo el sistema, con el fin de disminuir el ruido que causa las interferencias electromagnéticas, que en este caso son muy peligrosas, ya que la conexión del módulo central es por medio de *WiFi* y es indispensable que tenga la mejor señal posible.
- En cuanto al ensamblaje y la soldadura de los diferentes dispositivos en la placa PCB, se utilizaron conectores macho/hembra, esto con el fin de dar mayor facilidad al momento de extraer los microcontroladores para su respectivo mantenimiento y corrección de errores.
- Las pruebas realizadas en el circuito de control de la cortina permitieron corregir errores y afinar los tiempos para abrir y cerrar dicha cortina, con esto se logró que, al momento de activarse el motor, este produzca la apertura y cierre suavemente, obteniendo un nivel de confort adecuado.
- Las pruebas realizadas en el control de acceso de la puerta ayudaron a determinar errores en la comunicación entre el circuito de control y el módulo central, para corregir estos errores se reajustaron los tiempos de envío de datos entre estas dos etapas para conseguir un funcionamiento adecuado.
- Después de instalar la cámara de seguridad se pudo confirmar que la zona más vulnerable de la casa, quedó protegida de cierta manera y adicional a esto las cerraduras tanto en la ventana como en la puerta hacen que el nivel de seguridad de la casa aumente, a un precio relativamente más bajo frente a otros sistemas de seguridad.
- Con la interfaz diseñada, el usuario puede controlar los dispositivos de manera remota, de esta forma, no necesita desplazarse por la casa para accionarlos, además, se pueden activar los dispositivos sin la necesidad de conectarse a la red de su hogar y controlar el sistema con cualquier otra red o con datos móviles.

4.2 Recomendaciones

- Si bien es cierto, los identificadores que indican si las luces están encendidas o apagadas son de gran ayuda, pero tienen un problema y es que estos solo indican el estado dentro de la página, no tienen la capacidad de mostrar el estado de las luces cuando son accionadas de manera manual (física) con los interruptores instalados en la pared, debido a que estos últimos son dispositivos simples que

no poseen la función de conectarse a la red *WiFi* de la residencia. Para mejorar el sistema se recomienda reemplazar los interruptores simples por interruptores inteligentes que si tengan conectividad *WiFi*.

- La cámara de video vigilancia instalada en la residencia posee una resolución de 1080p, si lo que se quiere, es mejorar el sistema, se recomienda instalar una cámara con mayor resolución o incluso con funcionalidad *PTZ*, esta función permite que el usuario pueda manipular el movimiento de la cámara de manera remota, cubriendo así mucha más área de vigilancia.
- El motor que ayuda al control de la cortina tiene baja potencia, esto causa que la cortina no se abra por completo, para poder mejorar esto, se recomienda instalar un motor con mayor potencia para que la cortina se pueda abrir completamente.
- Al momento de realizar las respectivas pruebas del sistema, se pudo constatar ciertos retardos de la comunicación entre el módulo central y la página de "*Thingspeak*". Para mejorar esto, se recomienda mejorar la conectividad de Internet adquiriendo un mejor plan con su respectivo distribuidor de servicios y adicionalmente un mejor plan de la página "*Thingspeak*" para tener acceso a más beneficios que brinda la misma.
- Para que el sistema se potencialice aún más y genere un ahorro energético mayor, se recomienda instalar adicionalmente un medidor inteligente (*smart meter*) que muestre el consumo y costo de energía del hogar y que a su vez envíe avisos, que permita a los usuarios conocer en tiempo real sus niveles de consumo para que puedan tomar las acciones respectivas.
- La cámara de seguridad instalada posee su debida protección, sin embargo, se recomienda revisarla cada mes o cada dos meses para evitar que la humedad provoque algún fallo.
- Para el sistema de iluminación se recomienda revisar continuamente el estado de las luminarias, tomar en cuenta que para reemplazar dichas luminarias es necesario interrumpir la alimentación del circuito afectado; así se asegura la integridad del sistema y el bienestar del usuario.

5 REFERENCIAS BIBLIOGRÁFICAS

- [1] E. Terán, «Coronavirus en Ecuador: Un ABC de la cuarentena, restricciones, teletrabajo.,» 24 Marzo 2020.
- [2] «comparaiso,» 25 SEPTIEMBRE 2020. [En línea]. Available: <https://comparaiso.es/domotica>. [Último acceso: SEPTIEMBRE 2020].
- [3] laarcom, «laarcom,» laarcom, [En línea]. Available: <https://www.laarcom.com/que-es-la-domotica>. [Último acceso: 15 Septiembre 2021].
- [4] «¿Qué es un Hosting y cómo funciona?,» Webempresa, 2021. [En línea]. Available: <https://www.webempresa.mx/hosting/hosting-que-es-como-funciona.html>. [Último acceso: 08 17 2021].
- [5] «Hosting web,» 000webhost, 2021. [En línea]. Available: <https://www.000webhost.com/>. [Último acceso: 23 Agosto 2021].
- [6] E. S. I. Team, «adafruit.com,» 2015. [En línea]. Available: cdn-shop.adafruit.com/product-files/2471/0A-ESP8266__Datasheet__EN_v4.3.pdf. [Último acceso: 20 AGOSTO 2021].
- [7] naylampmechatronics, «naylampmechatronics,» [En línea]. Available: <https://naylampmechatronics.com/espressif-esp/153-nodemcu-v2-esp8266-wifi.html>. [Último acceso: 20 AGOSTO 2021].
- [8] «API: qué es y para qué sirve,» Xataka, 13 Agosto 2019. [En línea]. Available: <https://www.xataka.com/basics/api-que-sirve>. [Último acceso: 20 Agosto 2021].
- [9] «ThingSpeak,» MatLab, 2021. [En línea]. Available: https://es.libre.university/lesson/HkS_Bx8kZ/5.%20ThingSpeak:%20Internet%20o%20f%20Things. [Último acceso: 22 Agosto 2021].
- [10] «Autocad,» Autodesk, 2020. [En línea]. Available: <https://www.autodesk.es/products/autocad/overview?plc=ACDIST&term=1-YEAR&support=ADVANCED&quantity=1>. [Último acceso: Agosto 2020].

- [11] «Internet protocol,» Digital Guide IONOS, 02 Abril 2019. [En línea]. Available: <https://www.ionos.es/digitalguide/servidores/know-how/internet-protocol-definicion-y-fundamentos/>. [Último acceso: Agosto 2020].
- [12] Futrera, «smartfreehosting,» [En línea]. Available: <https://smartfreehosting.com/hosting-gratuito-con-000webhost/>. [Último acceso: Septiembre 2020].
- [13] Naylampmechatronics, «Naylampmechatronics,» [En línea]. Available: <https://naylampmechatronics.com/espressif-esp/153-nodemcu-v2-esp8266-wifi.html>. [Último acceso: Agosto 2021].
- [14] «Storage.setItem,» Developer Mozilla, 2021. [En línea]. Available: <https://developer.mozilla.org/es/docs/Web/API/Storage/setItem>. [Último acceso: 31 AGOSTO 2021].
- [15] «Fundamentos de JavaScript,» Developer Mozilla, 2021. [En línea]. Available: https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/JavaScript_basics. [Último acceso: 20 Agosto 2021].
- [16] N. C. Deisy Sánchez, «Introducción a la Programación,» 2021. [En línea]. Available: <http://files.sanjo2014.webnode.es/200000001-c34cac445e/INTRODUCCION%20A%20LA%20PROGRAMACION.pdf>. [Último acceso: 22 Agosto 2021].
- [17] «APIs para el Internet de las Cosas: ThingSpeak, Pachube y Fitbit,» BBVA API_Market, 21 Marzo 2016. [En línea]. Available: <https://www.bbvaapimarket.com/es/mundo-api/apis-para-el-internet-de-las-cosas-thingspeak-pachube-y-fitbit/>. [Último acceso: 10 Noviembre 2021].
- [18] «Ajax introduction,» W3Schools, 2021. [En línea]. Available: https://www.w3schools.com/js/js_ajax_intro.asp. [Último acceso: 31 AGOSTO 2021].
- [19] «Cómo usar LocalStorage con JavaScript y HTML5,» Platzi, 2015. [En línea]. Available: <https://platzi.com/blog/local-storage-html5/>. [Último acceso: 20 Agosto 2021].

- [20] «Códigos de estado de respuesta HTTP,» Developer Mozilla, 2021. [En línea]. Available: <https://developer.mozilla.org/es/docs/Web/HTTP/Status>. [Último acceso: 17 Septiembre 2021].
- [21] «Cerradura Electromagnética,» Novelec, 2018 Agosto 14. [En línea]. Available: <https://blog.gruponovelec.com/redes-vdi/cerradura-electrica-tipos-y-caracteristicas/>. [Último acceso: 20 Septiembre 2021].
- [22] Arduino, «ARDUINO,» ARDUINO, 24 DICIEMBRE 2019. [En línea]. Available: <https://www.arduino.cc/en/reference/SPI>. [Último acceso: 20 AGOSTO 2021].
- [23] E. J. Carletti, «Comunicación - Bus I2C Descripción y funcionamiento».
- [24] C. M. N. T. Felipe Castillo, «DISEÑO E IMPLEMENTACIÓN DE UN VEHÍCULO RC MEDIANTE TECNOLOGÍA WIFI,» 2013.
- [25] D. Alicia, Diseño Web con CSS, CreateSpace Independent Publishing Platform, 2014.
- [26] «Consulta planillas,» Empresa Eléctrica Quito, 2021. [En línea]. Available: <http://190.120.76.177:8080/consultaplanillas/servlet/gob.ec.sapconsultas>. [Último acceso: Septiembre 2021].
- [27] «hard zone,» [En línea]. Available: <https://hardzone.es/reportajes/tema/esp8266-2n2222-arduino/>. [Último acceso: SEP 2021].
- [28] «Aprendiendo Arduino,» [En línea]. Available: <https://aprendiendoarduino.wordpress.com/2018/11/23/thingspeak/>. [Último acceso: SEP 2021].
- [29] «Luis Llamas,» [En línea]. Available: <https://www.luisllamas.es/arduino-rfid-mifare-rc522/>. [Último acceso: SEP 2021].
- [30] «AV Electronics,» [En línea]. Available: <https://avelectronics.cc/producto/modulo-rfid-rc522/>. [Último acceso: SEP 2021].
- [31] «5hertz,» [En línea]. Available: https://www.5hertz.com/index.php?route=tutoriales/tutorial&tutorial_id=9. [Último acceso: SEP 2021].

- [32] «Arduino para 4º de ESO,» [En línea]. Available: https://angelmicelti.github.io/4ESO/ARD/el_circuito_integrado_l293d.html. [Último acceso: SEP 2021].
- [33] «AliExpress,» [En línea]. Available: <https://es.aliexpress.com/i/32856080561.html>. [Último acceso: SEP 2021].

ANEXOS

ANEXO 1: CERTIFICADO DE FUNCIONAMIENTO



ESCUELA POLITECNICA NACIONAL

Campus Politécnico "J. Rubén Orellana R

Quito, 01 de octubre de 2021

CERTIFICADO DE FUNCIONAMIENTO DE PROYECTO DE TITULACIÓN

Yo, Alan Daniel Cuenca Sánchez, docente a tiempo completo de la Escuela Politécnica Nacional y como director de este trabajo de titulación, certifico que he constatado el correcto funcionamiento de la **"IMPLEMENTACIÓN DE UN SISTEMA DOMÓTICO CONTROLADO MEDIANTE UN SERVIDOR IP"** el cual fue implementado por los estudiantes Ronal Vinicio Cordonez Arguello y Angel Armando Vela Tituaña.

El proyecto cumple con los requerimientos de diseño y parámetros necesarios para que los usuarios puedan usar el sistema con seguridad.

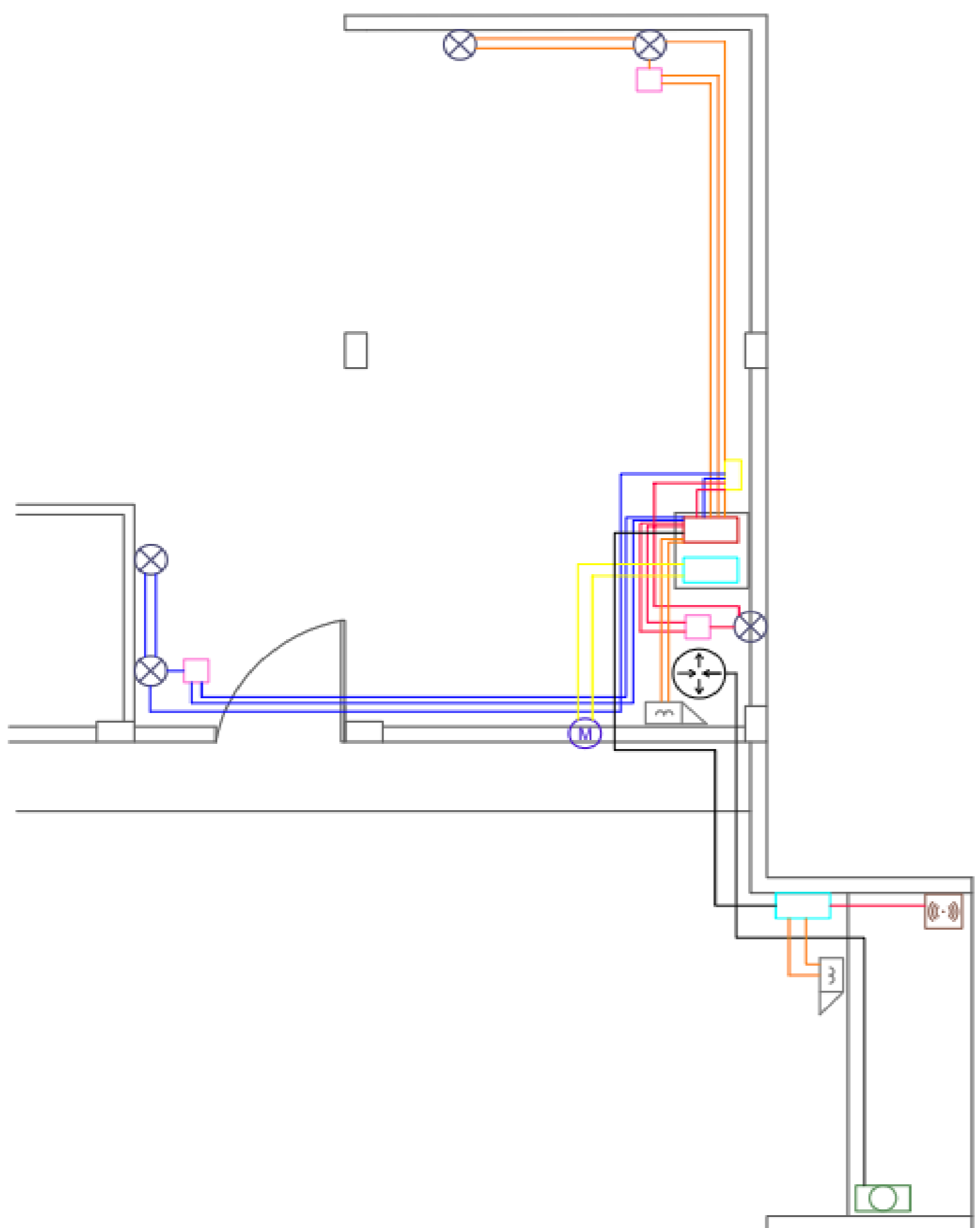
DIRECTOR

Ing. Alan Daniel Cuenca Sánchez

Ladrón de Guevara E11-253, Escuela de Formación de Tecnólogos, Oficina 22. EXT: 2736
email: alan.cuenca@epn.edu.ec

Quito-Ecuador

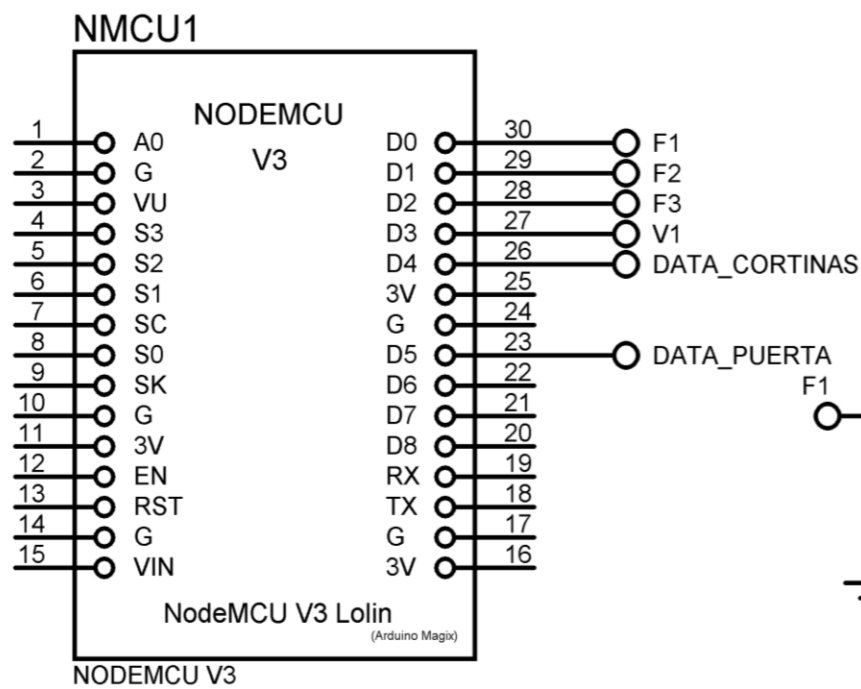
ANEXO 2: PLANO ESQUEMÁTICO



SIMBOLOGÍA	
	Cámara
	Router
	Focos
	Conmutador
	Alimentación
	Arduino nano
	Node MCU
	Motor
	Cerradura
	Rfid

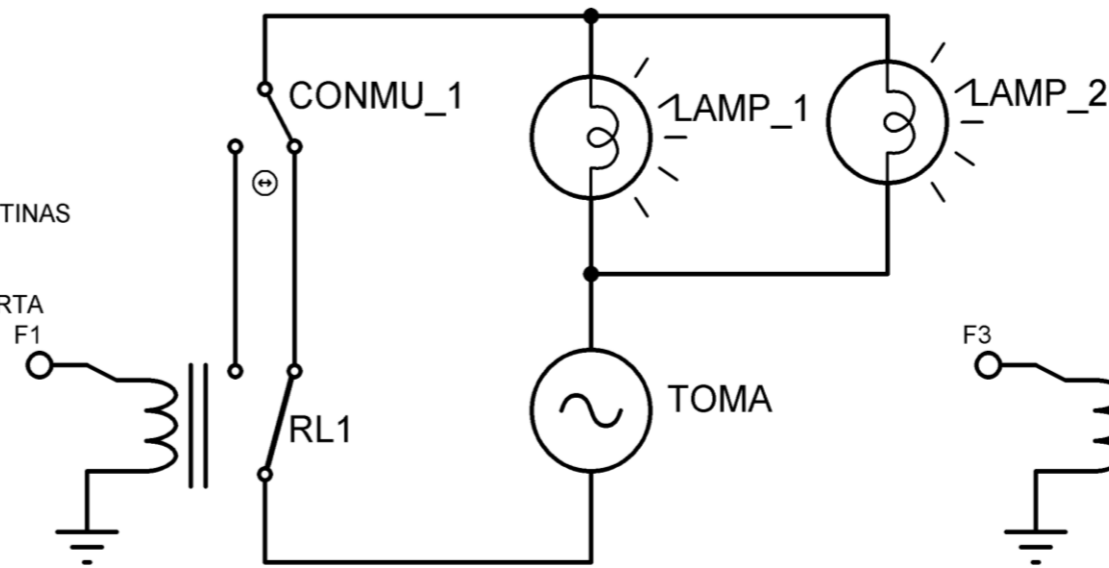
ESCUELA POLITÉCNICA NACIONAL - ESFOT		
PLANO ESQUEMÁTICO	TRATAMIENTO TÉRMICO: N/A	MATERIAL: N/A
	ESCALA: N/A	FORMATO: A3
DISEÑADO POR: Cordonez Ronal Vela Angel	FECHA: 01-10-2021	LÁMINA: 1

ANEXO 3: DIAGRAMAS DE CIRCUITOS

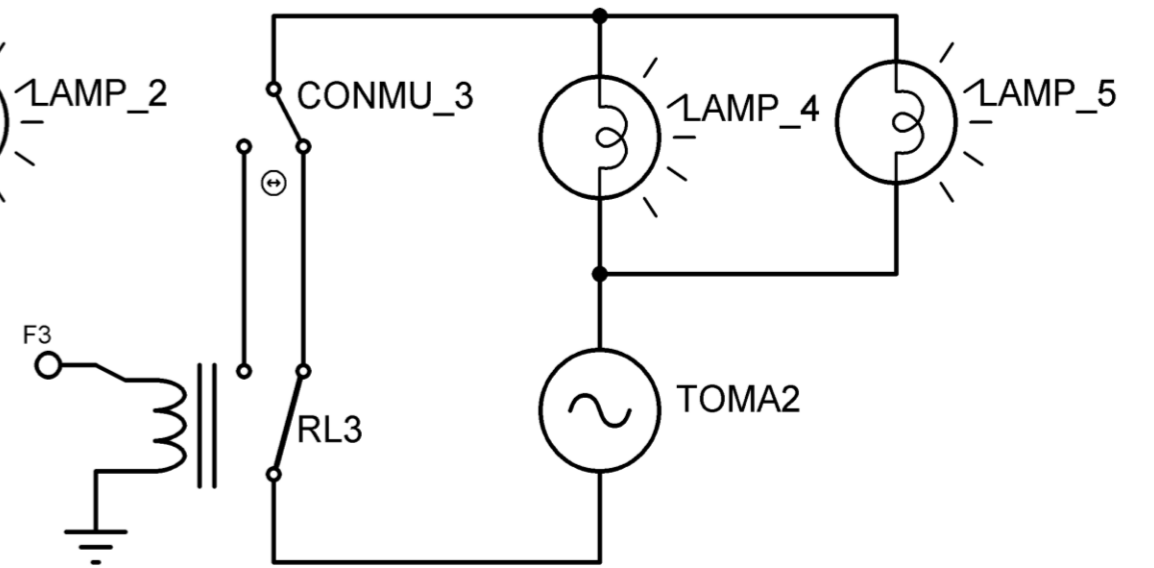


NODE MCU 6266

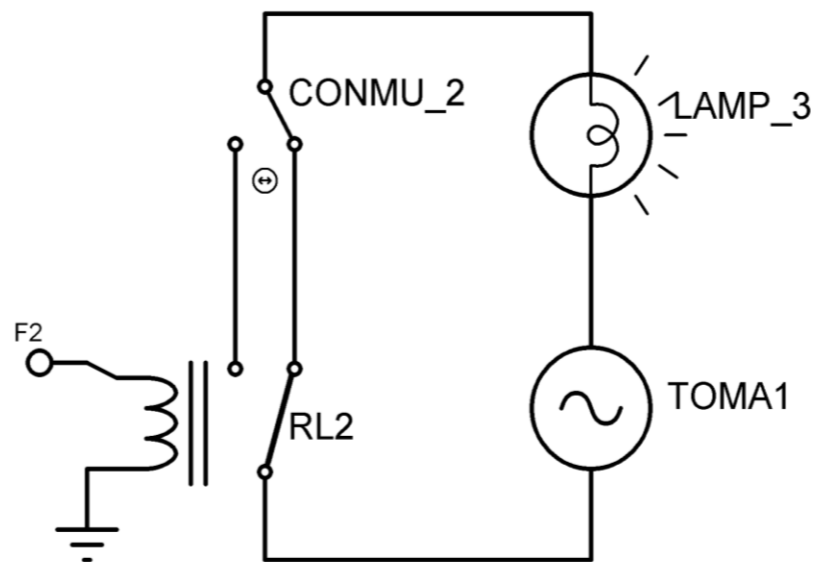
CIRCUITO DE COMEDOR



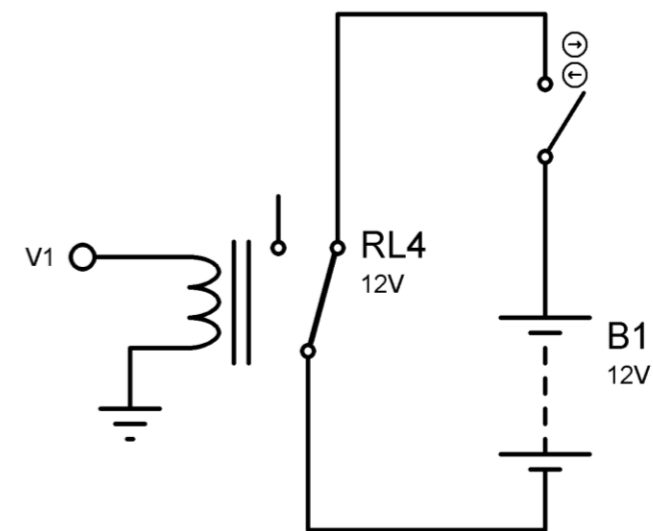
CIRCUITO DE PASILLO



CIRCUITO DE SALA



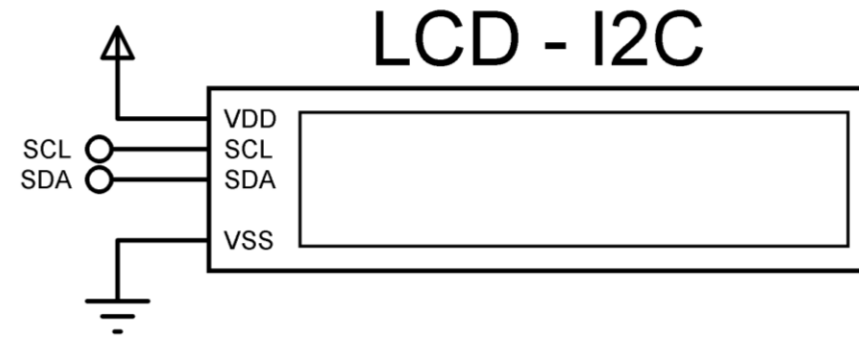
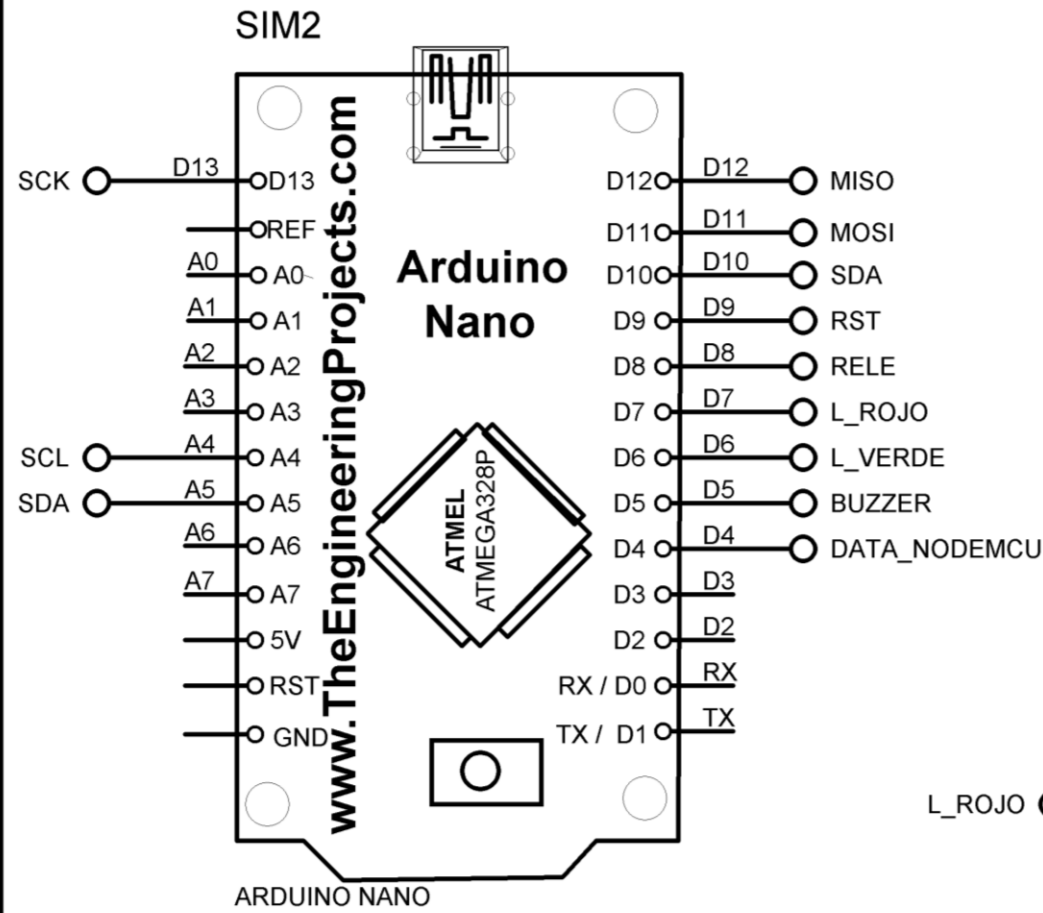
ACCESO VENTANA



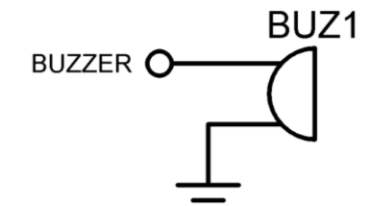
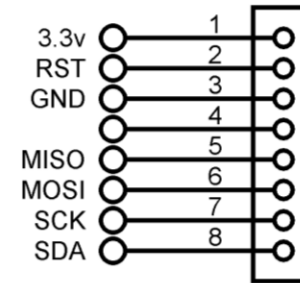
ESCUELA POLITÉCNICA NACIONAL - ESFOT

ESQUEMA CIRCUITAL MÓDULO CENTRAL	TRATAMIENTO TERMICO: N/A	MATERIAL: N/A
	ESCALA: 1:1	FORMATO: A3
DISEÑADO POR: Cordonez Ronal Vela Angel	FECHA: 01-10-2021	LÁMINA: 1

ARDUINO NANO

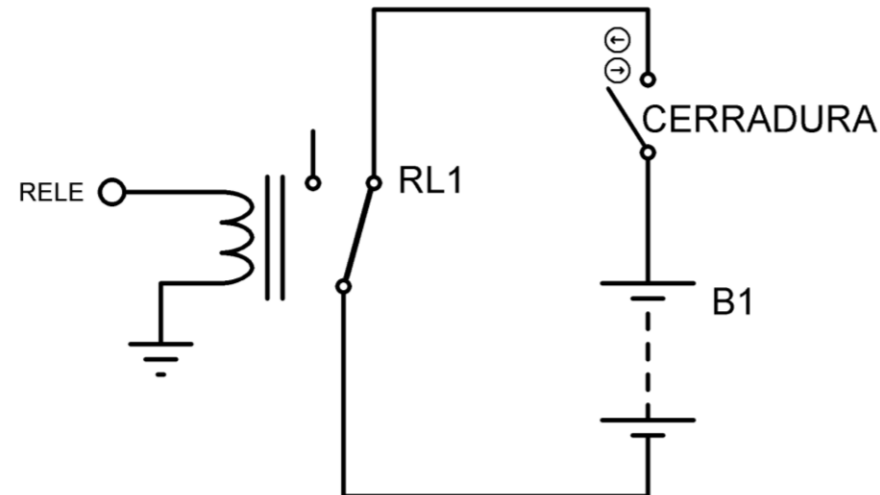
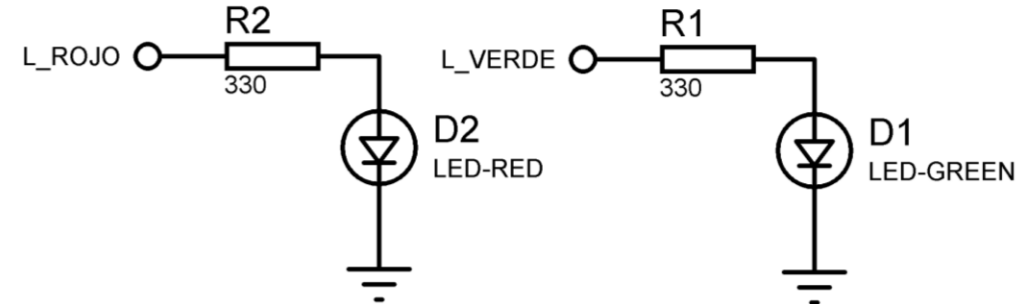


MÓDULO RFID



BUZZER

LEDS INDICADORES



MÓDULO RELÉ Y CERRADURA

ESCUELA POLITÉCNICA NACIONAL - ESFOT

ESQUEMA CIRCUITAL
ACCESO DE PUERTA

TRATAMIENTO
TERMICO: N/A

MATERIAL: N/A

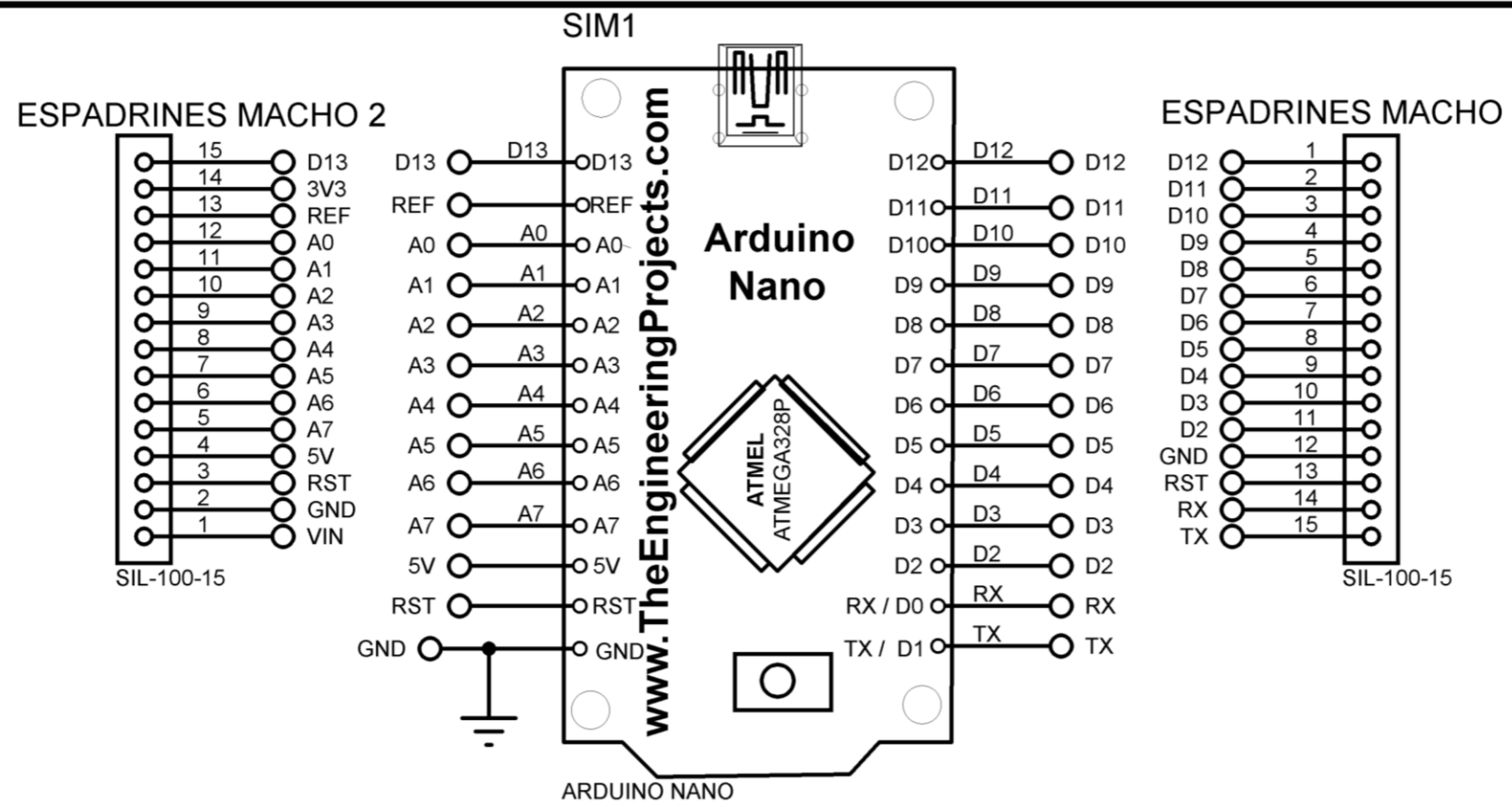
ESCALA: 1:1

FORMATO: A3

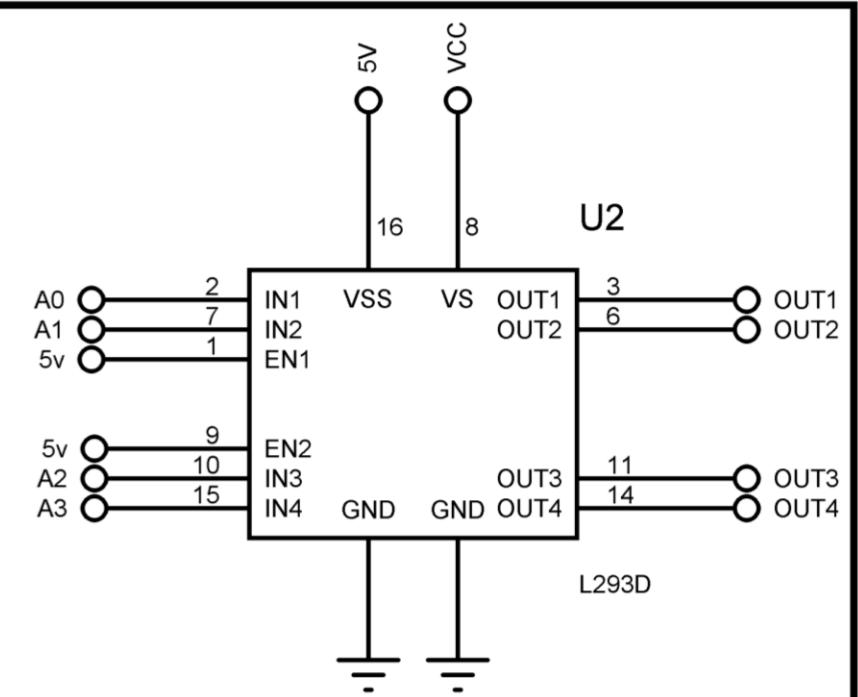
DISEÑADO POR:
Cordonez Ronal
Vela Angel

FECHA: 01-10-2021

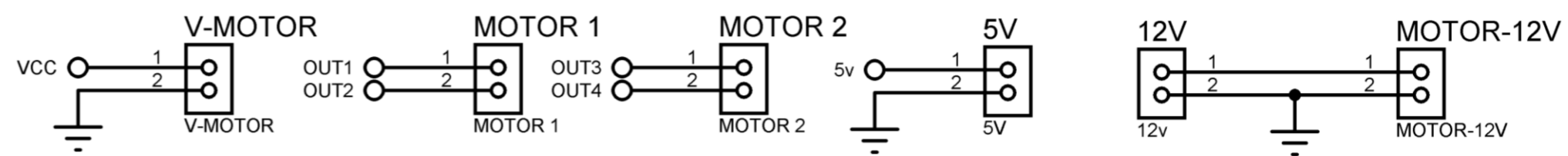
LÁMINA: 2



EXPANSOR-ARDUINO NANO



PUENTE H



SALIDAS DE VOLTAJE Y MOTORES

ESCUELA POLITÉCNICA NACIONAL - ESFOT		
ACCIONAMIENTO CORTINA	TRATAMIENTO TERMICO: N/A	MATERIAL: N/A
	ESCALA: 1:1	FORMATO: A3
DISEÑADO POR: Cordonez Ronal Vela Angel	FECHA: 01-10-2021	LÁMINA: 3

ANEXO 4: DIAGRAMAS DE FLUJO

DIAGRAMA DE FLUJO DEL MÓDULO CENTRAL

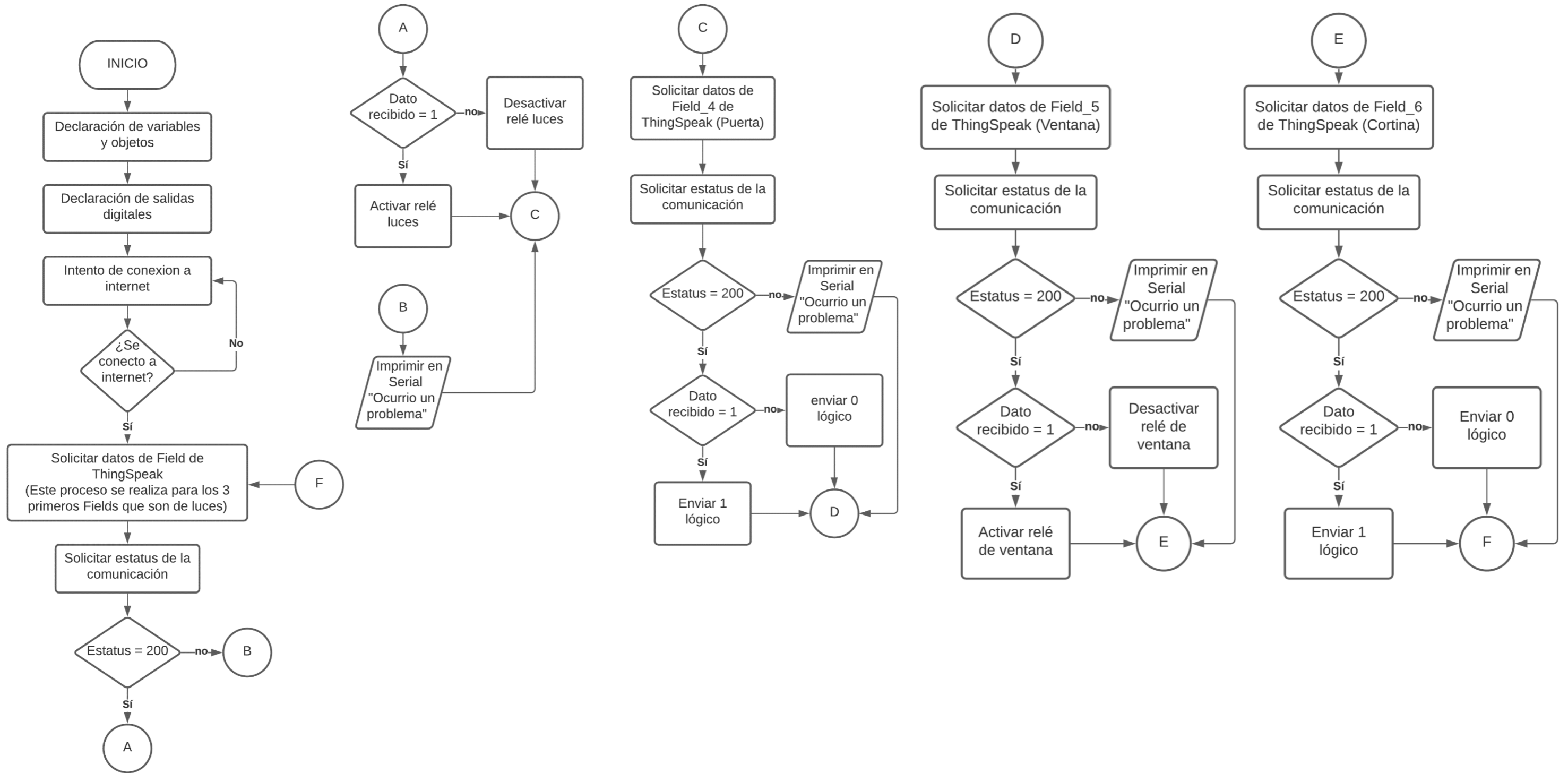


DIAGRAMA DE FLUJO DEL MÓDULO DEL CONTROL DE ACCESO

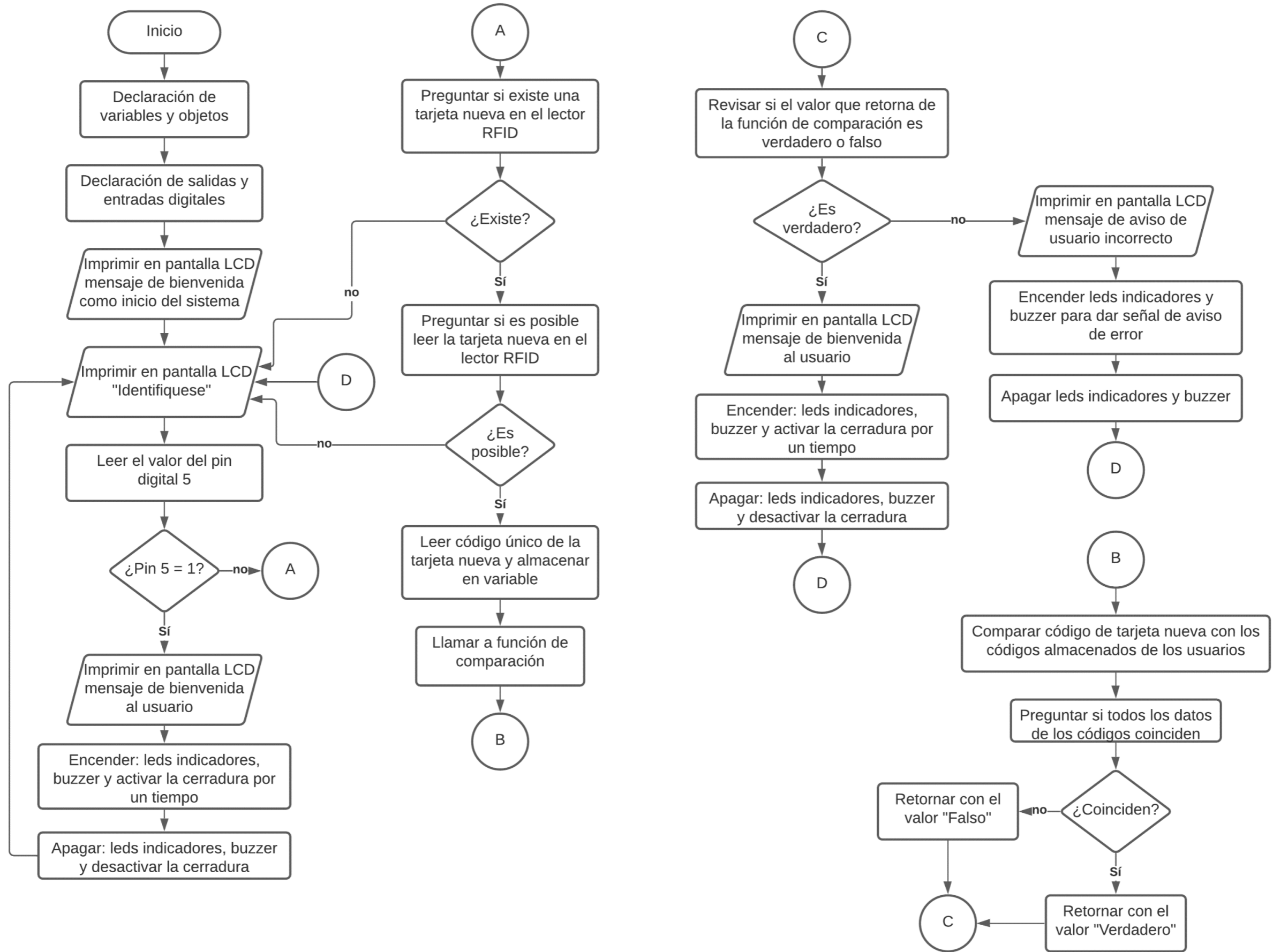
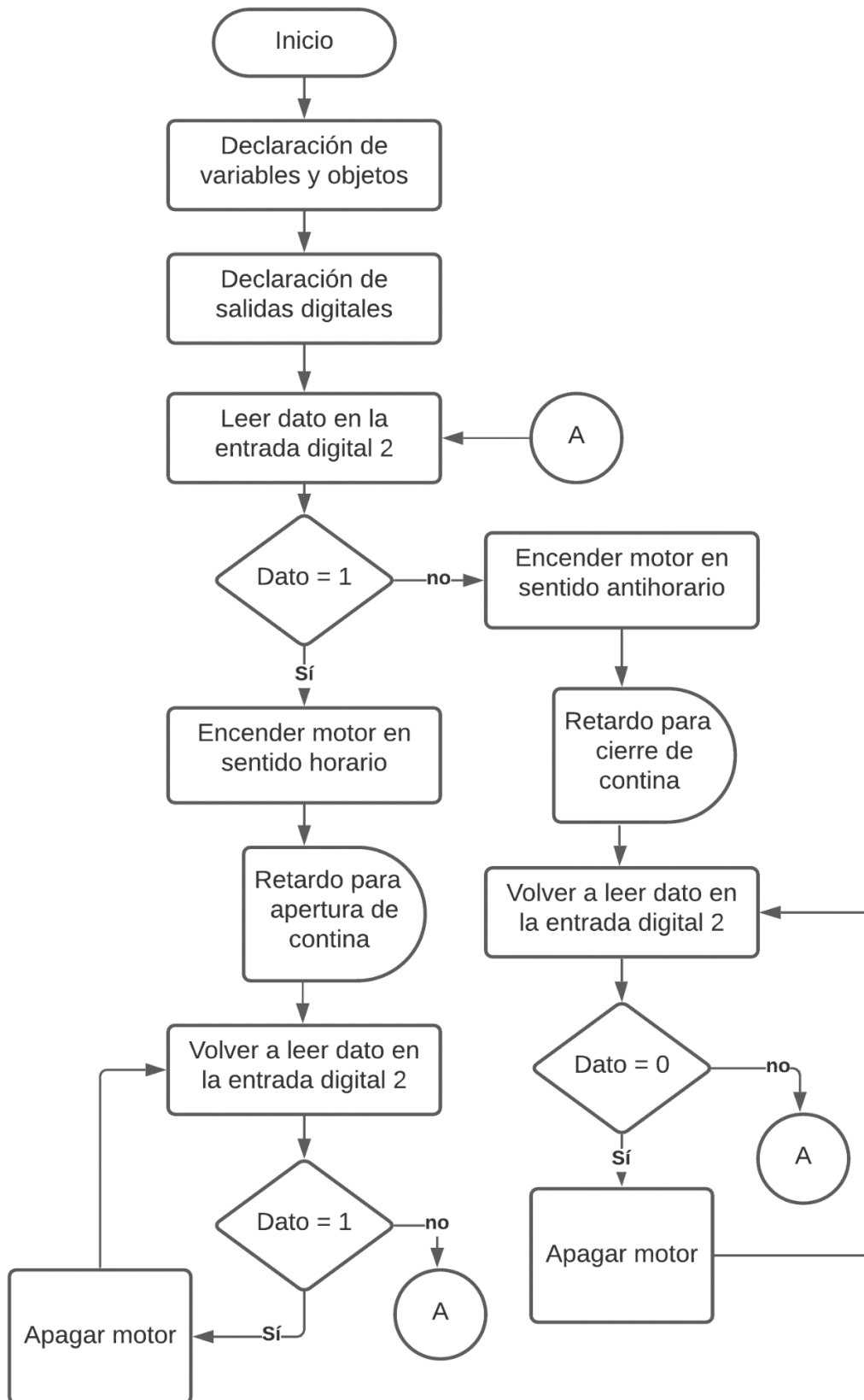


DIAGRAMA DE FLUJO DEL ACCIONAMIENTO DE LA CORTINA



ANEXO 5: COSTO TOTAL DEL PROYECTO

LISTA DE MATERIALES UTILIZADOS				
ETAPA	NOMBRE	CANTIDAD	VALOR U. (\$)	VALOR T. (\$)
MÓDULO CENTRAL	LÁMPARAS LEDS	3	3,90	11,70
	CONMUTADORES	3	2,50	7,50
	NODE MCU 8266	1	7,00	7,00
	CERRADURA ELECTROMAGNÉTICA PEQUEÑA	1	11,00	11,00
	BORNERAS X2	7	0,20	1,40
	HEADERS MACHO	1	0,45	0,45
	HEADERS HEMBRA	1	0,45	0,45
	MATERIAL PARA PCB	1	8,00	8,00
	MÓDULO RELÉ DE 4	1	5,00	5,00
	TOTAL			
CONTROL DE CORTINA	MOTOR DC	1	3,00	3,00
	ARDUINO NANO	1	7,50	7,50
	L293D	1	1,00	1,00
	BORNERAS X2	6	0,20	1,20
	HEADERS MACHO	1	0,45	0,45
	HEADERS HEMBRA	1	0,45	0,45
	MATERIAL PARA PCB	1	8,00	8,00
	TOTAL			
CONTROL DE ACCESO DE PUERTA	ARDUINO NANO	1	7,50	7,50
	EXPANSOR NANO	1	4,00	4,00
	MÓDULO RELÉ DE 1	1	2,50	2,50
	BORNERAS X2	2	0,20	0,40
	CERRADURA ELECTROMAGNÉTICA GRANDE	1	11,00	11,00
	LCD - I2C	1	7,00	7,00
	MÓDULO RFID	1	4,50	4,50
	TAGS PARA RFID	2	1,00	2,00
	LEDS	2	0,15	0,30
	BUZZER	1	0,75	0,75
	TOTAL			
SISTEMA DE SEGURIDAD	CÁMARA EXTERIOR HIKVISION	1	60,00	60,00
	PROTECTOR DE CÁMARA EXTERIOR	1	5,00	5,00
	TOTAL			
	CABLE GEMELO AWG 14	55	0,85	46,75
	CABLE TIMBRE	45	0,16	7,20
	CABLE FTP CAT 5 EXTERIOR	25	0,55	13,75

INSTALACIÓN	CABLE UTP CAT 5 EXTERIOR	3	0,45	1,35	
	CORTA PICOS	1	3,50	3,50	
	ENCHUFES	3	1,21	3,63	
	CAJA SOBREPUESTA PARA CONMUTADOR	3	1,10	3,30	
	CANALETA	2	0,80	1,60	
	TOMA CORRIENTE	1	1,70	1,70	
	CAPUCHÓN	2	0,10	0,20	
	RJ45	2	0,25	0,50	
	TAIPE	1	1,00	1,00	
	AISLANTE TERMORETRÁCTIL	2	0,30	0,60	
	CAJAS CONTENEDORAS	4	6,00	24,00	
	PEGAMENTO UHU	1	3,30	3,30	
	BISAGRAS	8	0,20	1,60	
	CANDADO	1	2,00	2,00	
	ALDABA	1	1,00	1,00	
	CARGADORES	2	5,00	10,00	
	OTRO	1	10,00	10,00	
	TOTAL				136,98
	TOTAL DEL PROYECTO				316,03