



**I** NSTITUTO  
**U** NIVERSITARIO  
**A** ERONAUTICO

**Instituto Universitario Aeronáutico**  
**Facultad de Ciencias de la Administración**  
**Carrera Ing. en Sistemas**



**Trabajo Final de Grado**

**Domótica 1.0- Casas Inteligentes – Innovadoras - Revolucionarias**

Autores:

Cascone Gabriel Amadeo  
Rizzo Gabriel Tomas

Tutor:

Ing. Juan Luis Castagnola

Córdoba –Argentina  
2018



# Índice

## Contenido

Resumen del Proyecto de Grado .....	12
Glosario de Palabras y/o Listado de símbolos .....	13
CAPITULO I .....	16
1 Capítulo 1 - Marco Referencial .....	17
1.1 Introducción.....	17
1.2 Problema.....	19
1.2.1 Situación Problemática .....	19
1.2.2 Antecedentes .....	20
1.3 Objetivos.....	21
1.3.1 Objetivo General.....	21
1.3.2 Objetivos Específicos .....	21
1.4 Hipótesis .....	22
1.5 Justificación.....	22
1.6 Alcance del Trabajo.....	23
1.7 Metodologías Aplicadas.....	23
1.7.1 Metodología Aplicada al Diseño y Creación del Ambiente Domótico. ....	24
1.7.2 Metodología Aplicada para el Desarrollo de la Interfaz del Sistema Domótico. ....	28
1.7.3 Metodología Aplicada para el Diseño de la Interfaz Electrónica y los Módulos de Control. ....	29
1.7.4 Metodología de V para el Testeo del Sistema .....	30
CAPITULO II .....	32
2 Capítulo II - Marco Teórico .....	33
2.1 Relevamiento y/o Diagnóstico y sus Conclusiones.....	33
2.1.1 Introducción.....	33
2.2 Domótica.....	34
2.2.1 Ventajas de un Sistema Domótico .....	35
2.2.2 Desventajas de un Sistema Domótico.....	36
2.3 Características Generales de un Sistema Domótico .....	36
2.3.1 Programación y Ahorro Energético.....	36

2.3.2	Confort .....	37
2.3.3	Seguridad .....	38
2.3.4	Comunicaciones .....	39
2.3.5	Accesibilidad .....	39
2.4	Arquitecturas .....	40
2.4.1	Arquitectura Centralizada:.....	40
2.4.2	Arquitectura Distribuida: .....	41
2.4.3	Arquitectura Mixta.....	42
2.5	Elementos de una Instalación Domótica .....	42
2.5.1	Central de gestión:.....	43
2.5.2	Nodo.....	45
2.5.3	Microcontroladores .....	45
2.5.4	Medios de interconexión .....	53
2.5.5	Protocolos de comunicación.....	55
2.5.6	Sensores.....	59
2.5.7	Actuadores.....	60
2.5.8	Software para control domótico.....	61
2.5.9	Interfaz de usuario .....	62
2.6	Elección Del Sistema Domótico .....	64
2.7	Conclusiones .....	65
CAPITULO III .....		67
3	Capítulo III - Propuesta de Sistema.....	68
3.1	Resolución de Etapas según Metodología Adoptada: Etapa de Relevamiento .....	68
3.1.1	Análisis de los objetivos del sistema y requisitos previos.....	68
3.1.2	Componentes para el Armado del Sistema Domótico.....	74
3.2	Resolución de Etapas según Metodología Adoptada: Etapas de Propuestas.....	84
3.2.1	Definir los pines de salida y entrada de datos del microcontrolador. ....	84
3.2.2	Localización de actuadores y sensores. ....	86
3.2.3	Diagrama de ubicación y cableado de nodos .....	90
CAPITULO IV.....		136
4	Capítulo IV – Etapa de Solución .....	137
4.1	Etapa de Solución.....	137

4.1.1	Maqueta finalizada con el sistema domótico funcionando.....	137
4.1.2	Interfaz para la manipulación del sistema.....	145
4.1.3	Código fuente del sistema domótico ( microcontrolador) .....	149
4.1.4	Riesgos del sistema.....	194
CAPITULO V.....		197
5	Capítulo V – Conclusión del Proyecto .....	198
5.1	Introducción.....	198
5.2	Conclusiones .....	198
5.2.1	Objetivo General Propuesto .....	198
5.2.2	Objetivos Específicos Propuestos .....	199
5.3	RECOMENDACIONES.....	200
Bibliografía .....		201
Anexos .....		203

## Índice de Imágenes

Figura 1	Proceso de Instalación Domótica: .....	24
Figura 2	Casa Domótica .....	35
Figura 3	Arquitectura Domótica Centralizada .....	41
Figura 4	Arquitectura Domótica Distribuida.....	41
Figura 5	Arquitectura Domótica Híbrida/Mixta.....	42
Figura 6	Gestión y Control Informático de Viviendas .....	44
Figura 7	Microcontrolador Arduino Mega.....	50
Figura 8	Sensores .....	60
Figura 9	Actuadores .....	61
Figura 10	Interfaz para el Control Domotico .....	62
Figura 11	Arquitectura Domótica Centralizada .....	75
Figura 12	Microcontrolador Arduino Mega.....	78
Figura 13	IDE Arduino .....	83
Figura 14	IDE Eclipse .....	83
Figura 15	Sistema de Gestión de Base de Datos.....	84
Figura 16	Fritzing.....	84
Figura 17	Diagrama de Conexión de Nodos.....	90

Figura 18 Diagrama General de la Vivienda.....	91
Figura 19 Baño de la Vivienda.....	92
Figura 20 Garaje de la Vivienda .....	93
Figura 21 Garaje de la Vivienda .....	93
Figura 22 Cocina de la Vivienda .....	94
Figura 23 Cocina de la Vivienda .....	94
Figura 24 Jardín de la Vivienda .....	95
Figura 25 Jardín de la Vivienda .....	95
Figura 26 Habitación de la Vivienda.....	96
Figura 27 Habitación de la Vivienda.....	96
Figura 28 Sala de la Vivienda .....	97
Figura 29 Sala de la Vivienda.....	97
Figura 30 Pasillo de la Vivienda.....	98
Figura 31 Pasillo de la Vivienda.....	98
Figura 32 Diagrama Ubicación de Dispositivos en el Garaje.....	99
Figura 33 Diagrama Ubicación de Dispositivos en la Cocina.....	100
Figura 34 Diagrama Ubicación de Dispositivos en la Habitación .....	101
Figura 35 Diagrama Ubicación de Dispositivos en el Salón .....	102
Figura 36 Diagrama Ubicación de Dispositivos en el Pasillo .....	103
Figura 37 Diagrama de Conexión de Sensores y Actuadores en el Garaje .....	104
Figura 38 Diagrama de Conexión de Sensores y Actuadores en el Garaje .....	105
Figura 39 Diagrama de Conexión de Sensores y Actuadores en la Cocina .....	106
Figura 40 Diagrama de Conexión de Sensores y Actuadores en la Cocina .....	107
Figura 41 Diagrama de Conexión de Sensores y Actuadores en la Habitación.....	108
Figura 42 Diagrama de Conexión de Sensores y Actuadores en la Habitación.....	109
Figura 43 Diagrama de Conexión de Sensores y Actuadores en el Salón .....	110
Figura 44 Diagrama de Conexión de Sensores y Actuadores en el Salón .....	111
Figura 45 Diagrama de Conexión de Sensores y Actuadores en el Pasillo.....	112
Figura 46 Diagrama de Conexión de Sensores y Actuadores en el Pasillo.....	113
Figura 74 Bosquejo Login.jsp .....	115
Figura 75 Bosquejo Index.jsp .....	115
Figura 76 Bosquejo Acciones.jsp.....	115
Figura 77 Bosquejo Miconfiguracion.jsp.....	116
Figura 78 Bosquejo Estadistica.jsp.....	116
Figura 79 Bosquejo Logs.jsp.....	116
Figura 80 Bosqujo Contacto.jsp .....	117
Figura 81 Bosquejo Usuario.jsp.....	117
Figura 48 Caso de Uso Consultar Microcontrolador .....	120
Figura 49 Caso de Uso Cambiar Estado Actuadores Web.....	121
Figura 50 Caso de Uso Cambiar Estado Actuadores Automático .....	122
Figura 51 Caso de Uso Cambiar Estado Actuadores Manuales .....	123
Figura 52 Caso de Uso Activar Alarma .....	124
Figura 53 Caso de Uso ABM Tareas .....	125

Figura 54 Caso de Uso Consular Sensores .....	126
Figura 55 Caso de Uso ABM Usuarios .....	127
Figura 56 Diagrama de Actividad Consultar Estado Sensores .....	128
Figura 57 Diagrama de Actividad Activar Alarma .....	129
Figura 58 Diagrama de Actividad Cambiar Estado de Actuadores Web .....	130
Figura 59 Diagrama de Actividad Cambiar Estado de Actuadores Manual .....	131
Figura 60 Diagrama de Actividad Cambiar Estado de Actuadores Automáticos .....	132
Figura 61 Diagrama de Actividad Cambiar Estado de Actuadores Web .....	135
Figura 62 Imagen Maqueta Completa .....	137
Figura 63 Imagen Maqueta Garaje	Figura 64 Imagen Maqueta Conexión .....
Figura 65 Imagen Maqueta Pasillo 1	Figura 66 Imagen
Maqueta Pasillo 2	139
Figura 67 Imagen Maqueta Pasillo 3.....	140
Figura 68 Imagen Maqueta Salón 1	Figura 69 Imagen Maqueta Salón 2.....
Figura 70 Imagen Maqueta Dormitorio 1	Figura 71 Imagen Maqueta Dormitorio 2.....
Figura 72 Imagen Maqueta Frente 1 .....	143
Figura 73 Imagen Maqueta Frente 2 .....	143
Figura 74 Captura Login.jsp .....	145
Figura 75 Captura Index.jsp .....	145
Figura 76 Captura Acciones.jsp.....	146
Figura 77 Captura Miconfiguracion.jsp.....	146
Figura 78 Captura Estadistica.jsp .....	147
Figura 79 Captura Logs.jsp .....	147
Figura 80 Captura Contacto.jsp .....	148
Figura 81 Captura Usuario.jsp.....	148

## Índice de Tablas

Tabla 1 Características Microcontrolador.....	49
Tabla 2 Características Arduino Mega .....	77
Tabla 3 Sensores Domóticos .....	80
Tabla 4 Actuadores Domóticos .....	82
Tabla 5 Pines de Entrada y Salida .....	85
Tabla 6 Sensores Utilizados .....	87
Tabla 7 Actuadores Utilizados.....	87
Tabla 8 Otros Dispositivos Utilizados.....	89
Tabla 9 Distribución Dispositivos Garaje .....	99
Tabla 10 Distribución Dispositivos Cocina .....	100
Tabla 11 Distribución Dispositivos Dormitorio .....	101
Tabla 12 Distribución Dispositivos Salón .....	102
Tabla 13 Distribución Dispositivos Pasillo.....	103



## **Dedicatoria**

El presente trabajo está dedicado con mucho cariño a mis abuelos, padres, compañera e hijos por el apoyo incondicional que me brindaron en cada momento de mi carrera.

Cascone Gabriel Amadeo

## **Dedicatoria**

Para mi madre, hermana y en especial a mi padre, por haberme apoyado incondicionalmente con mi carrera y desarrollo profesional.

Rizzo, Gabriel Tomas

## **Agradecimientos**

Un agradecimiento muy especial a todos nuestros mis familiares por el apoyo y la confianza que depositaron en nosotros, permitiéndonos lograr la culminación de este proyecto de investigación.

A todos los docentes de clases por transmitir sus conocimientos y permitir que tengamos una formación profesional.

Los compañeros y compañeras con los que compartimos de la carrera, gracias por el apoyo brindado.

Gracias.

## **Resumen del Proyecto de Grado**

En el presente trabajo se introduce al lector a conocer la estructura y las diferentes herramientas hardware y software para el desarrollo de un sistema domótico para el hogar.

La Domótica utiliza dos aspectos tecnológicos de gran desarrollo como son la electrónica y la informática. Estas dos tecnologías logran dos cosas muy importantes, el control y la automatización de los diferentes dispositivos o aparatos que funcionan dentro de una vivienda familiar o edificio inteligente.

El sistema a desarrollar engloba varios elementos como son: el confort, seguridad, accesibilidad, comunicación y la gestión energética. Este informe describe y explica el trabajo desarrollado como tesis de grado, cuyo objetivo es lograr la implantación de un sistema domótico en una vivienda a escala.

Para ello se utilizan 2 herramientas principales:

El hardware a utilizar es un microcontrolador Arduino MEGA, adaptando sensores y actuadores según los requerimientos del sistema.

En cuanto a software vamos a utilizar; Eclipse como entorno de desarrollo para la creación de la interfaz web y con la que usuario manipulara el sistema; y Arduino IDE para programar las funciones básicas a nuestro microcontrolador.

Estos son los equipos básicos para implementar las funciones que debe cumplir el sistema.

Los resultados obtenidos en cuanto al aspecto técnico y teórico en que se sustenta la domótica son de un valor tal que las aplicaciones prácticas pueden ser inmediatas. El uso de una maqueta con algunos elementos de control, junto al software de gestión, permite demostrar la viabilidad técnica del sistema domótico.

## **Glosario de Palabras y/o Listado de símbolos**

- Domótico: los sistemas capaces de automatizar una vivienda o edificación de cualquier tipo, aportando servicios de gestión energética, seguridad, bienestar y comunicación, y que pueden estar integrados por medio de redes interiores y exteriores de comunicación, cableadas o inalámbricas, y cuyo control goza de cierta ubicuidad, desde dentro y fuera del hogar.
- Arduino: es una compañía open source y open hardware, así como un proyecto y comunidad internacional que diseña y manufactura placas de desarrollo de hardware para construir dispositivos digitales y dispositivos interactivos que puedan censar y controlar objetos del mundo real.
- Microcontroladores: Un microcontrolador es un circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria. Está compuesto de varios bloques funcionales, los cuales cumplen una tarea específica. Un microcontrolador incluye en su interior las tres principales unidades funcionales de una computadora: unidad central de procesamiento, memoria y periféricos de entrada/salida.
- Telecontrolados: Consiste en el envío de indicaciones a distancia mediante un enlace de transmisión por ejemplo, a través de cables, radio, dirección IP, etc. utilizando órdenes enviadas para controlar un sistema o sistemas remotos que no están directamente conectados al lugar desde donde se envía el telecontrol.
- TIC's: La tecnología de la información y comunicación son un conjunto de servicios de redes y aparatos que tiene como objetivo mejorar la calidad de

vida del ser humano dentro de un entorno, la tecnología de la información son aquellas herramientas computacionales e informáticas que procesan, almacenan y recuperan información, es muy útil para el estudiante pero ellos deben de saber.

- SPI: El Bus SPI ( Serial Peripheral Interface) es un estándar de comunicaciones, usado principalmente para la transferencia de información entre circuitos integrados en equipos electrónicos. El bus de interfaz de periféricos serie o bus SPI es un estándar para controlar casi cualquier dispositivo electrónico digital que acepte un flujo de bits serie regulado por un reloj.
- Radiofrecuencia:denominado espectro de radiofrecuencia, es un término que se aplica a la porción menos energética del espectro electromagnético, situada entre los 3 hercios (Hz) y 300 gigahercios (GHz).
- TCP/IP:La familia de protocolos de internet es un conjunto de protocolos de red en los que se basa internet y que permiten la transmisión de datos entre computadoras.

En ocasiones se le denomina conjunto de protocolos TCP/IP, en referencia a los dos protocolos más importantes que la componen, que fueron de los primeros en definirse, y que son los dos más utilizados de la familia:

TCP: protocolo de control de transmisión.

IP: protocolo de internet.

- Shield: Las placas Arduino pueden conectarse con módulos adicionales denominados shields, dichos shields aumentan las características técnicas de la placa Arduino en uso, debido a que poseen circuitos específicos que añaden una o más funcionalidades extras a la placa Arduino nativa en la cual se utilice, también se les conoce como placas de expansión. La mayoría de estos shields se conectan a través de un bus serie, aunque existen

también aquellas que emplean conexión mediante el bus UART (Universal Asynchronous Receiver-Transmitter), así como con el bus SPI.

- Relay: El relé es un dispositivo electromagnético. Funciona como un interruptor controlado por un circuito eléctrico en el que, por medio de una bobina y un electroimán, se acciona un juego de uno o varios contactos que permiten abrir o cerrar otros circuitos eléctricos independientes.
- Driver: Un controlador de dispositivo o manejador de dispositivo es un programa informático que permite al sistema operativo interactuar con un periférico, haciendo una abstracción del hardware y proporcionando una interfaz para utilizar el dispositivo.

**CAPITULO I**  
**MARCO REFERENCIAL**



# 1 Capítulo 1 - Marco Referencial

## 1.1 Introducción

La revolución tecnológica que el mundo ha venido experimentando durante los últimos 50 años ha impactado cada una de las facetas del diario vivir de los seres humanos. Tras el marcado desarrollo de la electrónica, a partir de la década de los 60 y el desarrollo de nuevas tecnologías orientadas a las comunicaciones y aplicadas a la vida, generaron un cambio en la mayoría de las personas que conviven en un entorno globalizado, donde la conectividad juega un papel preponderante y hasta el más simple elemento como un dispositivo móvil funciona como transición a la dinámica de inter conectividad global.

En la actualidad los escenarios donde los seres humanos interactúan son del tipo inteligente, lo que quiere decir que están dotados de un sinnúmero de funciones para su bienestar. Esta palabra, inteligente, ha llegado al hogar o vivienda transformando su simple definición según el Diccionario de la Real Academia Española: “Centro de ocio en el que se reúnen personas que tienen en común una actividad, una situación personal o una procedencia”, en escenarios vivos, donde el hombre es el principal actor.

Las viviendas con la última tecnología pueden ser denominadas viviendas inteligentes. Una vivienda inteligente es aquella que permite manejar dispositivos tal como iluminación, riego, climatización y seguridad, de manera manual o automática. Así también, el manejo de artefactos del hogar como la cafetera, el lavarropas, el equipo de música, la tv, entre otros, a través de teléfonos inteligentes con sistemas operativos conectados a internet o pantallas con sistemas diseñadas específicamente para el manejo de estos dispositivos.

La tecnología incluida en el hogar es algo que hoy en día está avanzando con rapidez ya que encuentra posibilidades apasionantes para todo tipo de usuario, entre estas nuevas tecnologías se encuentra la Domótica.

Para entender lo que es la domótica se explica que deriva de la unión de la palabra “domo” y el sufijo “tica” que provienen del latín y significan casa y automática respectivamente. Algunos autores difieren entre “tic”, proveniente de tecnologías de la informática y comunicación; y “a” de automatización. Las TIC’s se encargan del desarrollo, procedimiento, almacenamiento y distribución de los datos por medio de sistemas informáticos como computadoras, teléfonos celulares, televisión, radio, etc.

Esta nueva área de la ingeniería conocida como domótica o como ciencia de la innovación, está creciendo a niveles exponenciales, y las principales empresas diseñadoras de electrodomésticos están dotándolos de inteligencia para aumentar las funciones que tiene cada uno de ellos y cada vez más suplir las necesidades que se presentan en el hogar.

La domótica es un concepto que se refiere a la integración de las distintas tecnologías en el hogar mediante el uso simultáneo de la electricidad, la electrónica, la informática y las telecomunicaciones. Su fin es mejorar la seguridad, el confort, la flexibilidad, las comunicaciones, el ahorro energético, facilitar el control integral de los sistemas para los usuarios y ofrecer nuevos servicios, creando ambientes para adaptarlos al estilo y forma de vida de cada persona o grupo familiar.

Por otra parte, cabe señalar que en la domótica, las redes están interconectadas y han ido creciendo con la aparición de internet, y por tanto este desarrollo ha permitido el crecimiento de los procesos de domotización. Una de las grandes características que se tiene en las redes de telecomunicaciones es que desde cualquiera de ellas es posible comunicarse con un host que posea el protocolo IP, que es el medio básico de conexión en el mundo de las telecomunicaciones electrónicas.

Así, mediante protocolos, tales como IP, es posible implementar sistemas de comunicación para sistemas domóticos, como por ejemplo los procesos telecontrolados de climatización, apertura de puertas, iluminación automática, entre otros.

## **1.2 Problema**

### **1.2.1 Situación Problemática**

En la actualidad la dinámica productiva enfrenta a las personas e instituciones al desafío de la eficiencia, la velocidad y la ubicuidad; en el mundo moderno cualquier actividad que no pueda ser realizada en forma eficiente mediante herramientas remotas representa una enorme desventaja, tanto competitivamente como de calidad de vida.

Dentro de esta perspectiva, la automatización de los espacios, su dinamización con la incorporación de herramientas, las tecnologías de la información y las telecomunicaciones son una necesidad inminente.

Tareas tan simples como controlar el acceso al hogar, mantener climatizada las habitaciones y actividades comunes como abrir o cerrar una ventana o una puerta para personas con alguna dificultad física son arduas de realizar pero con la implementación de la domótica en el hogar se facilitan estos quehaceres.

En el mercado actual se cuenta con un gran número de aplicaciones de muy alto nivel sobre el tema, las cuales en su mayoría tienen costos elevados y están pensadas más en generar una experiencia de lujo al cliente que en facilitar el desarrollo de una actividad específica a las personas.

Las características anteriormente descritas ponen en evidencia un desafío mayor: poder implementar un sistema domótico orientado a facilitar actividades domésticas o empresariales asociados al control de acceso y seguridad de los espacios contando con un bajo costo, alto nivel de calidad, confiabilidad y seguridad.

## **1.2.2 Antecedentes**

Las edificaciones comúnmente son construidas de forma tradicional, es decir poseen una infraestructura con instalaciones no monitoreadas, ni controladas que puede representar aparentemente un servicio de calidad pero en comparación con los avances que existen en la actualidad en el campo de la tecnología de sistemas de monitoreo y control, existe una gran diferencia por la eficiencia que éstos representan tanto para la calidad de vida y el ahorro económico que implica para los usuarios en su vida diaria.

Tomando en cuenta esta consideración aparece la edificación domótica, que de un tiempo atrás, comienzan a surgir paulatinamente edificaciones tanto públicas como privadas, con ciertas características inteligentes aún en un grado menor, consiguiendo así una mayor calidad de vida para sus usuarios a través de la implementación de tecnología obteniendo una reducción del tiempo empleado en tareas, un aumento de la seguridad y ahorros en los consumos de agua, electricidad, etc.

Para conseguir todos estos beneficios, una edificación domótica tendrá una instalación especial de nuevos componentes conectados entre sí y que serán los encargados de recoger información del entorno como temperatura, iluminación, etc., procesarla y actuar en consecuencia dotando a la edificación de cierta inteligencia y automatizando tareas que hasta ahora se venían haciendo de forma manual.

En Argentina la domótica surge de la mano de empresas de tecnología que incorporan el concepto y lo desarrollan.

A comienzo de la década de 1990, estas empresas comienzan a hablar de domótica al referirse a la casa del futuro, y a realizar algunas aplicaciones de carácter parcial, participando en ferias y notas periodísticas que colaboran con la difusión del nuevo concepto. Conforme avanzan los años 90, las instalaciones se hacen más frecuentes e importantes comenzando a expandirse el mercado argentino, lo cual posibilita, llegado el fin del milenio, la aparición de otras

compañías que comienzan a incorporarlo entre sus servicios o realizan desarrollos propios.

La crisis económica Argentina de fines del 2001 paraliza este desarrollo que recién se recupera con la expansión que se da en el área de la construcción casi tres años después.

En la actualidad una edificación domótica es una herramienta que toma cada día más fuerza, al ser más segura, más cómoda, con mayores posibilidades de comunicación y que consume menos energía que una edificación tradicional.

## **1.3 Objetivos**

### **1.3.1 Objetivo General**

El objetivo general del proyecto es realizar el estudio, diseño y aplicación de un sistema domótico a escala que permita en una vivienda estándar reducir el consumo de energía, aumentar el confort, accesibilidad, comunicación y seguridad de la vivienda.

### **1.3.2 Objetivos Específicos**

- ✓ Identificar y desarrollar los servicios que el sistema controlará.
- ✓ Diseñar e implementar la interface electrónica que permita la recolección de datos y accionar de dispositivos electrónicos.
- ✓ Diseñar e implementar la interface visual mediante *software* con indicadores gráficos y sonoros que permitan un inmediato conocimiento de las condiciones reales del sistema en general.
- ✓ Diseñar e implementar las pantallas, formularios y reportes requeridos para el exacto control y supervisión del sistema.
- ✓ Implantar el sistema en una maqueta diseñada para tal fin donde las señales de salida sean visualizadas en tiempo real y las señales de entrada sean un fiel patrón de las originadas por sensores reales.

## **1.4 Hipótesis**

Realización de un sistema de control domótico utilizando una aplicación web que permite controlar a través de la red, el encendido y/o apagado de luces, apertura/cierre de puertas, regular la calefacción o ventilación del hogar y activar/desactivar el sistema de alarmas.

## **1.5 Justificación**

En la actualidad se han desarrollado nuevas tecnologías aplicables a las edificaciones, las cuales permiten que las personas disfruten de un estilo de vida mejor. Por cuanto la domótica es la integración de las tecnologías aplicada para la automatización en viviendas.

Se realizará por tanto un estudio de los diferentes sistemas de monitoreo y control para la automatización de la edificación, como son: sistema de iluminación, sistema de monitorización de la vivienda, sistema de climatización de ambientes, entre otros.

La domótica es una de las áreas que facilita los procesos de automatización y operación de mecanismos electrónicos, logrando de esta manera integrar las diferentes tecnologías que se encuentran al interior de la edificación, enfocado en mejorar la calidad de vida de las personas cuyo propósito principal es ofrecer servicios de confort, seguridad y ahorro energético.

Por esta razón, y como parte del proceso de formación se ha elegido este proyecto orientado a implementar un sistema domótico para el hogar, que para fines académicos se centrará en un sistema de control de accesos, iluminación y seguridad, aprovechando la integración de actuadores, sensores y sistemas de comunicación.

## **1.6 Alcance del Trabajo**

El sistema llevara el control de funciones en la casa, se podrá realizar por una aplicación web, donde el usuario podrá ingresar con un nombre y contraseña, con el cual administrara su casa a través de una red interna o externa.

El sistema domótico puede monitorear y controlar específicamente los elementos que a continuación se detallan para cada servicio básico dado:

- Apertura/Cierre de puertas y portones de ingreso.
- Encendido/Apagado de equipos eléctricos.
- Climatización de ambientes según temperatura.
- Programar encendido/apagado de equipos como sistemas de riego o limpieza de piletas entre otros.
- Monitorear de consumo de equipos.
- Consulta de actividades realizadas en el hogar según usuario, fecha o actividad.
- Registro de actividades a realizar en el hogar.

De igual manera, los programas elaborados tanto para la tarjeta de interfaz como para la computadora, que sirve como central de gestión, se diseñan en función de los elementos utilizados. Cualquier cambio de los programas, obligará a la revisión y adaptación de los sensores y actuadores. Pudiendo inclusive, si el caso lo exige, al diseño y elaboración de nuevos elementos.

## **1.7 Metodologías Aplicadas**

Para el desarrollo del sistema domótico, realizamos la división del sistema en tres partes a las cuales le aplicaremos distintas metodologías para el desarrollo e implementación:

- ✓ Diseño y creación del ambiente domótico.
- ✓ Diseño y desarrollo de interfaz web.
- ✓ Diseño y programación de la interfaz electrónica para el microcontrolador.

### 1.7.1 Metodología Aplicada al Diseño y Creación del Ambiente Domótico.

La metodología a seguir para el diseño y creación es:



Figura 1. Proceso de una instalación Domótica.

Figura 1 Proceso de Instalación Domótica:

Cada paso de ésta metodología, realizada correctamente, nos conduce a la realización efectiva de un sistema domótico donde la automatización y control sean evidentes, así también donde los detalles de instalación, manipulación y usabilidad formen parte integral del mismo.

En líneas generales ésta metodología involucra tres etapas para la correcta implementación de una vivienda Domótica:

Los pasos necesarios para lograr con éxito cada etapa se mencionan a continuación:

#### 1.7.1.1 Etapas de Relevamiento.

1. Análisis de los objetivos del sistema y requisitos previos.
2. Selección del sistema domótico idóneo.



## **Descripción de las Etapas del Relevamiento**

### 1. Análisis de los objetivos del sistema y requisitos previos.

Realizar un análisis detallado de las necesidades, que se conviertan en “requisitos”. Determinar los objetivos principales que cumplirá el sistema para evitar problemas en las etapas de propuesta y solución.

En este análisis comprobar que funciones son efectivas de realizar por el sistema, para luego desarrollarlas.

Conocer el equipamiento domestico a controlar es también básico para orientar las aplicaciones, determinando sus posibilidades efectivas y prácticas.

### 2. Selección del sistema domótico.

Conocidos y cerrados los objetivos y requisitos del sistema es el momento de elegir la mejor solución que permita dar respuesta a dichos requisitos. Para ello, es básico:

- Definir el tipo de arquitectura
- Central de gestión
- Microcontrolador
- Medios de interconexión
- Protocolo
- Sensores
- Actuadores
- Interfaz
- Software para el control domótico

### ***1.7.1.2 Etapa de Propuesta.***

Tras la fase de relevamiento, y disponiendo de una descripción detallada de la toda instalación domótica, es el momento de llevarla a acabo. Para ello es preciso considerar cuatro sub-etapas:

1. Definir los pines de salida y entrada de datos del microcontrolador.
2. Localización de actuadores y sensores.
3. Diagrama de ubicación y cableado de nodos
4. Bosquejo de la interfaz.
5. Casos de Uso
6. Diagramas de Actividades
7. Lógica del controlador

#### **Descripción de las Etapas del Relevamiento**

1. Definición los pines de salida y entrada de datos Del microcontrolador a utilizar:

Según el microcontrolador a utilizar definir cuáles serán los pines analógicos y digitales que se utilizaran para la interconexión con los sensores y actuadores.

2. Localización de actuadores y sensores.

Análisis de la mejor ubicación de sensores y actuadores en la vivienda, para los cuales es preciso un conocimiento técnico de las prestaciones de los sensores elegidos por el sistema domótico. Prestar especial atención a dichas prestaciones es fundamental para asegurar que el sistema domótico pueda funcionar de una forma adecuada.

3. Diagrama de ubicación y cableado de nodos

Consta de los siguientes puntos a tener en cuenta:

- Diagrama del sistemas de cableado.
- Disposición de todos los dispositivos (sensores, actuadores, elementos auxiliares de actuación, etc.).
- Elementos propios de redes domésticas (por ejemplo, dispositivos de red, estaciones Wi-Fi, etc.).
- Central de gestión.
- Conexión de todo el sistema domótico.

#### 4. Bosquejo de la interfaz.

Maquetación de las pantallas de la página web, donde el usuario interactúa con el sistema domótico, ubicación de los controles, menús y sub-menú

#### 5. Casos de Uso

Esquema que facilita la comprensión en el funcionamiento del sistema domótico, actividades que puede realizar el usuario y el control domótico.

#### 6. Diagramas de Actividades

Estos diagramas ayudan a comprender las acciones básicas para el funcionamiento del sistema domótico, mensajes en los cuales interactúan el usuario-web-control domótico y los sensores.

#### 7. Lógica del controlador

Diagrama de flujo que explica la lógica implementada en el microcontrolador para la implementación del sistema domótico.

### ***1.7.1.3 Etapa de Solución***

1. Maqueta finalizada con el sistema domótico funcionando.
2. Interfaz para la manipulación del sistema.
3. Código fuente del sistema domótico (web y microcontrolador)

4. Riesgos del sistema.

## **Descripción de las Etapas del Relevamiento**

1. Sub-Etapa de Maqueta finalizada con el sistema domótico funcionando.

Realizar la puesta en marcha definitiva del sistema domótico con los suministros básicos (electricidad, agua y gas) comprobando el correcto funcionamiento de las funcionalidades descritas.

2. Sub-Etapa Interfaz para la manipulación del sistema.

Demostración de las interfaces con que cuenta el sistema domótico para que el usuario interactúe, realizando el ingreso al sistema, demostrando funcionalidad y facilidades en el uso del sistema, así como también ingreso a los menús y sub-menús con que cuenta el sistema.

3. Sub-Etapa Código fuente del sistema domótico (web y microcontrolador)

Se anexa el código fuente del sistema, tanto de la web que funciona como interfaz, así como también las tablas y scripts de la base de datos y la lógica implementada para que el microcontrolador funcione correctamente.

4. Sub-Etapa Riesgos del sistema.

Se listan los posibles riesgos con su ponderación que tiene el sistema, los cuales no se tuvieron en cuenta en el momento del desarrollo.

### **1.7.2 Metodología Aplicada para el Desarrollo de la Interfaz del Sistema Domótico.**

La elaboración de la interfaz para el control y monitoreo del sistema domótico seguirá los siguientes pasos:

- ✓ Requerimientos funcionales necesarios para el nuevo sistema.

- ✓ Definición de las entradas y salidas.
- ✓ Diseño de los formularios y ventanas para las funciones definidas.
- ✓ Diseño de los controles de seguridad y resguardo de la información.
- ✓ Ajustes finales de los errores detectados, cambios definitivos.
- ✓ Implantación del sistema
- ✓ Pruebas de funcionamiento de todo el programa.
- ✓ Elaboración de toda la documentación necesaria para el programa; manual de usuario, manual de sistema y manual de programación.

La elaboración de la lógica del microcontrolador, una vez seleccionado por su capacidad de memoria y el número de E/S, seguirá los siguientes pasos:

- Determinar el número de salidas y entradas que manejará el microcontrolador.
- Definir qué tipo de señales que manejarán los puertos del microcontrolador; analógicas o digitales.
- Editar el programa principal en un programa de edición de texto.
- Compilar el programa para detectar posibles errores.
- Corregir los errores detectados en la compilación.
- Elaborar pruebas de funcionamiento del programa cargado en el microcontrolador en un protoboard (tabla de conexiones).
- Una vez verificado conectar los dispositivos de salida necesarios para la activación de los diferentes módulos de control.

### **1.7.3 Metodología Aplicada para el Diseño de la Interfaz Electrónica y los Módulos de Control.**

Para los módulos de control se seguirán los siguientes pasos

- ✓ Realizar el esquema del circuito en papel con todas sus entradas y salidas identificadas. Sus elementos, valores nominales y dimensiones.
- ✓ Montar el circuito en un protoboard, tal como señala el esquema dibujado.
- ✓ Energizar con todos los voltajes requeridos y probar el circuito tanto para entradas simuladas como para las salidas obtenidas con dichas entradas.
- ✓ Realizar las correcciones necesarias en caso de que las salidas no se ajusten a lo establecido en el diseño.
- ✓ Realizar la edición del circuito con todos sus elementos utilizando para ello un programa de edición de circuitos electrónicos.
- ✓ Probar todos los componentes en conjunto, alimentándola con los voltajes necesarios y sometiéndola a pruebas de entrada y verificación de salidas.

#### **1.7.4 Metodología de V para el Testeo del Sistema**

El modelo en V dice que las pruebas necesitan empezarse lo más pronto posible en el ciclo de vida. También muestra que las pruebas no son sólo una actividad basada en la ejecución. Hay una variedad de actividades que se han de realizar antes del fin de la fase de codificación. Estas actividades deberían ser llevadas a cabo en paralelo con las actividades de desarrollo, y los técnicos de pruebas necesitan trabajar con los desarrolladores y analistas de negocio de tal forma que puedan realizar estas actividades y tareas y producir una serie de entregables de pruebas. Los productos de trabajo generados por los desarrolladores y analistas de negocio durante el desarrollo son las bases de las pruebas en uno o más niveles.

El modelo en V es un modelo que ilustra cómo las actividades de prueba verificación y validación se pueden integrar en cada fase del ciclo de vida. Dentro del modelo en V, las pruebas de validación tienen lugar especialmente durante las etapas tempranas, por ejemplo, revisando los requisitos de usuario y después por ejemplo, durante las pruebas de aceptación de usuario.

El modelo en V es un proceso que representa la secuencia de pasos en el desarrollo del ciclo de vida de un proyecto. Describe las actividades y resultados

que han de ser producidos durante el desarrollo del producto. La parte izquierda de la V representa la descomposición de los requisitos y la creación de las especificaciones del sistema. El lado derecho de la V representa la integración de partes y su verificación. V significa "Validación y Verificación".

El modelo en V define las siguientes etapas o fases de desarrollo:

- a. Definición de especificaciones: Se deben definir y documentar los diferentes requisitos del sistema a desarrollar, identificando los valores numéricos más concretos posibles. Entre ellos debe estar la especificación del nivel de integridad, o SIL, en caso de ser requerido.
- b. Diseño global: También llamado diseño de alto nivel. Su objetivo es obtener un diseño y visión general del sistema.
- c. Diseño en detalle: Consiste en detallar cada bloque de la fase anterior.
- d. Implementación: Es la fase en la que se materializa el diseño en detalle.
- e. Test unitario: En esta fase se verifica cada módulo Hardware y Software de forma unitaria, comprobando su funcionamiento adecuado.
- f. Integración: En esta fase se integran los distintos módulos que forman el sistema. Como en el caso anterior, ha de generarse un documento de pruebas. Por una parte, se debe comprobar en todo el sistema el funcionamiento correcto, y por otra, en caso de tratarse con un sistema tolerante a fallos, debe verificarse que ante la presencia de un fallo persiste el funcionamiento correcto. Se comprueba el cumplimiento de los requisitos establecidos.
- g. Test operacional del sistema: Se realizan las últimas pruebas pero sobre un escenario real, en su ubicación final, anotando una vez más las pruebas realizadas y los resultados obtenidos.

**CAPITULO II**  
**MARCO TEÒRICO**



## **2 Capítulo II - Marco Teórico**

### **2.1 Relevamiento y/o Diagnóstico y sus Conclusiones**

#### **2.1.1 Introducción**

Resulta imposible llegar a procesar una fecha exacta para el nacimiento de la domótica, ya que ésta es resultado de todo un proceso evolutivo que comenzó con las redes de control en edificios que se han ido adaptando cada vez más a las necesidades de las viviendas

Con los años van surgiendo cada vez más necesidades y esto se convierte en una problemática social, antes ésta tecnología considerada un privilegio o algo exclusivo para la gente de clase alta; hoy día se convierte en la necesidad de facilitar o suplir falencias presentadas en el diario vivir, y con el avance de la tecnología a un ritmo tan acelerado, el acceso a esta tecnología se convierte en algo más tangible para la sociedad en general.

Hoy en día, cuando se habla de domótica, se esboza inmediatamente el término control remoto que es muy utilizado para cualquier tipo de proceso, logrando con ello resultados muy satisfactorios en el manejo del dispositivo que se quiere controlar.

Desde el punto de vista de automatización, la domótica es un concepto interdisciplinario que se refiere a la integración de las distintas tecnologías en el hogar mediante el uso simultáneo de las telecomunicaciones, la electrónica, la informática y la electricidad. Estas tecnologías pueden estar integradas gracias a redes interiores y exteriores de comunicación, cableadas o inalámbricas y cuyo control puede ser gestionado desde dentro y fuera de la casa mediante dispositivos móviles.

En el caso de una vivienda inteligente basada en protocolos de comunicación en el ámbito doméstico, sus habitantes pueden controlar desde una computadora o un

celular, elementos como los sistemas de iluminación, la climatización, así como los distintos dispositivos (electrodomésticos) o utilizar internet para vigilar los movimientos de la casa a través de una cámara *web*.

La flexibilidad de este tipo de control permite a las personas un mejor desempeño en sus actividades cotidianas a niveles tanto familiar, como tecnológico, promoviendo con ello el bienestar social y técnico cuando se habla de automatización, área fundamental para mejorar la efectividad de los procesos.

## **2.2 Domótica**

Se llama domótica a los sistemas capaces de automatizar una vivienda o edificación de cualquier tipo, aportando servicios de gestión energética, seguridad, bienestar y comunicación, y que pueden estar integrados por medio de redes interiores y exteriores de comunicación, cableadas o inalámbricas, y cuyo control goza de cierta ubicuidad, desde dentro y fuera del hogar. Se podría definir como la integración de la tecnología en el diseño inteligente de un recinto cerrado.

Las claves para el éxito de un sistema domótico son:

- Cubrir las necesidades (ni quedarse corto, ni pasarse)
- Facilidad de ampliación e incorporación de nuevas funciones.
- Simplicidad de uso.
- Alto grado de estandarización
- Estética de la instalación
- Costo



Figura 2. Casa Domótica

Figura 2 Casa Domótica

## 2.2.1 Ventajas de un Sistema Domótico

Las ventajas en conjunto son claras:

- ✓ Ahorro en la instalación ya que se evita la duplicidad en el dispositivos: Instalando una central domótica no son necesarios otros componentes adicionales como son: central de riego, central de incendios, central de alarma, central de accesos, central de climatización, etc. Todas las demás centrales, no son necesarias ya que la central DOMÓTICA controla, supervisa y gestiona esto y mucho más.
- ✓ Comodidad en la gestión cotidiana de su vivienda: La central domótica se encarga por usted de muchas tareas que deberíamos hacer todos los días por ejemplo: cuando nos vayamos a dormir el sistema conectara el sistema de seguridad, apagará todas las luces que no sean necesarias, regulará la climatización, bajará las persianas, recogerá los toldos.. Cualquier proceso se puede realizar automáticamente.

- ✓ Seguro y de buen funcionamiento. La ausencia de partes mecánicas, hacen que el mantenimiento sea fácil y predictivo. El sistema posee una unidad central que supervisa y gestiona el correcto funcionamiento de todas las partes eléctricas de la instalación, y de todos los elementos asociados a ella.

## **2.2.2 Desventajas de un Sistema Domótico**

Las desventajas son realmente pocas con respecto a las ventajas pero se pueden mencionar las siguientes:

- El costo inicial de instalación es alto.
- Al ser relativamente nueva su aplicación se pueden experimentar fallos en los sistemas, etc.
- Se puede producir el aislamiento del usuario.
- Se puede dar un entorpecimiento del usuario, dependiendo del grado de automatización del sistema

## **2.3 Características Generales de un Sistema Domótico**

Los servicios que ofrece la domótica se pueden agrupar según cinco aspectos principales:

### **2.3.1 Programación y Ahorro Energético**

El ahorro energético no es algo tangible, sino legible con un concepto al que se puede llegar de muchas maneras. En muchos casos no es necesario sustituir los aparatos o sistemas del hogar por otros que consuman menos energía sino una gestión eficiente de los mismos.

- Climatización y calderas: programación y zonificación, pudiéndose utilizar un termostato.
  - \* Se pueden encender o apagar la caldera usando un control de enchufe, mediante telefonía móvil, fija, Wi-Fi o Ethernet.
- Control de toldos y persianas eléctricas, realizando algunas funciones repetitivas automáticamente o bien por el usuario manualmente mediante un mando a distancia:
  - \* Proteger automáticamente el toldo del viento, con un mismo sensor de viento que actúe sobre todos los toldos.
  - \* Protección automática del sol, mediante un mismo sensor de sol que actúe sobre todos los toldos y persianas.
  - \* Con un mando a distancia o control central se puede accionar un producto o agrupación de productos y activar o desactivar el funcionamiento del sensor.
- Gestión eléctrica:
  - \* Racionalización de cargas eléctricas: desconexión de equipos de uso no prioritario en función del consumo eléctrico en un momento dado.
  - \* Contadores electrónicos que informan el consumo electrónico.

### **2.3.2 Confort**

El confort conlleva todas las actuaciones que se puedan llevar a cabo que mejoren la comodidad en una vivienda. Dichas actuaciones pueden ser de carácter tanto pasivo, como activo o mixtas.

- Iluminación:

- \* Apagado general de todas las luces de la vivienda.
  - \* Automatización del apagado/encendido en cada punto de luz.
  - \* Regulación de la iluminación según el nivel de luminosidad ambiente.
- Automatización de todos los distintos sistemas/instalaciones/dotándolos de control eficiente y de fácil manejo.
  - Integración del portero al teléfono, o del video-portero al televisor.
  - Control vía Internet.
  - Gestión Multimedia y del ocio electrónico.
  - Generación de macros y programas de forma sencilla para el usuario y automatización.

### **2.3.3 Seguridad**

Consiste en una red de seguridad encargada de proteger tanto los bienes patrimoniales, como la seguridad personal y la vida.

- Alarmas de intrusión (anti-intrusión): Se utilizan para detectar o prevenir la presencia de personas extrañas en una vivienda o edificio:
  - \* Detección de un posible intruso (Detectores volumétricos o perimetrales).
  - \* Cierre de persianas puntual y seguro.
  - \* Simulación de presencia.
- Detectores y alarmas de detección de incendios (detector de calor, detector de humo), detector de gas (fugas de gas, para cocinas no eléctricas), escapes de agua e inundación, concentración de monóxido de carbono en garajes cuando se usan vehículos de combustión.
- Alerta médica y tele asistencia.
- Acceso a cámaras IP.

### **2.3.4 Comunicaciones**

Son los sistemas o infraestructuras de comunicaciones que posee el hogar.

- Ubicuidad en el control tanto externo como interno, control remoto desde Internet, PC, mandos inalámbricos (p.ej. PDA con Wi-Fi) de aparatos eléctricos.
- Tele asistencia.
- Tele mantenimiento.
- Informes de consumo y costes.
- Transmisión de alarmas.
- Intercomunicaciones.
- Teléfonos y video porteros.

### **2.3.5 Accesibilidad**

Bajo este epígrafe se incluyen las aplicaciones o instalaciones de control remoto del entorno que favorecen la autonomía personal de personas con limitaciones funcionales, o discapacidad.

El concepto diseño para todos es un movimiento que pretende crear la sensibilidad necesaria para que al diseñar un producto o servicio se tengan en cuenta las necesidades de todos los posibles usuarios, incluyendo las personas con diferentes capacidades o discapacidades, es decir, favorecer un diseño accesible para la diversidad humana. La inclusión social y la igualdad son términos o conceptos más generalistas y filosóficos. La domótica aplicada a favorecer la accesibilidad es un reto ético y creativo pero sobre todo es la aplicación de la tecnología en el campo más necesario, para suplir limitaciones funcionales de las personas, incluyendo las personas discapacitadas o mayores. El objetivo no es que las personas con discapacidad puedan acceder a estas tecnologías, porque las tecnologías en si no

son un objetivo, sino un medio. El objetivo de estas tecnologías es favorecer la autonomía personal. Los destinatarios de estas tecnologías son todas las personas, independientemente de su condición de enfermedad, discapacidad o envejecimiento.

Un sistema domótico orientado hacia el uso de personas con discapacidad incluye:

- El registro y control del consumo de servicios en tiempo real: agua, energía eléctrica, gas, aire acondicionado o caldera.
- La vigilancia remota de lugares distantes o inaccesibles para esa persona.
- La transmisión de la información del usuario con sus familiares o cuidadores de forma constante y automatizada.
- La posibilidad de emitir mensajes de emergencia o activar alarmas en caso necesario.
- La programación de ambientes pre-configurados con varios dispositivos enlazados.

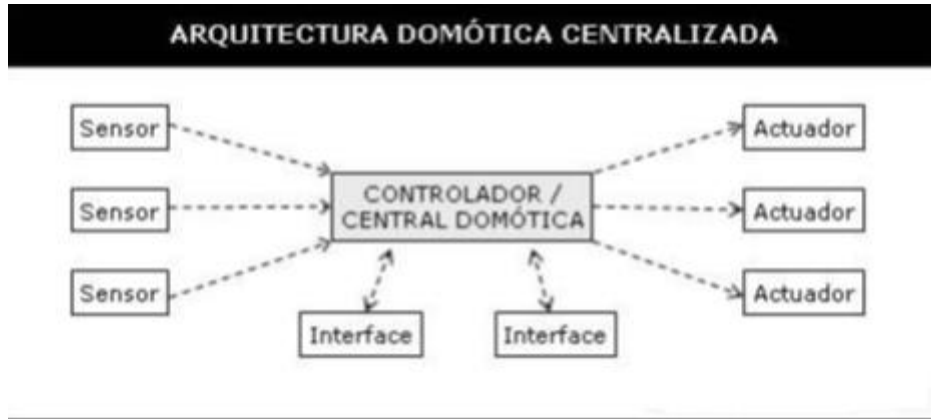
## **2.4 Arquitecturas**

Desde el punto de vista de donde reside la inteligencia del sistema domótico, hay varias arquitecturas diferentes: centralizada, distribuida y mixta. Esta clasificación puede ser considerada tanto desde un punto de vista físico (distribución del cableado o medio físico entre los dispositivos) como lógico (distribución de las comunicaciones que tiene lugar entre dispositivos).

### **2.4.1 Arquitectura Centralizada:**

Es cuando la topología de la red es en estrella, el sistema de control central sería el centro de ésta, de la que están colgando los distintos sensores y actuadores. Esta topología no permite la comunicación directa entre los dispositivos, ya que debe pasar por el sistema de control centralizado



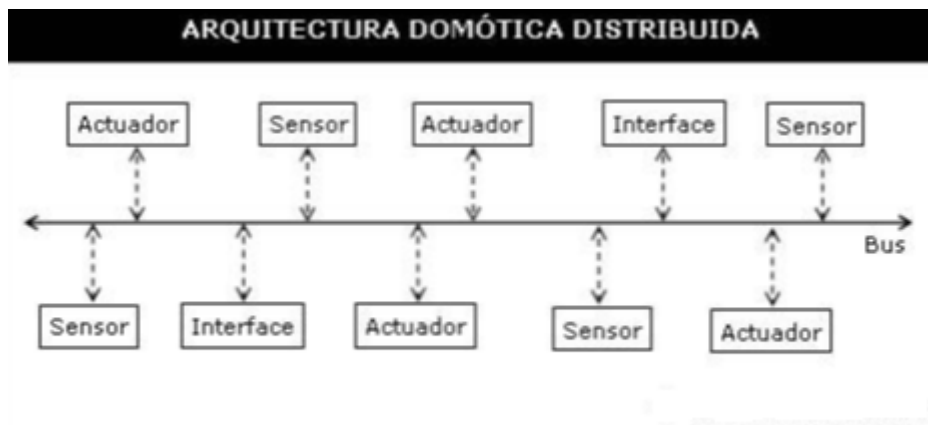


**Figura 3. Arquitectura Domótica Centralizada**

Figura 3 Arquitectura Domótica Centralizada

### 2.4.2 Arquitectura Distribuida:

Para esta arquitectura, el sistema de control se sitúa próximo al elemento a controlar, dando al sistema domótico gran flexibilidad, porque si uno de los dispositivos no puede ser controlado no significa que los demás fallen. Los factores más influyentes para la utilización de este tipo de arquitectura son los medios de transmisión, la velocidad en las comunicaciones, el tipo de protocolo; por lo tanto, estas son algunas características a tener en cuenta si se quiere implementar una arquitectura de esta índole.



**Figura 4. Arquitectura Domótica Distribuida**

Figura 4 Arquitectura Domótica Distribuida

### 2.4.3 Arquitectura Mixta.

En un sistema de domótica de arquitectura híbrida (también denominado arquitectura mixta) se combinan las arquitecturas de los sistemas centralizadas y distribuidas. A la vez que puede disponer de un controlador central o varios controladores descentralizados, los dispositivos de interfaces, sensores y actuadores pueden también ser controladores (como en un sistema “distribuido”) y procesar la información según el programa, la configuración, la información que capta por sí mismo, y tanto actuar como enviarla a otros dispositivos de la red, sin que necesariamente pase por otro controlador.

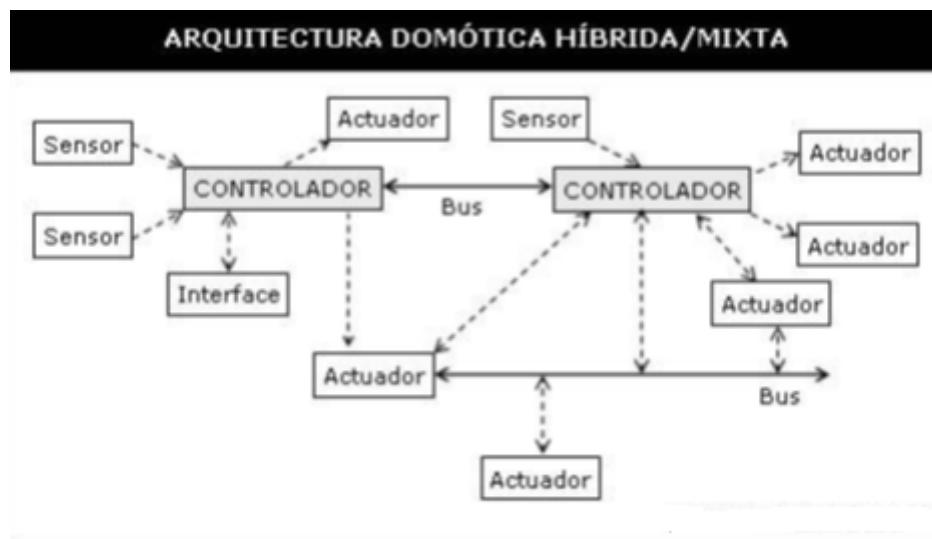


Figura 5. Arquitectura Domótica Híbrida/Mixta

Figura 5 Arquitectura Domótica Híbrida/Mixta

## 2.5 Elementos de una Instalación Domótica

Los elementos que comúnmente nos encontraremos en una instalación domótica y que intervienen en un sistema domótico son:

- Central de gestión
- Nodo
- Microcontrolador

- Medios de interconexión
- Protocolo
- Sensores
- Actuadores
- Interfaz
- Software para el control domótico

Algunos de estos elementos se conectan entre ellos mediante lo que podríamos denominar un bus de datos, que es por donde viaja la información. En la actualidad podemos encontrar sistemas domóticos basados en un bus (uso de cable) por radio frecuencia (sin hilos) o mixtos (cable y radio frecuencia).

Para conseguir un sistema integrado las diferentes partes de este sistema deben comunicarse a través de protocolos.

A continuación se detalla cada uno de los puntos antes mencionados:

### **2.5.1 Central de gestión:**

Se trata del centro de control para gestionar toda la instalación domótica desde un único aparato. No es necesario tener un centro de control, pero nos permitirá la programación de los dispositivos y una gran flexibilidad en el uso de la domótica. Pueden ser aparatos sencillos con unos cuantos botones, hasta pequeñas consolas con pantalla táctil con un menú de opciones para el control de los dispositivos.

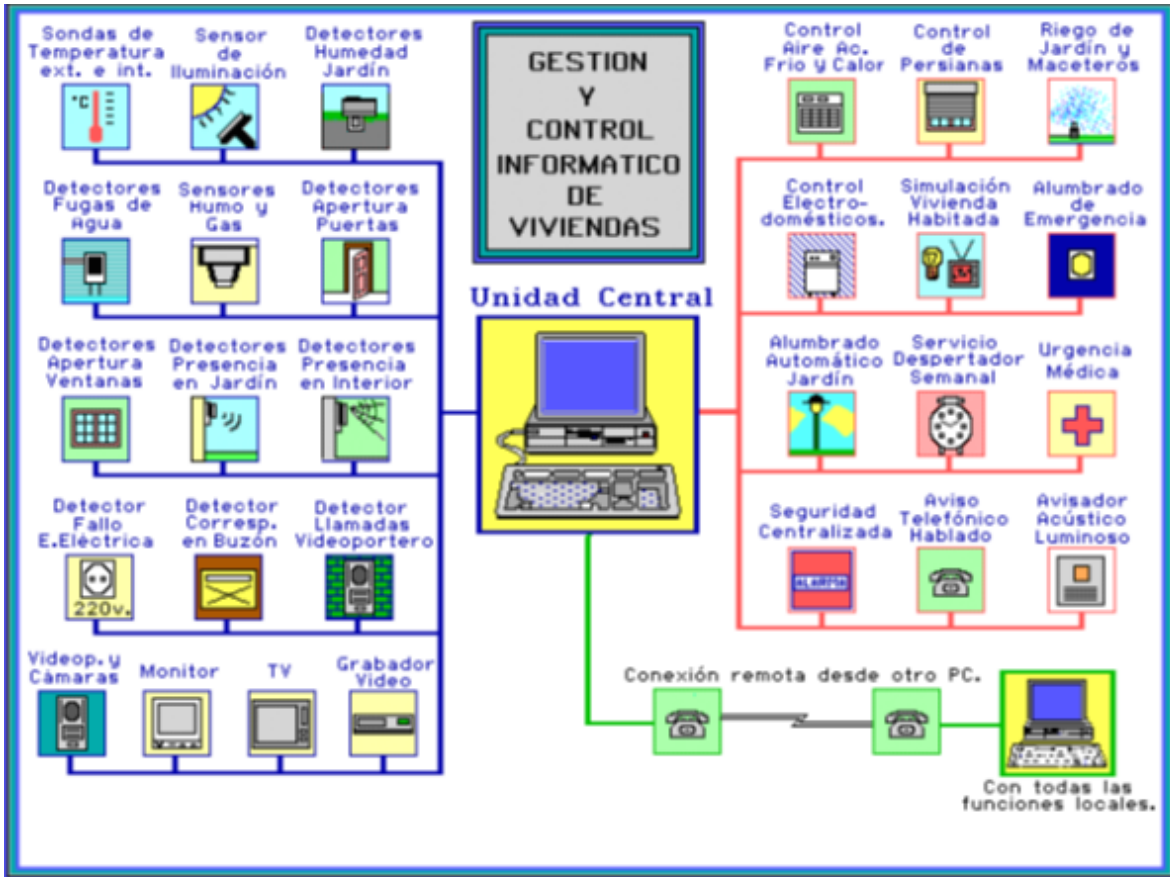


Figura 6. Gestión y Control Informático de Viviendas

Figura 6 Gestión y Control Informático de Viviendas

La central de gestión posee distintos tipos de interfaces:

- Interfaz local: La centralita incorpora una pantalla y un teclado.
- Interfaz de voz: Permite programar o conocer el estado del edificio desde cualquier teléfono. Debe autenticarse con una contraseña.
- Interfaz de mensajes móviles: Utilizan la red GSM. Si se produce una incidencia, envía un SMS.
- Interfaz Web: El sistema dispone de un servidor Web que permite configurar o conocer el estado actual de una forma gráfica.

## **2.5.2 Nodo**

Es el dispositivo que recibe, procesa y envía las señales domóticas procedentes de los sensores hacia los actuadores.

Un sistema domótico puede disponer de uno o más nodos interconectados entre sí, de los cuales tienen sus respectivos sensores y actuadores.

Cuando el nodo es único y todos los sensores y actuadores de la instalación están conectados a él, se dice que es un sistema centralizado.

Sin embargo, cuando existen varios nodos interconectados entre sí a través de un bus de datos común, se dice que es un sistema distribuido.

## **2.5.3 Microcontroladores**

Para el hogar inteligente es necesario contar con una central de gestión que se encargue del control de los distintos dispositivos que se hallen dentro del hogar; esta central es llamada microcontrolador, que es el común denominador en cuanto a mando o accionamiento de los electrodomésticos. También es el encargado de recibir la información y procesarla de acuerdo a la petición que haga el usuario al encender o apagar un dispositivo en la vivienda.

Los microcontroladores son el nuevo orden mundial en cuanto a control. Estos chips reducidos están presentes en muchas campos de interacción humana como el trabajo, la casa, y en general, en muchas faceta de la vida. Se pueden encontrar controlando el funcionamiento de mouse y teclados de los computadores, en los teléfonos, en los hornos microondas y los televisores.

Pero la proliferación está en su apogeo, y en el siglo XXI se observará la conquista masiva de estos diminutos chips, que gobernarán la mayor parte de los aparatos que se fabrican y usan en la humanidad.

En la actualidad es necesario tener pleno control sobre los procesos; para llegar a ello se usan los microcontroladores, que dependiendo de sus características permiten realizar múltiples actividades, por tal motivo es necesario conocer el

manejo de los microcontroladores para el desarrollo y diseño de cualquier dispositivo digital moderno o, para este, caso un dispositivo domótico.

Un microcontrolador es un circuito integrado de alta escala de integración que incorpora la mayor parte de los elementos que configuran un controlador.

Un microcontrolador dispone normalmente de los siguientes componentes:

- ✓ Procesador o UCP (Unidad Central de Proceso).
- ✓ Memoria RAM para contener los datos. Sistemas digitales de mando y control Memoria para el programa tipo ROM/PROM/EPROM.
- ✓ Líneas de E/S para comunicarse con el exterior.
- ✓ Diversos módulos para el control de periféricos (temporizadores, puertas serie y paralelo, CAD: conversores analógico/digital, CDA: conversores Digital/Analógico, etc.).
- ✓ Generador de impulsos de reloj que sincronizan el funcionamiento de todo el sistema.

El microcontrolador es, en definitiva, un circuito integrado que incluye todos los componentes de un computador. Debido a su reducido tamaño es posible montar el controlador en el propio dispositivo al que gobierna.

Características de los microcontroladores:

- Buena información, fácil de conseguir y económica.
- Poseen una elevada velocidad de funcionamiento gracias a su operatividad a alta frecuencia.
- Herramientas de desarrollo fáciles y baratas. Muchas herramientas software se pueden recoger libremente a través de internet desde microchip, Atmel, Motorola, Intel, etc.
- Existe una gran variedad de herramientas hardware que permiten grabar, depurar, borrar y comprobar el comportamiento de los microcontroladores.

## **2.5.3.1 Microcontrolador: Arduino Mega**

### **2.5.3.1.1 Introducción**

Arduino es una plataforma electrónica de hardware libre basada en una placa con un microcontrolador. Con software y hardware flexibles y fáciles de utilizar, Arduino ha sido diseñado para adaptarse a las necesidades de todo tipo de público, desde aficionados, hasta expertos en robótica o equipos electrónicos.

Ante todo y sobre todo es un microcontrolador, es decir un ordenador completo integrado en un chip, con su CPU, memoria de programa, memoria de datos y circuitos para el control de periféricos.

El microcontrolador necesita para su correcto funcionamiento, de algunos circuitos auxiliares y complementos tales como:

- La entrada de alimentación
- El oscilador de trabajo
- Circuito de RESET
- La conexión USB
- Los accesos a las líneas de entrada y salida, etc

También consta de un simple, pero completo, entorno de desarrollo, que nos permite interactuar con la plataforma de manera muy sencilla. Se puede definir por tanto como una sencilla herramienta de contribución a la creación de prototipos, entornos, u objetos interactivos destinados a proyectos multidisciplinarios y multitecnología

La placa Arduino está capacitada para incorporar hardware adicional, contiene una matriz de terminales en la que se puede añadir hardware de acuerdo al requerimiento del prototipo a desarrollar.

### 2.5.3.1.2 Características

El Arduino Mega está basado en el microcontrolador ATmega2560. Tiene 54 pines de entradas/salidas digitales (14 de las cuales pueden ser utilizadas como salidas PWM), 16 entradas análogas, 4 UARTs (puertos serial por hardware), cristal oscilador de 16 Mhz, conexión USB, jack de alimentación, conector ICSP y botón de reset.

Incorpora todo lo necesario para que el microcontrolador trabaje; simplemente conéctalo a tu PC por medio de un cable USB o con una fuente de alimentación externa. El Arduino Mega es compatible con la mayoría de los shields diseñados para Arduino Duemilanove, diecimila o UNO.

La versión de Arduino Mega 2560 adicionalmente a todas las características de su sucesor, utiliza un microcontrolador ATmega8U2 en vez del chip FTDI. Esto permite mayores velocidades de transmisión por su puerto USB y no requiere drivers para Linux o MAC (archivo inf es necesario para Windows) además cuenta con la capacidad de ser reconocido por el PC como un teclado, mouse, joystick, etc.

ELEMENTO	INFORMACIÓN
Microcontrolador	Atmega2560
Voltaje de operación	5 V
Voltaje de entrada (Recomendado)	7 – 12 V
Voltaje de entrada (Límite)	6 – 20 V
Pines para entrada-salida digital.	54 (de los cuales 15 proporcionan salida PWM)



Pines de entrada analógica.	16
Corriente continua por pin IO	20 mA
Corriente continua en el pin 3.3 V	50 mA
Memoria Flash	256 KB, 8 KB utilizado por el gestor de arranque
SRAM	8 KB
EEPROM	4 KB
Velocidad de reloj	16 MHz
LED_BUILTIN	8
Longitud	101.52 mm
Anchura	53,3 mm
Peso	37 g

**Tabla 1 Características Microcontrolador**

Arduino Mega

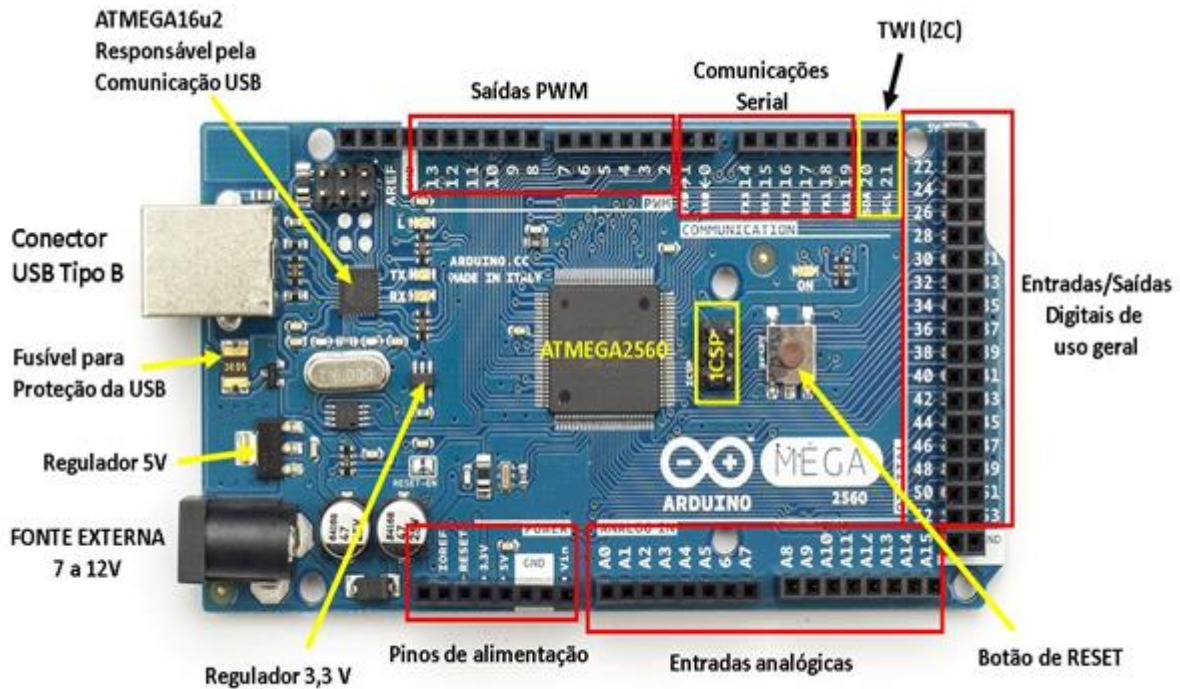


Figura 7. Microcontrolador Arduino Mega

Figura 7 Microcontrolador Arduino Mega

### 2.5.3.1.3 Alimentación de un Arduino

El Arduino Mega puede ser alimentado vía la conexión USB o con una fuente de alimentación externa. El origen de la alimentación se selecciona automáticamente.

Las fuentes de alimentación externas (no USB) pueden ser tanto un transformador o una batería. El transformador se puede conectar usando un conector macho de 2.1mm con centro positivo en el conector hembra de la placa. Los cables de la batería pueden conectarse a los pines Gnd y Vin en los conectores de alimentación (POWER)

La placa puede trabajar con una alimentación externa de entre 6 a 20 voltios. Si el voltaje suministrado es inferior a 7V, el pin de 5V puede proporcionar menos de 5 Voltios y la placa puede volverse inestable; si se usan más de 12V los reguladores de voltaje se pueden sobrecalentar y dañar la placa. El rango recomendado es de 7 a 12 voltios.

Los pines de alimentación son los siguientes:

- VIN. La entrada de voltaje a la placa Arduino cuando se está usando una fuente externa de alimentación (en oposición a los 5 voltios de la conexión USB). Se puede proporcionar voltaje a través de este pin, o, si se está alimentando a través de la conexión de 2.1 mm acceder a ella a través de este pin.
- 5V. La fuente de voltaje estabilizado usado para alimentar el microcontrolador y otros componentes de la placa. Esta puede provenir de VIN a través de un regulador integrado en la placa, o proporcionada directamente por el USB u otra fuente estabilizada de 5V.
- 3V3. Una fuente de voltaje de 3.3 voltios generada por un regulador integrado en la placa. La corriente máxima soportada 50mA.
- GND. Pines de toma de tierra.

#### **2.5.3.1.4 Memoria**

El ATmega2560 tiene 256KB de memoria flash para almacenar código (8KB son usados para el arranque del sistema). El ATmega2560 tiene 8 KB de memoria SRAM y 4KB de EEPROM, a la cual se puede acceder para leer o escribir con la librería EEPROM.

#### **2.5.3.1.5 Entradas y Salidas**

Cada uno de los 54 pines digitales en el Mega pueden utilizarse como entradas o como salidas usando las funciones `pinMode()`, `digitalWrite()`, y `digitalRead()`. Las E/S operan a 5 voltios. Cada pin puede proporcionar o recibir una intensidad máxima de 40mA y tiene una resistencia interna de pull-up (desconectada por defecto) de 20-50kOhms.

Además, algunos pines tienen funciones especializadas:

- Serie 0 (RX) y 1 (TX),
- Serie 1: 19 (RX) y 18 (TX);
- Serie 2: 17 (RX) y 16 (TX);
- Serie 3: 15 (RX) y 14 (TX).

- Usados para recibir (RX) transmitir (TX) datos a través de puerto serie TTL.

Los pines Serie: 0 (RX) y 1 (TX) están conectados a los pines correspondientes del chip FTDI USB-to-TTL.

- PWM: de 0 a 13. Proporciona una salida PWM (Pulse Wave Modulation, modulación de onda por pulsos) de 8 bits de resolución (valores de 0 a 255) a través de la función `analogWrite()`.
- SPI: 50 (SS), 51 (MOSI), 52 (MISO), 53 (SCK). Estos pines proporcionan comunicación SPI, usando la librería SPI.
- LED: 13. Hay un LED integrado en la placa conectado al pin digital 13, cuando este pin tiene un valor HIGH(5V) el LED se enciende y cuando este tiene un valor LOW(0V) este se apaga.

El Mega tiene 16 entradas analógicas, y cada una de ellas proporciona una resolución de 10bits (1024 valores). Por defecto se mide desde 0V a 5V, aunque es posible cambiar la cota superior de este rango usando el pin AREF y la función `analogReference()`.

- I2C: 20 (SDA) y 21 (SCL). Soporte para el protocolo de comunicaciones I2C (TWI) usando la librería Wire.
- AREF. Voltaje de referencia para la entradas analógicas. Usado por `analogReference()`.
- Reset. Suministrar un valor LOW (0V) para reiniciar el microcontrolador. Típicamente usado para añadir un botón de reset a los shields que no dejan acceso a este botón en la placa.

## **2.5.4 Medios de interconexión**

Los medios de interconexión entre la central de gestión, sensores y actuadores se clasifican en dos tipos alámbricos e inalámbricos, a continuación se detalla cada uno de ellos:

### ***2.5.4.1 Alámbrico:***

Son aquellos que utilizan componentes físicos y sólidos para la transmisión de datos. Estos se pueden transmitir a través de:

- Línea de distribución de energía eléctrica: Si bien no es el medio más adecuado para la transmisión de datos, es una alternativa a tener en cuenta para las comunicaciones domésticas, dado el bajo coste que implica su uso al tratarse de una instalación existente. En aquellos casos en los que las necesidades de los sistemas no impongan requerimientos muy exigentes en cuanto a la velocidad de transmisión de datos, la línea de distribución de energía eléctrica puede ser suficiente como soporte de dicha transmisión.
- Cable coaxial: El cable coaxial tiene gran utilidad por su propiedad idónea de transmisión de voz, audio y video. Los factores a tener en cuenta a la hora de elegir un cable coaxial son su ancho de banda, su resistencia o impedancia característica, su capacidad y su velocidad de propagación. La velocidad de transmisión del cable coaxial puede llegar a 10 Mbps.
- Cable de par trenzado: El cable consiste en dos alambres de cobre aislados. Los alambres se trenzan con el propósito de reducir la interferencia eléctrica de pares similares cercanos. Los pares trenzados se agrupan bajo una cubierta común de PVC. A pesar que las propiedades de transmisión de cables de par trenzado son inferiores, y en especial la sensibilidad ante perturbaciones extremas, a las del cable coaxial, su gran adopción se debe al costo, su flexibilidad y facilidad de instalación, así como las mejoras tecnológicas constantes introducidas en enlaces de mayor velocidad,

longitud, etc. La velocidad de transmisión de este cable puede llegar a 100 Mbps.

- **Cable de fibra óptica:** Es una de las tecnologías más avanzadas que se utilizan como medio de interconexión. Se puede lograr una mayor velocidad y disminuir casi en su totalidad ruidos e interferencias, hasta multiplicar las formas de envío en comunicaciones y recepción por vía telefónica. La fibra óptica está compuesta por filamentos de vidrio de alta pureza muy compactos. El grosor de una fibra es como la de un cabello humano aproximadamente. Fabricadas a alta temperatura con base en silicio, su proceso de elaboración es controlado por medio de computadoras, para permitir que el índice de refracción de su núcleo, que es la guía de la onda luminosa, sea uniforme y evite las desviaciones.

#### ***2.5.4.2 Inalámbrico.***

Los medios inalámbricos son especiales para cubrir grandes distancias y hacia cualquier dirección. De manera general podemos definir las siguientes características de este tipo de medio. La transmisión y recepción se realiza por medio de antenas, las cuales deben estar alineadas cuando la transmisión es direccional, o si es omnidireccional la señal se propaga en todas las direcciones. Entre los medios inalámbricos existentes en la actualidad se puede definir algunas de ellas como son:

- **Radiofrecuencia:** Esta tecnología puede parecer en principio, idóneo para el control a distancia de los sistemas domóticos, dada la gran flexibilidad que supone su uso. Sin embargo la este medio resulta particularmente sensible a las perturbaciones electromagnéticas producidas, tanto por los medios de transmisión, como por los equipos ya sea domésticos o industriales.
- **Bluetooth:** Es un enlace radio de corto alcance que aparece asociado a las redes de área personal inalámbricas, o sus siglas en inglés WPAN (wireless

personal area network). Este concepto hace referencia a una red sin cables que se extiende a un espacio de funcionamiento personal con un radio de hasta 10 metros. Las WPAN constituyen un esquema de red de bajo coste que permite conectar entre sí equipos informáticos y de comunicación portátil y móvil, como ordenadores, PDA, impresoras, ratones, micrófonos, auriculares, cuyo objetivo principal es que todos los equipos puedan interconectarse entre sí, sin interferencias.

- Infrarrojos: La comunicación se realiza entre un diodo que emite una luz en la banda de IR, sobre la que se superpone una señal modulada con la información de control, y un fotodiodo receptor cuya misión consiste en extraer de la señal recibida la información de control. Al tratarse de un medio de transmisión óptico, son inmunes a las radiaciones electromagnéticas producidas por los equipos domésticos o por los demás medios de transmisión.

### **2.5.5 Protocolos de comunicación**

Existen varios tipos de sistemas de comunicación empleados en la domótica, los cuales se clasifican dependiendo del canal que utilizan para la transmisión de datos, se detallan a continuación los principales.

- ✓ X-10
- ✓ EIB
- ✓ LONWORKS
- ✓ JINI
- ✓ UNPN
- ✓ TCP/IP

Ahora continuación se dará una descripción sobre cada uno de los protocolos que son usados en la implementación de los sistemas domóticos y con ello se observa cómo es la interacción de los diversos dispositivos dentro de la vivienda inteligente.

### ***2.5.5.1 Protocolo X-10:***

El protocolo X-10 es aquel orientado hacia la utilización de la red eléctrica de las viviendas. Allí se utilizan corrientes portadoras para controlar cualquier dispositivo a través de la línea de corriente doméstica (120 ó 220 v. y 50 ó 60 hz), y se hace modulando impulsos de 120 khz,

Este protocolo posee tramas de datos que son ceros y unos agrupados formando comandos; con esta agrupación se pueden formar hasta seis acciones para el dispositivo que son: encendido, apagado, reducir, aumentar, todo encendido, todo apagado.

Las señales que se envían son recibidas por todos los módulos, pero de acuerdo a la dirección sólo actúan sobre el dispositivo al que va dirigida la acción (los primeros bits de la señal son el identificador del módulo, de este modo se sabe dónde está el dispositivo a controlar).

El protocolo X-10 no es propietario, es decir, cualquier fabricante puede producir dispositivos X-10 y venderlos, pero está obligado a usar los circuitos de control que han sido diseñados por el fabricante de esta tecnología.

### ***2.5.5.2 EIB (European Installation Bus)***

European Installation Bus o EIB, es un sistema domótico desarrollado bajo el aval y supervisión de la Unión Europea con el objetivo de evitar las importaciones de productos similares.

Es un estándar europeo que define una relación extremo a extremo entre los dispositivos, permitiendo distribuir la inteligencia entre los sensores y los actuadores instalados en la vivienda.

El sistema EIB permite, mediante la instalación de algunos amplificadores o repetidores, optimizar su sistema hasta 256 dispositivos por línea, pero para evitar las colisiones entre telegramas enviados se utiliza la técnica CSMA/CA (Carrier



Sense Multiple Access), garantizando el acceso libre de colisiones fortuitas en la línea Bus y sin ninguna pérdida de datos.

### **2.5.5.3 Konnex**

El modelo Konnex es el paso evolutivo lógico que trata de concentrar toda la experiencia y conocimientos de los principales estándares europeos EIB, EHS, BatiBus.

En sí el modelo Konnex se basa en el modelo EIB y expande su funcionalidad añadiendo nuevos medios físicos y modos de configuración BatiBus y EHS.

Contempla tres modos de trabajo que pueden seleccionarse dependiendo del nivel de competencia de cada instalador:

- Modo S (Modo sistema): este modelo sigue la misma configuración que el EIB actual, esto es, los diversos dispositivos o nidos de la nueva instalación son instalados por profesionales con ayuda de software.
- Modo E (Modo fácil Easy): aquí los dispositivos son programados en fábrica para realizar una función concreta. También requieren alguna configuración por parte de un controlador central.
- Modo A (Modo automático): se sigue una filosofía plug and play, es decir, que no se tiene que configurar el dispositivo.

Por otro lado, Konnex puede funcionar por cuatro medios físicos distintos:

- Par trenzado (TP1 y TP0)
- Corrientes portadoras (PL100 y PL132)
- Ethernet (IP)
- Radiofrecuencia (RF)

La posibilidad de utilizar medios físicos distintos permite a los instaladores adaptar la red a condiciones más favorables para el usuario final. Konnex está basado en el modelo de EIB, siendo compatible con los distintos productos elaborados por varias de las firmas productoras de componentes EIB.

#### ***2.5.5.4 Lonworks***

Es un sistema de control domótico propietario presentado por la firma Echelon en 1992. Debido a su costo los dispositivos Lonworks no han tenido una implantación masiva en los hogares.

Los fabricantes de dispositivos Lonworks deben emplear en cada uno de sellos un microcontrolador especial que le denominan neuron chip. Este circuito integrado, al igual que el firmware que implementa el protocolo Lontalk utilizado en los nodos Lonworks para comunicarse, fueron desarrollados por Echelon en 1990. Una ventaja importante de Lonworks respecto a otras tecnologías es que implementa el modelo de referencia OSI. De este modo, servicios tales como el reenvío automático tras una pérdida de trama o la autenticación del emisor de la trama, están completamente implementados en la solución Lonworks.

#### ***2.5.5.5 JINI***

Esta tecnología de Sun Microsystems permite descubrir nuevos dispositivos que se van incorporando a la red del hogar mediante cualquier medio.

JINI permite utilizar servicios y dispositivos de red de manera tan fácil como utilizar una conexión telefónica, esto es, permite conectarse y participar por medio de un tono de marcado en red. La meta de JINI es simplificar interacción en la red.

JINI aprovecha la tecnología JAVA y consiste en una pequeña cantidad de este código en forma de librerías de clases y algunas convenciones para crear una «federación» de máquinas JAVA virtuales en la red, similar a la creación de una comunidad.

Los navegantes de la red, tales como usuarios, dispositivos, datos y aplicaciones, se conectan dinámicamente para compartir información y realizar tareas, convirtiéndose en una unidad lógica de información de red.

Cada aparato de esta red tiene en principio dos conexiones: el enchufe a la red eléctrica, y una ficha de conexión del tipo RJ-45. Está pensado para ejecutarse en

periféricos, pero puede hacerse en potentes ordenadores, cámaras, teléfonos, o cualquier dispositivo electrónico que se nos ocurra.

### **2.5.5.6 *Tcp/ip***

La mayoría de los protocolos domóticos han sido especialmente creados para implementar redes de control distribuidas (Lonworks, EIB, X-10), las tramas fueron diseñadas de forma que el espacio útil para datos de las aplicaciones fuera el máximo. Por ejemplo, para encender y apagar una luz basta con una orden codificada en un par de octetos. Por lo tanto, se trata de minimizar los campos de control que el protocolo necesita para transferir estos dos octetos al dispositivo destino.

TCP/IP está siendo usado en infinidad de computadoras y aplicaciones, de forma que ha conseguido un volumen de negocio tal que ha hecho de este protocolo la herramienta ideal para asegurar la interconectividad total entre máquinas en cualquier parte del mundo, y con esto se puede hablar cada vez más de viviendas inteligentes.

Luego de haber definido algunos de los protocolos que forman parte de la domótica, se entrará en los microprocesadores de control que son usados para todos los dispositivos que hay en la vivienda inteligente.

### **2.5.6 Sensores**

Los sensores son los elementos que utiliza el sistema para conocer el estado de ciertos parámetros (la temperatura ambiente, la existencia de un escape de agua, etc.) y enviarla al sistema de control centralizado para que actúe en consecuencia.

Entre los Sensores o detectores más comunes tenemos:

- Termostato ambiente.
- Detector de gas.
- Detector de incendios.

- Sonda de humedad.
- Sensores de presencia (volumétricos y perimetrales)
- Sensor de iluminación ambiente.

Ejemplo:

Sensor de corriente registra en el sistema las variaciones en el consumo de los distintos equipos eléctricos en el hogar.



**Figura 8. Sensores**

Figura 8 Sensores

## 2.5.7 Actuadores

Es el dispositivo encargado de ejecutar una acción. Los sensores que son capaces de ejecutar una acción, en realidad son también actuadores, son elementos que utiliza el sistema para modificar el estado de ciertos equipos e instalaciones.

Entre los actuadores más comunes encontramos:

- Contactores (relés de actuación) de carril DIN,
- Electroválvulas de corte de suministro (gas y agua),
- Válvulas para zonificar la calefacción por agua caliente,
- Sirenas o elementos zumbadores para el aviso de alarmas.

Ejemplo:

Si un sensor de humo detecta un incendio, avisará al sistema de control. Este hará las llamadas telefónicas programadas y actuará sobre la válvula de corte de gas a la vivienda, y sobre la sirena.



**Figura 9. Actuadores**

Figura 9 Actuadores

### 2.5.8 Software para control domótico

El software es el encargado de la programación, puesta en marcha, monitorización y mantenimiento de un sistema domótico. Se comunica directamente con el hardware (sensor y actuador) para la realización de la tarea de control (iluminación, flujo de agua, alarmas etc.) y puede estar basado en los diversos sistemas operativos existentes en el mercado.

Normalmente se ha utilizado el lenguaje de programación C para desarrollar las distintas aplicaciones que controlan todo el sistema, pero con la llegada de la programación orientada a objetos y del avance en las distintas áreas de la informática se hace más adaptable a las necesidades actuales en cuanto a facilidad de programación y operación.

Cada sistema domótico ha desarrollado diversos tipos de software compatibles con sus productos, encontrándose versiones para los diferentes sistemas operativos (Windows, Linux, Android) que se los puede manipular desde un computador hasta un teléfono inteligente.

El software que vaya a ser utilizado debería tener como mínimo las siguientes recomendaciones:

- ✓ Comunicación con el circuito de acceso a la red domótica, vía USB.
- ✓ Visualización de los módems conectados a la red y las aplicaciones de cada uno.
- ✓ Sub-aplicaciones de monitorización y control de los sensores y actuadores de las aplicaciones.
- ✓ Interfaz visual y amigable.
- ✓ Sección que ofrezca conectividad con el exterior.

## 2.5.9 Interfaz de usuario

El sistema de control centralizado puede ofrecer varias interfaces para que los usuarios puedan conectarse a ella, controlar y programar todos los dispositivos de la vivienda. Las interfaces de los sistemas de control centralizado es que suele constar de un teclado y pantalla para su programación local y requerirá el uso de otros aparatos PC o dispositivo móvil.



**Figura 10. Interfaz para el Control Domotico**

**Figura 10 Interfaz para el Control Domotico**

### Interfaz local

La central consta de una pantalla y un teclado que permiten su instalación y configuración local, por lo general, la interfaz que presentan es muy básica, es decir, no es una interfaz demasiado amigable para realizar operaciones complejas. Por ello, también suele ser posible su programación de una forma más sencilla mediante un software gráfico específico en un pc, con un sistema operativo

Windows. El PC se conectará a la central a través de un puerto tipo USB o Ethernet, o bien a través de tecnologías inalámbricas como Bluetooth o Wlan.

Para realizar operaciones sencillas, como por ejemplo activar o desactivar el sistema de alarma cuando se sale o se entra de la vivienda, también se utilizan pequeños pulsadores o mandos a distancia con infrarrojos.

### Interfaz de Voz

La forma de control remoto tradicionalmente más utilizada era la telefónica. Para ello, el sistema de control centralizado debe estar conectado a una toma telefónica fija (típicamente, RJ-11 en la RTC y RJ-45 en la RDSI) o disponer de una tarjeta-modulo-GSM en su interior.

Los sistemas de control centralizado suelen integrar una interfaz de voz que permite al usuario conocer o programar el estado de la vivienda en cualquier momento y desde cualquier teléfono fijo o móvil, con tan sólo marcar un número de teléfono. Cuando sea el usuario el que llame al sistema, deberá autenticarse mediante la inserción de una contraseña numérica mediante el teclado del teléfono.

Cuando se produzca una alarma o incidencia (como un escape de gas o de agua) es el propio sistema de control el que llama automáticamente al usuario informándole del problema.

### Interfaz de Mensajes Móviles.

Puesto que la red telefónica fija puede, en muchas ocasiones, ser fácilmente sabotada, los sistemas centralizados de seguridad incorporan cada vez más conexión con la red celular GSM. Para ello es necesario introducir una tarjeta SIM (subscriberidentity module) GSM dentro de la central.

En caso de detectarse una anomalía en la vivienda, la central puede enviar un SMS al usuario, con el fin de que sea informado de la incidencia y pueda tomar las acciones pertinentes. Las centrales capaces de comunicarse con video cámaras, podrían enviar incluso mensajes con imágenes o sonido al usuario.

Cuando el usuario quiera acceder remotamente al sistema, podrá llamar al número que tiene la tarjeta GSM utilizada por la central. También podría enviar un SMS en el que estuviese un determinado comando a ejecutar sobre la central.

### Interfaz web.

Ofrece una mayor facilidad de uso para el usuario. El sistema de control centralizado dispone de un servidor web, el cual permite acceder a través de intuitivos menús gráficos a toda configuración y estado actual.

Es, además, una interfaz muy potente, pues permite realizar más operaciones a mayor velocidad, esta interfaz se utiliza únicamente cuando el usuario establece las comunicaciones, como en el caso de una alarma importante; el usuario sería avisado por el sistema de control centralizado mediante una sirena, una llamada a su móvil o mensaje SMS a su móvil.

Esta conexión puede realizarse en el propio hogar desde un PC, Pad o móvil. No obstante, el verdadero potencial está en poder ser utilizada por un PC remoto a través de internet, para lo cual es necesario conectar la central a internet mediante alguna tecnología de banda ancha con conexión permanente y de la velocidad (como ADSL, cable o LMDS). El propio usuario podría utilizar cualquier tipo de conexión (RTC, RDSI, ADSL, cable, LMDS, etc.) para conectarse a la interfaz web de la pasarela ubicada en su vivienda.

## **2.6 Elección Del Sistema Domótico**

La elección del sistema domótico representa en esencia uno de los aspectos más importantes para administrar el la vivienda inteligente, ya que el sistema elegido formará parte integral de la construcción y por lo tanto deberá tener una duración de muchos años y un grado de satisfacción adecuada al usuario, para ello se debe hacer una evaluación muy minuciosa del sistema.

Para su elección se debe tomar en consideración varios aspectos como:



- \* Tipo y tamaño de la edificación a ser intervenida: Se refiere al servicio que brinda la edificación como por ejemplo, una vivienda, un edificio, oficinas, centros de recreación, centros educativos, etc.
- \* Nueva o construida: Si la vivienda aún no se ha construido existe la libertad para la instalación de cualquier sistema, pero si ya está construida buscar un sistema que se adapte mejor a la vivienda, pues interviene posibles modificaciones en la estructura del inmueble.
- \* Las funcionalidades: Las funcionalidades toman en cuenta los diferentes servicios que necesitarán los usuarios, como pueden ser en ahorro energético (consumo energético), seguridad tanto a las personas como los bienes y confort para un ambiente adecuado de trabajo, o cualquier actividad que se requiera realizar.
- \* La integración: Se basa en el análisis de los diferentes dispositivos para interactuar, además de aparatos y sistemas que se controla directamente con el sistema de domótica, con que otro sistema se quiere interactuar.
- \* Los interfaces: Se los realiza mediante controles remotos, pantallas táctiles, computadores, Smartphone, vía internet, etc. Cada sistema domótico maneja sus propios interfaces en mayor o menor grado de complejidad.
- \* Sencillez: Tanto de preinstalación, instalación como programación. El sistema tiene que ser sencillo e intuitivo tanto para el instalador como para el usuario.
- \* Costos del sistema: El costo varía entre los diferentes tipos de sistemas domóticos y tiene que ver mucho con el presupuesto del usuario.
- \* Reconfiguración y mantenimiento: Hay que tener en cuenta la facilidad con la que se puede reconfigurar el sistema por parte del usuario y los servicios de mantenimiento.

## **2.7 Conclusiones**

La domótica es una de las formas como la automatización busca mejorar el estado actual de la sociedad, aumentando la calidad de vida de la humanidad.

El acceso a internet de banda ancha aporta la conexión permanente de la vivienda inteligente a las redes públicas de datos. Con este acceso, los propietarios podrán controlar las viviendas en todo momento, podrán recibir correos electrónicos o mensajes en los móviles cuando sucedan eventos o alarmas.

Gracias a la domótica se integran las diversas áreas, como lo son las telecomunicaciones, la electrónica, la electricidad y la informática, para lograr la gestión del hogar inteligente.

En los sistemas domóticos se tiene un sin fin de posibilidades, haciendo de la cotidianidad algo menos monótono, permitiendo la interacción remota con el hogar para optimizar la variable tiempo según la necesidad presentada en él, reduciendo así retardos en el cumplimiento de las labores familiares.

## **CAPITULO III**

### **PROPUESTA DE SISTEMA**

## **3 Capítulo III - Propuesta de Sistema**

### **3.1 Resolución de Etapas según Metodología Adoptada: Etapa de Relevamiento**

#### **3.1.1 Análisis de los objetivos del sistema y requisitos previos.**

##### ***3.1.1.1 Objetivo General***

El objetivo general del proyecto es realizar el estudio, diseño y aplicación de un sistema domótico a escala que permita en una vivienda estándar reducir el consumo de energía, aumentar el confort, accesibilidad, comunicación y seguridad de la vivienda.

##### ***3.1.1.2 Objetivos Específicos***

- ✓ Identificar y desarrollar los servicios que el sistema controlará.
- ✓ Diseñar e implementar la interface electrónica que permita la recolección de datos y accionar de dispositivos electrónicos.
- ✓ Diseñar e implementar la interface visual mediante software con indicadores gráficos y sonoros que permitan un inmediato conocimiento de las condiciones reales del sistema en general.
- ✓ Diseñar e implementar las pantallas, formularios y reportes requeridos para el exacto control y supervisión del sistema.
- ✓ Implantar el sistema en una maqueta diseñada para tal fin donde las señales de salida sean visualizadas en tiempo real y las señales de entrada sean un fiel patrón de las originadas por sensores reales.

##### ***3.1.1.3 Límites y Alcances Del Sistema***

El sistema domótico llevara el control de funciones en la casa, este control se podrá realizar a través aplicación web, donde el usuario podrá ingresar con un nombre y contraseña, con el cual administrara las acciones de su casa a por medio de una red interna o externa.

El sistema domótico puede monitorear y controlar específicamente los elementos que a continuación se detallan para cada servicio básico dado:

- Apertura/Cierre de puertas y portones de ingreso.
- Programar Encendido/Apagado de equipos eléctricos.
- Programar la iluminación de los ambientes.
- Climatización de ambientes según temperatura.
- Monitorear de consumo de equipos.
- Automatización del sistema de iluminación, tomando en consideración la presencia de personas.
- Programar un horario para que se realicen ciertas tareas específicas
- Gestionar mediante software el control remoto y automático de todo el sistema domótico.
- Ingreso al hogar mediante tarjeta electrónica
- Activar/Desactivar Alarma
- Detección de fugas de gas/humo/llamas y cierre de electroválvulas
- Consulta de actividades realizadas en el hogar según usuario, fecha o actividad.
- Registro de actividades a realizar en el hogar.
- Monitoreo de la vivienda, interior y exteriormente por medio de un circuito cerrado de cámaras de video instalado en lugares estratégicos

De igual manera, los programas elaborados tanto para la tarjeta de interfaz (microcontrolador) como para la computadora, (central de gestión), se diseñan en función de los elementos utilizados. Cualquier cambio de los programas, obligará a la revisión y adaptación de los sensores y actuadores. Pudiendo inclusive, si el caso lo exige, al diseño y elaboración de nuevos elementos.

Cualquier otro elemento distinto a los mencionados no puede ser controlado ni monitoreado por el sistema domótico. Esto debido a que cada actuador y sensor, se diseñan exclusivamente para los elementos señalados.

El sistema de control domótico puede ser adaptado en otros espacios, distintos a la vivienda, donde los requerimientos de ahorro energético, confort, seguridad y accesibilidad sean un requisito a cumplir.

### ***3.1.1.4 Factibilidades***

A pesar de que una implementación en escala real resulta bastante costosa, después de mucho investigar con diferentes tipos de sistemas de domótica, se llegó a la conclusión de que un modelo a escala es más fiable para ejecutar pruebas y tomar resultados para posteriormente llevar a cabo este proyecto a escala real.

La factibilidad de la solución está basada en lo siguiente:

#### **3.1.1.4.1 Factibilidad Teórica**

Aquí la Domótica nos proporciona todas las herramientas metodológicas y experiencias relacionadas con la aplicación de la tecnología en el hogar. Todo el marco teórico de la Domótica, aún no formalizado ni estandarizado, señala un camino a seguir para la correcta implantación mediante pasos y estrategias de un hogar domótico basado en las pautas de diseño que lo identifican como tal.

El segmento teórico de la Domótica actualmente ha sido construido por innumerables empresas que en el ámbito internacional hacen de la Domótica su principal oferta como alternativa de automatización y confort aplicado en el hogar.

La Domótica abarca muchos aspectos de automatización del hogar; seguridad, gestión energética, automatización de las tareas domésticas, formación, cultura, entretenimiento, monitorización de salud, comunicación con servidores externos, operación y mantenimiento de instalaciones.

#### **3.1.1.4.2 Factibilidad Técnica**

El detalle técnico involucra todos los elementos de hardware y software que se necesitan para controlar, ajustar o modificar las funciones del sistema domótico.

A continuación se describen los elementos de hardware y software disponibles y que son necesarios para la solución considerada:

#### **3.1.1.5 Hardware**

Dentro de esta categoría se tiene en orden de importancia:

- Computadora: La cual actuará como centro de control de la información.
- Sensores: Los cuales permiten detectar los cambios en las variables controladas.
- Actuadores: Son elementos encargados de actuar sobre las variables controladas.
- Interfaz Electrónica: Permite la comunicación con el ambiente a controlar.
- Medio físico de comunicación: Es la vía que utiliza el sistema domótico para monitorear y actuar sobre las variables a controlar.

#### **3.1.1.6 Software**

El software aplicado al sistema domótico está presente en dos formas:

- El Sistema encargado de monitorear y actuar sobre las variables a controlar por el sistema domótico.
- El programa en que gestiona las entradas y salidas de los datos en la interfaz electrónica.

Los lenguajes de programación para el software de gestión y el programa del microcontrolador están son de licencia pública (GNU) como así las herramientas auxiliares para el desarrollo del proyecto domótico.

### ***3.1.1.7 Servicios en la Vivienda***

#### **3.1.1.7.1 Confort.**

Esta aplicación busca específicamente el bienestar y la comodidad de las personas que utilicen las instalaciones. Para cumplir con este objetivo de aplicación se ha considerado la automatización de las funciones como:

- Control de puerta y portones apertura/cierre
- Control de cortinas apertura/cierre
- Control de iluminación encendido, apagado y regulación (en algunos ambientes)
- Control de equipos eléctricos encendido/apagado
- Climatización de los ambientes

#### **3.1.1.7.2 Seguridad**

Dentro de los sistemas de aplicación en la domótica uno de las aplicaciones más importantes está el sistema de seguridad en la cual se ha considerado para nuestro criterio emplear las siguientes funciones:

- Control de acceso inteligente monitoreo y ejecución
- Alarmas anti intrusión monitoreo y ejecución
- Sistema de vigilancia (circuito cerrado de cámara de video) monitoreo y ejecución
- Alarmas para detección de fugas de gas monitoreo y ejecución



### **3.1.1.7.3 Gestión de energía eléctrica**

La gestión de energía nos permite manejar eficazmente y de manera óptima el consumo energético de las instalaciones eléctricas, lo que contribuye a la protección del medio ambiente y permite reducir significativamente el consumo energético.

Dentro de las funciones de esta aplicación consideramos las siguientes funciones:

- Automatización de sistema de iluminación ejecución y regulación
- Control de consumo de energía eléctrica Ejecución
- Control de aparatos eléctricos encendido/apagado.

### **3.1.1.7.4 Comunicaciones**

Los elementos de campo (detectores, transmisores y sensores), transmitirán las señales a una unidad central inteligente que tratará y elaborará la información recibida. En función de dicha información y de una determinada programación, la unidad central actuará sobre determinados dispositivos relacionados con las señales recogidas por los elementos sensores de campo correspondientes.

Las funciones de esta aplicación está dada por:

- Emisión de señal (actuadores)
- Procesamiento (placa microcontrolador)
- Recepción de señal (sensores)
- Ejecución
- Transmisión – Recepción de datos

## **3.1.2 Componentes para el Armado del Sistema Domótico**

### ***3.1.2.1 Sistema a utilizar***

El sistema a utilizar se va a encargar del control y regulación de los dispositivos de la vivienda en las diferentes aplicaciones como: confort, seguridad, accesibilidad, gestión de energía eléctrica y comunicación.

En base al relevamiento realizado en el punto anterior, proponemos el desarrollo de un sistema domótico, para ser presentado en una maqueta a escala, en base a:

1. Arquitectura: Centralizada
2. Centro de control: PC
3. Microcontrolador: Arduino Mega
4. Sensores: Gas- Humo– Movimiento – Sonido –Temperatura – Humedad, etc.
5. Actuadores: Pulsadores – Motor paso a paso – Motor DC –Parlante, etc.
6. Protocolo: TPC/IP para la comunicación entre Web y Microcontrolador
7. Interfaz: Interfaz Web accesible desde PC o móvil
8. Software para control domótico: Eclipse para el desarrollo de la interfaz – Arduino IDE para programar el microcontrolador.

\* Como se nombró en otros puntos anteriores el sistema domótico, con todas las funcionalidades plantadas, serán implementadas en una maqueta a escala esto es para la defensa del proyecto de grado.

### ***3.1.2.2 Arquitectura: Centralizada***

Es cuando la topología de la red es en estrella, el sistema de control central sería el centro de ésta, de la que están colgando los distintos sensores y actuadores. Esta topología no permite la comunicación directa entre los dispositivos, ya que debe pasar por el sistema de control centralizado

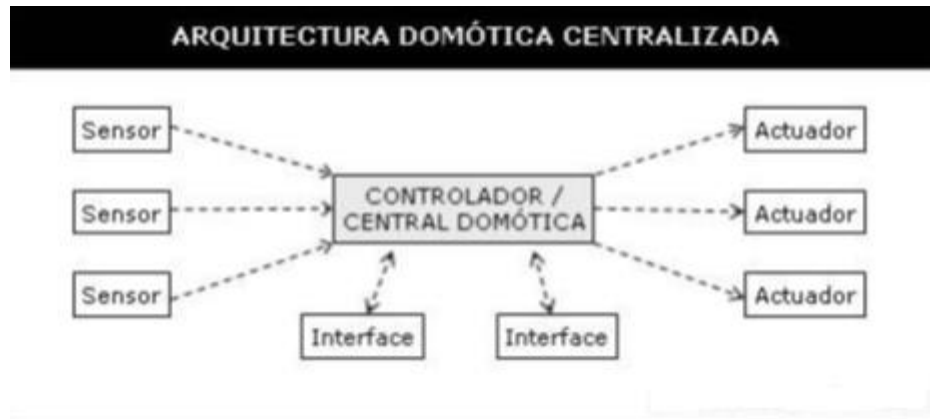


Figura 11. Arquitectura Domótica Centralizada

Figura 11 Arquitectura Domótica Centralizada

### 3.1.2.3 Centro de control: PC

#### Requerimientos mínimos de PC:

- Sistemas Operativos:  
Windows Vista, Windows 7, Windows 8, Windows 10  
Mac OS X 10.6 y versiones posteriores  
Ubuntu 12.04 y versiones posteriores, Debian 7 y versiones posteriores, OpenSuSE 12.2 y versiones posteriores, Fedora Linux 17
- Procesador:  
Intel Pentium 4 o superior.
- Espacio en disco:  
350 MB o superior.
- Memoria RAM:  
512 MB o superior.
- Navegadores Web:  
Chrome – Internet Explorer – Opera – Safari
- Conexión a Internet  
1MB o superior

### **3.1.2.4 Microcontrolador: Arduino Mega**

#### **3.1.2.4.1 Características**

El Arduino Mega está basado en el microcontrolador ATmega2560. Tiene 54 pines de entradas/salidas digitales (14 de las cuales pueden ser utilizadas como salidas PWM), 16 entradas análogas, 4 UARTs (puertos serial por hardware), cristal oscilador de 16 Mhz, conexión USB, jack de alimentación, conector ICSP y botón de reset.

Incorpora todo lo necesario para que el microcontrolador trabaje; simplemente conéctalo a tu PC por medio de un cable USB o con una fuente de alimentación externa. El Arduino Mega es compatible con la mayoría de los shields diseñados para Arduino Duemilanove, diecimila o UNO.

ELEMENTO	INFORMACIÓN
Microcontrolador	Atmega2560
Voltaje de operación	5 V
Voltaje de entrada (Recomendado)	7 – 12 V
Voltaje de entrada (Límite)	6 – 20 V
Pines para entrada- salida digital.	54 (de los cuales 15 proporcionan salida PWM)
Pines de entrada analógica.	16
Corriente continua por pin IO	20 mA

Corriente continua en el pin 3.3 V	50 mA
Memoria Flash	256 KB, 8 KB utilizado por el gestor de arranque
SRAM	8 KB
EEPROM	4 KB
Velocidad de reloj	16 MHz
LED_BUILTIN	8
Longitud	101.52 mm
Anchura	53,3 mm
Peso	37 g

Tabla 2 Características Arduino Mega

Arduino Mega

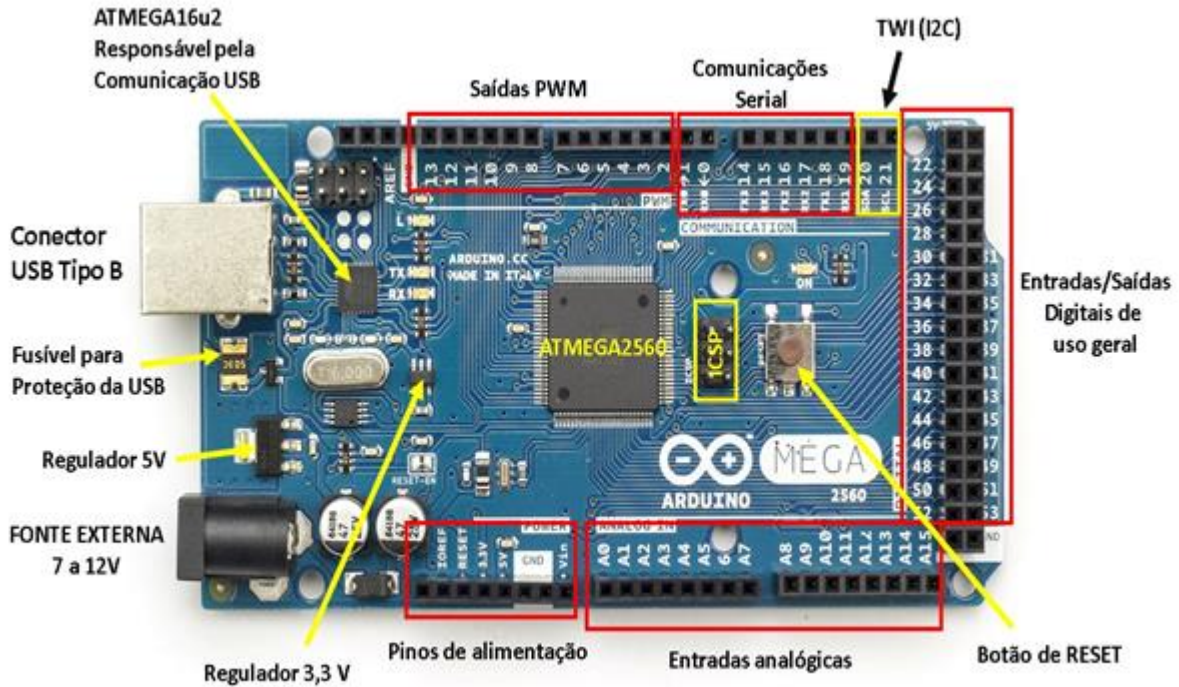


Figura 12. Microcontrolador Arduino Mega

Figura 12 Microcontrolador Arduino Mega

### 3.1.2.5 Sensores:

Los sensores a utilizar en el sistema domótico son:

Sensores	Funcionalidad	Ubicación
Sensor de humo y gas	Detectar humo en los distintas ambientes para alertar al usuario e iniciar actuadores correspondientes	Sala – Cocina – Habitaciones - Garaje
Sensor de llama	Detectar llamas en los distintos ambientes para alertar al usuario e iniciar actuadores correspondientes	Sala – Cocina – Habitaciones – Garaje
Sensor de Carga y	Detectar el consumo de	Sala – Cocina –

Consumo	cada uno de los equipos eléctricos conectados para medir el consumo	Habitaciones
Sensores flujo de aire	Mide la calidad del aire en caso de ser nociva alerta al usuario	Sala – Cocina – Habitaciones – Garaje
Sensor de temperatura	Mide la temperatura en los distintos ambientes de la casa y acondiciona el ambiente según parámetros	Sala – Cocina – Habitaciones
Sensor de humedad	Mide la humedad en el ambiente	Sala – Cocina – Habitaciones
Sensor de luz (fotorresistencia LDR)	Mide la intensidad de la luz en el ambiente e inicia actuadores según parámetros	Sala – Cocina – Habitaciones - Patio y Jardín
Sensor de movimiento infrarrojo	Utilizado para disparar alerta de intruso en caso de que la alarma este activada	Sala – Cocina – Habitaciones – Garaje
Sensor de sonido	Utilizado para disparar alerta de intruso en caso de que la alarma este activada	Sala – Cocina – Habitaciones – Garaje
Modulo Radio Frecuencia con Tarjeta	Utilizado como llave para ingresar al hogar e identificar que usuario ingreso	Puerta de entrada – Garaje
Sensores de calidad de agua	Mide el PH del agua y alerta al usuario sobre	Cocina – Baño

	los niveles de esta	
Sensor huellas digitales	Utilizado como llave para ingresar al hogar e identificar que usuario ingreso	Puerta de entrada – Garaje
Sensores de humedad de suelo	Ubicado en el suelo para activar los regadores automáticos	Patio y Jardín

**Tabla 3 Sensores Domóticos**



### 3.1.2.6 Actuadores:

Los actuadores a utilizar en el sistema domótico son:

Actuador - Dispositivo	Funcionalidad	Ubicación
Cámara DVR	Monitorear los distintos ambientes del hogar	Sala – Cocina – Habitaciones – Garaje – Puerta de Ingreso
Relé con Wifi	Activa/Desactiva la alimentación eléctrica de los dispositivos a controlar	Sala – Cocina – Habitaciones – Garaje
Avisadores Acústicos	Emite sonido según funcionalidad activada	Sala
Teclado para activar Alarma	Ingresa clave para activar/desactivar alarma	Sala
Electro-Válvulas	Abre/cierra las válvulas para el paso de agua o gas	Cocina – Patio Jardín
Motores Eléctricos	Abre/cierra para la apertura de puertas – portones y ventanas	Sala – Cocina – Habitaciones – Garaje – Puerta de Ingreso
Cerradura Eléctrica	Abre/Cierra el pestillo de la puerta principal y portón	Puerta de Ingreso – Garaje
Pantalla LCD	Muestra mensajes de la situación actual del hogar, temperatura, humedad, estado alarma, etc.	Sala – Cocina – Habitaciones
Ventiladores	Regula la temperatura del ambiente según parámetros	Sala – Cocina – Habitaciones
Extractores de Aire	Inicia al detectar fugas de gas en el hogar	Cocina

Calo-ventores	Regula la temperatura del ambiente según parámetros	Sala – Cocina – Habitaciones – Garaje
---------------	---	---------------------------------------

Tabla 4 Actuadores Domóticos

### **3.1.2.7 Protocolo de Comunicación:**

Los protocolos de comunicación a implementar en el sistema domótico son:

- TPC/IP: para las comunicaciones entre:
  - ✓ Microcontrolador y PC (para modulo Wifi).
  - ✓ PC y la página Web.
  - ✓ DVR y PC
- Radio Frecuencia: utilizado en el módulo lector de tarjetas para leer los códigos de las mismas
- SPI: comunica al microcontrolador con los módulos RFID, el cual lee las tarjetas para ingreso a la vivienda.
- RX-TX: para las comunicaciones entre el módulo WIFI y el controlador

### **3.1.2.8 Interfaz:**

Interfaz Web – Móvil ver módulos de la web con Gaby

La interfaz web cuenta con los siguientes módulos:

- Login de Usuario: Pantalla para el registro del usuario para la manipulación de la página web
- Mi Casa: Modulo en el que cambian los estados de los equipos del Arduino
- Mi Configuración: Muestra la configuración actual del controlador que administra la casa y la dirección IP
- Estadísticas: Muestra los estados actuales de los sensores de la casa, temperatura humedad, luces encendidas, sensor de gas, etc
- Contacto: Para enviar solicitudes al administrador del proyecto.
- Usuario: Modulo para el ABM de los usuarios

### 3.1.2.9 Software:

La herramienta computacional que se utiliza para realizar las funciones anteriores sobre cada servicio, fue escogida después de hacer un análisis de los existentes en el mercado y de los que están al alcance de los autores, a continuación se detallan los lenguajes y entornos de desarrollo a utilizar:

Lenguaje que se implementa para el desarrollo de la interfaz del sistema

#### 3.1.2.9.1 IDE Arduino:

Lenguaje de programación propio de Arduino Ver. 1.8.3 de 64 bits para programar las instrucciones en el microcontrolador. Dispone de bibliotecas para manipular los sensores y actuadores a utilizar en el sistema domótico

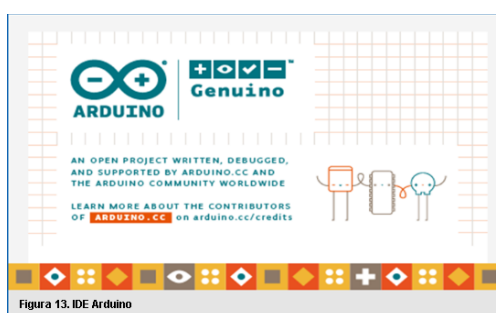


Figura 13 IDE Arduino

#### 3.1.2.9.2 Eclipse:

Es una plataforma de software compuesto por un conjunto de herramientas de programación de código abierto multiplataforma para desarrollar lo que el proyecto llama "Aplicaciones de Cliente Enriquecido". Esta plataforma, típicamente ha sido usada para desarrollar entornos de desarrollo integrados (del inglés IDE), como el IDE de Java llamado Java Development Toolkit (JDT) y el compilador (ECJ) que se entrega como parte de Eclipse.



Figura 14 IDE Eclipse

#### 3.1.2.9.3 MySQL

MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual: Licencia pública general/Licencia comercial por Oracle Corporation y está considerada como la base de datos de código abierto más popular para entornos de desarrollo web.



Figura 15 Sistema de Gestión de Base de Datos

### 3.1.2.9.4 Fritzing

Fritzing es el programa por excelencia para la realización de esquemas eléctricos en proyectos con Arduino y es open source. Dispone bibliotecas con la mayoría de componentes incluido los propios Arduino, placas de conexiones, led, motores, displays, etc. Además permite hacer esquemas eléctricos y diseñar nuestro PCB final.

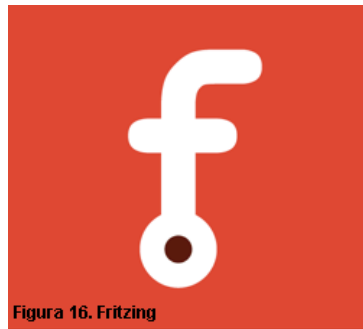


Figura 16 Fritzing

## 3.2 Resolución de Etapas según Metodología Adoptada: Etapas de Propuestas.

Por falta de pines decidimos colocar los siguientes dispositivos.

### 3.2.1 Definir los pines de salida y entrada de datos del microcontrolador.

Ambiente	Dispositivos	Pines	Entrada/Salida
Garaje	Servo	D10	Salida
	Luz	D11	Salida
	Pulsador Luz	D12	Entrada
Cocina	Pulsador	D37	Entrada
	Luz	D38	Salida
	Sensor Gas	A4	Entrada
	Calefacción	D39	Salida
	Sensor Temperatura	D40	Entrada

	Extractor	D41	Salida
	Luz Gas	D42	Salida
	Ventilador	D43	Salida
	Sensor Luz	A5	Entrada
<b>Habitación</b>	Calefacción	D32	Salida
	Luz	D33	Salida
	Ventilador	D34	Salida
	Pulsador	D35	Entrada
	Sensor Temperatura	D36	Entrada
	Sensor Luz	A3	Entrada
<b>Salón</b>	Pulsador	D44	Entrada
	Sensor Movimiento	D45	Entrada
	Sensor Temperatura	D46	Entrada
	Ventilador	D47	Salida
	Cortina	D4-D5-D6-D7	Salida
	Sensor Luz	A0	Entrada
	Micrófono	A1	Entrada
	Luz	D48	Salida
	Calefacción	D49	Salida
<b>Pasillo</b>	Sensor Movimiento	D22	Entrada
	Sensor Temperatura y Humedad	A2	Entrada
	Parlante	D23	Salida
	Pulsador Luz Interior	D24	Entrada
	Pulsador Luz Exterior	D25	Entrada
	Pulsador Puerta	D26	Entrada
	Luz Pasillo	D27	Salida
	Luz Exterior	D28	Salida
	Lector Tarjeta	D50-D51-52-53-D9	Entrada
	Servo Puerta	D29	Salida
	Luz Tarjeta Verde	D30	Salida
	Luz Tarjeta Roja	D31	Salida
<b>Sensor corriente</b>		D13	Entrada

Tabla 5 Pines de Entrada y Salida

\*D: refiere a pines Digitales

\*A: refiere a pines Analógico

### 3.2.2 Localización de actuadores y sensores.

Por falta de pines y presupuesto para el acceso a determinados sensores decidimos colocar los siguientes dispositivos.

#### 3.2.2.1 Sensores

Imagen	Sensores	Ubicación
	Sensor de humo y gas	Sala – Cocina – Habitaciones - Garaje
	Sensor de Carga y Consumo	Sala – Cocina – Habitaciones
	Sensor de temperatura	Sala – Cocina – Habitaciones
	Sensor de humedad	Pasillo
	Sensor de luz (fotorresistencia LDR)	Sala – Cocina – Habitaciones
	Sensor de movimiento infrarrojo	Sala – Pasillo


	Sensor de sonido	Sala
	Modulo Radio Frecuencia con Tarjeta	Puerta de entrada







Tabla 6 Sensores Utilizados

### 3.2.2.2 Actuadores


Imagen	Actuador - Dispositivo	Ubicación
	Cámara DVR (web cam)	Sala
	Avisadores Acústicos	Pasillo
	Motores Eléctricos (motor paso a paso y servo)	Sala - Garaje – Puerta de Ingreso
	Ventiladores (cooler en maqueta)	Sala – Cocina – Habitaciones
	Calo-ventores (led color rojo en maqueta)	Sala – Cocina – Habitaciones – Garaje
	Extractores de Aire (cooler en maqueta)	Cocina

Tabla 7 Actuadores Utilizados

### 3.2.2.3 Otros Dispositivos

Imagen	Dispositivo	Función
	<p>Arduino MEGA</p>	<p>Controlado/Central Domótica</p>
	<p>Relé</p>	<p>Encendido/Apagado de dispositivos eléctricos</p>
	<p>Módulo WiFi</p>	<p>Conexión Wifi de placa Arduino a router</p>
	<p>Leds</p>	<p>Simulación de Lámparas de luz</p>
	<p>Teclas Switch</p>	<p>Simulación de llave de luz</p>
	<p>Resistencias</p>	<p>Necesario para algunos dispositivos</p>



	Protoboard	Necesario para ordenar el circuito completo
---	------------	---

**Tabla 8 Otros Dispositivos Utilizados**

### 3.2.3 Diagrama de ubicación y cableado de nodos

#### 3.2.3.1 Diagrama Conexión de Nodos

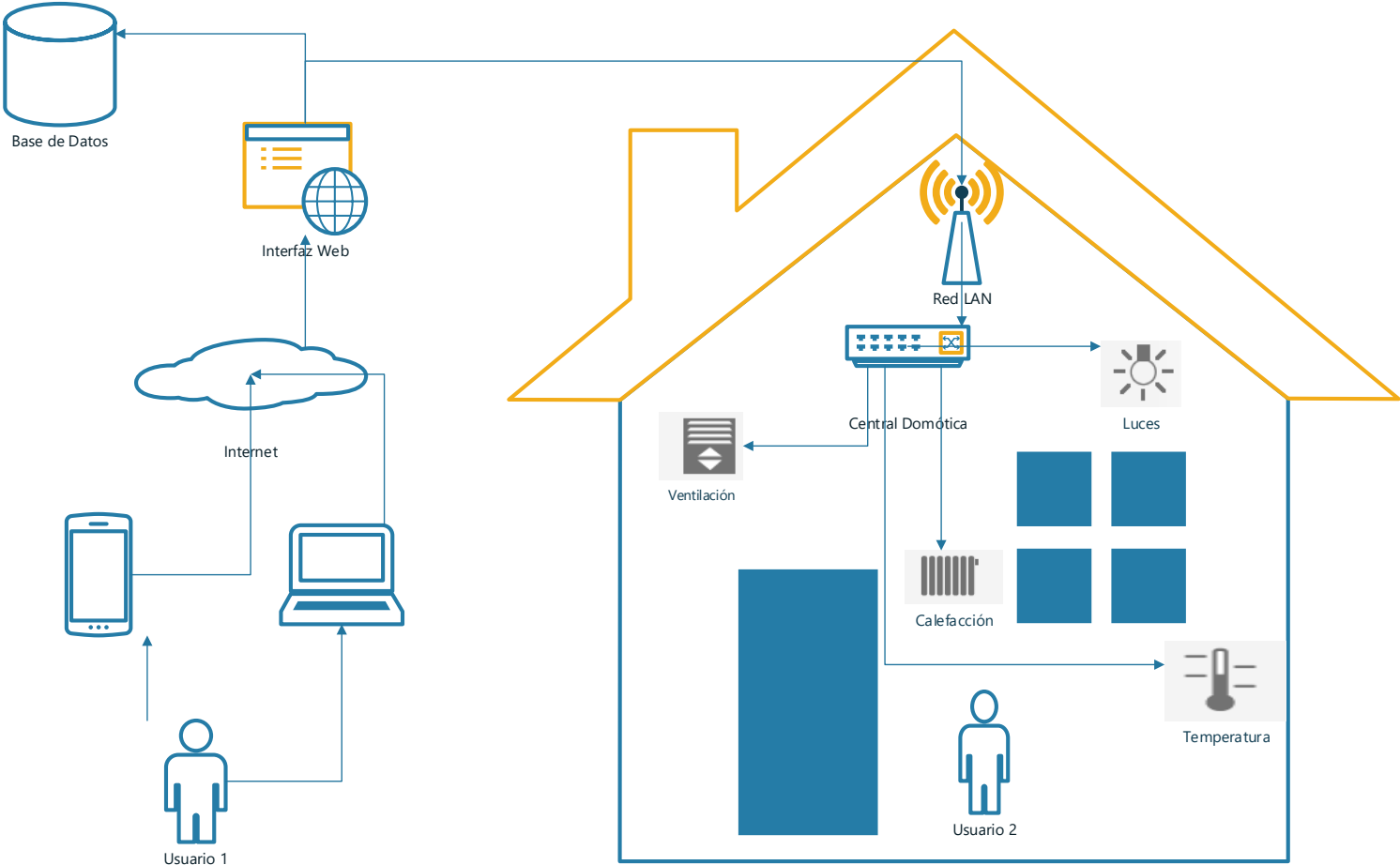


Figura17. Diagrama Conexión de Nodos

Figura 17 Diagrama de Conexión de Nodos

**3.2.3.2 Diagrama General de la Vivienda**



Figura 18 Diagrama General de la Vivienda

### 3.2.3.3 Ambientes

#### 3.2.3.3.1 Baño



Figura 19 Baño de la Vivienda

### 3.2.3.3.2 Garaje



Figura 20 Garaje de la Vivienda

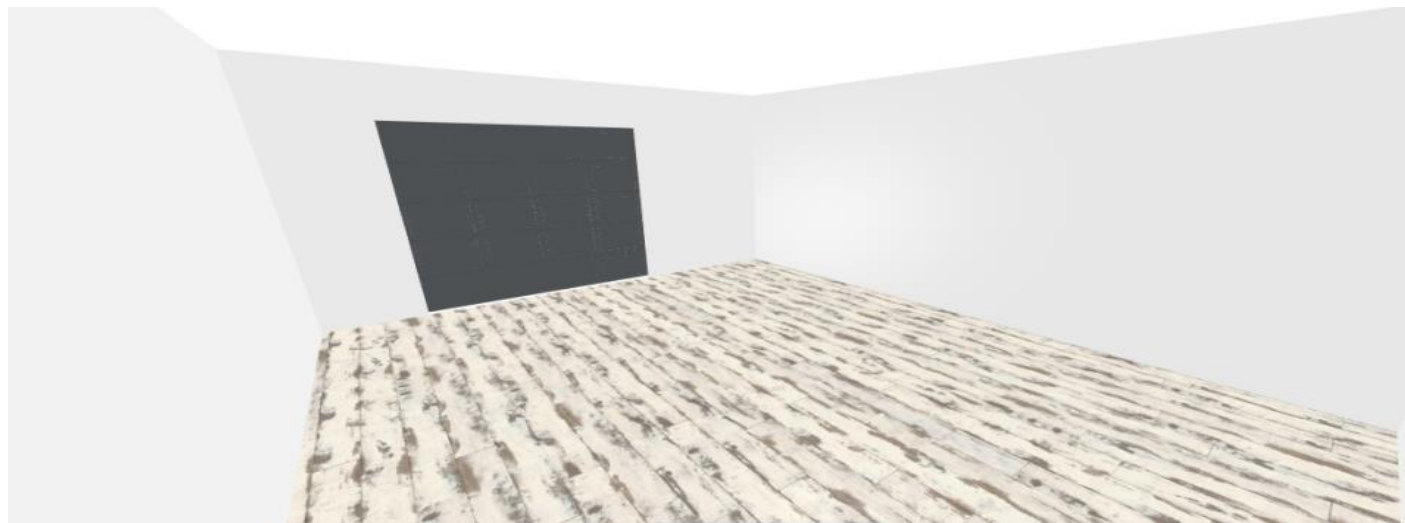


Figura 21 Garaje de la Vivienda

### 3.2.3.3.3 Cocina /Comedor



Figura 22 Cocina de la Vivienda



Figura 23 Cocina de la Vivienda

### 3.2.3.3.4 Jardín/Patio



Figura 24 Jardín de la Vivienda



Figura 25 Jardín de la Vivienda

### 3.2.3.3.5 Habitación



Figura 26 Habitación de la Vivienda



Figura 27 Habitación de la Vivienda



### 3.2.3.3.6 Sala



Figura 28 Sala de la Vivienda



Figura 29 Sala de la Vivienda

### 3.2.3.3.7 Puerta Entrada/Pasillo



Figura 30 Pasillo de la Vivienda



Figura 31 Pasillo de la Vivienda

### 3.2.3.4 Disposición de Sensores y Actuadores por Ambiente

#### 3.2.3.4.1 Distribución de Sensores Garaje

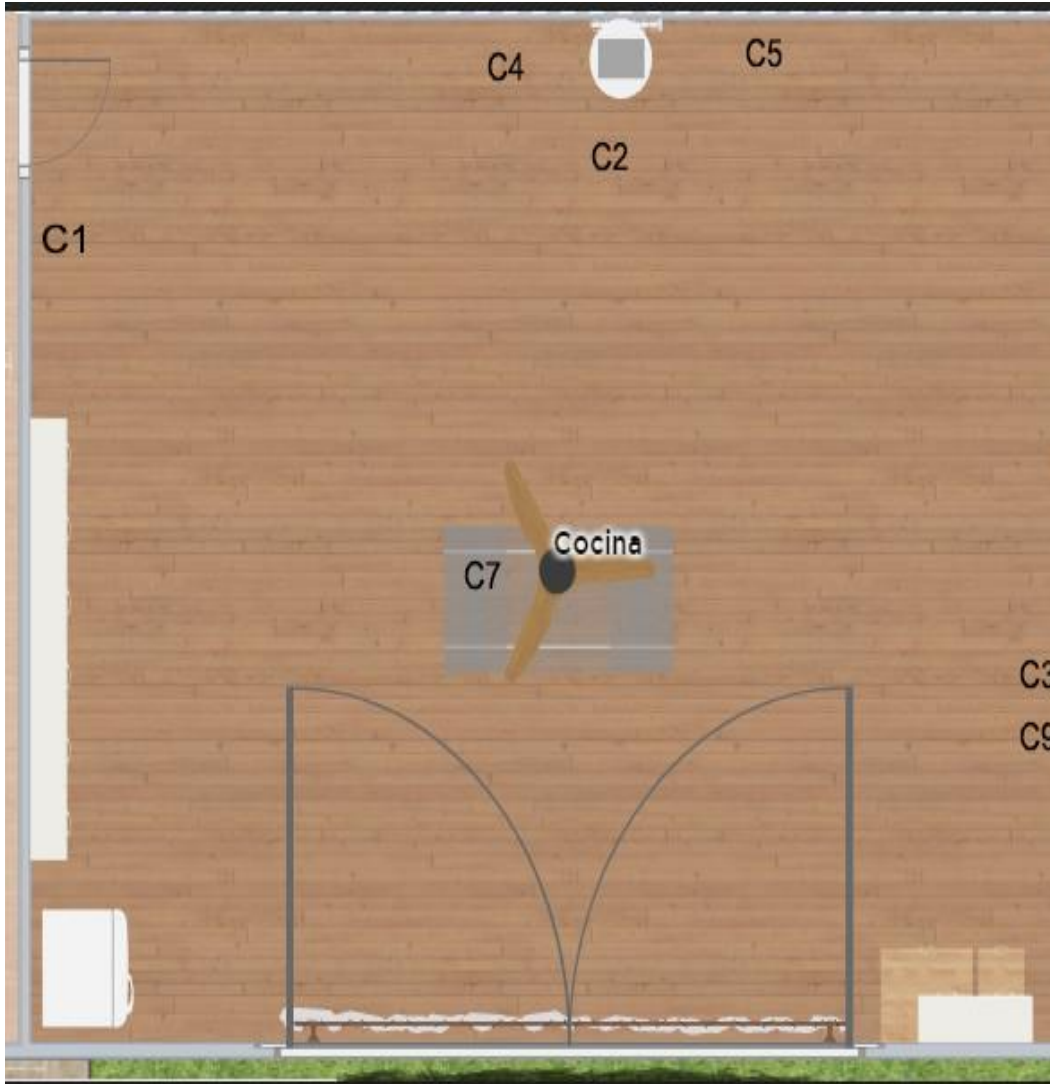


Figura 32 Diagrama Ubicación de Dispositivos en el Garaje

Referencia	Descripción
G1	Pulsador Portón/Luz
G2	Luz
G3	Motor apertura/cierre portón

Tabla 9 Distribución Dispositivos Garaje

### 3.2.3.4.2 Distribución de Sensores Cocina



Referencia	Descripción
C1	Pulsador Luz
C2	Luz
C3	Extractor
C4	Sensor Temperatura
C5	Calefacción
C6	Luz Testigo Gas
C7	Ventilador
C8	Sensor de Luz
C9	Sensor de Gas

Tabla 10 Distribución Dispositivos Cocina

Figura 33 Diagrama Ubicación de Dispositivos en la Cocina

**3.2.3.4.3 Distribución de Sensores Dormitorio**

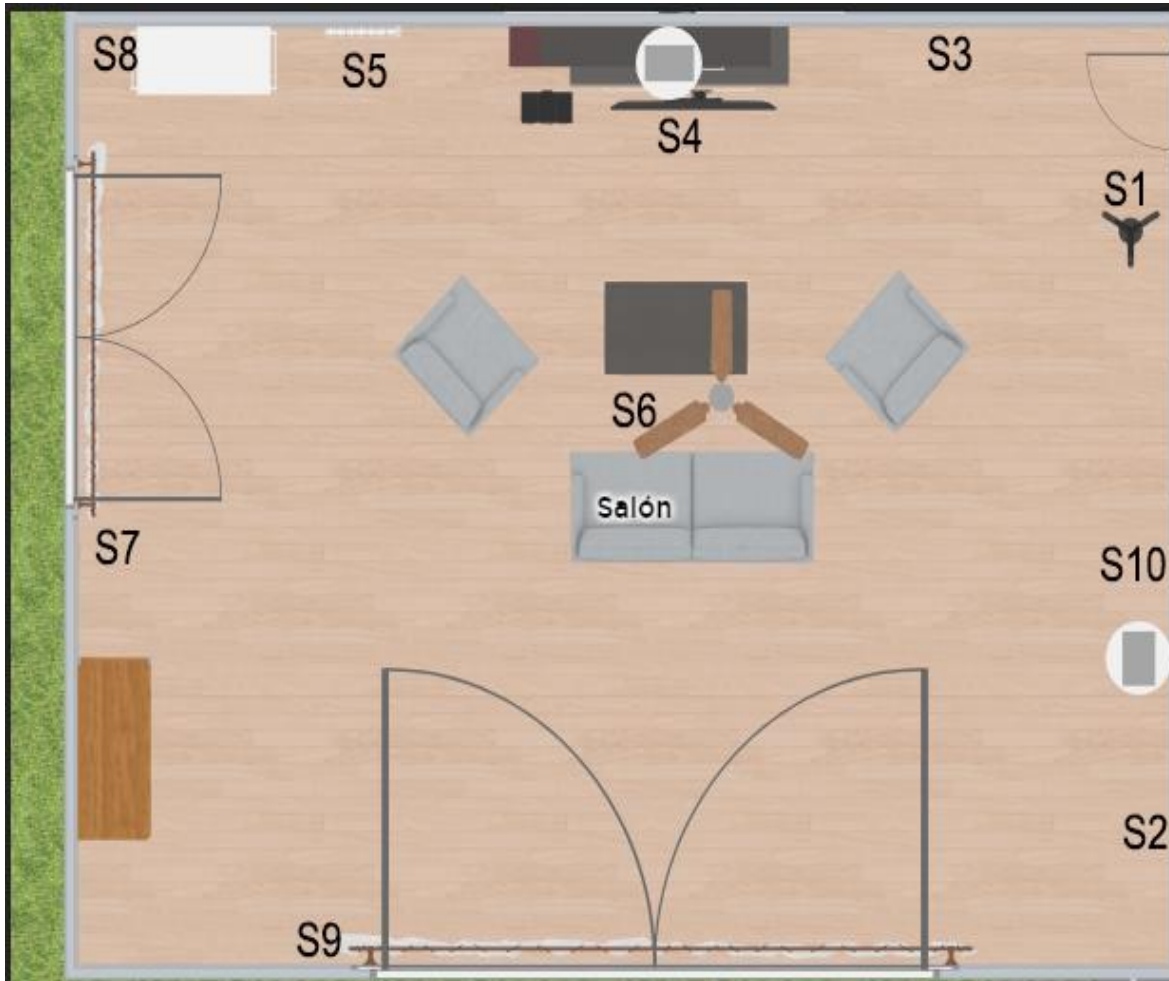


Referencia	Descripción
H1	Motor para movimiento de cortinas
H2	Calefacción
H3	Luz
H4	Ventilador
H5	Pulsador Luz
H6	Sensor Temperatura
H7	Sensor Luz(fotorresistencia)

**Tabla 11 Distribución Dispositivos Dormitorio**

**Figura 34 Diagrama Ubicación de Dispositivos en la Habitación**

### 3.2.3.4.4 Distribución de Sensores Salón



Referencia	Descripción
S1	Pulsador Luz
S2	Sensor Sonido (micrófono)
S3	Sensor Temperatura
S4	Luz
S5	Calefacción
S6	Ventilador
S7	Motor para movimiento de cortinas
S8	Sensor Luz(fotorresistencia)
S9	Motor para movimiento de cortinas
S10	Sensor Movimiento

Tabla 12 Distribución Dispositivos Salón

Figura 35 Diagrama Ubicación de Dispositivos en el Salón

### 3.2.3.4.5 Pasillo



Figura 36 Diagrama Ubicación de Dispositivos en el Pasillo

Referencia	Descripción
C1	Sensor Movimiento
C2	Sensor Temperatura y Humedad
C3	Parlante
C4	Pulsador Luz Interior
C5	Pulsador Luz Exterior
C6	Pulsador Puerta
C7	Luz Pasillo
C8	Luz Exterior
C9	Lector Tarjeta
C10	Servo Puerta
C11	Luz Tarjeta Verde
C12	Luz Tarjeta Roja

### 3.2.3.5 Conexión de todo el sistema domótico.

Sub-Sistemas de conexiones de los ambientes domóticos:

#### 3.2.3.5.1 Sub-Sistema Garaje

- Vista Protoboard

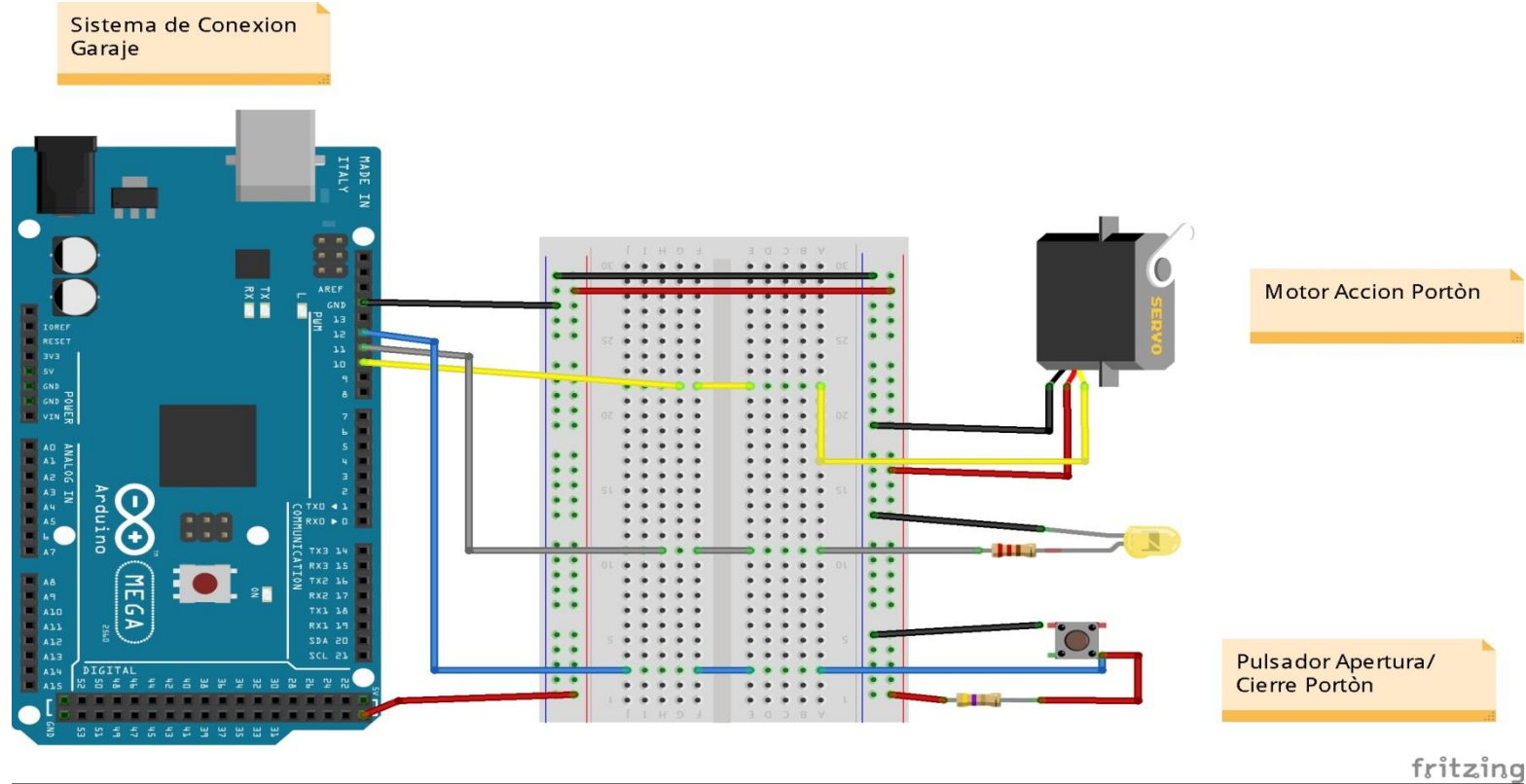


Figura 37 Diagrama de Conexión de Sensores y Actuadores en el Garaje



- Vista Esquemática

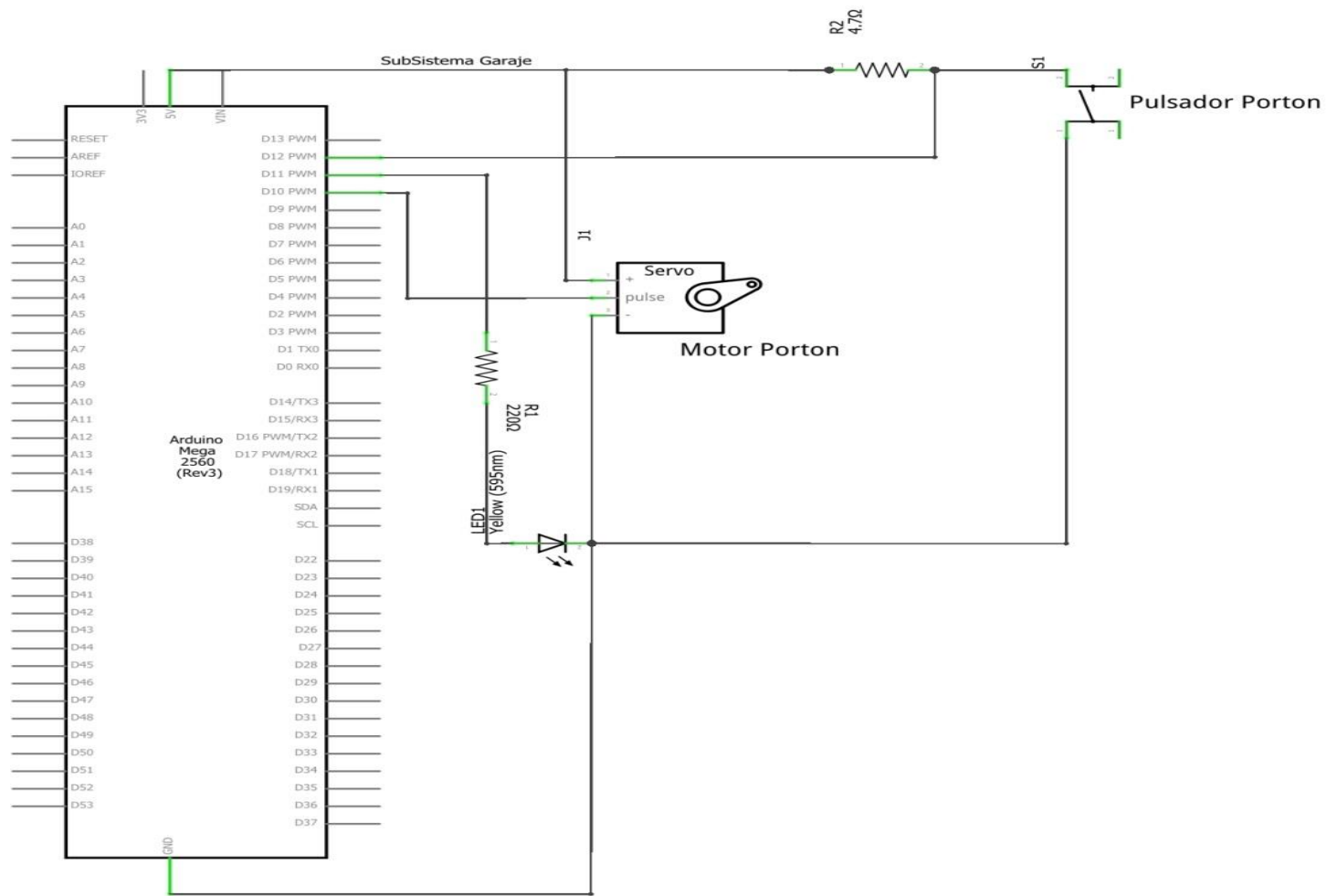


Figura 38 Diagrama de Conexión de Sensores y Actuadores en el Garaje

### 3.2.3.5.2 Sub-Sistema Cocina

- Vista Protoboard

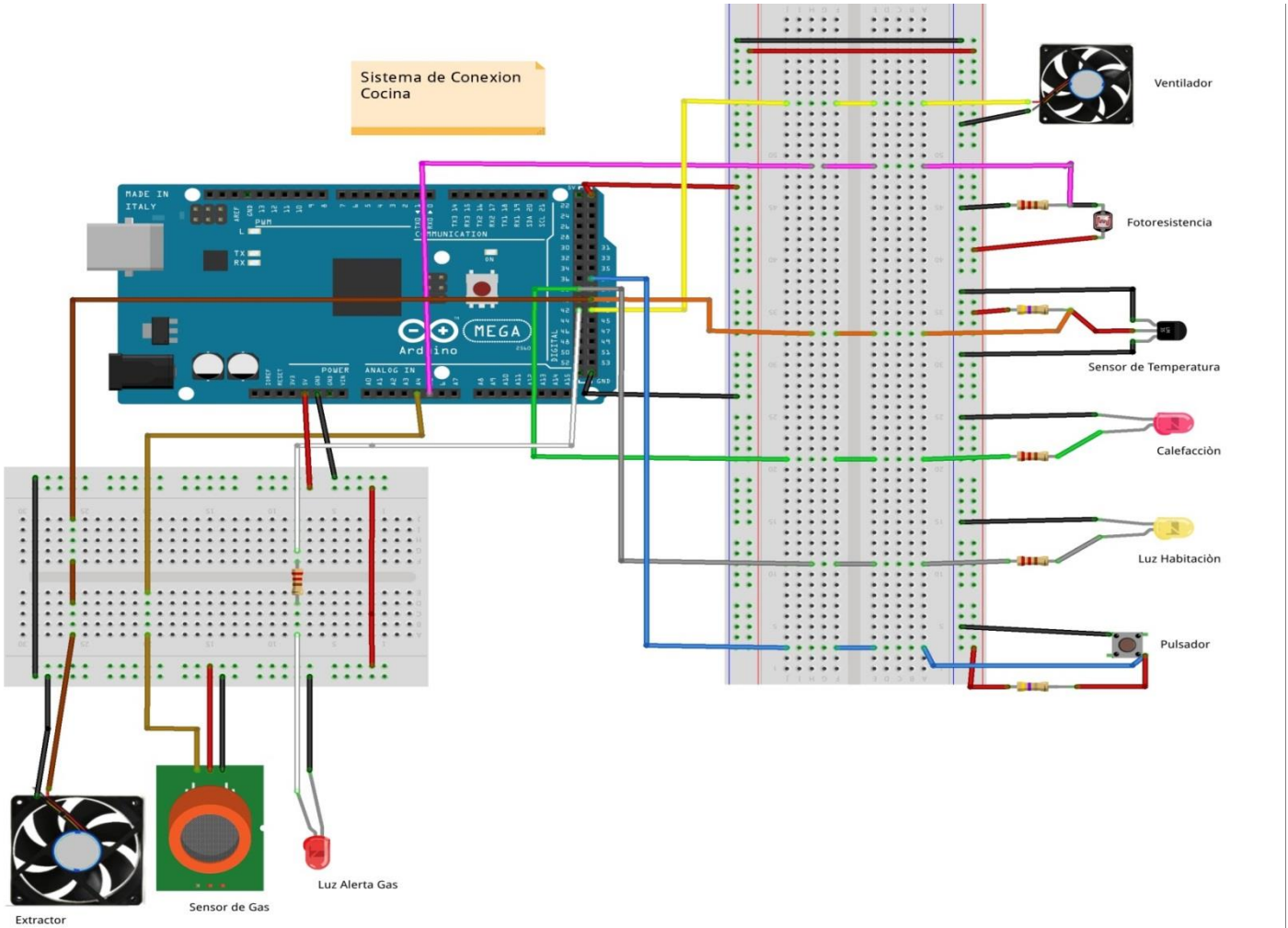


Figura 39 Diagrama de Conexión de Sensores y Actuadores en la Cocina

- Vista Esquemática

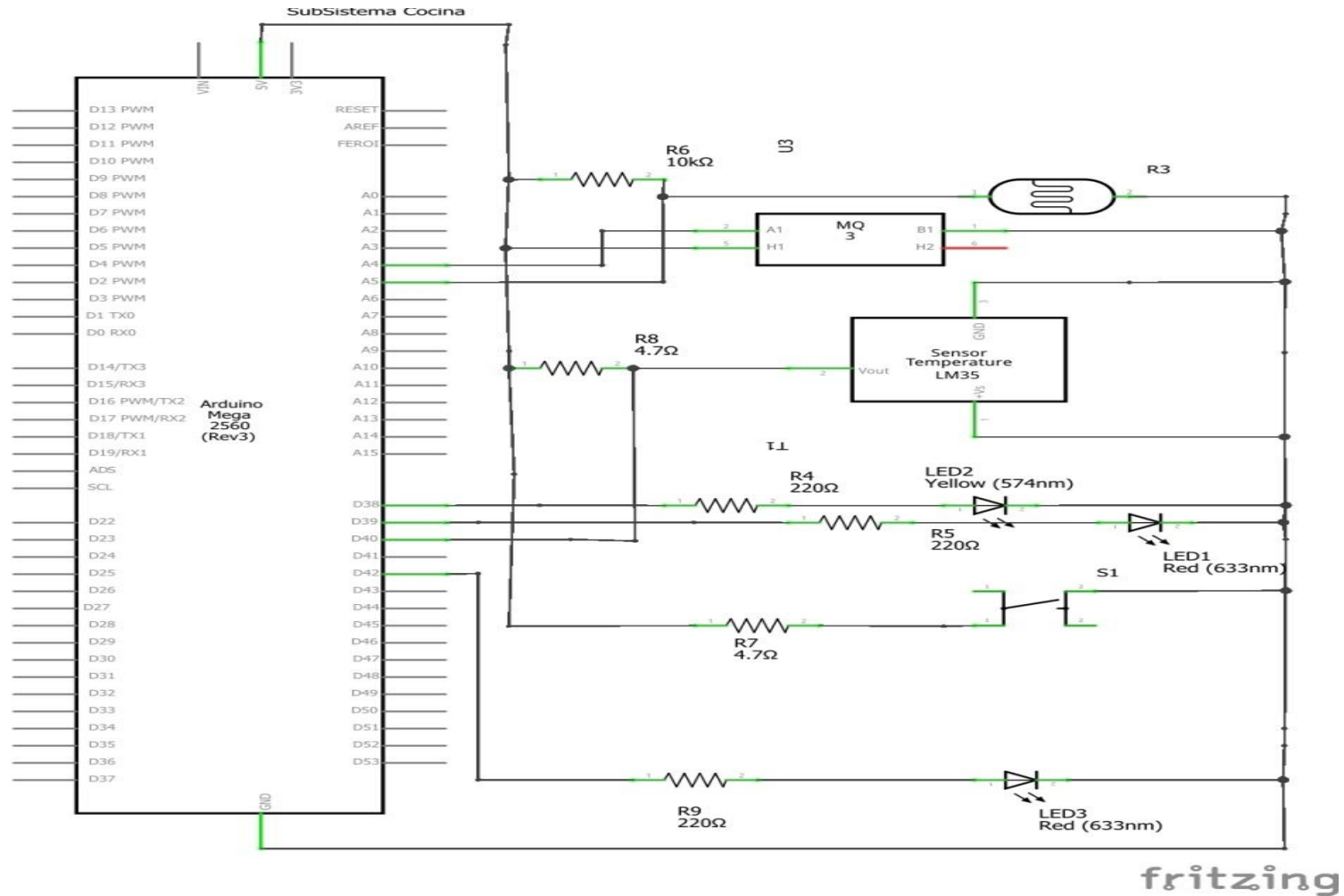


Figura 40 Diagrama de Conexión de Sensores y Actuadores en la Cocina

### 3.2.3.5.3 Sub-Sistema Habitación

- Vista Protoboard

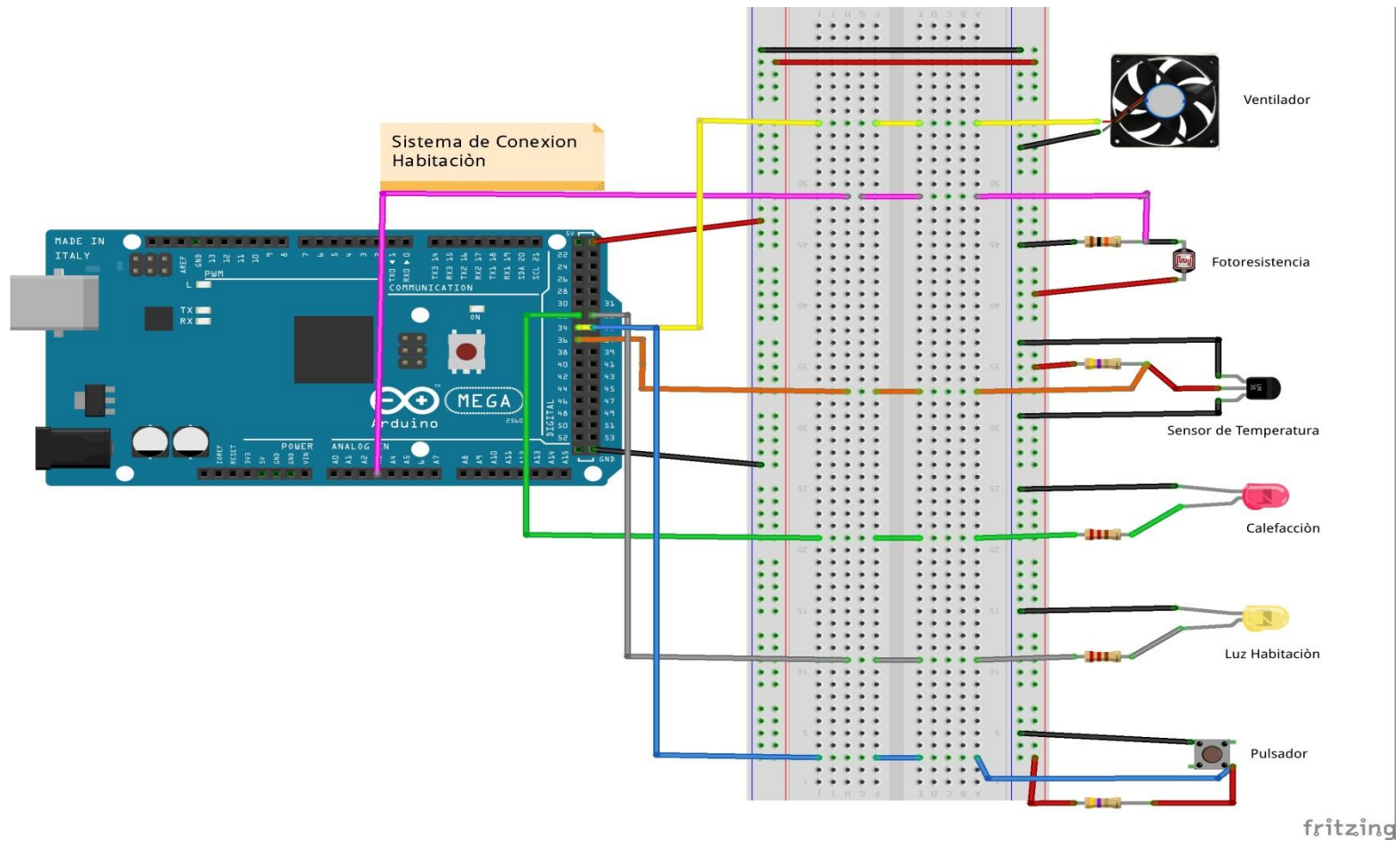


Figura 41 Diagrama de Conexión de Sensores y Actuadores en la Habitación

- Vista Esquemática

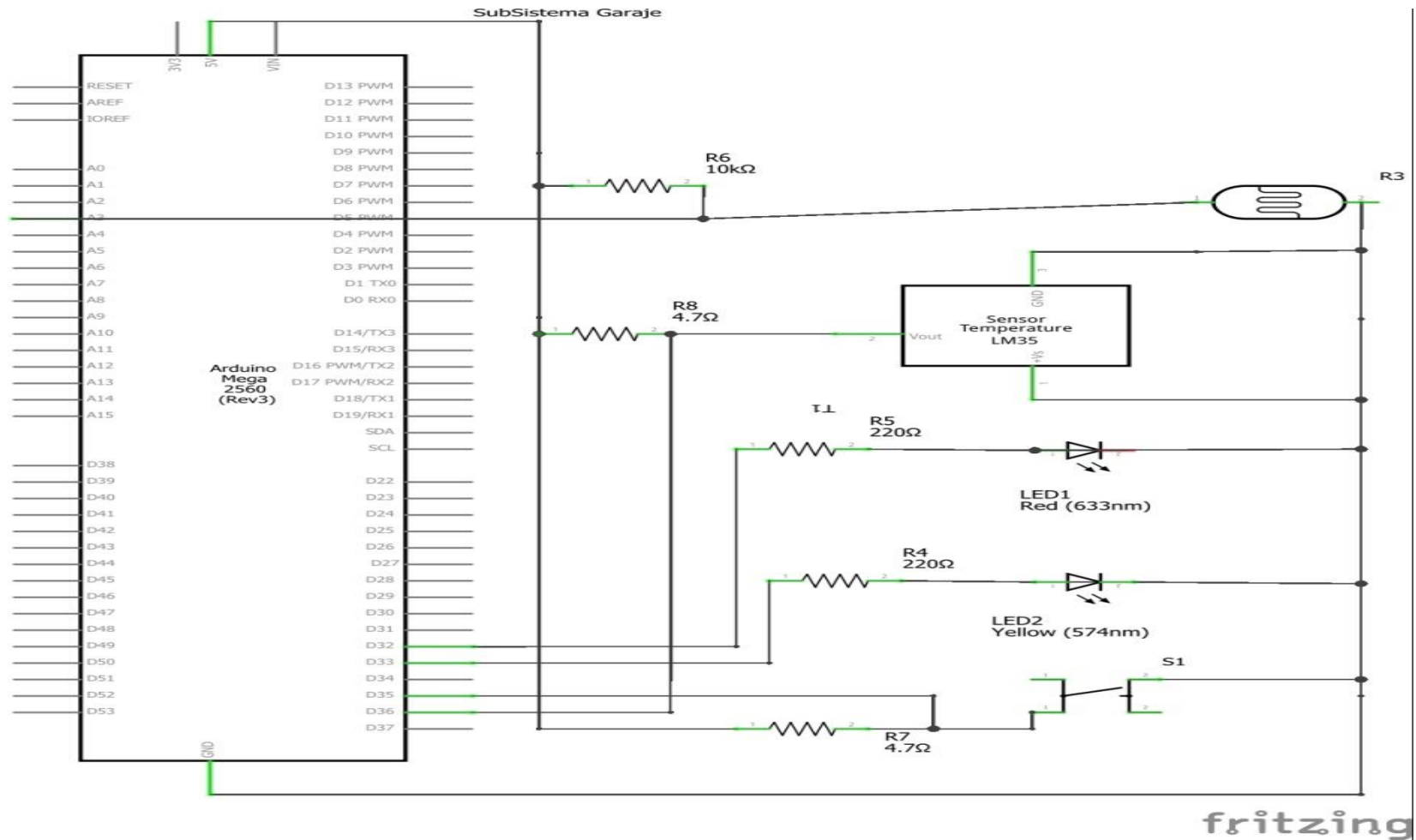


Figura 42 Diagrama de Conexión de Sensores y Actuadores en la Habitación

### 3.2.3.5.4 Sub-Sistema Salón

- Vista Protoboard

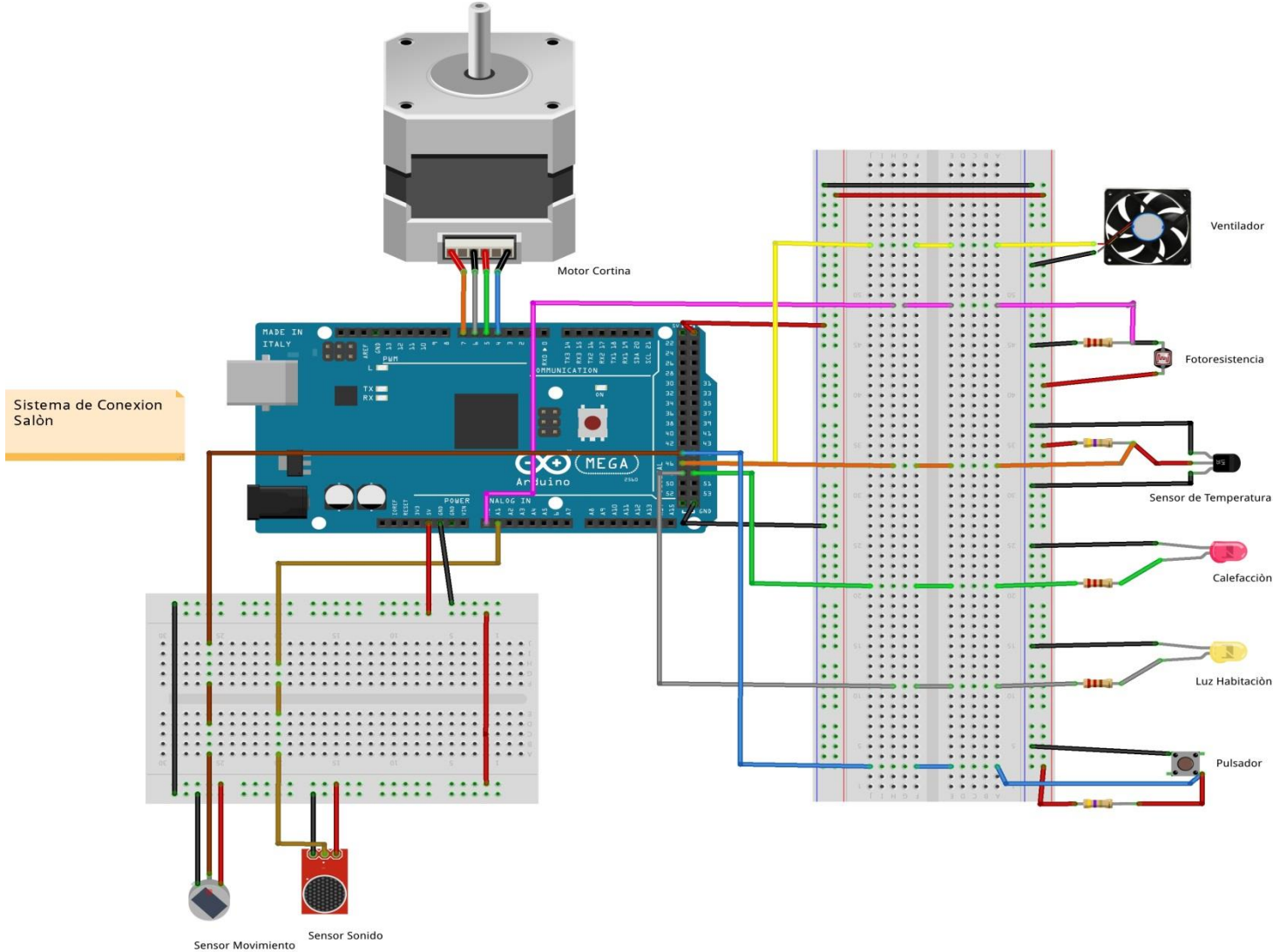
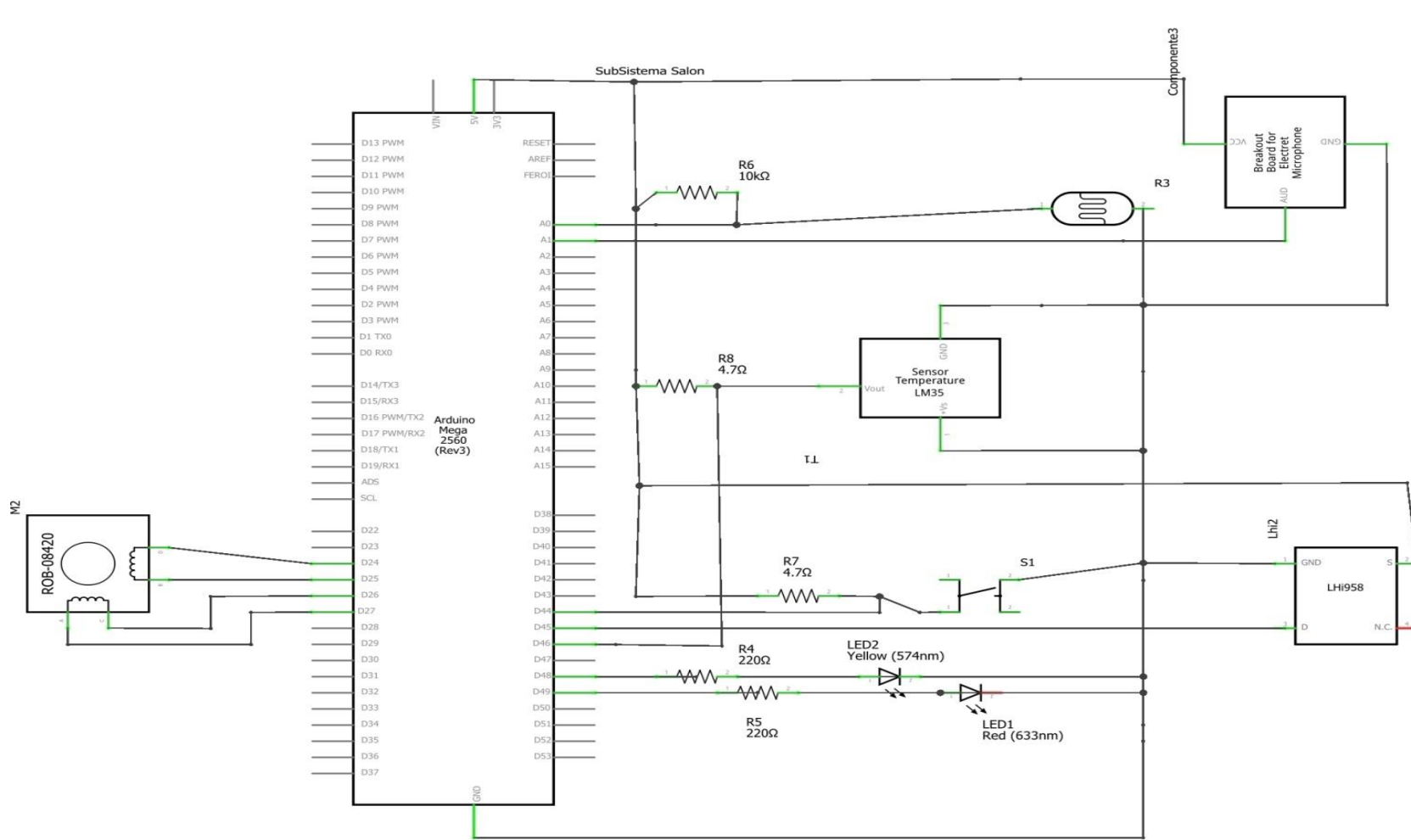


Figura 43 Diagrama de Conexión de Sensores y Actuadores en el Salón

- Vista Esquemática



fritzing

Figura 44 Diagrama de Conexión de Sensores y Actuadores en el Salón

### 3.2.3.5.5 Sub-Sistema Pasillo

- Vista Protoboard

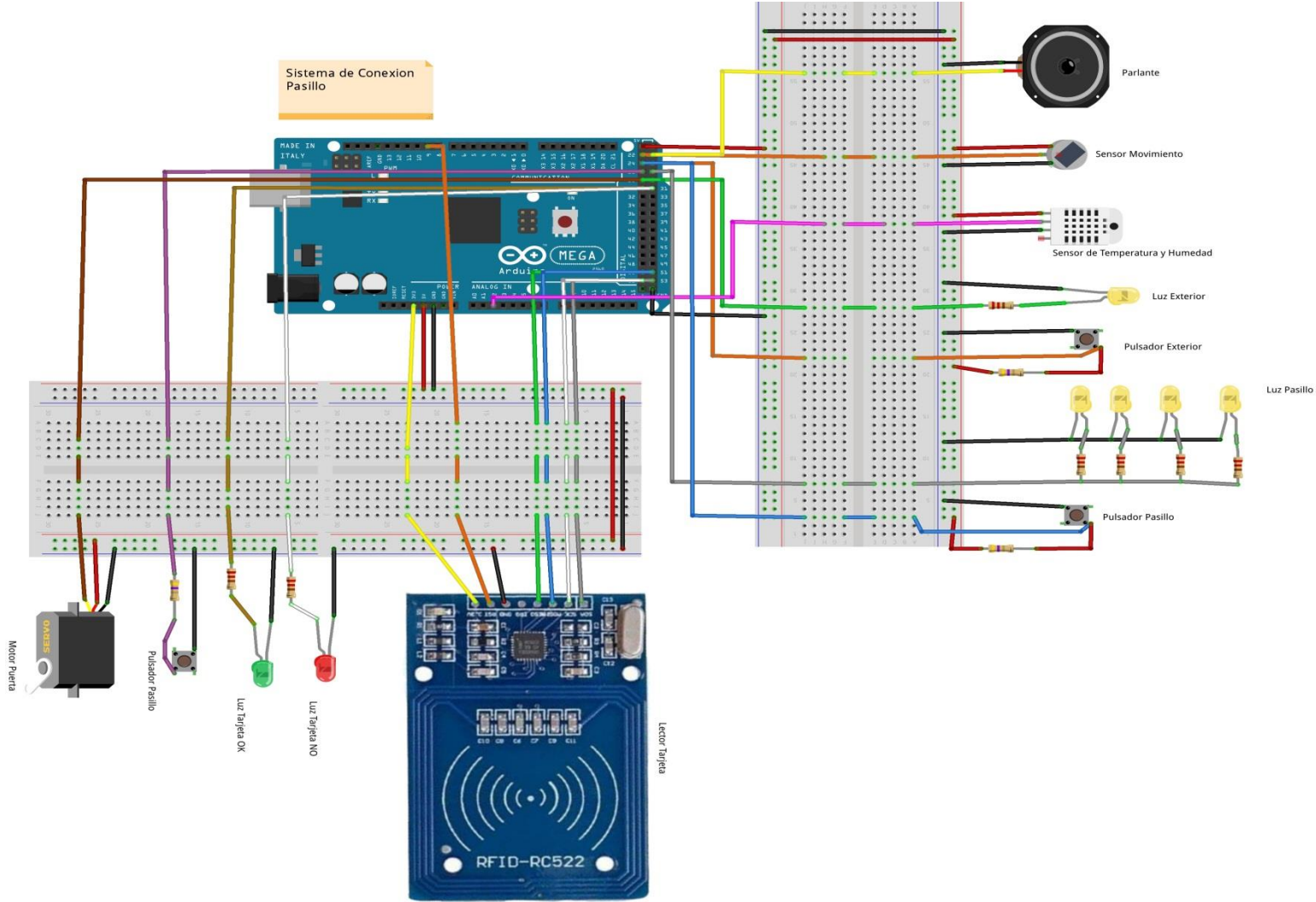


Figura 45 Diagrama de Conexión de Sensores y Actuadores en el Pasillo



- Vista Esquemática

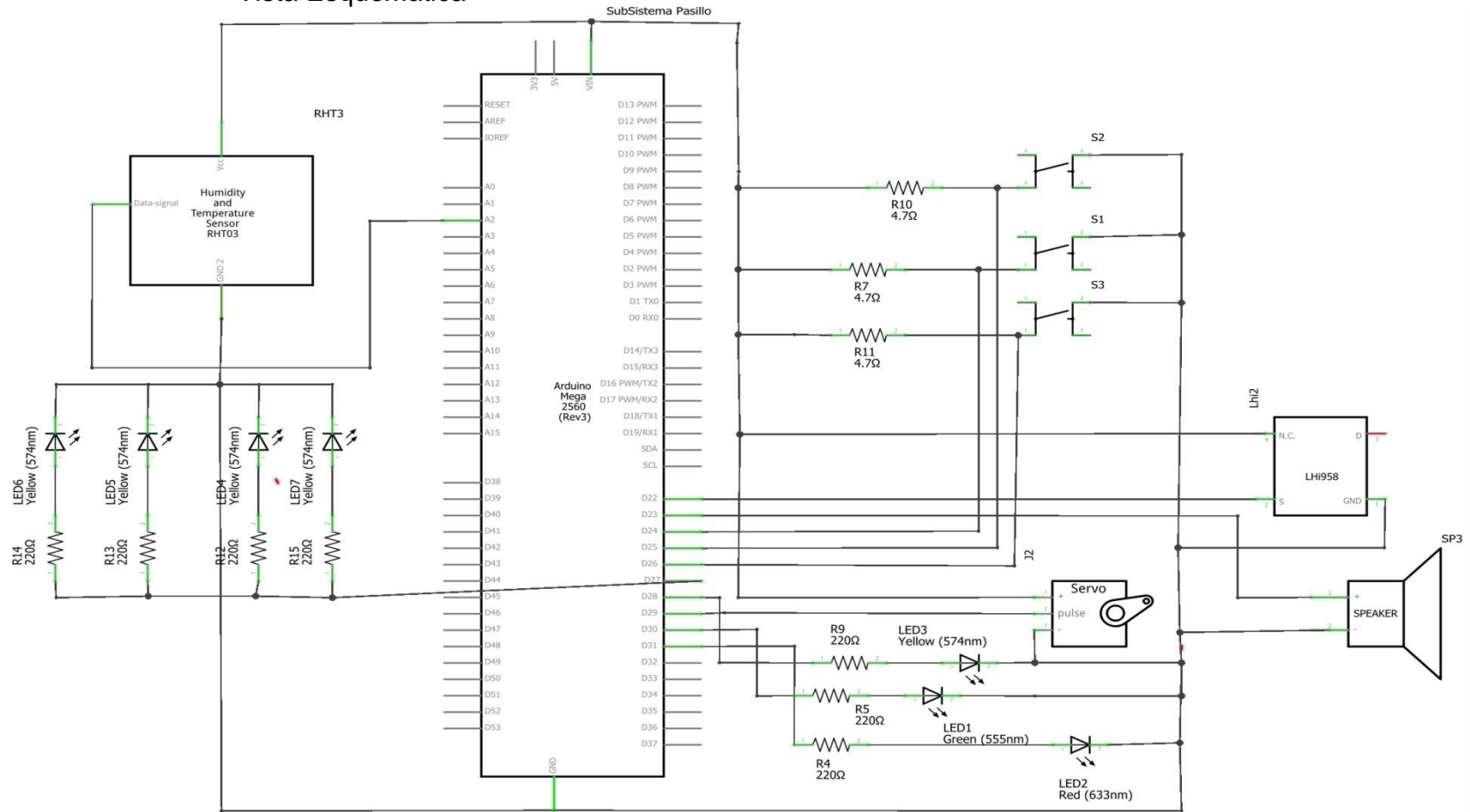


Figura 46 Diagrama de Conexión de Sensores y Actuadores en el Pasillo



### 3.2.3.6 Bosquejo de la interfaz

Maquetación de las pantallas de la página web, donde el usuario interactúa con el sistema domótico, ubicación de los controles, menús y sub-menú

#### 3.2.3.6.1 Login.jsp

Usuario	
Clave	

Figura 47 Bosquejo Login.jsp

#### 3.2.3.6.2 Index.jsp

Inciso de Sesión / Usuario		Menú Principal	
	Imagen		
Pie / Menú			

Figura 48 Bosquejo Index.jsp

#### 3.2.3.6.3 Acciones.jsp

Inciso de Sesión / Usuario		Menú Principal									
	<table border="1"><thead><tr><th>Dispositivo</th><th>Estado</th></tr></thead><tbody><tr><td></td><td></td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr></tbody></table>		Dispositivo	Estado							
Dispositivo	Estado										
Pie / Menú											

Figura 49 Bosquejo Acciones.jsp

### 3.2.3.6.4 Miconfiguracion.jsp

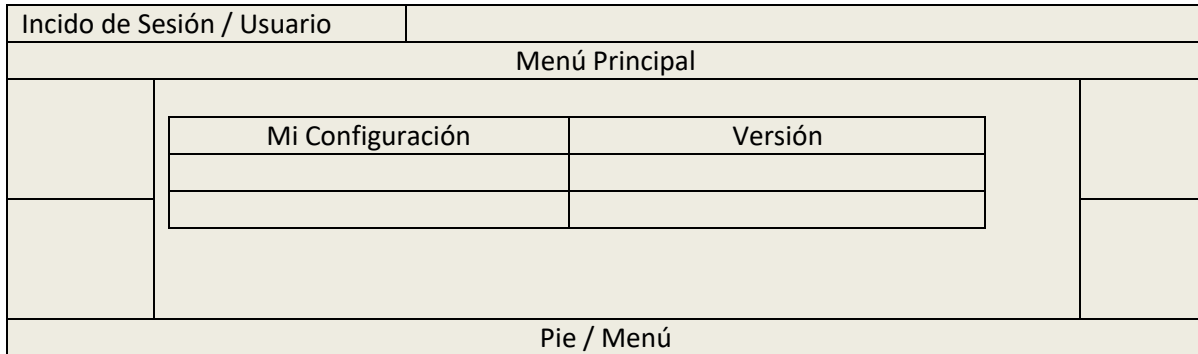


Figura 50Bosquejo Miconfiguracion.jsp

### 3.2.3.6.5 Estadística.jsp

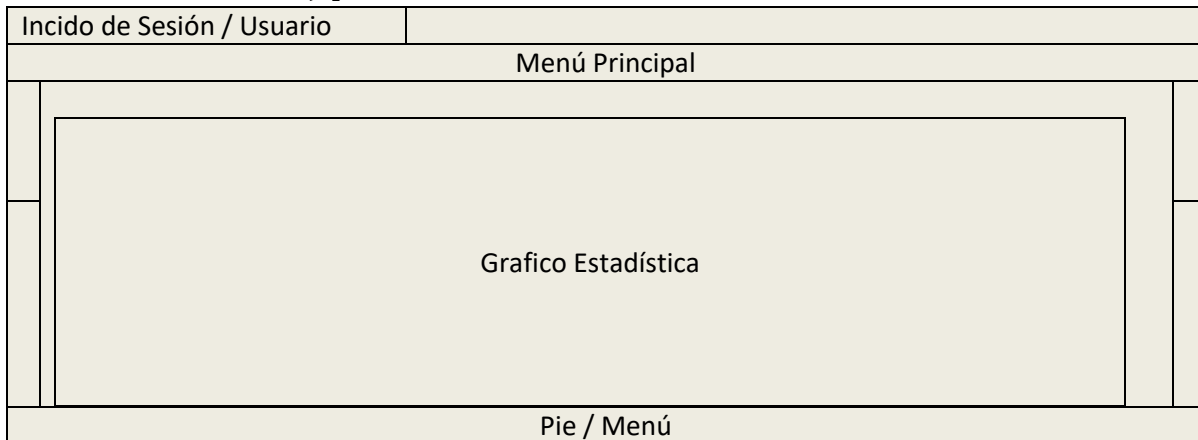


Figura 51Bosquejo Estadistica.jsp

### 3.2.3.6.6 Logs.jsp

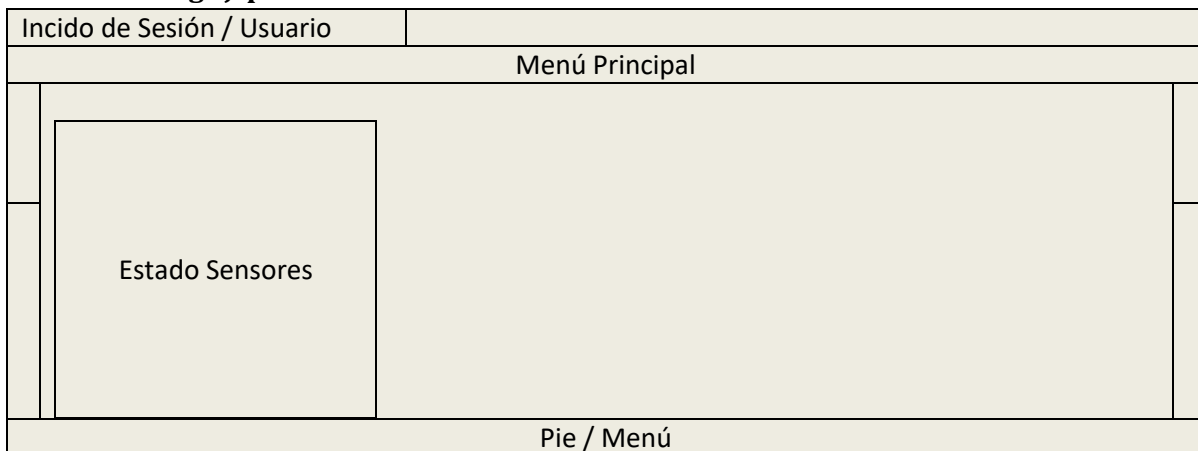


Figura 52Bosquejo Logs.jsp

### 3.2.3.6.7 Contacto.jsp

Incido de Sesión / Usuario		Menú Principal	
Menú Principal			
	Nombre		
	Apellido		
	Mail		
	Teléfono		
	Dirección		
	Ciudad		
	Postal		
	Comentario		
Pie / Menú			

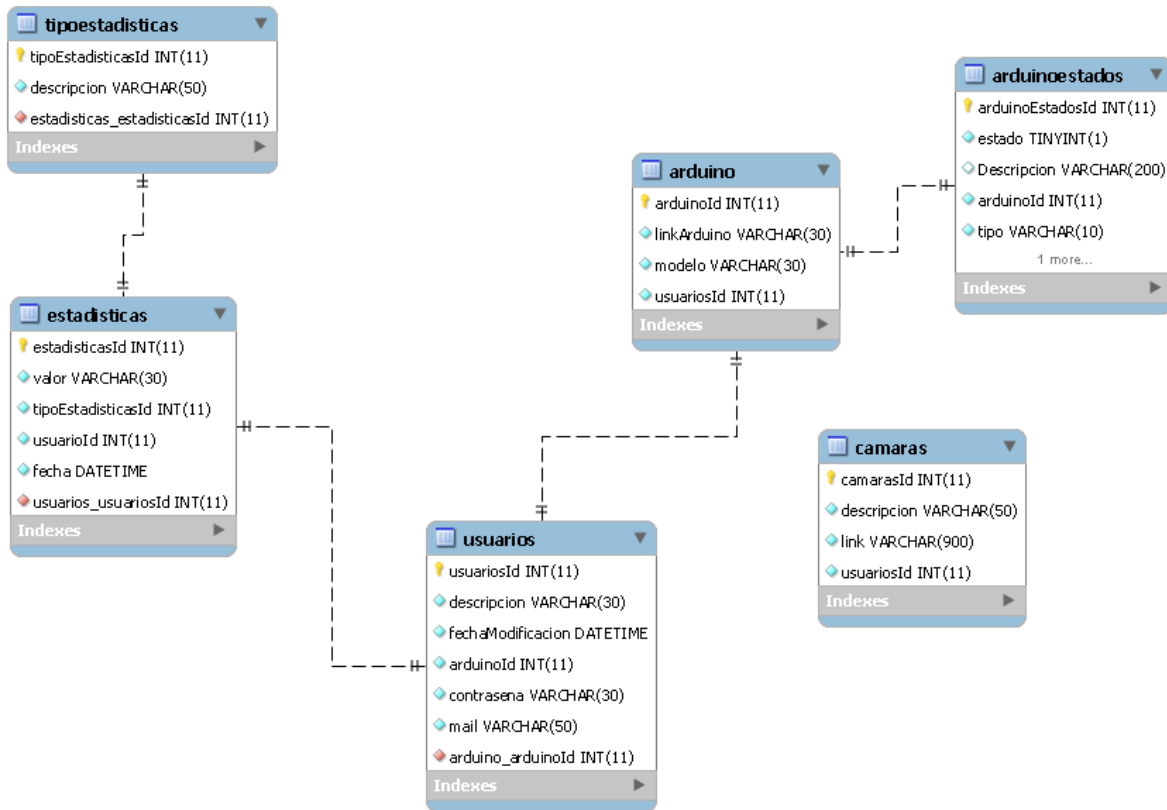
Figura 53Bosqujo Contacto.jsp

### 3.2.3.7 Usuario.jsp

Incido de Sesión / Usuario		Menú Principal	
Menú Principal			
	Usuario	Estado	
Pie / Menú			

Figura 54Bosquejo Usuario.jsp

### 3.2.3.8 Diagrama Entidad Relación



### 3.2.3.9 Casos de Uso

#### 3.2.3.9.1 Lista de Actores

Usuario: Operador que interactúa con el sistema domótico realizando las acciones de encendido/ apagado, apertura/ cierres en los elementos domóticos, así como también consultar el estado actual de la vivienda

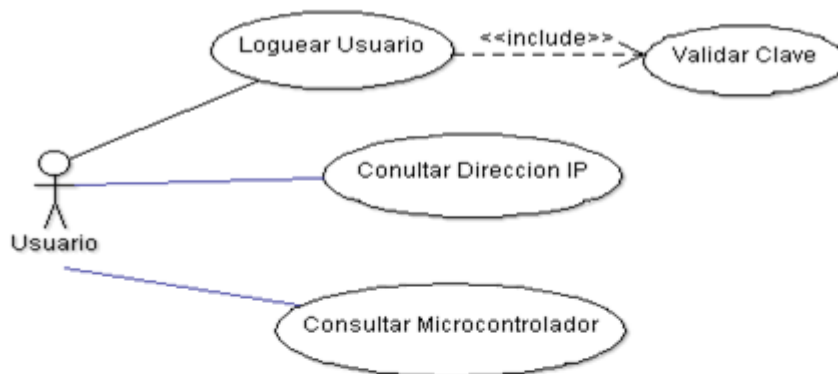
Control Domótico: Hardware encargado de tomar las peticiones del usuario y cambiar los estados de los actuadores (motores, luces, ventiladores, calefacción, etc), también envía información del estado actual de los sensores en los distintos ambientes de la vivienda

### **3.2.3.9.2 Listado de Casos de Uso**

- ✓ Activar Actuadores Alarma
- ✓ Activar Alarma
- ✓ Agregar Tarea
- ✓ Agregar Usuario
- ✓ Alertar Usuario
- ✓ Apagar Calefacción
- ✓ Apagar Luces
- ✓ Apagar Ventilador
- ✓ Apertura de Cortinas
- ✓ Apertura de Puertas
- ✓ Cambiar Estado Calefacción
- ✓ Cambiar Estado Luces
- ✓ Cambiar Estado Ventilador
- ✓ Cierre de Puerta
- ✓ Cierre de Cortina
- ✓ Consultar Microcontrolador
- ✓ Consultar Sensor Alarma
- ✓ Consultar Sensor Corriente
- ✓ Consultar Sensor Gas
- ✓ Consultar Sensor Luz
- ✓ Consultar Sensor Temperatura
- ✓ Consultar Dirección IP
- ✓ Desactivar Alarma
- ✓ Eliminar Tarea
- ✓ Eliminar Usuario
- ✓ Encender Calefacción
- ✓ Encender Extractor
- ✓ Encender Luces
- ✓ Encender Luz Aviso
- ✓ Ingresar Datos
- ✓ Loguear Usuario
- ✓ Modificar Tarea
- ✓ Modificar Usuario
- ✓ Pulsador Luces

- ✓ Pulsador Puerta
- ✓ Registrar Evento
- ✓ Validar Usuario

### Caso de Uso: Consultar Microcontrolador



Caso de Uso N°1	Consultar Microcontrolador	
Actores	Usuario	
Tipo		
Descripción	Consulta estado actual del Microcontrolador	
Propósito	Informar al usuario las características del microcontrolador	
Precondición	Ingresar en el sistema web	
Resultado Exito	Muestra información del microcontrolador	
Resultado Fracaso	Error al comunicarse con el microcontrolador	
Curso Normal de Eventos		
Acciones de los Actores	Respuestas del sistema	Alternativa
1 El usuario ingresa en el sistema	2 El sistema valida el nombre de usuario y contraseña.	2.1 Falla nombre de usuario o clave
3 El usuario ingresa al menú de consultar estado de microcontrolador	4 El sistema muestra la Dirección IP del microcontrolador y características de este	4.1 Error al comunicarse con el microcontrolador

Figura 55 Caso de Uso Consultar Microcontrolador



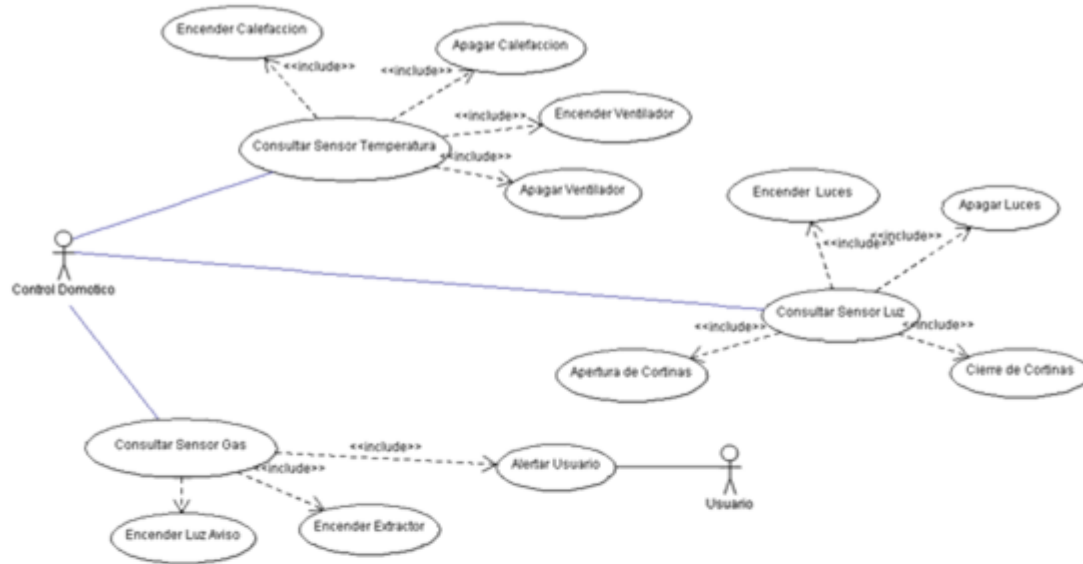
## Caso de Uso: Cambiar Estado Actuadores Web



Caso de Uso Nº2	Cambiar Estado Actuadores Web	
Actores	Usuario –Control Domótico	
Tipo		
Descripción	Cambiar el estado de las luces, calefacción, ventilación, cortinas y puertas de acceso a través de la interfaz web	
Propósito	Cambiar estados de los distintos actuadores en la vivienda	
Precondición	Ingresar en el sistema web	
Resultado Exitoso	Cambia el estado de los actuadores seleccionados	
Resultado Fracaso	Error al comunicarse con el microcontrolador	
Curso Normal de Eventos		
Acciones de los Actores	Respuestas del sistema	Alternativa
1 El usuario ingresa en el sistema	2 El sistema valida el nombre de usuario y contraseña.	2.1 Falla nombre de usuario o clave
3 El usuario ingresa al menú "Mi Casa"	4 El sistema muestra todos los cambios posibles de realizar en los actuadores, luce, ventilación, calefacción, apertura/cierre de puertas o cortinas, etc.	4.1 Error al comunicarse con el microcontrolador 4.2 Los actuadores se han dañado con el tiempo o un desperfecto, ejemplo luz de la habitación quemada.
5 El usuario prende/apaga o abre/cierra uno de los dispositivos desde la web	6 El sistema recibe la petición del dispositivo a cambiar de estado y envía el cambio de estado al actuador.	6.1 Error el sistema no puede comunicarse con el microcontrolador

Figura 56 Caso de Uso Cambiar Estado Actuadores Web

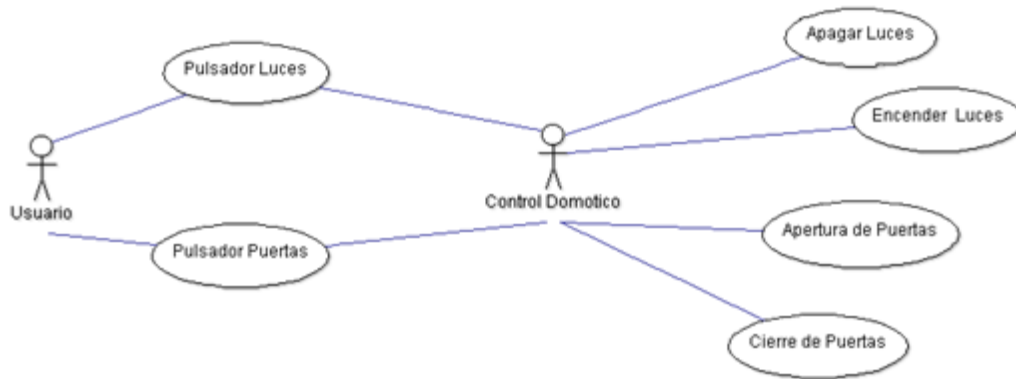
### Caso de Uso: Cambiar Estado Actuadores Automático



Caso de Uso Nº3	Cambiar Estado Actuadores Automático	
Actores	Usuario –Control Doméstico	
Tipo		
Descripción	El microcontrolador cambiar el estado de las luces, calefacción, ventilación y cortinas según condiciones preestablecidas	
Propósito	Cambiar estados de los distintos actuadores en la vivienda	
Precondición	Habilitar vía web es cambio automático de actuadores	
Resultado Exito	Cambia el estado de los actuadores seleccionados	
Resultado Fracaso	Error al cambiar de estado los actuadores	
Curso Normal de Eventos		
Acciones de los Actores	Respuestas del sistema	Alternativa
<p>1 El microcontrolador, a través de los sensores, toma lectura de los cambios en temperatura, iluminación, pérdidas de gas, etc</p> <p>3 El usuario recibe una alerta en caso de pérdidas de gas en la casa</p>	<p>2 Según condiciones preestablecidas en la lógica del microcontrolador, este cambia los estados de las luces en caso de poca iluminación y bajo/levanta las cortinas. Prende/ apaga la calefacción o ventilación según la temperatura. Alerta al usuario en caso de pérdida de gas.</p>	<p>2.1 Los actuadores se han dañado con el tiempo o un desperfecto, ejemplo luz de la habitación quemada.</p> <p>2.2 Error el sistema no puede alertar al usuario</p>

Figura 57 Caso de Uso Cambiar Estado Actuadores Automático

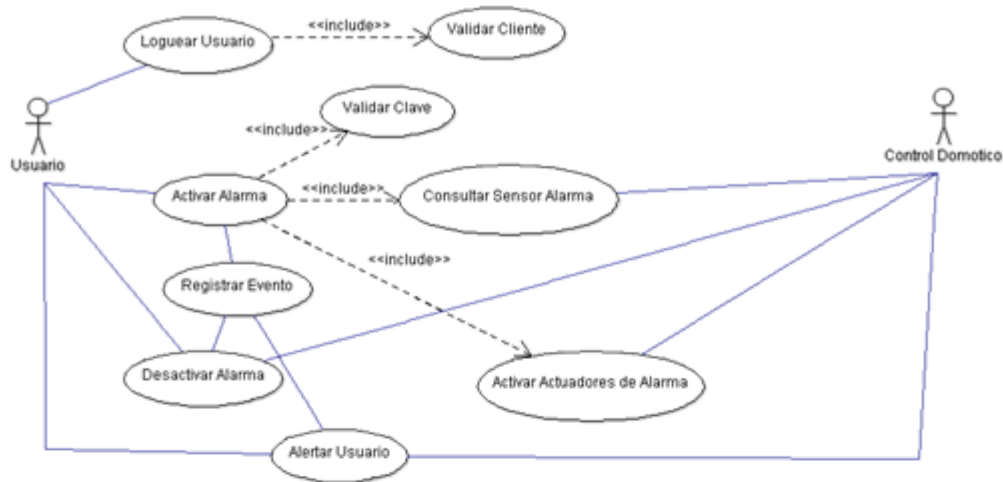
### Caso de Uso: Cambiar Estado Actuadores Manual



Caso de Uso Nº4	Cambiar Estado Actuadores Manual	
Actores	Usuario –Control Domótico	
Tipo		
Descripción	El Usuario prende las luces desde de los pulsadores, abre/cierra las puertas de forma manual	
Propósito	Cambiar el estado de los actuadores de forma manual	
Precondición		
Resultado Exito	Prender/Apagar las luces de la vivienda	
Resultado Fracaso	Las luces no prenden	
Curso Normal de Eventos		
Acciones de los Actores	Respuestas del sistema	Alternativa
1 El usuario prende/apaga las luces desde el pulsador	2 El sistema recibe la petición y cambia el estado del actuador.	2.1 Error al comunicarse con el sistema.
3 El usuario utiliza su tarjeta magnética para abrir/ cerrar la puerta de ingreso	4 El sistema valida la clave de la tarjeta y abre/cierra la puerta de ingreso.	4.1 Falla al leer la tarjeta de ingreso 4.2 Falla al activar los actuadores de cierre/apertura de la puerta

Figura 58 Caso de Uso Cambiar Estado Actuadores Manuales

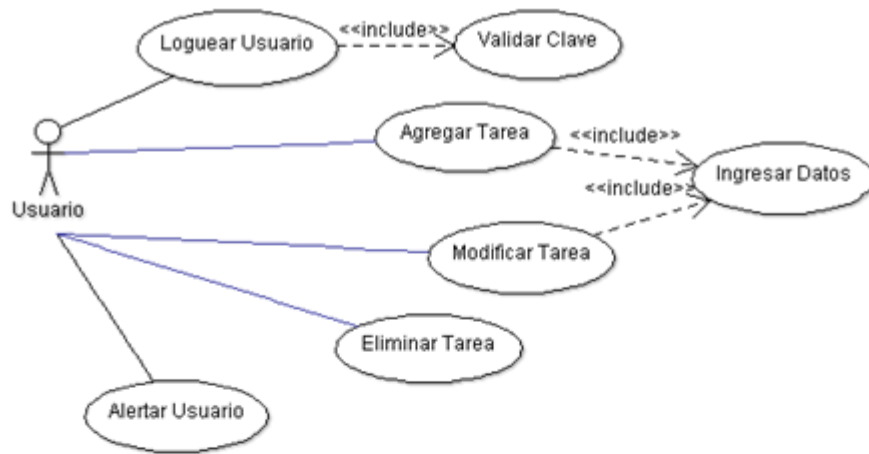
## Caso de Uso: Activar Alarma



Caso de Uso Nº5	Activar Alarma	
Actores	Usuario –Control Domótico	
Tipo		
Descripción	El Usuario activa la alarma de la vivienda y recibe una alerta en caso de que esta detecte un intruso	
Propósito	Alertar al usuario en caso de irrupción en la vivienda	
Precondición	Al activar la alarma la vivienda debe estar vacía	
Resultado Éxito	Alertar al usuario vía web sobre intruso en la vivienda, activar los actuadores pertinentes	
Resultado Fracaso	Error al cambiar de estado los actuadores y alertar al usuario	
Curso Normal de Eventos		
Acciones de los Actores	Respuestas del sistema	Alternativa
1 El usuario ingresa en el sistema	2 El sistema valida el nombre de usuario y contraseña.	2.1 Falla nombre de usuario o clave
3 El usuario se dirige al menú Alarma y coloca el código para activar/desactivar la alarma	4 El sistema valida la clave para activar/desactivar la alarma. 5 Se activan los sensores de movimiento y sonido de la vivienda.	4.1 Falla la clave para activar la alarma 5.1 Error activar los sensores 5.2 Error al comunicarse con el sistema.
7 El usuario recibe una alerta de hay un intruso en el hogar	6 Si el sistema detecta movimiento o sonidos en el interior de la vivienda activa el sonido de alerta, enciende las luces, cierra las puertas y alerta al usuario	6.1 Falla el envío del alerta al usuario 6.2 Falla al activar los actuadores de sonido

Figura 59 Caso de Uso Activar Alarma

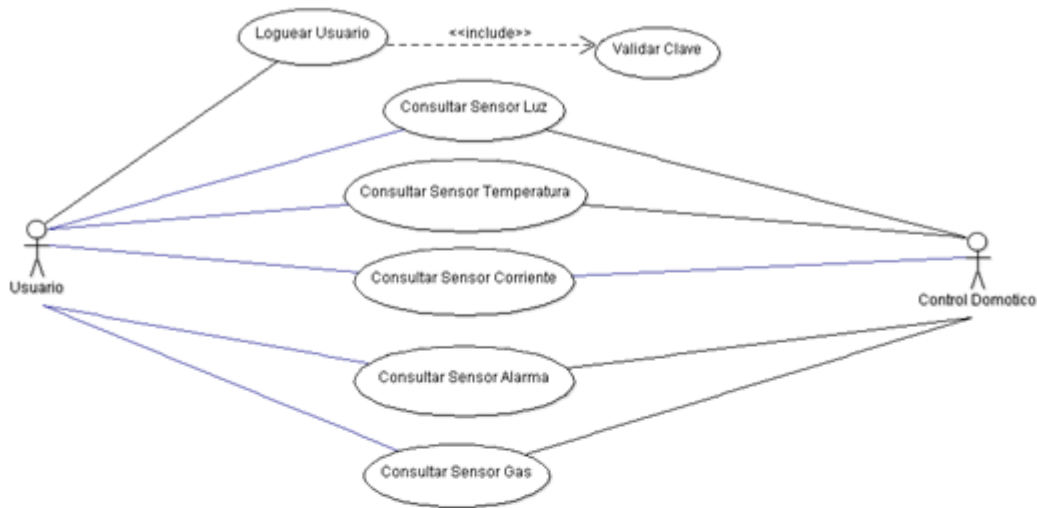
## Caso de Uso: ABM Tareas



Caso de Uso N°6	ABM Tareas	
Actores	Usuario	
Tipo		
Descripción	Agregar – Borrar- Modificar Tareas a realizar en la vivienda	
Propósito	Informar al usuario las tareas pendientes	
Precondición	Ingresar en el sistema web	
Resultado Éxito	Muestra las tareas próximas a realizar	
Resultado Fracaso	Error al acceder a la Base de Datos	
Curso Normal de Eventos		
Acciones de los Actores	Respuestas del sistema	Alternativa
1 El usuario ingresa en el sistema	2 El sistema valida el nombre de usuario y contraseña	2.1 Falla nombre de usuario o clave
3 El usuario ingresa al menú de tareas	4 El sistema muestra las pendientes de realizar.	4.1 Error al comunicarse con la base de datos
5 El usuario puede agregar o modificar una tarea existente	6 El sistema valida el ingreso o modificación de la nueva tarea a realizar	6.1 Error al comunicarse con la base de datos 6.2 Error en los datos ingresados

Figura 60 Caso de Uso ABM Tareas

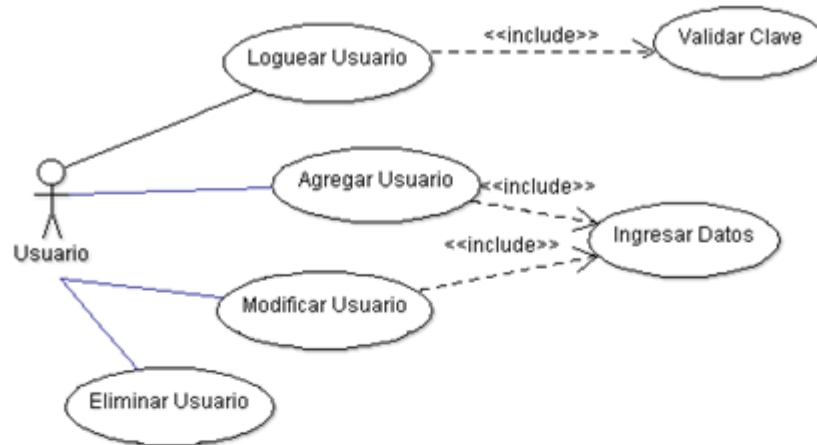
### Caso de Uso: Consultar Sensores



Caso de Uso N°7	Consultar Sensores	
Actores	Usuario – Control Domótico	
Tipo		
Descripción	Mostrar información actual de captura de sensores	
Propósito	Informar al usuario sobre los estados de temperatura, consumos de electrodomésticos, etc.	
Precondición	Ingresar en el sistema web	
Resultado Exitoso	Muestra información de todos los sensores de la vivienda	
Resultado Fracaso	Error al comunicarse con el control domótico	
Curso Normal de Eventos		
Acciones de los Actores	Respuestas del sistema	Alternativa
1 El usuario ingresa en el sistema	2 El sistema valida el nombre de usuario y contraseña	2.1 Falla nombre de usuario o clave
3 El usuario ingresa al menú de "consultar sensores"	4 El sistema muestra en tiempo real los estados actuales de temperatura y humedad en los ambientes, iluminación, estado de la alarma, posibles pérdidas de gas y consumo de los aparatos eléctricos.	4.1 Error al comunicarse con los sensores. 4.2 Falla en la lectura de sensores.

Figura 61 Caso de Uso Consultar Sensores

### Caso de Uso: ABM Usuarios



Caso de Uso Nº8	ABM Usuarios	
Actores	Usuario	
Tipo		
Descripción	Agregar – Borrar- Modificar Usuarios para el acceso y control del sistema domótico	
Propósito	Agregar nuevos usuarios	
Precondición	Ingresar en el sistema web	
Resultado Exitoso	Agregar/Modifica/Elimina nuevos Usuarios	
Resultado Fracaso	Error al acceder a la Base de Datos	
Curso Normal de Eventos		
Acciones de los Actores	Respuestas del sistema	Alternativa
1 El usuario ingresa en el sistema	2 El sistema valida el nombre de usuario y contraseña	2.1 Falla nombre de usuario o clave
3 El usuario ingresa al menú de Usuarios	4 El sistema muestra los usuarios registrados.	4.1 Error al comunicarse con la base de datos
5 El usuario puede agregar, modificar o eliminar un usuario existente	6 El sistema valida el ingreso o modificación del nuevo usuario	6.1 Error al comunicarse con la base de datos 6.2 Error en los datos ingresados

Figura 62 Caso de Uso ABM Usuarios

### 3.2.3.10 Diagramas de Actividades

#### Diagrama de Actividad: Consultar Estado Sensores

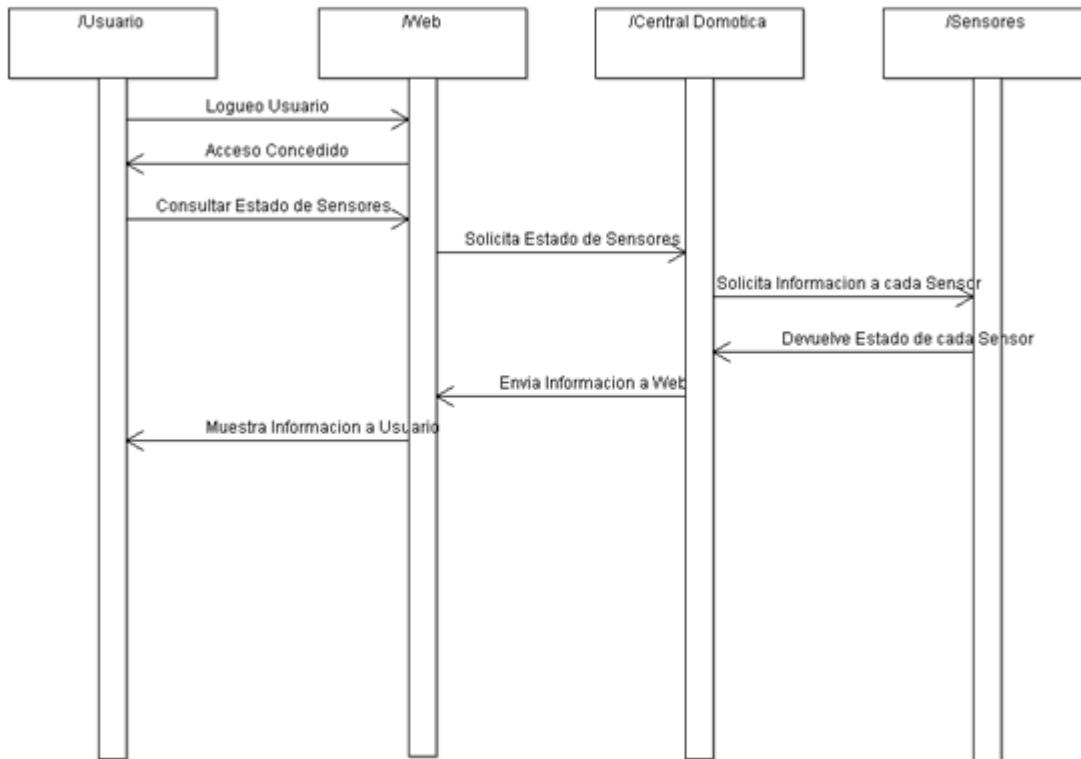


Figura 63 Diagrama de Actividad Consultar Estado Sensores



**Diagrama de Actividad: Activar Alarma**

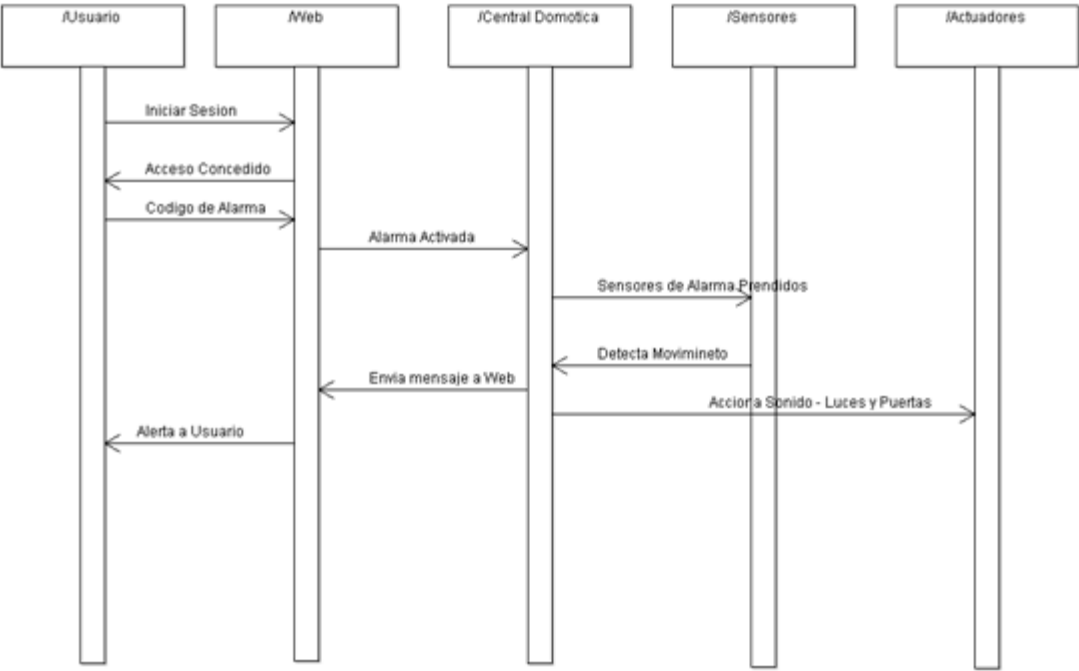


Figura 64 Diagrama de Actividad Activar Alarma

### Diagrama de Actividades: Cambiar Estado de Actuadores Web

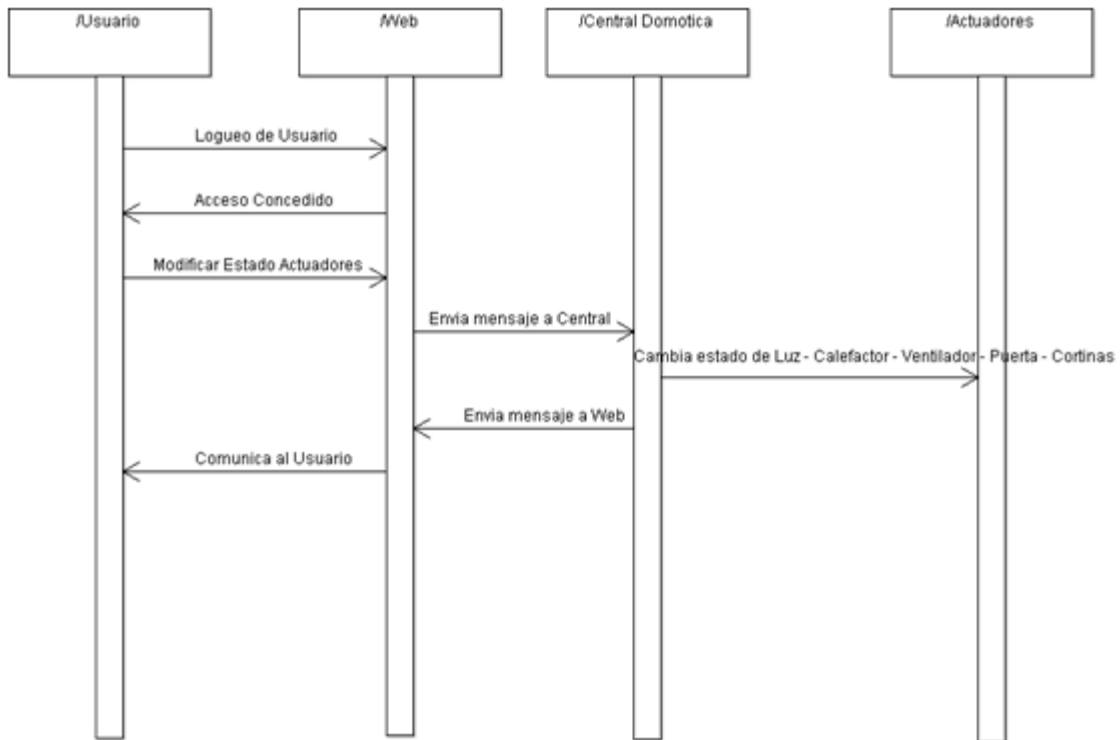


Figura 65 Diagrama de Actividad Cambiar Estado de Actuadores Web

### Diagrama de Actividad: Cambiar Estado Actuadores Manual

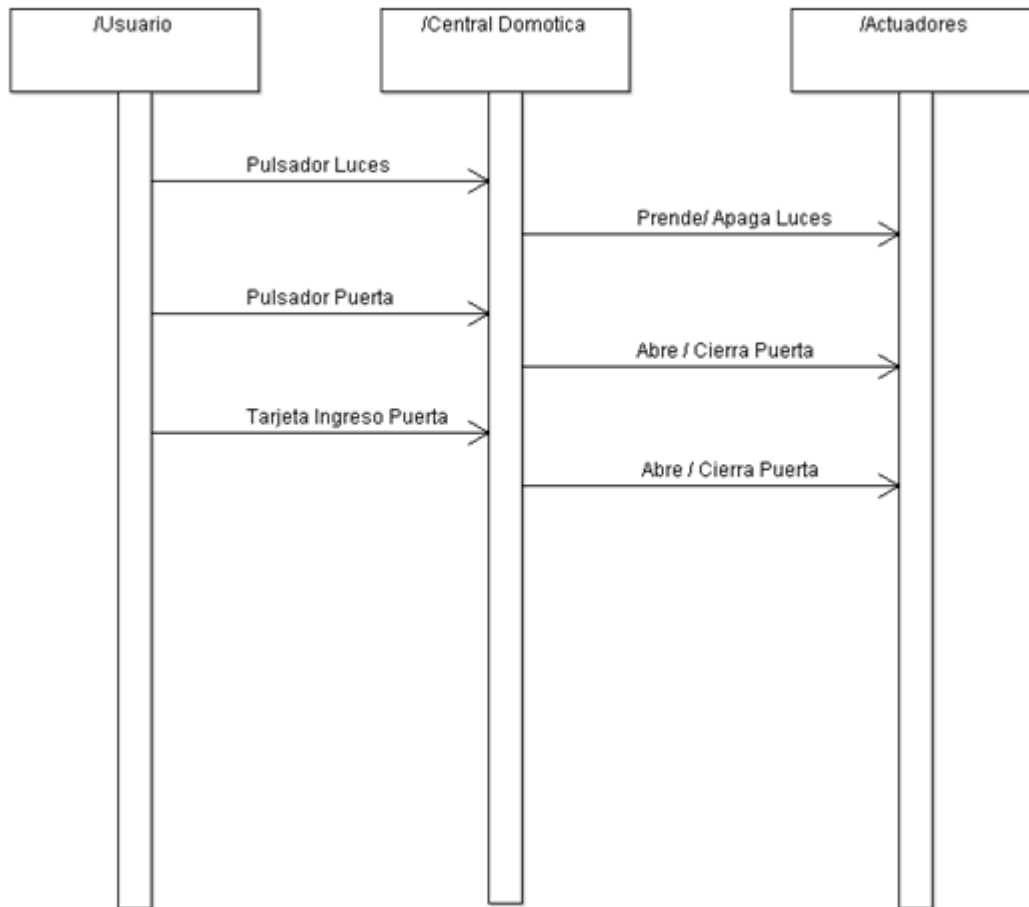


Figura 66 Diagrama de Actividad Cambiar Estado de Actuadores Manual

### Diagrama de Actividad: Cambiar Estado Actuadores Automático

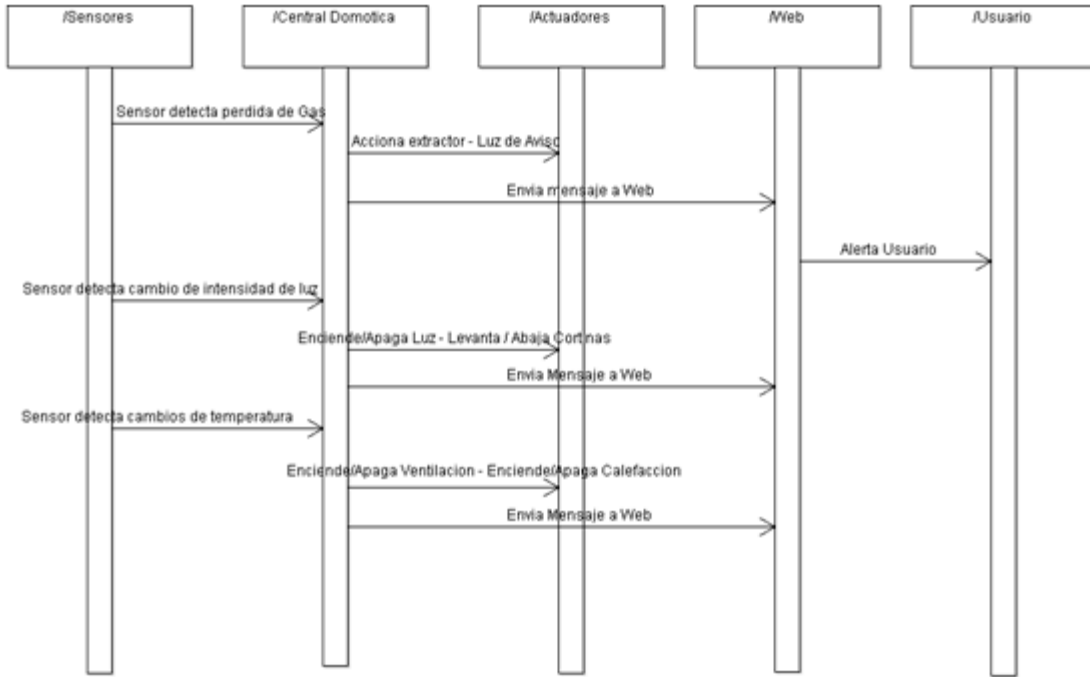
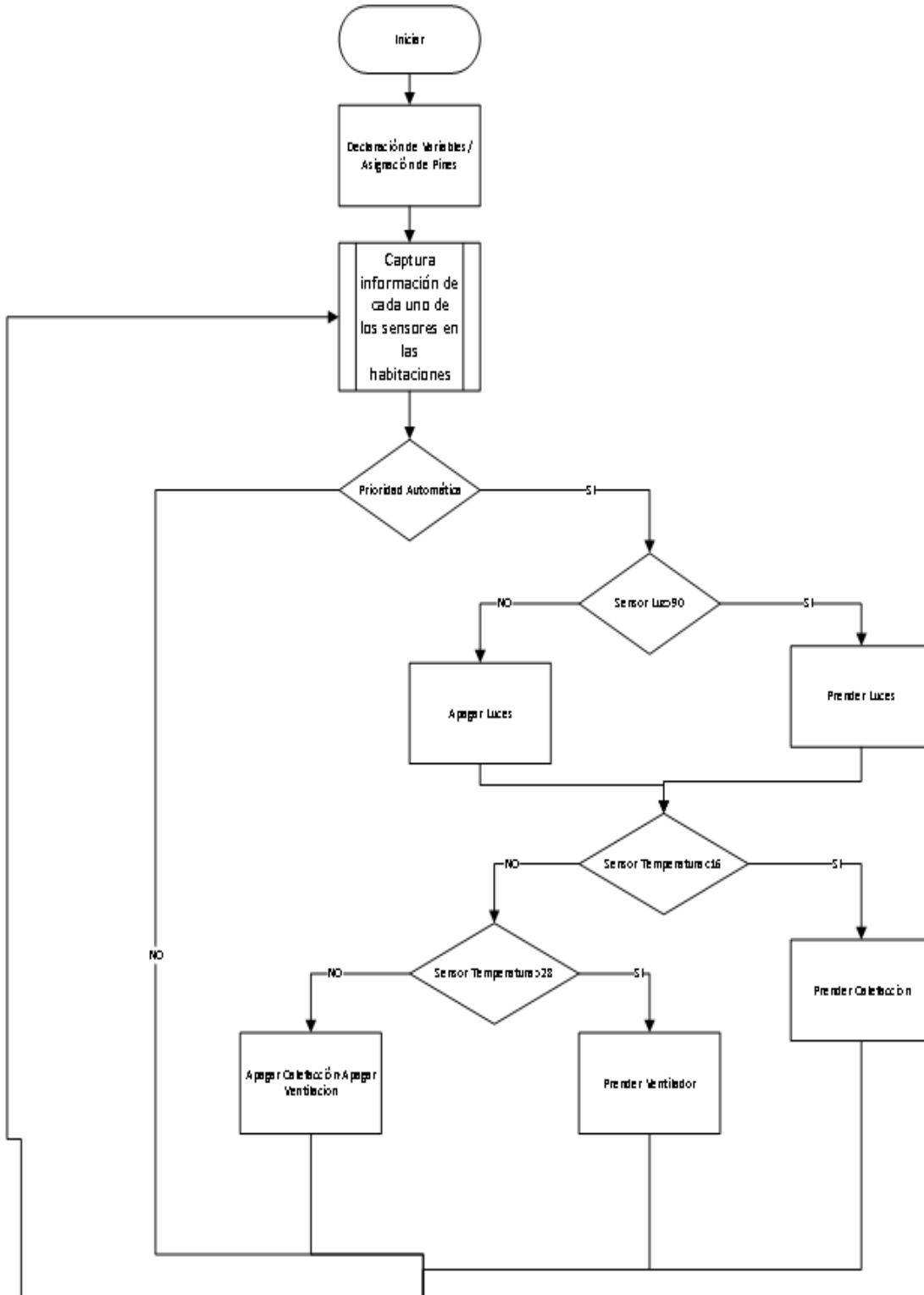
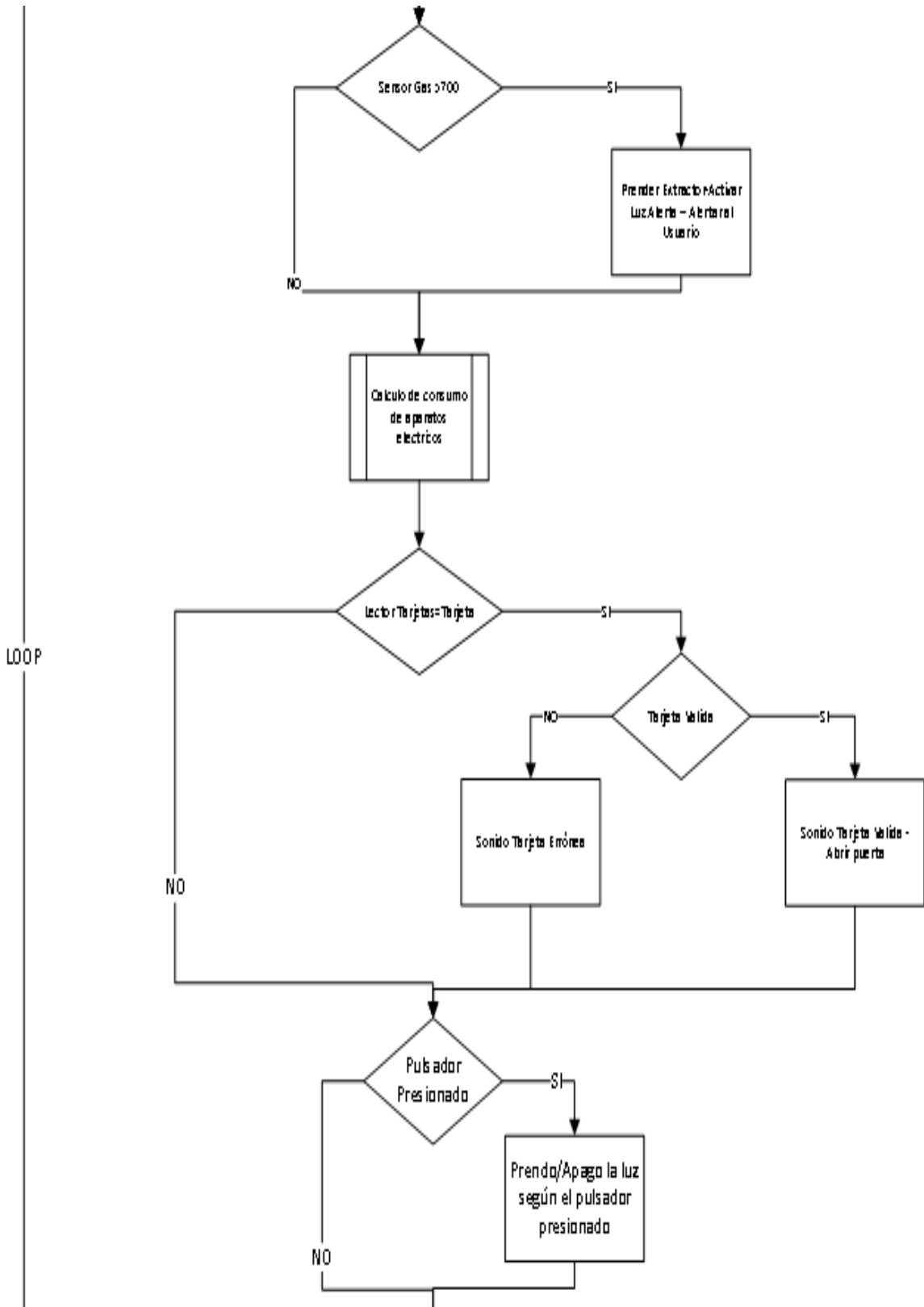


Figura 67 Diagrama de Actividad Cambiar Estado de Actuadores Automáticos

### 3.2.3.11 Lógica del controlador





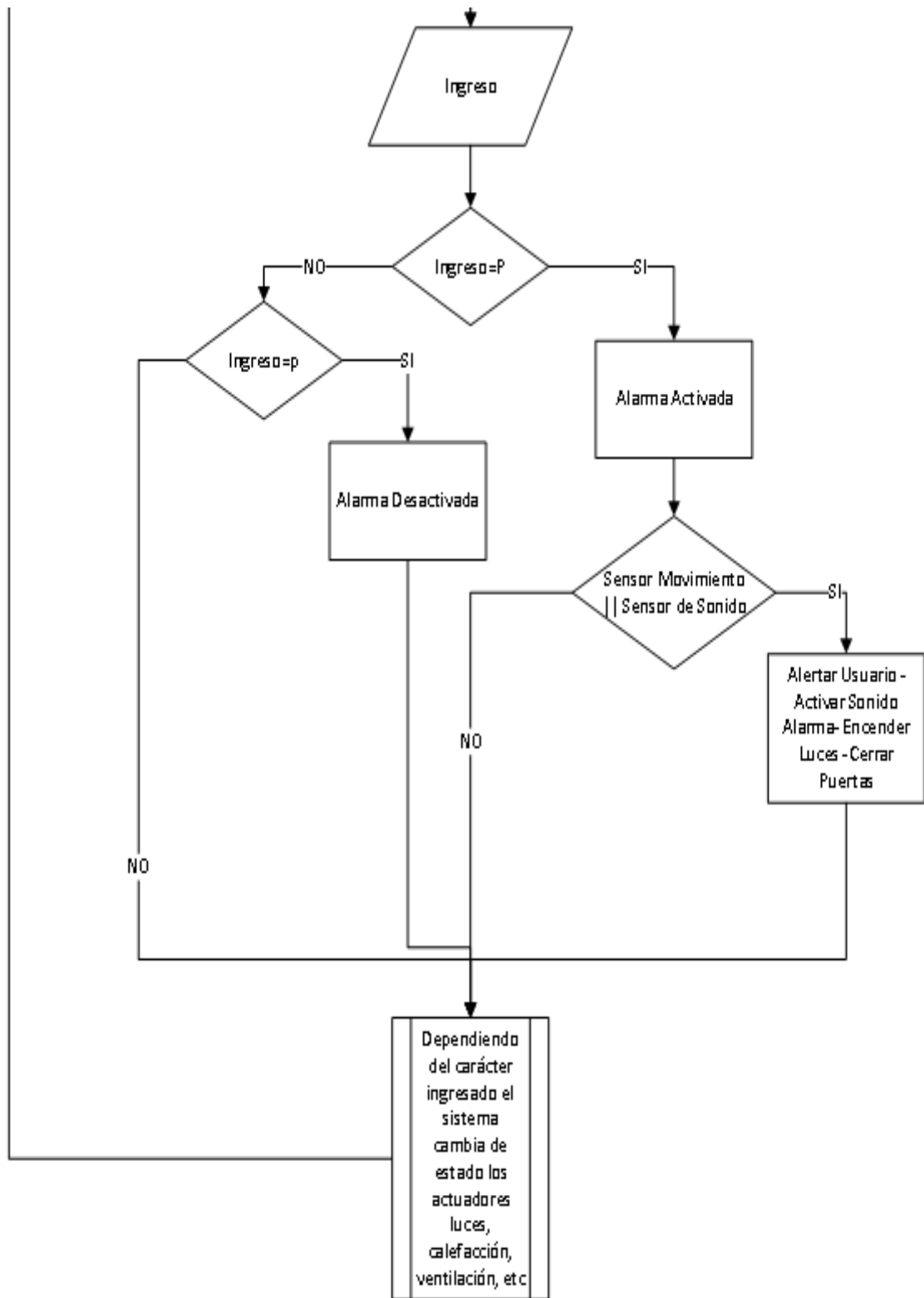


Figura 68 Diagrama de Actividad Cambiar Estado de Actuadores Web

**CAPITULO IV**  
**ETAPA DE SOLUCIÒN**



## 4 Capítulo IV–Etapa de Solución

### 4.1 Etapa de Solución

#### 4.1.1 Maqueta finalizada con el sistema domótico funcionando.



Figura 69 Imagen Maqueta Completa

### 4.1.1.1 Garage



Figura 70 Imagen Maqueta Garaje



Figura 71 Imagen Maqueta Conexión

#### 4.1.1.2 Pasillo



Figura 72 Imagen Maqueta Pasillo 1



Figura 73 Imagen Maqueta Pasillo 2



Figura 74 Imagen Maqueta Pasillo 3

### 4.1.1.3 Sal3n



Figura 75 Imagen Maqueta Sal3n 1



Figura 76 Imagen Maqueta Sal3n 2

#### 4.1.1.4 Dormitorio



Figura 77 Imagen Maqueta Dormitorio 1

Figura 78 Imagen Maqueta Dormitorio 2

#### 4.1.1.5 Frente Maqueta



Figura 79 Imagen Maqueta Frente 1



Figura 80 Imagen Maqueta Frente 2





## 4.1.2 Interfaz para la manipulación del sistema.

### 4.1.2.1 Login.jsp



Figura 81 Captura Login.jsp

### 4.1.2.2 Index.jsp




Figura 82 Captura Index.jsp

### 4.1.2.3 Acciones.jsp

DOMOTICA Mi Casa Mi Configuración Estadísticas Logs Contacto Usuarios SALIR / Usuario actual: GTR

Descripcion	Estado
Calefaccion cocina	<input type="checkbox"/>
Calefaccion habitacion	<input type="checkbox"/>
Calefaccion salon	<input checked="" type="checkbox"/>
Ventilador cocina	<input type="checkbox"/>
Ventilador salon	<input type="checkbox"/>
Ventilador habitacion	<input type="checkbox"/>
Cortina salon	<input checked="" type="checkbox"/>
Puerta pasillo	<input type="checkbox"/>
Alarma	<input checked="" type="checkbox"/>
Modo manual/automatico	<input type="checkbox"/>



Web diseñada para la tesis de Domotica por Cascone Gabriel y Rizzo Gabriel

Figura 83 Captura Acciones.jsp

### 4.1.2.4 Miconfiguracion.jsp

DOMOTICA Mi Casa Mi Configuración Estadísticas Logs Contacto Usuarios SALIR / Usuario actual: GTR

Mi configuración	Valores
Arduino ID:	2
Modelo de Arduino:	Arduino Mega 101
IP Arduino:	GTR.com



Web diseñada para la tesis de Domotica por Cascone Gabriel y Rizzo Gabriel

Figura 84 Captura Miconfiguracion.jsp

## 4.1.2.5 Estadística.jsp

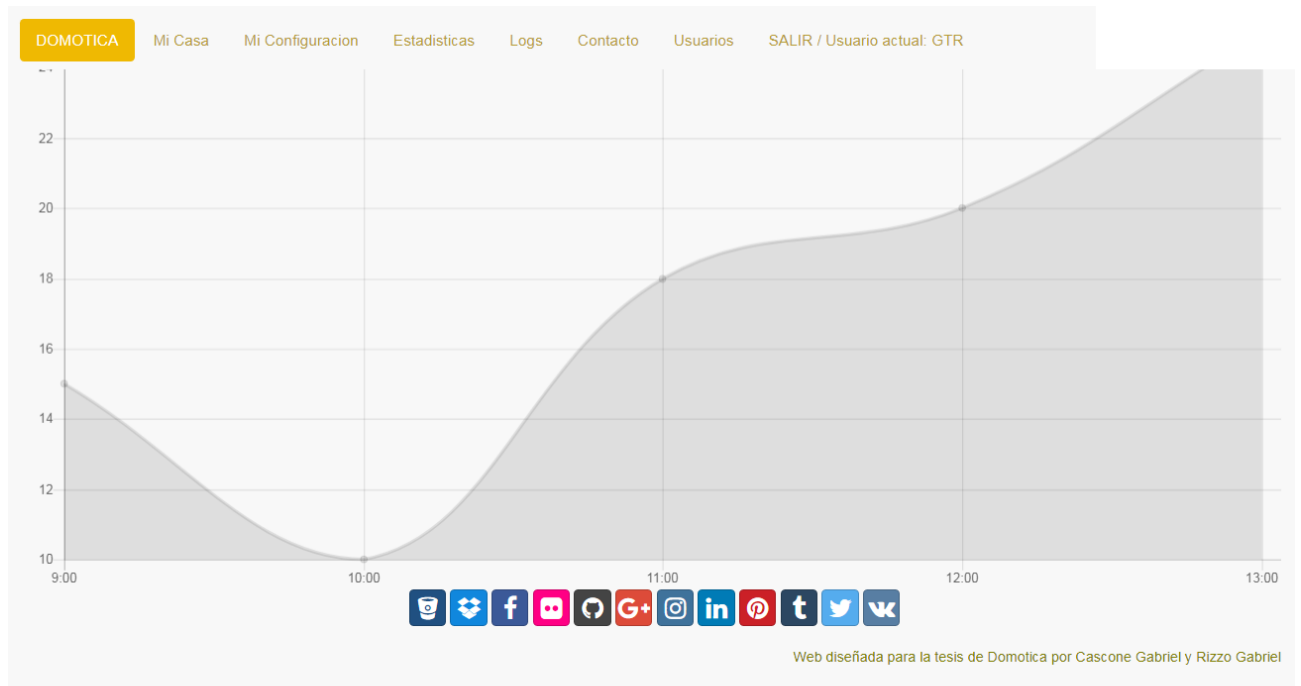


Figura 85 Captura Estadística.jsp

## 4.1.2.6 Logs.jsp

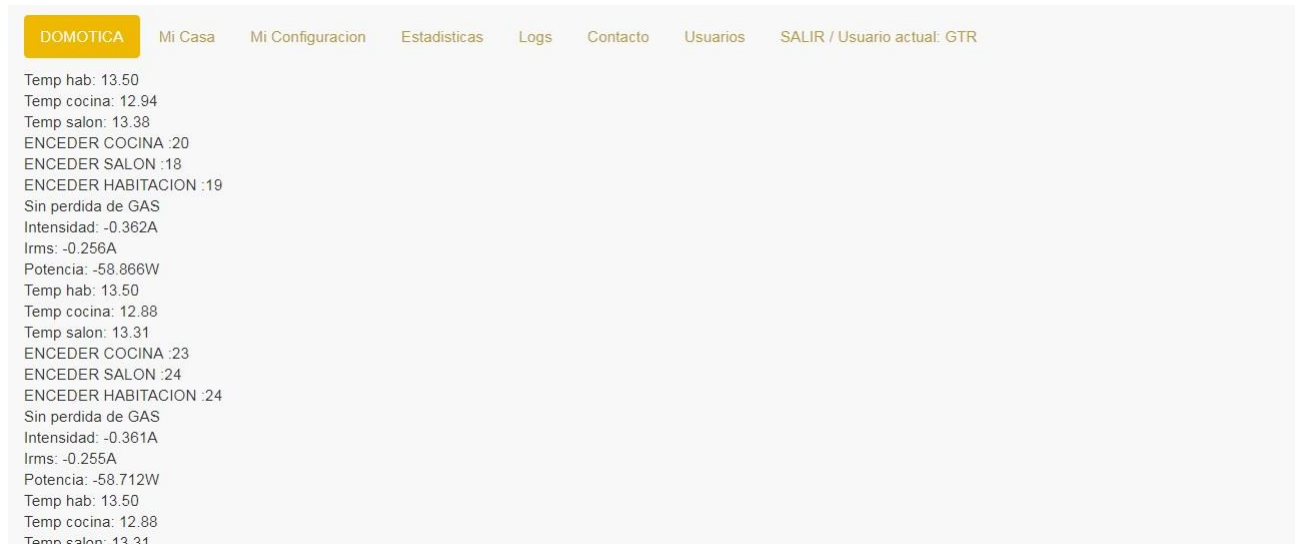


Figura 86 Captura Logs.jsp

## 4.1.2.7 Contacto.jsp

DOMOTICA Mi Casa Mi Configuración Estadísticas Logs Contacto Usuarios SALIR / Usuario actual: GTR

### Contacto

Nombre

Apellido

E-Mail

Teléfono

Dirección

Ciudad

Código postal

Comentario

Web diseñada para la tesis de Domotica por Cascone Gabriel y Rizzo Gabriel

Figura 87 Captura Contacto.jsp

## 4.1.2.8 Usuario.jsp

DOMOTICA Mi Casa Mi Configuración Estadísticas Logs Contacto Usuarios SALIR / Usuario actual: GTR

Mi configuración	Valores
Usuario ID:	2
Nombre de usuario:	GTR
Ultima modificación:	2018-05-01

Web diseñada para la tesis de Domotica por Cascone Gabriel y Rizzo Gabriel

Figura 88 Captura Usuario.jsp

## 4.1.3 Código fuente del sistema domótico ( microcontrolador)

### 4.1.3.1 Código Fuente Microcontrolador

#### 4.1.3.1.1 Modulo Principal

```
#include <Servo.h>// para porton y puerta

#include <OneWire.h>

#include <DallasTemperature.h>// temperatura de ambientes

#include "SPI.h"//tarjeta de acceso

#include "MFRC522.h"//tarjeta de acceso

#include <pitches.h>//Sonidos

//#include <Wire.h>

//#include <DHT11.h>

//variable toma caracteres web

char pag = 'q';

int luz = 0; // para la funcion del pulsador

int tiempoSensorAutomatico = 0.0;

char prioridadSensor = 'a'; //prioridad automatica

char estadoCortinas = 'b';

boolean alarmaActivada = false;
```

```
//// Variables para el sensor de corriente

float SENSIBILITY = 0.066; // Modelo 30A

int SAMPLESNUMBER = 10000;

/// Pines para Cocina ///

const int pulsadorLuzCocina = 37;

const int luzCocina = 38;

const int sensorGas = A4;

const int calefaccionCocina = 39;

OneWire oneWireCocina(40);

DallasTemperature sensorTemperaturaCocina(&oneWireCocina);

const int extractor = 41;

const int luzGas = 42;

const int ventiladorCocina = 43;

const int pinLDRCocina = A5;

//Variables

int estadoLuzCocina = LOW;

int estadoLuzGas = LOW; // extractor

String estadoGas = "";

int estadoCalefaccionCocina = LOW;

int estadoVentiladorCocina = LOW;

int temperaturaCocina = 0;
```

```
int valorLDRCocina = 0;

//int MQ_DELAY = 2000;

int iluminacionCocina;

boolean anterior_estadoLuzCocina = LOW;

///Pines para Salon ///

const int pulsadorLuzSalon = 44;

const int sensorMovimientoSalon = 45;

OneWire oneWireSalon(46);

DallasTemperature sensorTemperaturaSalon(&oneWireSalon);

const int ventiladorSalon = 47;

const int pinLDRSalon = A0;

const int sensorSonidoSalon = A1;

const int luzSalon = 48;

const int calefaccionSalon = 49;

//Cortina

// Definimos los pines donde tenemos conectadas las bobinas

#define IN1 4

#define IN2 5

#define IN3 6

#define IN4 7

// Secuencia de pasos (par máximo)
```

```
int paso [4][4] =
{
    {1, 0, 0, 0},
    {0, 1, 0, 0},
    {0, 0, 1, 0},
    {0, 0, 0, 1}
};

//Variables

int valorLDRSalon = 0;

int iluminacionSalon;

int estadoCalefaccionSalon = LOW;

int estadoVentiladorSalon = LOW;

int estadoLuzSalon = LOW;

int estadoCortinaSalon;

int temperaturaSalon;

boolean anterior_estadoLuzSalon = LOW;

boolean anterior_estadoPuertaFrente = false;

/// Habitacion ///

const int calefaccionHabitacion = 32;

const int luzHabitacion = 33;

const int ventiladorHabitacion = 34;
```



```
const int pulsadorLuzHabitacion = 35;

OneWire oneWireHabitacion(36);

DallasTemperature sensorTemperaturaHabitacion(&oneWireHabitacion);

const int pinLDRHabitacion = A3;

//Variables

int valorLDRHabitacion = 0;

int estadoLuzHabitacion = LOW;

int estadoCalefaccionHabitacion = LOW;

int estadoVentiladorHabitacion = LOW;

int iluminacionHabitacion;

int temperaturaHabitacion;

boolean anterior_estadoLuzHabitacion = LOW;

///Pasillo ///

const int sensorMovimientoPasillo = 22;

const int sensorTemperaturaHumedadPasillo = A2;

const int parlantePasillo = 23;

const int pulsadorLuzPasilloInterior = 24;

const int pulsadorPuertaPasillo = 26;

const int luzPasillo = 27;

const int luzVerdePasillo = 30;

const int luzRojaPasillo = 31;
```

```
#define SS_PIN 53

#define RST_PIN 9

Servo myServoPuerta; // crea el objeto servo

MFRC522 rfid(SS_PIN, RST_PIN); // seteos los pines sda y rst

MFRC522::MIFARE_Key key;

bool intruso=false;

//Variables

//capturar los numeros de tarjeta de la base de datos

String miTarjeta = "45:01:38:BB";

// Sonido de acceso y falla

int access_melody[] = {NOTE_G4, 0, NOTE_A4, 0, NOTE_B4, 0, NOTE_A4, 0, NOTE_B4,
0, NOTE_C5, 0};

int access_noteDurations[] = {8, 8, 8, 8, 8, 4, 8, 8, 8, 8, 8, 4};

int fail_melody[] = {NOTE_G2, 0, NOTE_F2, 0, NOTE_D2, 0};

int fail_noteDurations[] = {8, 8, 8, 8, 8, 4};

//sonido alerta intruso

int intruso_melody[] = {NOTE_G2, 0, NOTE_F2, 0, NOTE_D2, 0};

int intruso_noteDurations[] = {8, 8, 8, 8, 8, 4};
```

```
int estadoLuzPasillo = LOW;

boolean anterior_estadoLuzFrente = LOW;

boolean anterior_estadoPuertaPasillo = false; //cerrada

boolean anterior_estadoLuzPasillo = LOW;
```

```
///Garage///
```

```
Servo myServoPorton; // crea el objeto servo d10
```

```
const int luzGarage = 11;
```

```
const int pulsadorLuzGarage = 12;
```

```
//Variables
```

```
int estadoLuzGarage = 0;
```

```
int estadoPorton = 0;
```

```
boolean anterior_estadoLuzGarage = LOW;
```

```
///Exterior///
```

```
const int luzExterior = 28;
```

```
const int pulsadorLuzExterior = 25;
```

```
int estadoLuzExterior = LOW;
```

```
//int pinLDRExterior=A2;
```

```
void setup() {  
  
  ///Cocina///  
  
  pinMode(luzCocina, OUTPUT);  
  
  pinMode(pulsadorLuzCocina, INPUT);  
  
  pinMode(calefaccionCocina, OUTPUT);  
  
  pinMode(extractor, OUTPUT);  
  
  pinMode(ventiladorCocina, OUTPUT);  
  
  pinMode(luzGas, OUTPUT);  
  
  sensorTemperaturaCocina.begin();// Inicia el sensor de temperatura  
  
  digitalWrite(luzCocina, LOW);  
  
  
  ///Salon///  
  
  pinMode(luzSalon, OUTPUT);  
  
  pinMode(pulsadorLuzSalon, INPUT);  
  
  pinMode(ventiladorSalon, OUTPUT);  
  
  pinMode(calefaccionSalon, OUTPUT);  
  
  sensorTemperaturaSalon.begin();  
  
  pinMode(sensorMovimientoSalon, INPUT);  
  
  
  digitalWrite(luzSalon, LOW);  
  
  
  // Todos los pines de la cortina en modo salida  
  
  pinMode(IN1, OUTPUT);
```

```
pinMode(IN2, OUTPUT);
```

```
pinMode(IN3, OUTPUT);
```

```
pinMode(IN4, OUTPUT);
```

```
////Habitacion////
```

```
pinMode(luzHabitacion, OUTPUT);
```

```
pinMode(pulsadorLuzHabitacion, INPUT);
```

```
pinMode(calefaccionHabitacion, OUTPUT);
```

```
pinMode(ventiladorHabitacion, OUTPUT);
```

```
sensorTemperaturaHabitacion.begin();
```

```
digitalWrite(luzHabitacion, LOW);
```

```
////Pasillo////
```

```
pinMode(luzPasillo, OUTPUT);
```

```
pinMode(pulsadorLuzPasilloInterior, INPUT);
```

```
pinMode(pulsadorPuertaPasillo, INPUT);
```

```
pinMode(sensorMovimientoPasillo, INPUT);
```

```
pinMode(luzVerdePasillo, OUTPUT);
```

```
pinMode(luzRojaPasillo, OUTPUT);
```

```
myServoPuerta.attach(29); // vincula el servo al pin digital 8
```

```
myServoPuerta.write(105); //inicio mi motor en pos 0
```

```
pinMode(parlantePasillo, OUTPUT);
```

```
digitalWrite (luzPasillo, LOW);
```

```
digitalWrite (luzExterior, LOW);

SPI.begin();

rfid.PCD_Init();

////Garage////

pinMode(luzGarage, OUTPUT);

pinMode(pulsadorLuzGarage, OUTPUT);

myServoPorton.attach(10); // vincula el servo al pin digital 8

myServoPorton.write(0); //inicio mi motor en pos 0

////Exterior////

pinMode(luzExterior, OUTPUT);

pinMode(pulsadorLuzExterior, INPUT);

//////// Monitor Serie//

Serial.begin(9600);

}

void loop() {

    if (Serial.available() > 0 )//seteos desde web

    {

        pag = Serial.read();
```

```
//prioridad

if (pag == 'q' || pag == 'Q') {

prioridadSensor = prioridad(pag);

    // identifico si desde la web se seteo la prioridad manual o la automatica para los
    sensores luz - temperatura

//Luces

    if (pag == 'A' || pag == 'a' || pag == 'B' || pag == 'b' || pag == 'C' || pag == 'c' || pag == 'D'
|| pag == 'd' || pag == 'E' || pag == 'e' || pag == 'F' || pag == 'f') {

        luces(pag);

    }

//Calefacciones

if (pag == 'G' || pag == 'g' || pag == 'H' || pag == 'h' || pag == 'I' || pag == 'i') {

    calefaccion(pag);

}

//Ventiladores

if (pag == 'J' || pag == 'j' || pag == 'K' || pag == 'k' || pag == 'L' || pag == 'l') {

    ventilador(pag);

}

//CortinaSalon

if (pag == 'M' || pag == 'm') {
```

```
CortinaSalon(pag);
```

```
}
```

```
//Puertas
```

```
if (pag == 'N' || pag == 'n' || pag == 'O' || pag == 'o') {
```

```
    puerta(pag);
```

```
    porton(pag);
```

```
}
```

```
    //si al ingresar la clave en la web esta ok manda la variable para activar o desactivar la  
    alarma
```

```
if (pag == 'P' || pag == 'p') {
```

```
    alarmaActivada = alarma(pag);
```

```
}
```

```
}
```

```
//Sensores AUTOMATICOS
```

```
if (prioridadSensor == 'a') {
```

```
    sensoresTemperatura();
```

```
    sensoresLuz();
```

```
}
```



```

sensoresGas();

sensorCorriente();

sensoresTarjeta();

// ALARMA

if (alarmaActivada == true) { //activa la alarma

    //puerta cerrada

    //   if (anterior_estadoPuertaPasillo==true){puerta('n') ;}

    //actuar sensor de sonido

    boolean sonido = activarSensorSonido();

    //activar sensor movimiento

    boolean sensorPasillo = activarSensorMovimiento(sensorMovimientoPasillo);

    boolean sensorSalon = activarSensorMovimiento(sensorMovimientoSalon);

    alarmaActivada = true;

    if (sonido == true || sensorPasillo == true || sensorSalon == true) {

        luces('A'); luces('B'); luces('C'); luces('D'); luces('E'); luces('F');

        puerta('n');

        intruso=true;

    }

    if(intruso==true){// Suena mientras no se ingrese otro caracter o se desactive la alarma

        sonidoAlarma();}

}

```

```
else if (pag == 'p') { //desactiva la alarma
```

```
    Serial.println("**Alarma OFF");
```

```
    alarmaActivada = false;
```

```
    luces('a'); luces('b'); luces('c'); luces('d'); luces('e'); luces('f');
```

```
    sonidoAlarmaDesactivada();
```

```
    intruso=false;
```

```
    pag='z';//sesactivo el pag de la alarma
```

```
}
```

```
// seteos manuales
```

```
// Pulsadores de Luz - leo el estado de la luz -- // Pulsadores de Puertas
```

```
anterior_estadoLuzHabitacion = digitalRead(luzHabitacion);
```

```
anterior_estadoLuzSalon = digitalRead(luzSalon);
```

```
anterior_estadoLuzCocina = digitalRead(luzCocina);
```

```
anterior_estadoLuzFrente = digitalRead(luzExterior);
```

```
anterior_estadoLuzGarage = digitalRead(luzGarage);
```

```
anterior_estadoLuzPasillo = digitalRead(luzPasillo);
```

```
int flanco_Habitacion = detectaFlanco(pulsadorLuzHabitacion, luzHabitacion,  
anterior_estadoLuzHabitacion);
```

```
int flanco_Salon = detectaFlanco(pulsadorLuzSalon, luzSalon, anterior_estadoLuzSalon);
```

```

int flanco_Cocina = detectaFlanco(pulsadorLuzCocina, luzCocina,
anterior_estadoLuzCocina);

int flanco_Frente = detectaFlanco(pulsadorLuzExterior, luzExterior,
anterior_estadoLuzFrente);

int flanco_Pasillo = detectaFlanco(pulsadorLuzPasilloInterior, luzPasillo,
anterior_estadoLuzPasillo);

int flanco_Garage = detectaFlanco(pulsadorLuzGarage, luzGarage,
anterior_estadoLuzGarage);

int flanco_Puerta = detectaFlanco(pulsadorPuertaPasillo, luzGarage,
anterior_estadoPuertaFrente);

//int Pulsador_Puerta =
detectaEstadoPuerta(pulsadorPuertaPasillo, estadoPuerta, anterior_estadoPuertaFrente); //
acciones puerta -> abre c tarjeta y cierra pulsador - abre con pulsador cierra pulsador -
abre pulsador cierra tarjeta
}

```

```

char prioridad(char pag) {

if (pag == 'Q') {

prioridadSensor = 'a';

Serial.println("**Prioridad automatica");

Serial.print("**$");

Serial.println(pag);

return prioridadSensor;

} //automatico

```

```

    else if (pag == 'q') {
prioridadSensor = 'm';
Serial.println("*Prioridad manual ARR");
Serial.print("*$");
    Serial.println(pag);
    return prioridadSensor;
}
} //manual

void printMeasure(String prefix, float value, String postfix)
{
    Serial.print(prefix);
    Serial.print(value, 3);
Serial.println(postfix);
}

```

#### **4.1.3.1.2 Modulo Alarma**

```

boolean alarma(char pag) {

if (pag == 'P') { //activa la alarma
    //Serial.println("Alarma activada");
    // sonido alarma activada
    sonidoAlarmaActivada();
}
}

```

```
        alarmaActivada = true;

Serial.print("*$");

        Serial.println(pag);

return alarmaActivada ;

    }

else if (pag == 'p') { //desactiva la alarma

        //Serial.println("Alarma desactivada");

        // sonido alarma activada

        sonidoAlarmaDesactivada();

        alarmaActivada = false;

Serial.print("*$");

        Serial.println(pag);

return alarmaActivada ;

    }

}
```

```
boolean activarSensorSonido() {

    bool state = false;

    int lectura = analogRead(A1) ;

    if (lectura > 500)
```

```
{  
    Serial.println("*Sonido detectado");  
state = ! state;  
}  
return state;  
}  
  
boolean activarSensorMovimiento(int sensorMovimiento) {  
    int pirState = LOW; //sin movimiento  
int valSensor = digitalRead(sensorMovimiento);  
    if (valSensor == HIGH)  
    {  
        if (pirState == LOW)  
        {  
            pirState = HIGH; Serial.println("*movimiento detectado");  
        }  
    }  
    else {  
        if (pirState == HIGH) {  
            pirState = LOW;  
        }  
    }  
}
```

```
    return pirState;
}

void sonidoAlarmaActivada() { //contraseña bien colocada p
    for (int i = 0; i < 6; i++) { // toca música de alarma activada
        int access_noteDuration = 1000 / access_noteDurations[i];

        tone(parlantePasillo, access_melody[i], access_noteDuration);

        int access_pauseBetweenNotes = access_noteDuration * 1.30;

        delay(access_pauseBetweenNotes);

        noTone(parlantePasillo);
    }

    Serial.println("*Alarma Activada");
}
```

```
void sonidoAlarmaDesactivada() { //contraseña bien colocada p desactivar
    for (int i = 0; i < 6; i++) { // toca música de alarma desactivada
        int fail_noteDuration = 1000 / fail_noteDurations[i];

        tone(parlantePasillo, fail_melody[i], fail_noteDuration);

        int fail_pauseBetweenNotes = fail_noteDuration * 1.30;

        delay(fail_pauseBetweenNotes);

        noTone(parlantePasillo);
    }

    Serial.println("*Alarma Desactivada");
}
```

```
}
```

```
void sonidoAlarma() { //Sonido de intruso  
  
//for (int i = 0; i < 6; i++) { // toca música de intruso  
  
//int fail_noteDuration = 1000 / fail_noteDurations[i];  
  
    //tone(parlantePasillo, fail_melody[i], fail_noteDuration);  
  
    tone(parlantePasillo, 400, 3000);  
  
    //int fail_pauseBetweenNotes = fail_noteDuration * 1.30;  
  
    // delay(fail_pauseBetweenNotes);  
  
    //noTone(parlantePasillo);  
  
//}  
  
    Serial.println("*Alerta Intruso");  
  
}
```

#### **4.1.3.1.3 Modulo Luces**

```
void luces(char pag)  
  
{  
  
    if (pag == 'A')//luzcocina prendida  
  
    {  
  
digitalWrite(luzCocina, HIGH);  
  
        estadoLuzCocina = HIGH;  
  
    }  
  
}
```



```
if (pag == 'a')//luzcocina apagada
```

```
{
```

```
digitalWrite(luzCocina, LOW);
```

```
estadoLuzCocina = LOW;
```

```
}
```

```
if (pag == 'B')//luzSalon prendida
```

```
{
```

```
digitalWrite(luzSalon, HIGH);
```

```
estadoLuzSalon = HIGH;
```

```
}
```

```
if (pag == 'b')//luzSalon apagada
```

```
{
```

```
digitalWrite(luzSalon, LOW);
```

```
estadoLuzSalon = LOW;
```

```
}
```

```
if (pag == 'C')//luzHabitacion prendida
```

```
{
```

```
digitalWrite(luzHabitacion, HIGH);
```

```
estadoLuzHabitacion = HIGH;
```

```
}
```

```
if (pag == 'c')//luzHabitacion apagada
```

```
{  
    digitalWrite(luzHabitacion, LOW);  
    estadoLuzHabitacion = LOW;  
}
```

```
if (pag == 'D')//luz garage prendida
```

```
{  
    digitalWrite(luzGarage, HIGH);  
    estadoLuzGarage = HIGH;  
}
```

```
if (pag == 'd')//luz garage apagada
```

```
{  
    digitalWrite(luzGarage, LOW);  
    estadoLuzGarage = LOW;  
}
```

```
if (pag == 'E') //luzPasillo prendida
```

```
{  
    digitalWrite(luzPasillo, HIGH);  
    estadoLuzPasillo = HIGH;  
}
```

```
if (pag == 'e')//luzPasillo apagada
```

```
{
```

```
digitalWrite(luzPasillo, LOW);

estadoLuzPasillo = LOW;

}

if (pag == 'F')//luzJardin prendida
{

digitalWrite(luzExterior, HIGH);

estadoLuzExterior = HIGH;

}

if (pag == 'f')//luzJardin apagada
{

digitalWrite(luzExterior, LOW);

estadoLuzExterior = LOW;

}

Serial.print("*$");

Serial.println(pag);

}

void calefaccion(char pag)
{

if (pag == 'G')//calefaccion cocina prendida
{
```

```
digitalWrite(calefaccionCocina, HIGH);  
estadoCalefaccionCocina = HIGH;  
}  
  
if (pag == 'g')//calefaccion cocina apagada  
{  
    digitalWrite(calefaccionCocina, LOW);  
    estadoCalefaccionCocina = LOW;  
}  
  
if (pag == 'H')//calefaccion habitacion prendida  
{  
    digitalWrite(calefaccionHabitacion, HIGH);  
    estadoCalefaccionHabitacion = HIGH;  
}  
  
if (pag == 'h')//calefaccion habitacion apagada  
{  
    digitalWrite(calefaccionHabitacion, LOW);  
    estadoCalefaccionHabitacion = LOW;  
}  
  
if (pag == 'l')//calefaccion salon prendida  
{  
    digitalWrite(calefaccionSalon, HIGH);
```

```
    estadoCalefaccionSalon = HIGH;
}

if (pag == 'i')//calefaccion salon apagada
{
    digitalWrite(calefaccionSalon, LOW);
    estadoCalefaccionSalon = LOW;
}

Serial.print("**$");
Serial.println(pag);
}
```

```
void ventilador(char pag)
{
    if (pag == 'J')//ventiladores cocina prendida
    {
        digitalWrite(ventiladorCocina, LOW);
        estadoVentiladorCocina = HIGH;
    }

    if (pag == 'j')//ventiladores cocina apagada
    {
        digitalWrite(ventiladorCocina, HIGH);
        estadoVentiladorCocina = LOW;
    }
}
```

```
}
```

```
if (pag == 'L')//ventiladores habitacion prendida
```

```
{
```

```
digitalWrite(ventiladorHabitacion, LOW);
```

```
estadoVentiladorHabitacion = HIGH;
```

```
}
```

```
if (pag == 'I')//ventiladores habitacion apagada
```

```
{
```

```
digitalWrite(ventiladorHabitacion, HIGH);
```

```
estadoVentiladorHabitacion = LOW;
```

```
}
```

```
if (pag == 'K')//ventiladores salon prendida
```

```
{
```

```
digitalWrite(ventiladorSalon, LOW);
```

```
estadoVentiladorSalon = HIGH;
```

```
}
```

```
if (pag == 'k')//ventiladores salon apagada
```

```
{
```

```
digitalWrite(ventiladorSalon, HIGH);
```

```
estadoVentiladorSalon = LOW;
```

```
}
```

```
Serial.print("$");  
  
Serial.println(pag);  
  
}
```

#### **4.1.3.1.4 Modulo Puertas**

```
void CortinaSalon(char pag)  
{  
  
  if (pag == 'M') {  
  
    if (estadoCortinas != 'l') {  
  
      for (int t = 0; t < 1500; t++) { //levantar cortinas  
  
  
        for (int i = 0; i < 4; i++) {  
digitalWrite(IN1, paso[i][0]);  
  
          digitalWrite(IN2, paso[i][1]);  
  
          digitalWrite(IN3, paso[i][2]);  
  
          digitalWrite(IN4, paso[i][3]);  
  
          delay (2);  
  
        }  
  
      }  
  
      estadoCortinas = 'l';  
  
      Serial.println("$Cortina Salon Levantada");  
  
    }  
  
  }  
  
}
```

```
}

if (pag == 'm') {

  if (estadoCortinas != 'b') {

    for (int t = 0; t < 1500; t++) { //bajar cortinas

      for (int i = 0; i < 4; i++) {

digitalWrite(IN4, paso[i][0]);

        digitalWrite(IN3, paso[i][1]);

        digitalWrite(IN2, paso[i][2]);

        digitalWrite(IN1, paso[i][3]);

        delay (2);

      }

    }

    estadoCortinas = 'b';

    Serial.println("**Cortina Salon bajada");

  }

}

Serial.print("**$");

Serial.println(pag);

}
```



```

void puerta(char pag) {

    //varia la posicion de 0 a 180, con esperas de 15ms

    int pos = 0;

    if (pag == 'n') {

    for (pos = 0; pos <= 105; pos += 1)

        {

            myServoPuerta.write(pos);

            delay(15);

        }

        anterior_estadoPuertaFrente = false;

        Serial.println("**Puerta Principal Cerrada");

    Serial.print("**$");

        Serial.println(pag);

    }

    if (pag == 'N') {

        //varia la posicion de 0 a 180, con esperas de 15ms

        for (pos = 105; pos >= 0; pos -= 1)

            {

                myServoPuerta.write(pos);

                delay(15);

            }

    }

```

```
    anterior_estadoPuertaFrente = true;

    Serial.println("*Puerta Principal Abierta");

Serial.print("$");

    Serial.println(pag);

}

}

void porton(char pag) {

    //varia la posicion de 0 a 180, con esperas de 15ms

    int pos = 0;

    if (pag == 'O') {

    for (pos = 0; pos <= 160; pos += 1)

        {

        myServoPorton.write(pos);

            delay(15);

        if (pos == 10) {

            {

                digitalWrite(luzGarage, HIGH);

                estadoLuzGarage = HIGH;

            }

        }

    }

}
```

```

    }

}

Serial.println("*Porton Garage Abierto");

Serial.print("$");

Serial.println(pag);

}

if (pag == 'o') {

    //varia la posicion de 0 a 180, con esperas de 15ms

    for (pos = 160; pos >= 0; pos -= 1)

    {

myServoPorton.write(pos);

        delay(15);

if (pos == 10) {

            {

                digitalWrite(luzGarage, LOW);

estadoLuzGarage = LOW;

            }

        }

    }

Serial.print("$");

Serial.println(pag);

}

```

```
}
```

#### 4.1.3.1.5 Modulo Pulsadores

```
int detectaFlanco(int pulsador, int luz, boolean anterior_estadoLuz) {  
  
    //Devuelve 1 flanco ascendente, -1 flanco descendente y 0 si no hay nada  
  
    static boolean anterior_estadoPulsador = digitalRead(pulsador);  
  
    boolean estadoPulsador = digitalRead(pulsador);  
  
    delay (200);  
  
    if ( estadoPulsador == HIGH) {  
  
        if (anterior_estadoLuz == LOW) {  
  
            anterior_estadoPulsador = estadoPulsador;  
  
            digitalWrite(luz, HIGH);  
  
            if (pulsador == 44) {  
  
                luces ('B');  
  
            }  
  
            if (pulsador == 37) {  
  
                luces ('A');  
  
            }  
  
            if (pulsador == 35) {  
  
                luces ('C');  
  
            }  
  
            if (pulsador == 24) {  
  
                luces ('E');
```

```
}  
  
if (pulsador == 25) {  
    luces ('F');  
}  
  
if (pulsador == 12) {  
    porton('O');  
    luces('D');  
}  
  
if (pulsador == 26) {  
    puerta('N');  
}  
  
return 1;  
}  
  
else {  
  
anterior_estadoPulsador = estadoPulsador;  
  
    if (pulsador == 12) {  
        porton('o');  
        luces('d');  
    }  
  
    if (pulsador == 44) {  
        luces ('b');  
    }  
  
    if (pulsador == 37) {
```

```
    luces ('a');  
}  
  
if (pulsador == 35) {  
    luces ('c');  
}  
  
if (pulsador == 24) {  
    luces ('e');  
}  
  
if (pulsador == 25) {  
    luces ('f');  
}  
  
if (pulsador == 26) {  
    puerta('n');  
}  
}  
  
digitalWrite(luz, LOW);  
  
return -1;  
}  
}
```

#### **4.1.3.1.6 Modulo Sensor Corriente**

```
void sensorCorriente()  
{
```

```

float current = getCorriente(SAMPLESNUMBER);

float currentRMS = 0.707 * current;

float power = 230.0 * currentRMS;

printMeasure("Intensidad: ", current, "A");

printMeasure("Irms: ", currentRMS, "A");

printMeasure("Potencia: ", power, "W");

}

///// metodo para el sensor de corriente////////////////////////////////////

float getCorriente(int samplesNumber)

{

float voltage;

float corrienteSum = 0;

for (int i = 0; i < samplesNumber; i++)

{

voltage = analogRead(A7) * 5.0 / 1023.0;

corrienteSum += (voltage - 2.5) / SENSIBILITY;

}

return (corrienteSum / samplesNumber);

}

```

#### 4.1.3.1.7 Modulo Sensor Gas

```
void sensoresGas() {  
  
    int sensorValue = analogRead(sensorGas);  
  
    //Serial.println(sensorValue, DEC);  
  
    if (sensorValue > 700)  
  
    {  
  
digitalWrite(luzGas, HIGH); digitalWrite(extractor, LOW); //ESTA AL REVES EL RELE NO  
LO PUEDO CAMBIAR X LOS TORNILLOS  
  
        Serial.println("*Perdida de GAS");  
  
        estadoGas = "Perdida";  
  
        //ver valvula gas  
  
    } else {  
  
        if (estadoGas == "Perdida") {  
  
            digitalWrite(extractor, LOW);  
  
            delay (5000);  
  
        }  
  
        digitalWrite(luzGas, LOW); digitalWrite(extractor, HIGH); //ESTA AL REVES EL RELE  
NO LO PUEDO CAMBIAR X LOS TORNILLOS  
  
        Serial.println("*Sin perdida de GAS");  
  
        estadoGas = "Normal";  
  
    }  
  
}
```



#### 4.1.3.1.8 Modulo Sensor Luz

```
void sensoresLuz() {  
  
    valorLDRHabitacion = analogRead(pinLDRHabitacion);  
  
    valorLDRCocina = analogRead(pinLDRCocina);  
  
    valorLDRSalon = analogRead(pinLDRSalon);  
  
    //ilum = ((long)(1024-V)*A*10)/((long)B*Rc*V); //usar si LDR entre GND y A0  
  
    iluminacionCocina = valorLDRCocina; //(((long)valorLDRCocina*A*10)/((long)B*Rc*(1024-  
valorLDRCocina)); //usar si LDR entre A0 y Vcc (como en el esquema anterior)  
  
    iluminacionSalon = valorLDRSalon; //(((long)valorLDRSalon*A*10)/((long)B*Rc*(1024-  
valorLDRSalon)); //usar si LDR entre A0 y Vcc (como en el esquema anterior)  
  
    iluminacionHabitacion =  
valorLDRHabitacion; //(((long)valorLDRHabitacion*A*10)/((long)B*Rc*(1024-  
valorLDRHabitacion)); //usar si LDR entre A0 y Vcc (como en el esquema anterior)  
  
    if (iluminacionCocina < 90) {  
  
        digitalWrite(luzCocina, HIGH);  
  
        Serial.print("*ENCEDER COCINA :");  
  
        Serial.println(iluminacionCocina);  
  
    }  
  
    else {  
  
        digitalWrite(luzCocina, LOW);  
  
        Serial.print("*APAGAR COCINA: ");  
  
    }  
}
```

```
    Serial.println(iluminacionCocina);
}

if (iluminacionSalon < 90) {
    digitalWrite(luzSalon, HIGH);
    Serial.print("*ENCEDER SALON :");
    Serial.println(iluminacionSalon);
    //CortinaSalon('M');
}
else {
    digitalWrite(luzSalon, LOW);
    Serial.print("*APAGAR SALON: ");
    Serial.println(iluminacionSalon);
    //CortinaSalon('m');
}

if (iluminacionHabitacion < 90) {
    digitalWrite(luzHabitacion, HIGH);
    Serial.print("*ENCEDER HABITACION :");
    Serial.println(iluminacionHabitacion);
}
else {
    digitalWrite(luzHabitacion, LOW);
    Serial.print("*APAGAR HABITACION: ");
```

```
        Serial.println(iluminacionHabitacion);  
    }  
}
```

#### 4.1.3.1.9 Modulo Sensor Tarjeta

```
void sensoresTarjeta() {  
    //lee si hay una tarjeta  
    if (!rfid.PICC_IsNewCardPresent() || !rfid.PICC_ReadCardSerial())  
        return;  
  
    //Identifica e tipo de tarjeta  
    MFRC522::PICC_Type piccType = rfid.PICC_GetType(rfid.uid.sak);  
  
    // Chequea q el tipo de tarjeta sea compatible  
    if (piccType != MFRC522::PICC_TYPE_MIFARE_MINI &&  
        piccType != MFRC522::PICC_TYPE_MIFARE_1K &&  
        piccType != MFRC522::PICC_TYPE_MIFARE_4K) {  
        Serial.println(F("*Tarjeta invalida"));  
        return;  
    }  
  
    //capturo el codigo de la tarjeta  
    String strID = "";
```

```

for (byte i = 0; i < 4; i++) {

    strID +=

        (rfid.uid.uidByte[i] < 0x10 ? "0" : "") +

        String(rfid.uid.uidByte[i], HEX) +

(i != 3 ? ":" : "");

}

strID.toUpperCase();

Serial.print("**El codigo de la tarjeta es : ");

Serial.println(strID);

//valido que el codio de la tarjeta sea correcto

boolean numero_card = true; // meu cartão

if (miTarjeta != strID) {

    numero_card = false;

}

if (numero_card) { // numero coincide

    if (anterior_estadoPuertaFrente) {

        puerta('n');

    }

    else {

```

```

puerta('N');

Serial.println("**Bienvenido!"); // imprime mensagem

digitalWrite(luzVerdePasillo, HIGH); // acende LED verde

// myServoPuerta.write(0); // libera porta

for (int i = 0; i < 12; i++) { // toca música aceptado

int access_noteDuration = 1000 / access_noteDurations[i];

tone(parlantePasillo, access_melody[i], access_noteDuration);

int access_pauseBetweenNotes = access_noteDuration * 1.30;

delay(access_pauseBetweenNotes);

noTone(parlantePasillo);

}

//delay(5000); // espera

//myServoPuerta.write(105); // trava porta

digitalWrite(luzVerdePasillo, LOW); // apaga LED verde

anterior_estadoPuertaPasillo = true; //estado de puerta abierta

}

}

else { // numero no coincide

Serial.println("**Tarjeta no Reconocida!"); // imprime mensagem

digitalWrite(luzRojaPasillo, HIGH); // Encender Luz Roja

```

```

    for (int i = 0; i < 6; i++) { // toca música de rechazo
int fail_noteDuration = 1000 / fail_noteDurations[i];

    tone(parlantePasillo, fail_melody[i], fail_noteDuration);

    int fail_pauseBetweenNotes = fail_noteDuration * 1.30;

    delay(fail_pauseBetweenNotes);

    noTone(parlantePasillo);

}

delay(1000);

digitalWrite(luzRojaPasillo, LOW); // apaga luz rojo

}

delay(500);

rfid.PICC_HaltA();

rfid.PCD_StopCrypto1();

}

```

#### **4.1.3.1.10 Modulo Sensor Temperatura**

//seteos automaticos

```
void sensoresTemperatura() {
```

```

    sensorTemperaturaHabitacion.requestTemperatures(); // Send the command to get
temperatures

```

```
sensorTemperaturaCocina.requestTemperatures(); // Send the command to get
temperatures
```

```
sensorTemperaturaSalon.requestTemperatures(); // Send the command to get
temperatures
```

```
String tempHab = "*Temp hab: ";
```

```
float tempHabValue = sensorTemperaturaHabitacion.getTempCByIndex(0);
```

```
String tempCocina = "*Temp cocina: ";
```

```
float tempCocinaValue = + sensorTemperaturaCocina.getTempCByIndex(0);
```

```
String tempSalon = "*Temp salon: ";
```

```
float tempSalonValue = sensorTemperaturaSalon.getTempCByIndex(0);
```

```
Serial.println(tempHab + tempHabValue);
```

```
Serial.println(tempCocina + tempCocinaValue);
```

```
Serial.println(tempSalon + tempSalonValue);
```

```
temperaturaHabitacion = sensorTemperaturaHabitacion.getTempCByIndex(0);
```

```
temperaturaCocina = sensorTemperaturaCocina.getTempCByIndex(0);
```

```
temperaturaSalon = sensorTemperaturaSalon.getTempCByIndex(0);
```

```
if (temperaturaHabitacion > 26) {
```

```
    digitalWrite(ventiladorHabitacion, LOW); digitalWrite(calefaccionHabitacion, LOW); //
prender ventilador - apagar calefaccion

    //delay(1000);

}

else if (temperaturaHabitacion < 25) {

    digitalWrite(ventiladorHabitacion, HIGH); digitalWrite(calefaccionHabitacion, HIGH); //
prender ventilador - apagar calefaccion

    //delay(1000);

}

else {

    digitalWrite(ventiladorHabitacion, HIGH); digitalWrite(calefaccionHabitacion, LOW); //
prender ventilador - apagar calefaccion

    //delay(1000);

}

if (temperaturaCocina > 26) {

    digitalWrite(ventiladorCocina, LOW); digitalWrite(calefaccionCocina, LOW); // prender
ventilador - apagar calefaccion

    //delay(1000);

}

else if (temperaturaCocina < 25) {

    digitalWrite(ventiladorCocina, HIGH); digitalWrite(calefaccionCocina, HIGH); // prender
ventilador - apagar calefaccion
```



```
//delay(5000);  
  
}  
  
else {  
  
    digitalWrite(ventiladorCocina, HIGH); digitalWrite(calefaccionCocina, LOW); // prender  
ventilador - apagar calefaccion  
  
    //delay(1000);  
  
}  
  
  
if (temperaturaSalon > 26) {  
  
    digitalWrite(ventiladorSalon, LOW); digitalWrite(calefaccionSalon, LOW); // prender  
ventilador - apagar calefaccion  
  
    //delay(1000);  
  
}  
  
else if (temperaturaSalon < 25) {  
  
    digitalWrite(ventiladorSalon, HIGH); digitalWrite(calefaccionSalon, HIGH); // prender  
ventilador - apagar calefaccion  
  
    //delay(1000);  
  
}  
  
else {  
  
    digitalWrite(ventiladorSalon, HIGH); digitalWrite(calefaccionSalon, LOW); // prender  
ventilador - apagar calefaccion  
  
    //delay(1000);  
  
}  
  
}
```

## **4.1.4 Riesgos del sistema.**

### **4.1.4.1 Fallos en la energía eléctrica.**

#### **4.1.4.1.1 Magnitud del riesgo**

Nivel 3 – Riesgo Alto (gran impacto en el proyecto).

#### **4.1.4.1.2 Descripción**

Este sistema se basa en los sensores distribuidos a través de toda la casa y la conexión con su servidor para el manejo remoto. Aunque este previsto que la casa se pueda administrar de manera manual (apertura y cierre de puertas, cortinas, etc.). El sistema dejara de funcionar.

#### **4.1.4.1.3 Impactos**

- No se podrán realizar acciones desde las aplicaciones.
- Calidad de vida del usuario.
- La alarma contara con un tiempo limitado de encendido y no podrá comunicarse con el usuario en caso de alguna anomalía.

#### **4.1.4.1.4 Estrategia de mitigación**

- Prever ante históricos de la ubicación de la casa del usuario para conocer probabilidades y tiempos de cortes de energía eléctrica.
- En base a las probabilidades de cortes de energía eléctrica, proveer soluciones como UPS y/o generadores eléctricos.

#### **4.1.4.1.5 Plan de contingencia**

Si bien se podrá utilizar la casa de manera normal, mediante modos manuales. El sistema completo dejara de funcionar sin energía eléctrica, por lo que como plan de contingencia se puede implementar el uso de generadores de energía eléctrica externos.

#### **4.1.4.2 Fallos la comunicación de internet.**

##### **4.1.4.2.1 Magnitud del riesgo**

Nivel 2 – Riesgo Medio (impacto considerable en el proyecto).

##### **4.1.4.2.2 Descripción**

Si la comunicación a internet falla, no se podrá administrar las acciones al microcontrolador del usuario, aunque tendrá accionadores internos que le permitirán comunicarse con el microcontrolador. Este inconveniente será solo si se tiene el servidor en internet.

##### **4.1.4.2.3 Impactos**

- No se podrán realizar acciones desde las aplicaciones.
- Desincronización del microcontrolador con el servidor.
- Calidad de vida del usuario.

##### **4.1.4.2.4 Estrategia de mitigación**

- Prever ante históricos los cortes en cuanto a las conexiones de internet.
- Contar con un dispositivo USB 4G conectado al router. Mediante la configuración del router realizar un auto switching de interfaces de las conexiones a internet.

##### **4.1.4.2.5 Plan de contingencia**

En el caso que no exista la conexión a internet se podrá utilizar la casa de manera normal, mediante modos manuales, ya que habrá llaves distribuidas por las diferentes habitaciones. Cuando retorne la conexión a internet se podrá volver a utilizar la plataforma mediante la web (teléfono, tablet y/o pc).

### **4.1.4.3 Rotura en sensor.**

#### 4.1.4.3.1 Magnitud del riesgo

Nivel 1 – Riesgo Bajo (impacto pequeño en el proyecto).

#### 4.1.4.3.2 Descripción

Si falla algún sensor de la casa, este no se encontrara marcando los valores que debería.

#### 4.1.4.3.3 Impactos

- Errores en las estadísticas (valores no reales).
- Si el microcontrolador se encuentra en modo automático, podría tomar acciones no acordes a los valores reales.

#### 4.1.4.3.4 Estrategia de mitigación

- Realizar un mantenimiento preventivo de todos los dispositivos involucrados.

#### 4.1.4.3.5 Plan de contingencia

Se deberá cambiar el sensor que se ha roto o estaría tomando medidas incorrectas. Siempre se podrá desactivar el modo automático en el microcontrolador, para así poder tomar las acciones necesarias a través de la plataforma web.

**CAPITULO V**  
**CONCLUSIÓN DEL PROYECTO DE GRADO**

## **5 Capítulo V – Conclusión del Proyecto**

### **5.1 Introducción**

En el presente capítulo se detallan las conclusiones respecto al trabajo de investigación junto a los objetivos cumplidos tanto general como específicos.

Para llegar a cubrir las expectativas del objetivo general se realizaron pruebas utilizando el prototipo, de esa forma ir modificando el funcionamiento del mismo, de tal forma que el uso del sistema domótica sea óptimo.

### **5.2 Conclusiones**

Se logró diseñar e implementar un sistema de control domótico basado en Arduino con una aplicación web que maneje algunos elementos de una vivienda y así cumplir nuestro objetivo general.

Es posible a través de una interfaz web, manipular de forma remota los distintos dispositivos instalados en cada ambiente de la vivienda; por ejemplo encendido/apagado de luces, ventiladores o calefacción. También es posible programar ciertos parámetros para que los actuadores inicien de forma automática según los parámetros seteados.

#### **5.2.1 Objetivo General Propuesto**

“El objetivo general del proyecto es realizar el estudio, diseño y aplicación de un sistema domótico a escala que permita en una vivienda estándar reducir el consumo de energía, aumentar el confort, accesibilidad, comunicación y seguridad de la vivienda. “

Se logró diseñar e implementar un sistema domótico, a través del estudio de los distintos tipos de sistemas domóticos en el mercado, el cual por medio de sus funciones logra cumplir los objetivos de:

- La programación de la central domótica permite reducir el consumo de energía programando el encendido / apagado de luces, ventiladores y calefacción según los valores de los sensores de luz, temperatura y humedad
- Accediendo al sistema web es posible el control automático de casi todo tipo de dispositivo eléctrico, encendido/apagado de luces, calefacción y ventilación, apertura/cierre de puertas de ingreso, aumentando el confort y accesibilidad en la vivienda.
- El usuario puede informarse de todo tipo de eventos que ocurren el domicilio, ya sea accediendo a visualizar las cámaras del hogar, consultando el estado de luces, puertas, alarma y temperatura; también en caso de acontecimientos de emergencia el sistema comunica automáticamente al usuario como por ejemplo la alerta de intruso.
- Una vez activado el sistema de seguridad, los sensores se ponen en funcionamiento y en caso de detectar un intruso, avisa al usuario vía mail y pone en funcionamiento los mecanismos de sonido y alerta de la vivienda.

En los puntos nombrados anteriormente logramos cumplir con el objetivo propuestos para el proyecto de grado

## **5.2.2 Objetivos Específicos Propuestos**

En cuanto a los objetivos específicos planteados en el trabajo de investigación, a continuación se describe el cumplimiento de cada uno de ellos:

- Se desarrolló el software controlador del prototipo permitiendo el control de dispositivos como sensores y actuadores.
- Se desarrolló las páginas web necesarias para su navegación correspondiente.
- Se controlan luces, puertas, ventilador y alarma por medio de un teléfono móvil Inteligenteo PC con conexión a internet.

- Se reciben y envían datos por medio de Wifi o conexión USB y se procesa a través de Arduino.
- Se integraron los elementos software, hardware y aplicaciones satisfactoriamente.
- Se desarrolló una interfaz de control sencillo para el usuario de manera que sea auto suficiente.
- Se realizaron pruebas de instalación del prototipo, presentados sobre ello una maqueta que implementa los dispositivos domóticos.
- El presente trabajo de implementación muestra que la plataforma Arduino permite automatizar tareas de rutina control de dispositivos conectado a una vivienda.

### **5.3 RECOMENDACIONES**

El proyecto que presentamos es un prototipo, pero las aplicaciones a realizar en la domótica con Arduino son cientos y las que podemos emplear para mejorar el prototipo, son sensores de rotura de vidrio, control de persianas, sensor de fuego, sensores magnéticos en ventanas, sensor de lluvia, relé con módulos Wifi, módulo GSM para realizar llamadas o envío de SMS del sistema al usuario entre otros sensores y actuadores nombrados en los puntos 3.1.2.5 y 3.1.2.6.

Nosotros utilizamos los sensores y actuadores más comunes en el mercado por una cuestión de tiempos y costos.

Se recomienda en cuanto a trabajos posteriores el desarrollo de una aplicación móvil para el control manual y automático bajo plataformas Android

Es importante aclarar que en este tipo de proyectos se trabaja con tensiones de 5 voltios esto a nivel prototipo de maqueta, para la implementación del sistema en una vivienda con tensiones de 220 voltios se recomienda el uso de relés y otros dispositivos de amplitud en función al voltaje. Aunque es recomendable realizar la medición del proyecto



con un especialista en electrónica en lo referente a la alimentación externa ya que puede existir un sobrecalentamiento de los componentes sino se administra el voltaje y amperaje adecuado.

## **Bibliografía**

Huidrobo Moya J.M. (2004) Domótica Edificios Inteligentes. Guía aplicada.

Emilio Lledó Sánchez (2012), Diseño de un control domótico basado en Arduino: Universidad Técnica de Valencia, España

Jesús Rodarte Dávila, Jenaro Carlos Paz Gutiérrez, José Saúl González Campos, Ramsés Román García Martínez (2013), Casa inteligente y segura (fase 2), (Colección Textos Universitarios, Serie Investigación) Universidad Autónoma de Ciudad Juárez, México

Mario Rodríguez Cerezo, Sistema de Control remoto para aplicaciones domóticas a través de internet: Proyecto final de Carrera Universidad Autónoma de Madrid, Escuela Politécnica Superior

Coronel, R., Diseño e implementación de un sistema domótico para un control de energía eléctrica. Universidad Mayor de San Andrés, Bolivia.

Prieto Francisca, Martínez Eustaquio. Domotización con hardware abierto: Arduino & Shields, Facultad Politécnica, Universidad Nacional del Este. Ciudad del Este, Paraguay.  
CIEC - Colegio de Ingenieros Especialistas de Córdoba CD (2012), Guía de contenidos mínimos para la elaboración de un proyecto de domótica”, Argentina

Pérez V. (2010), Contribución al diseño de sistemas domóticos y de entretenimiento utilizando hardware libre y software de código abierto, Tesis, México

Emilio Lledo Sanchez (2012), Diseño de un control domótico basado en Arduinorecuperado: Universidad técnica de valencia.

Ferriz, D. S. (2015). Diseño y prototipo de una vivienda inteligente.

Garcia, D. G. (2014). Apunte de Arduino.

Aide, E. (2015). Desarrollo de proyectos domóticos. Universidad de Oviedo, España.

Carpio, M., Cárdenas, T., & Chavez, P. (2013). Desarrollo e implementación de un sistema de seguridad y confort para hogares monitoreado y administrado a través de una aplicación web. Escuela Superior Politécnica Litoral. Facultad de Ingeniería en Electricidad y Computación. Guayaquil, Ecuador.

Coronel, R. (2014). Diseño e implementación de un sistema domótico para un control de energía eléctrica. Universidad Mayor de San Andrés, Bolivia.

Diosdado, R. (2014). Manual de Arduino Programación y conceptos básicos. España: ZonaMaker.

Lledó, E. (2012). Diseño de un sistema de control domótico basado en la plataforma Arduino. Universidad Politécnica de Valencia. Valencia, España.

Yañez. La domótica un bien para todos.

Torrente, O. Arduino. Curso Práctico de formación. México: Editorial Alfaomega.

Sampieri, R. (2003). Metodologías de la Investigación.

Sensores actuadores –sistema de control de gestión  
<http://www.newsai.es/>

Domótica  
<https://es.wikipedia.org/wiki/Dom%C3%B3tica>

Arduino, Arduino Página Oficial: <http://www.arduino.cc>

Adesva Tecnología, Empresa en infraestructuras domótica  
<http://www.adesvatecnologia.com/home.php>.

Huidoboro J.M., Manual de Domótica,  
<http://www.ramonmillan.com/libros/libroManualDomotica.php>

Sensores, Sensores de movimiento,  
<http://sensmovimiento.blogspot.com/p/sensor-de-movimiento.html>.

CasaDomo, Domótica, Arquitectura Centralizada,  
<http://antoniopendolema.blogspot.com/2013/04/arquitectura-centralizada.html>

Electrónicos, Motores y servos  
<http://www.electronicoscaldas.com/motores-y-servos/468-micro-servo-motor-sg90.htm>

José Manuel Huidobro Moya, Ramón J Millar. Edificios Inteligentes  
<http://www.fenercom.com/pdf/publicaciones/la-domotica-solucion-de-futuro-feneron.pdf>

Kit de Sensores y Actuadores compatibles con Arduino. (2015). Sensores y Actuadores para Arduino. : <http://www.agspecinfo.com/pdfs/K/KITSENAT.PDF>.

<https://es.slideshare.net/danielramirez244/casas-inteligentes-o-domotica>

Soria, D. (2009). Metodologías de Desarrollo  
[http://ingenieriadesoftware.mex.tl/61885\\_Modelo-V.html](http://ingenieriadesoftware.mex.tl/61885_Modelo-V.html)

S.H.I.E.L.Ds  
<https://aprendiendoarduino.wordpress.com/2016/06/27/shields-arduino-2/>

Panamahitek – Uso de librerías y tutoriales  
<http://panamahitek.com/arduino/>

## **Anexos**

### **Concepto de electrónica**

Estudia y emplea sistemas cuyo funcionamiento se basa en la conducción y el control del flujo de los electrones u otras partículas cargadas eléctricamente.

El diseño y la gran construcción de circuitos electrónicos para resolver problemas prácticos forman parte de la electrónica y de los campos de la ingeniería electrónica, electromecánica y la informática en el diseño de software para su control.

La electrónica desarrolla en la actualidad una gran variedad de tareas. Los principales usos de los circuitos electrónicos son el control, el procesado, la distribución de información, la conversión y la distribución de la energía eléctrica. Estos dos usos implican la creación o la detección de campos electromagnéticos y corrientes eléctricas.

Mira a tu alrededor radio, televisor, PC, teléfono móvil, lavadora todos ellos tienen electrónica.

### **Computación física**

La Computación física, significa la construcción de sistemas inter-activos físicos mediante el uso de software y hardware que pueden sentir y responder al mundo analógico. Si bien esta definición es suficientemente amplia para abarcar aspectos como los sistemas inteligentes de control de tráfico de automóviles o los procesos de automatización de fábricas, en un sentido más amplio, la computación física es un marco creativo para la comprensión de la relación de los seres humanos en el mundo digital. En la práctica, a menudo el término describe el arte hecho a mano, diseño de proyectos DIY o pasatiempos que utilizan sensores y microcontroladores para traducir entradas analógicas a sistemas basados en software, y/o controlar dispositivos electromecánicos como motores, servos, iluminación u otro hardware.

Otras implementaciones de computación física trabajan con el reconocimiento de la voz, la cual se capta e interpretan sus ondas sonoras a través de micrófonos u otros dispositivos de detección de ondas sonoras, también la visión por computador, que aplica algoritmos a los videos detectados por algún tipo de cámara. Interfaces táctiles son también un ejemplo de la computación física.

El prototipado (crear montajes rápidos con ayuda de una protoboard y componentes básicos de electrónica) juega un papel importante en la computación física. Herramientas como Arduino y Fritzing son útiles para diseñadores, artistas, estudiantes y entusiastas porque ayudan a elaborar prototipos rápidamente.

### **Hardware libre**

Se llama hardware libre a los dispositivos de hardware cuyas especificaciones y diagramas esquemáticos son de acceso público, ya sea bajo algún tipo de pago o de forma gratuita. La filosofía del software libre (las ideas sobre la libertad del conocimiento) es aplicable a la del hardware libre. Se debe recordar en todo momento que libre no es sinónimo de gratis. El hardware libre forma parte de la cultura libre.

Dado que el hardware tiene asociados a él costos variables directos, ninguna definición de software libre se puede aplicar directamente sin modificación. En cambio, el término hardware libre se ha usado principalmente para reflejar el uso del software libre con el hardware y el lanzamiento libre de la información con respecto al hardware, a menudo incluyendo el lanzamiento de los diagramas esquemáticos, diseños y montajes.

El software libre (en inglés free software, aunque esta denominación también se confunde a veces con "gratis" por la ambigüedad del término "free" en el idioma inglés, por lo que también se usa "libre software" y "lógica libre") es la denominación del software que respeta la libertad de los usuarios sobre su producto adquirido y, por tanto, una vez obtenido puede ser usado, copiado, estudiado, modificado, y redistribuido libremente. Según la Free Software Foundation, el software libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, modificar el software y distribuirlo modificado.

## **Voltaje**

Una magnitud física que impulsa a los electrones a lo largo de un conductor (por ejemplo un cable) en un circuito eléctrico cerrado, provocando el flujo de una corriente eléctrica. Su unidad es el Voltio(V). El instrumento usado para medir el voltaje se conoce como voltímetro.

## **Voltaje DC**

Es el flujo continuo de electrones a través de un conductor entre dos puntos de distinto potencial. En la corriente continua las cargas eléctricas circulan siempre en la misma dirección, es continua la corriente mantiene siempre la misma polaridad. En la norma sistemática europea el color negro corresponde al negativo y el rojo al positivo o

sencillamente se simboliza para el positivo con VCC, +, VSS y para el negativo con 0V, -, GND.

Muchos aparatos necesitan corriente continua para funcionar, sobre todos los que llevan electrónica (equipos audiovisuales, computadores, etc.), para ello se utilizan fuentes de alimentación. Lo puedes encontrar en la baterías, pilas, salida de los carga-dores de computador.

### **Voltaje AC**

Es la corriente eléctrica en la que la magnitud y dirección varían cíclicamente. La forma de onda de la corriente alterna más comúnmente utilizada es la de una onda seno.

El voltaje AC es el que llega a la tomas de electricidad de los hogares y a las empresas, es muy común encontrarla en las tomas de corriente donde se co-nectan nuestros electrodomésticos. Sin embargo, las señales de audio y de radio transmitidas por los cables eléctricos son también ejemplos de corriente alterna. En estos usos, el fin más importante suele ser la transmisión y recuperación de la información codificada (o modulada) sobre la señal de la AC.

### **Corriente**

Es el flujo de electrones a través de un conductor o semiconductor en un sentido. La unidad de medida de ésta es el amperio (A). Una corriente eléctrica, puesto que se trata de un movimiento de cargas, produce un campo magnético, un fenómeno que puede aprovecharse en el electroimán, este es el principio de funciona-miento de un motor.

El instrumento usado para medir la intensidad de la corriente eléctrica es el galvanómetro que, calibrado en amperios, se llama amperímetro, colocado en serie con el conductor cuya intensidad se desea medir.

### **Resistencia**

Es la propiedad física mediante la cual todos los materia-les tienden a oponerse al flujo de la corriente. La unidad de este parámetro es el Ohmio ( $\Omega$ ). Puedes encontrar resistencias

en los calefactores eléctricos, tarjetas electrónicas, estufas son muy útiles para limitar el paso de la corriente.

### **Ley de Ohm**

La ley dice que la corriente (I) que circula por un conductor eléctrico es directamente proporcional al voltaje (V) e inversamente proporcional a la resistencia (R).

La pirámide de la derecha es muy útil para conocer la fórmula a la que es igual la variable que tapes con el dedo, por ejemplo: Tapa con tu dedo la V (voltaje), entonces voltaje va a ser igual a I (corriente) por R (resistencia), una más, tapa I (Corriente), I va ser igual a V dividido R.

### **Variable digital**

También llamadas variables discretas. Se caracterizan por tener dos estados diferenciados y por lo tanto se pueden llamar binarias. Siendo estas variables más fáciles de tratar (en lógica serían los valores Verdadero (V) y Falso (F) o podrían ser 1 ó 0 respectivamente).

Un ejemplo de una señal digital es el interruptor del timbre de tu casa, porque este interruptor tiene dos estados pulsado y sin pulsar

### **Variable análoga**

Son aquellas que pueden tomar un número infinito de valores comprendidos entre dos límites. La mayoría de los fenómenos de la vida real son señales de este tipo. (sonido, temperatura, voz, video, etc.)

Un ejemplo de sistema electrónico analógico es un parlante, que se emplea para amplificar el sonido de forma que éste sea oído por una gran audiencia. Las ondas de sonido que son analógicas en su origen, son capturadas por un micrófono y convertidas en una pequeña variación analógica de tensión denominada señal de audio.

## **Divisor de voltaje**

Un divisor de voltaje es una configuración de circuito eléctrico que reparte el voltaje de una fuente ( $V_{in}$ ) entre una o más resistencias ( $R_1$ ,  $R_2$ ) conectadas en serie (una a continuación de otra).

## **Conversor análogo-digital CAD ó ADC**

Un conversor (o convertidor) analógico-digital (CAD), (o también ADC del inglés "Analog-to-Digital Converter") es un dispositivo electrónico capaz de convertir una entrada analógica de voltaje en un valor binario. Se utiliza en equipos electrónicos como computadores, grabadores de sonido y de vídeo, y equi-pos de telecomunicaciones. La señal analógica, que varía de forma continua en el tiempo, se conecta a la entrada del dispositivo y se somete a un muestreo a una velocidad fija, obteniéndose así una señal digital a la salida del mismo.

## **Modulación por ancho del pulso PWM**

La modulación por ancho de pulsos (también conocida como PWM, siglas en inglés de pulse-width modulation) de una señal o fuente de energía es una técnica en la que se modifica el ciclo de trabajo de una señal periódica (una senoidal o una cuadrada, por ejemplo), ya sea para transmitir información a través de un canal de comunicaciones o para controlar la cantidad de energía que se envía a una carga.

Por ejemplo si le aplicamos PWM a un LED podemos variar su intensidad de brillo y si le aplicamos un PWM a un motor DC logramos variar la velocidad del mismo con la característica de mantener su par (fuerza) constante.

## **Comunicación serial**

Es una interfaz de comunicaciones de datos digitales, frecuentemente utilizada por computadores y periféricos, donde la información es transmitida bit a bit enviando un solo bit a la vez. Uno de sus usos es monitorear a través de la pantalla del computador el



estado del periférico conectado, por ejemplo al pulsar la letra A en el teclado se debe encender un LED conectado de manera remota al computador.

### **S-H.I.E.L.D Arduino**

Las shields son placas de circuitos modulares que se montan unas encima de otras para dar funcionalidad extra a un Arduino. Estas Shields son apilables.

Las shields se pueden comunicar con el Arduino bien por algunos de los pines digitales o analógicos o bien por algún bus como el SPI, I2C o puerto serie, así como usar algunos pines como interrupción. Además estas shields se alimentan generalmente a través del Arduino mediante los pines de 5V y GND.

En muchos casos nos encontraremos que los shields vienen sin soldar las patillas o en algunos casos sin algunos componentes soldados.

Cada Shield de Arduino debe tener el mismo factor de forma que el estándar de Arduino con un espaciado de pines concreto para que solo haya una forma posible de encajarlo.

Debemos tener en cuenta que cuando instalemos un shield, habrá que leer su documentación que nos dirá si inhabilita algunas de los pines I/O o si usa un bus y que requisitos tiene para su utilización. Generalmente las shields vienen con un ejemplo o una librería para su uso.

### **Tipos de SHIELDS:**

- Arduino WiFi Shield 101: Este shield usa el módulo WINC1500 y también añade funciones de cifrado hardware gracias al chip de cifrado ATECC508A diseñado para el mercado de IoT. Los pines 5 y 7 son usados por este shield. Usa una nueva librería llamada WiFi101 que también usan otros Arduinos con wifi integrado como el MKR1000. Esta librería es muy compleja y ocupa más del 60% de la memoria disponible en el Arduino UNO, dejando poco espacio para los sketches. Si se van a realizar programas complejos, este shield es recomendable usarlo con Arduino Zero, 101 o Mega.

- Arduino GSM Shield: Conecta Arduino a Internet mediante GPRS, usando una tarjeta SIM. También permite enviar y recibir mensajes y llamadas de voz (SMSs). Telefónica presentó la nueva versión del proyecto Arduino GSM/GPRS Shield, un dispositivo que incorpora conexión GPRS/GSM a una placa base de hardware libre, dando como resultado un dispositivo de bajo coste conectado gracias a la tecnología machine to machine(M2M) de Telefónica. En este caso por el consumo de esta shield, se hace necesario alimentar a Arduino mediante una fuente externa y no desde el USB ya que no es capaz de proporcionar suficiente energía. También es posible hacer llamadas de voz haciendo unas pequeñas modificaciones, añadiendo un micrófono y un altavoz.
- Shield Relay Driver de 8 canales: Este shield permite manejar hasta 8 relés (que no están en el shield) usando solo dos pines de I/O del Arduino. Se comunica con Arduino usando el bus I2C, de forma que apilando varias de estas shields podemos manejar 16, 24 y más relés.
- GPS Shield: Con un slot para micro SD donde guardar logs. No es compatible con Arduino Mega ni Due, RTC incluido. Conexión por puerto serie o conexión directa.
- Bluetooth Shield: *Comunicación inalámbrica por bluetooth. Conecta con la UART del Arduino o software serial con selector de puerto serie.*