



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**  
**ESCUELA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y REDES**  
**INDUSTRIALES**

**“DISEÑO DE UN SISTEMA PARA SEGURIDAD DE UNA  
VIVIENDA MEDIANTE PASARELA ACTIVADA POR VOZ Y  
VIDEO A DESARROLLARSE EN LA EMPRESA “RIO SOLAR  
SMART ENERGY” DE LA CIUDAD DE RIOBAMBA”**

**TRABAJO DE TITULACIÓN: DISPOSITIVO TECNOLÓGICO**

**Para optar al Grado Académico de:**

**INGENIERO EN ELECTRÓNICA, CONTROL Y REDES  
INDUSTRIALES**

**AUTOR: GUARANGA ALLAUCA JUAN ENRIQUE**

**TUTOR: Ing. HENRY ERNESTO VALLEJO VIZUETE Ms.**

Riobamba – Ecuador

2018

**©2018, Guaranga Allauca Juan Enrique**

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**  
**ESCUELA DE INGENIERIA ELECTRÓNICA EN CONTROL Y REDES**  
**INDUSTRIALES**

El Tribunal de Tesis certifica que: El dispositivo tecnológico: “DISEÑO DE UN SISTEMA PARA SEGURIDAD DE UNA VIVIENDA MEDIANTE PASARELA ACTIVADA POR VOZ Y VIDEO A DESARROLLARSE EN LA EMPRESA “RIO SOLAR SMART ENERGY” DE LA CIUDAD DE RIOBAMBA”, de responsabilidad del señor Guaranga Allauca Juan Enrique, ha sido minuciosamente revisado por los Miembros del Tribunal de Tesis, quedando autorizada su presentación.

Dr. Julio Santillán C.

**VICEDECANO DE LA FACULTAD DE**  
**INFORMÁTICA Y ELECTRÓNICA**

\_\_\_\_\_

Ing. Freddy Chávez

**DIRECTOR DE ESCUELA DE INGENIERÍA**  
**ELECTRÓNICA EN CONTROL Y**  
**REDES INDUSTRIALES**

\_\_\_\_\_

Ing. Henry Vallejo

**DIRECTOR DEL TRABAJO DE**  
**TITULACIÓN**

\_\_\_\_\_

Dr. Geovanny Vallejo

**MIEMBRO DEL TRABAJO DE**  
**TITULACIÓN**

\_\_\_\_\_

Yo, Juan Enrique Guaranga Allauca soy responsable de las ideas, doctrinas y resultados expuestos en este Trabajo de Titulación y el patrimonio intelectual del Trabajo de Titulación pertenece a la Escuela Superior Politécnica de Chimborazo.

---

Juan Enrique Guaranga Allauca

**ESTUDIANTE**

## **DEDICATORIA**

### **A mis padres.**

Porque cada día juntos era como un milagro que la vida nos regalaba, a mi madre que con solo un abrazo y la sonrisa que día a día se dibuja en su rostro, llena de luz mi mundo. A mi padre, desde el lejano cielo, se, que si escucho cuidadosamente incluso ahora podré escuchar el ritmo de sus pasos, que desde niño he intentado seguir.

*Juan Enrique Guaranga Allauca*

## **AGRADECIMIENTO**

A mis hermanos, maestros, tutores y amigos quienes fueron un apoyo constante e importante en el desarrollo de este proyecto; y gracias a la vida misma, que me ha dado tanto.

*Juan Enrique Guaranga Allauca*

## TABLA DE CONTENIDO

DEDICATORIA.....	V
AGRADECIMIENTO .....	VI
ÍNDICE DE ABREVIATURAS.....	XII
ÍNDICE DE TABLAS.....	XIII
ÍNDICE DE FIGURAS.....	XV
ÍNDICE DE GRÁFICOS.....	XVIII
RESUMEN.....	XIX
ABSTRACT .....	XX
INTRODUCCIÓN .....	1

## CAPÍTULO I

1.	MARCO TEÓRICO REFERENCIAL.....	4
1.1.	Sistema de seguridad.....	4
1.1.1.	<i>Definición</i> .....	4
1.1.2.	<i>Sistema de seguridad en el hogar</i> .....	4
1.1.3.	<i>Historia</i> .....	5
1.1.4.	<i>Elementos en un sistema de seguridad</i> .....	6
1.1.4.1.	<i>Elementos Activos</i> .....	6
1.1.4.2.	<i>Elementos Pasivos</i> .....	6
1.2.	Domótica .....	7
1.2.1.	<i>Definición</i> .....	7
1.2.2.	<i>Aplicación de la domótica</i> .....	7
1.2.3.	<i>Ventajas de la Domótica</i> .....	8
1.2.4.	<i>Desventajas de la Domótica</i> .....	8
1.2.5.	<i>Domótica y Pasarela Residencial</i> .....	9
1.2.5.1.	<i>Características de la Pasarela residencial</i> .....	9

<b>1.2.6.</b>	<b><i>Elementos de un Sistema Domótico</i></b> .....	<b>11</b>
1.2.6.1.	<i>Central de Gestión</i> .....	11
1.2.6.2.	<i>Nodos</i> .....	12
1.2.6.3.	<i>Actuadores</i> .....	12
1.2.6.4.	<i>Sirenas</i> .....	13
1.2.6.5.	<i>Sensores</i> .....	14
<b>1.3.</b>	<b><i>Cercas Eléctricas Domesticas</i></b> .....	<b>15</b>
<b>1.3.1.</b>	<b><i>Definición</i></b> .....	<b>15</b>
<b>1.3.2.</b>	<b><i>Composición y Funcionamiento</i></b> .....	<b>16</b>
1.3.2.1.	<i>Alambre con doble galvanización</i> .....	16
1.3.2.2.	<i>Postes</i> .....	17
1.3.2.3.	<i>Tensores</i> .....	17
1.3.2.4.	<i>Aros de supervisión</i> .....	18
1.3.2.5.	<i>Electrificador</i> .....	18
<b>1.3.3.</b>	<b><i>Uso de las cercas eléctricas</i></b> .....	<b>21</b>
<b>1.4.</b>	<b><i>Video vigilancia</i></b> .....	<b>21</b>
<b>1.4.1.</b>	<b><i>Introducción</i></b> .....	<b>21</b>
<b>1.4.2.</b>	<b><i>Utilidad de video vigilancia</i></b> .....	<b>21</b>
<b>1.4.3.</b>	<b><i>Componentes de un sistema de video vigilancia</i></b> .....	<b>22</b>
1.4.3.1.	<i>Cámaras de seguridad</i> .....	23
1.4.3.2.	<i>Medio de transmisión (Central de monitoreo)</i> .....	26
<b>1.5.</b>	<b><i>Arduino</i></b> .....	<b>27</b>
<b>1.5.1.</b>	<b><i>Introducción</i></b> .....	<b>27</b>
<b>1.5.2.</b>	<b><i>Elementos Básicos de programación C++</i></b> .....	<b>28</b>
<b>1.5.3.</b>	<b><i>Placa de Desarrollo Arduino Mega 2560 R3</i></b> .....	<b>29</b>
<b>1.5.4.</b>	<b><i>Placas y módulos de Expansión Arduino</i></b> .....	<b>29</b>
1.5.4.1.	<i>Módulo de Expansión Ethernet ENC28J60</i> .....	29
1.5.4.2.	<i>Módulo de relés</i> .....	30
1.5.4.3.	<i>Chip IoT (Esp8266)</i> .....	31
1.5.4.4.	<i>Módulo reconocimiento de voz Elechouse</i> .....	32

<b>1.6.</b>	<b>Fuentes de alimentación Arduino .....</b>	<b>33</b>
<b>1.6.1.</b>	<b>Regulador de voltaje.....</b>	<b>33</b>
<b>1.6.2.</b>	<b>Voltaje regulado .....</b>	<b>33</b>
<b>1.6.3.</b>	<b>Opciones de alimentación .....</b>	<b>34</b>
<b>1.6.3.1.</b>	<b>Pila de 9V.....</b>	<b>34</b>
<b>1.6.3.2.</b>	<b>Baterías recargables AA .....</b>	<b>34</b>
<b>1.6.3.3.</b>	<b>Baterías de litio.....</b>	<b>35</b>
<b>1.6.3.4.</b>	<b>Batería de polímero litio (LiPo).....</b>	<b>36</b>
<b>1.7.</b>	<b>Plataforma Blynk .....</b>	<b>37</b>
<b>1.7.1.</b>	<b>Introducción .....</b>	<b>37</b>
<b>1.7.2.</b>	<b>Funcionamiento .....</b>	<b>38</b>
<b>1.8.</b>	<b>Google Assistant e IFTTT .....</b>	<b>39</b>
<b>1.8.1.</b>	<b>Google Assistant.....</b>	<b>39</b>
<b>1.8.2.</b>	<b>IFTTT .....</b>	<b>40</b>

## CAPÍTULO II

<b>2.</b>	<b>MARCO METODOLÓGICO.....</b>	<b>42</b>
<b>2.1.</b>	<b>Metodología para el diseño de un sistema de seguridad para una vivienda ....</b>	<b>42</b>
<b>2.2.</b>	<b>Etapas de planificación.....</b>	<b>43</b>
<b>2.2.1.</b>	<b>Requerimientos del usuario .....</b>	<b>43</b>
<b>2.2.2.</b>	<b>Planificación.....</b>	<b>45</b>
<b>2.2.3.</b>	<b>Selección y ubicación de equipos.....</b>	<b>45</b>
<b>2.2.3.1.</b>	<b>Selección de equipos .....</b>	<b>46</b>
<b>2.2.3.2.</b>	<b>Ubicación de equipos.....</b>	<b>49</b>
<b>2.2.4.</b>	<b>Selección de placas de desarrollo, módulos y circuitos.....</b>	<b>53</b>
<b>2.2.5.</b>	<b>Características fuentes de alimentación.....</b>	<b>55</b>
<b>2.2.5.1.</b>	<b>Fuente de alimentación DC .....</b>	<b>56</b>

2.2.5.2.	<i>Batería LiPo</i> .....	56
2.2.6.	<i>Características de software para el desarrollo de la aplicación</i> .....	56
<b>2.3.</b>	<b>Etapa de desarrollo</b> .....	<b>57</b>
2.3.1.	<i>Desarrollo y disposición de Hardware</i> .....	57
2.3.1.1.	<i>Arduino mega 2650 y módulos de desarrollo</i> .....	67
2.3.2.	<i>Conexión equipos</i> .....	69
2.3.2.1.	<i>Conexión cámaras, NVR, Router y Central de Gestión</i> .....	72
2.3.2.2.	<i>Conexión Cerca eléctrica, electrificador</i> .....	72
2.3.3.	<i>Desarrollo de Software</i> .....	75
2.3.3.1.	<i>Vinculación placa arduino mega 2650 con el servidor Blynk</i> .....	77
2.3.3.2.	<i>Programación Arduino IDE</i> .....	79
2.3.3.3.	<i>Programación módulo de reconocimiento de voz</i> .....	82
2.3.3.4.	<i>Algoritmo central de sensores</i> .....	84
2.3.3.5.	<i>Desarrollo aplicación móvil</i> .....	86
2.3.3.6.	<i>Integración de servicios a través de IFTTT</i> .....	88
<b>2.4.</b>	<b>Etapa de implementación</b> .....	<b>90</b>
2.4.1.	<i>Cerca eléctrica y perímetro de la vivienda</i> .....	90
2.4.2.	<i>Sistema de video vigilancia</i> .....	91
2.4.3.	<i>Centrales de gestión domótica</i> .....	94

### CAPÍTULO III

<b>3.</b>	<b>ANÁLISIS Y RESULTADOS</b> .....	<b>98</b>
3.1.	<b>Determinación de indicadores</b> .....	<b>98</b>
3.1.1.	<i>Número de Pruebas</i> .....	98
3.2.	<b>Prueba de confiabilidad del sistema</b> .....	<b>99</b>
3.3.	<b>Prueba de rapidez del sistema</b> .....	<b>102</b>
3.3.1.1.	<i>Prueba de normalidad a las distribuciones</i> .....	104
3.3.1.2.	<i>Prueba de homogeneidad de varianzas</i> .....	105

3.3.1.3.	<i>Prueba T-para muestras independientes.....</i>	105
<b>3.4.</b>	<b>Prueba de precisión del sistema .....</b>	<b>107</b>
3.4.1.1.	<i>Prueba de normalidad de distribución.....</i>	109
3.4.1.2.	<i>Prueba de homogeneidad de varianzas .....</i>	109
3.4.1.3.	<i>Prueba T-para muestras independientes.....</i>	110
3.4.1.4.	<i>Comparación valores de tensión.....</i>	112
3.4.1.5.	<i>Cálculo error absoluto.....</i>	114
<b>3.5.</b>	<b>Análisis de costos .....</b>	<b>116</b>
3.5.1.	<i>Costos materiales y herramientas .....</i>	117
3.5.2.	<i>Costo de equipos .....</i>	117
3.5.3.	<i>Costo de implementación .....</i>	118
3.5.4.	<i>Costo total del proyecto .....</i>	118
	<b>CONCLUSIONES.....</b>	<b>119</b>
	<b>RECOMENDACIONES.....</b>	<b>120</b>
	<b>BIBLIOGRAFÍA</b>	
	<b>ANEXOS</b>	

## ÍNDICE DE ABREVIATURAS

<b>A</b>	Amperios
<b>cm</b>	Centímetros
<b>Cat</b>	Categoría
<b>dB</b>	Decibel
<b>DC</b>	Corriente Directa
<b>E/S</b>	Entrada o salida de datos
<b>ESPOCH</b>	Escuela Superior Politécnica de Chimborazo
<b>GND</b>	Tierra
<b>IEEE</b>	Instituto de Ingeniería Eléctrica y Electrónica
<b>IN</b>	Entrada
<b>In</b>	Pulgadas
<b>IoT</b>	Internet de las cosas
<b>IP</b>	Internet Protocol
<b>Lan</b>	Local area network
<b>Mbps</b>	Mega bits por segundo
<b>mm</b>	Milímetros
<b>m</b>	Metros
<b>ms</b>	Mili segundos
<b>NVR</b>	Network Video Recorder
<b>OUT</b>	Salida
<b>TCP</b>	Transmission Control Protocol
<b>s</b>	Segundos
<b>UTP</b>	Par trenzado no blindado
<b>USB</b>	Universal Serial Bus
<b>V</b>	Voltios
<b>Vin</b>	Voltaje de entrada
<b>Vout</b>	Voltaje de salida
<b>Wan</b>	Wide area network
<b>Wi-Fi</b>	Wireless Fidelity
<b>Wps</b>	Wi-Fi protect setup

## ÍNDICE DE TABLAS

<b>Tabla 1-1:</b> Valores nominales relés de bajo voltaje.....	19
<b>Tabla 1-2:</b> Requerimientos del usuario para el sistema de seguridad.....	41
<b>Tabla 2-2:</b> Identificación planos vivienda/ áreas de riesgo / fuentes de alimentación.....	41
<b>Tabla 3-2:</b> Características Router D-Link.....	47
<b>Tabla 4-2:</b> Ubicación de equipos.....	48
<b>Tabla 5-2:</b> Aspectos a tomar en cuenta.....	49
<b>Tabla 6-2:</b> Requerimientos en placa módulo de relés.....	51
<b>Tabla 7-2:</b> Requerimientos módulos: Ethernet enc28j60 – Reconocimiento de voz elechouse...	52
<b>Tabla 8-2:</b> Características placas de desarrollo.....	53
<b>Tabla 9-2:</b> Requerimientos totales.....	53
<b>Tabla 10-2:</b> Asignación de funciones pines Arduino Mega 2650.....	65
<b>Tabla 11-2:</b> Conexión Arduino Mega 2650 – Módulo reconocimiento de voz.....	65
<b>Tabla 12-2:</b> Distribución Pines Arduino Mega 2650 – Módulo Enc28j60.....	66
<b>Tabla 13-2:</b> Distribución Pines Arduino Mega 2650 – Módulo relés.....	67
<b>Tabla 14-2:</b> Librerías incluidas en la IDE de programación arduino.....	75
<b>Tabla 15-2:</b> Anomalías a tomar en cuenta en el diseño.....	76
<b>Tabla 16-2:</b> Entradas y salidas del sistema.....	93
<b>Tabla 1-3:</b> Desempeño esperado en un sistema de seguridad.....	96
<b>Tabla 2-3:</b> Muestra prueba de confiabilidad.....	98
<b>Tabla 3-3:</b> Contingencia: Recepción de datos.....	99
<b>Tabla 4-3:</b> Pruebas de chi-cuadrado.....	100
<b>Tabla 5-3:</b> Muestra rapidez del sistema.....	101
<b>Tabla 6-3:</b> Pruebas de normalidad.....	102
<b>Tabla 7-3:</b> Prueba de homogeneidad de varianzas.....	103
<b>Tabla 8-3:</b> Prueba T para la igualdad de medias.....	104
<b>Tabla 9-3:</b> Muestra precisión.....	106
<b>Tabla 10-3:</b> Pruebas de normalidad.....	107
<b>Tabla 11-3:</b> Prueba de homogeneidad de varianzas.....	108
<b>Tabla 12-3:</b> Prueba de muestras independientes.....	108
<b>Tabla 13-3:</b> Comparación voltajes cerca eléctrica.....	110
<b>Tabla 14-3:</b> Resumen estadístico sistema diseñado.....	111
<b>Tabla 15-3:</b> Resumen estadístico voltímetro alta tensión.....	111

<b>Tabla 16-3:</b> Diferencia sistema y voltímetro.....	113
<b>Tabla 17-3:</b> Resumen estadístico diferencia sistema-voltímetro.....	114
<b>Tabla 18-3:</b> Costos materiales y herramientas.....	115
<b>Tabla 19-3:</b> Costos de equipos.....	115
<b>Tabla 19-3:</b> Costos de implementación.....	116
<b>Tabla 20-3:</b> Costo total del proyecto.....	116

## ÍNDICE DE FIGURAS

<b>Figura 1-1:</b> Sistema de seguridad mediante huellas dactilares .....	4
<b>Figura 2-1:</b> Prototipo de Pope alarma electromagnética.....	5
<b>Figura 3-1:</b> Internet de las Cosas (IoT).....	6
<b>Figura 4-1:</b> Domótica: de un capricho para ricos a una herramienta de ahorro.....	7
<b>Figura 5-1:</b> Pasarela residencial.....	9
<b>Figura 6-1:</b> Central de gestión no programable .....	11
<b>Figura 7-1:</b> Control de electrodoméstico por relé.....	12
<b>Figura 8-1:</b> Sirena de alerta a 12V DC .....	14
<b>Figura 9-1:</b> Sensor de Presencia.....	14
<b>Figura 10-1:</b> Tarjeta de control sensor PIR.....	15
<b>Figura 11-1:</b> Esquema cerca eléctrica.....	16
<b>Figura 12-1:</b> Postes de 5 líneas .....	17
<b>Figura 13-1:</b> Tensor metálico de cerca .....	17
<b>Figura 14-1:</b> Aro de supervisión de cerca.....	18
<b>Figura 15-1:</b> Electrificador Hagroy i8 .....	19
<b>Figura 16-1:</b> Esquema circuito electrificador .....	19
<b>Figura 17-1:</b> Circuito electrificador .....	20
<b>Figura 18-1:</b> Cámara tipo tubo Hikvision .....	22
<b>Figura 19-1:</b> Planificación distribución de cámaras.....	23
<b>Figura 20-1:</b> Relación entre apertura de lente, área y longitud.....	24
<b>Figura 21-1:</b> Cámara ip para exteriores .....	25
<b>Figura 22-1:</b> Cámara PTZ para Exterior .....	26
<b>Figura 23-1:</b> Central de monitoreo.....	27
<b>Figura 24-1:</b> Arduino Started Kit.....	28
<b>Figura 25-1:</b> Módulo de expansión Ethernet ENC28J60.....	30
<b>Figura 26-1:</b> Módulo de relés.....	30
<b>Figura 27-1:</b> Esp8266 E01 .....	31
<b>Figura 28-1:</b> Módulo reconocimiento de voz Elechouse .....	32
<b>Figura 29-1:</b> Comparación tramas de datos .....	32
<b>Figura 30-1:</b> Pila de 9V.....	34
<b>Figura 31-1:</b> Baterías recargables AA .....	35
<b>Figura 32-1:</b> Batería de litio.....	36
<b>Figura 33-1:</b> Batería LiPo .....	37
<b>Figura 34-1:</b> Interfaz aplicación Blynk.....	38
<b>Figura 35-1:</b> Aplicación desarrollada con Blynk .....	38

<b>Figura 36-1:</b> Asistente de Google .....	39
<b>Figura 37-1:</b> Plataforma IFTTT .....	40
<b>Figura 1-2:</b> Plano parcial vivienda.....	42
<b>Figura 2-2:</b> Vista isométrica patio posterior.....	43
<b>Figura 3-2:</b> Angulo de visión cámaras de seguridad.....	44
<b>Figura 4-2:</b> Área cobertura sensor.....	45
<b>Figura 5-2:</b> NVR HIKVISION DS 7108.....	46
<b>Figura 6-2:</b> Ubicación tentativa cerca eléctrica.....	48
<b>Figura 7-2:</b> Ubicación tentativa cámaras de seguridad.....	49
<b>Figura 8-2:</b> Ubicación tentativa sensor de movimiento.....	50
<b>Figura 9-2:</b> Ubicación tentativa electrificador y central de control domótico.....	50
<b>Figura 10-2:</b> Circuito Divisor de Voltaje.....	57
<b>Figura 11-2:</b> Pulso de alto voltaje cerca eléctrica.....	57
<b>Figura 12-2:</b> Pulsos tiempo de cesado.....	58
<b>Figura 15-2:</b> Especificaciones opto acoplador 4N25.....	59
<b>Figura 14-2:</b> Circuito sincronización y cruce por cero de la cerca eléctrica.....	59
<b>Figura 15-2:</b> Sincronización: Disparo alto voltaje - Tiempo de censado.....	60
<b>Figura 16-2:</b> Hoja de datos 2W10 puente de diodos.....	61
<b>Figura 17-2:</b> Hoja de datos diodos Zener.....	62
<b>Figura 18-2:</b> Circuito reductor sensor de movimiento.....	63
<b>Figura 19-2:</b> Fuente 1.5A, alimentación placas y módulos.....	63
<b>Figura 20-2:</b> Fuente 1.5 A - Circuito lectura y cruce por cero cerca eléctrica.....	64
<b>Figura 21-2:</b> Circuito de distribución de energía.....	64
<b>Figura 22-2:</b> Conexión módulo reconocimiento de voz elechouse.....	65
<b>Figura 23-2:</b> Conexión Arduino Mega 2650 – Modulo Enc28j60.....	66
<b>Figura 24-2:</b> Prueba conexión módulo Enc28j60.....	66
<b>Figura 25-2:</b> Conexión Arduino Mega 2650 – Módulo Relés.....	67
<b>Figura 26-21:</b> Conexión cámaras, NVR, Router y central de gestión.....	70
<b>Figura 27-2:</b> Circuito Electrificador.....	71
<b>Figura 28-2:</b> Conexión electrificador y puesta a tierra.....	75
<b>Figura 29-2:</b> Desarrollo de Software .....	76
<b>Figura 30-2:</b> Librería Blynk.....	77
<b>Figura 31-2:</b> Creación nuevo proyecto Blynk.....	77

<b>Figura 32-2:</b> Auth Token proporcionado por Blynk.....	78
<b>Figura 33-2:</b> Conexión con el servidor Blynk a través del módulo ENC28j60.....	78
<b>Figura 34-2:</b> Declaración entradas directas.....	79
<b>Figura 35-2:</b> Declaración salidas módulo relés.....	79
<b>Figura 36-2:</b> Prueba lectura de sensores y voltaje de cerca.....	80
<b>Figura 37-2:</b> Función Timer Blynk.....	80
<b>Figura 38-2:</b> Código entrenamiento reconocimiento de voz.....	81
<b>Figura 39-2:</b> Entrenamiento reconocimiento de voz.....	82
<b>Figura 40-2:</b> Entrenamiento: Grabación de comandos.....	83
<b>Figura 41-2:</b> Diagrama de flujo esp8266 servidor.....	84
<b>Figura 42-2:</b> Diagrama de flujo esp8266 cliente.....	85
<b>Figura 43-2:</b> Espacio de trabajo Blynk.....	86
<b>Figura 44-2:</b> Lista de widgets disponibles.....	86
<b>Figura 45-2:</b> Selección variables a utilizar.....	87
<b>Figura 46-2:</b> Configuración Webhook.....	88
<b>Figura 47-2:</b> Cerca eléctrica en el perímetro de la vivienda.....	89
<b>Figura 48-2:</b> Varilla de puesta a tierra.....	90
<b>Figura 49-2:</b> Entradas central de gestión.....	90
<b>Figura 50-2:</b> Conexión cable PoE – cámaras de video.....	91
<b>Figura 51-2:</b> Cámaras de video vigilancia con cajas de revisión.....	92
<b>Figura 52-2:</b> Configuración NVR – Hik-Connect.....	92
<b>Figura 53-2:</b> Detección de movimiento cámaras de video vigilancia.....	93
<b>Figura 54-2:</b> Señal de salida de alerta NVR.....	93
<b>Figura 55-2:</b> Entradas central de gestión.....	94
<b>Figura 56-2:</b> Placa de manejo de señales de voltaje – fuente de alimentación.....	94
<b>Figura 57-2:</b> Central de gestión de sensores.....	95
<b>Figura 58-2:</b> Clientes IoT para sensores.....	95
<b>Figura 59-2:</b> Ubicación y cobertura central de sensores.....	96
<b>Figura 1-3:</b> Reconocimiento comandos de voz google assistant.....	96
<b>Figura 2-3:</b> Reconocimiento comandos de voz módulo elechouse.....	101
<b>Figura 3-3:</b> Detección de movimiento cámara.....	103
<b>Figura 4-3:</b> Recepción notificación de alerta. ....	105
<b>Figura 5-3:</b> Medida en voltios, tensión de la cerca eléctrica.....	106

## ÍNDICE DE GRÁFICOS

<b>Gráfico 1-2:</b> Etapas del diseño de un sistema de seguridad.....	40
<b>Gráfico 2-2:</b> Esquema conexión equipos y simbología domótica empleada.....	68
<b>Gráfico 3-2:</b> Simbología domótica empleada.....	69
<b>Gráfico 1-3:</b> Dispersión de voltajes. Sistema de puesta a tierra sin varilla.....	109
<b>Gráfico 2-3:</b> Dispersión de voltajes. Sistema de puesta a tierra con varilla.....	109
<b>Gráfico 3-3:</b> Dispersión de medidas de voltaje.....	112
<b>Gráfico 4-3:</b> Error entre sistema diseñado y voltímetro de alta voltaje.....	114

## RESUMEN

En el presente trabajo de titulación se diseñó un sistema de seguridad para una vivienda mediante pasarela activada por voz y video para la empresa “Rio Solar Smart Energy” de la ciudad de Riobamba, ante la necesidad de iniciar con la producción de una línea domótica destinada a la seguridad del hogar. A través del método heurístico y experimental se hallaron las principales necesidades de seguridad de una vivienda promedio, de esta manera fue diseñado un sistema que permita de manera sencilla al usuario, el control de un cerca eléctrica, proporcione acceso a video vigilancia y comandos de voz, controlado por placas de desarrollo y un dispositivo móvil, que permite un fácil manejo del sistema, desde cualquier parte del mundo, mediante pasarela, y un servidor para la recepción y envío de datos. Se elaboró un plan de diseño basado en la capacidad de manejo de la información de las placas de desarrollo, sin verse comprometido los tiempos de respuesta, o la desconexión del servidor en el momento de manejo de la información requerida por el sistema de seguridad. Con un tiempo de respuesta promedio de 1,5 segundos desde que se genera una petición al servidor por parte del sistema de seguridad, hasta la recepción de la alerta en el dispositivo móvil. Se puede concluir que el diseño es funcional en referencia a sistemas comerciales con tiempos de respuesta que oscilan entre los 0,8 y 1,5 segundos. Finalmente el sistema diseñado posee un costo de 962.90 dólares, incluyendo horas de programación, y costo de instalación los cuales son asumidos por la empresa, este costo está muy por debajo de sistemas comerciales disponibles, los cuales bordean precios de hasta 1244 dólares. Este diseño está enfocado a la seguridad del hogar, y se recomienda que sea instalado por personal previamente capacitado.

**Palabras clave:** <TECNOLOGÍA Y CIENCIAS DE LA INGENIERÍA>, <DOMÓTICA>, <SISTEMAS SEGURIDAD>, <SEGURIDAD EN VIVIENDA>, <CERCA ELÉCTRICA>, <PASARELA DE ACCESO>, <OPTIMIZACIÓN DE TIEMPO DE RESPUESTA>.

## **ABSTRACT**

A security system for a house was designed in the current research work by means of a voice-activated gateway and video record for "Rio Solar Smart Energy" company in Riobamba city. Faced with the necessity of introducing a domotic line production intended to improve home's safety. Through the heuristic and experimental method, the main security needs of an average home were found, in this way a system that allows the user the easily control of electric fence, provide access to a video security system and voice commands was designed. This is controlled by processing boards and a mobile device that allows an easy operation of the system from anywhere in the world by using a gateway and a server for receiving and sending data. A design plan was developed based on the information management capacity of the processing boards, without being compromised the response time or the disconnection of the server at the time of handling the required information by the security system. With an average response time of 1.5 seconds since a request is generated to the server by the security system until the alert is received on the mobile device. It can be concluded that the design is functional in reference to commercial systems with response times ranging between 0.8 and 1.5 seconds. Finally, the designed system has a cost of \$ 150 in the control center and another \$ 300 in sensors and actuators. This cost is far below commercially available systems which have prices of up to \$ 1244. This design is focused on home security and it is recommended that it be installed by previously trained personnel.

**KEYWORDS:** <TECHNOLOGY AND ENGINEERING SCIENCES>, <AUTOMATION>, <SECURITY SYSTEM>, <HOME SECURITY>, <ELECTRIC FENCE>, <GATEWAY>, <OPTIMIZATION OF RESPONSE TIME>.

# INTRODUCCIÓN

## **Antecedentes**

El evidente avance de la tecnología, y en consecuencia de las necesidades que la misma va cubriendo, ha provocado que los recursos empleados para su desarrollo sean cada vez mayores, así mismo la creciente demanda de tecnologías destinadas a una infinidad de tareas, ha permitido a los seres humanos desarrollar su vida diaria con mayor facilidad, optimizando tiempo en la ejecución de tareas, ahorrando recursos y sobre todo permitiendo la generación de nuevos campos de investigación, para el desarrollo de nuevas áreas para la aplicación de la tecnología.

Dentro de estas áreas, particularmente la domótica es una de las que mejor se ha desarrollado, no solo por su reciente auge de empleo en el sector inmobiliario, sino por la constante evolución en la seguridad del hogar y confort.

En la tesis “Estudio e implementación de Domótica activado por voz y comunicación zigbee”.(Alvarado Cadena & Arévalo Coronel, 2010), se menciona los protocolos existentes para envío y recepción inalámbrico de datos, que brindan confiabilidad y rapidez, sin embargo las tecnologías empleadas en dicho documento si bien aún son de mucha utilidad en unos campos, dentro de la domótica han sido remplazados por dispositivos con mayor confiabilidad y mejores prestaciones.

En la tesis “Diseño e implementación de un sistema domótico para el control y seguridad en tiempo real vía teléfono celular”(Urrutia Carrasco, 2015) se emplea tecnología gsm para la transmisión de comandos para control el sistema, como se puede evidenciar existen ya proyectos relacionados con el tema propuesto, sin embargo, no han sido diseñados para la producción a gran escala, para una comercialización en compañías de seguridad.

El estudio de los métodos y sistemas de seguridad en el hogar, ha sido objeto de investigación en los últimos años, dichas investigaciones han permitido desarrollar varias aplicaciones y dar varios usos a los beneficios que estos sistemas ofrecen. En este caso se describirá un sistema domótico de seguridad, cuyo uso resulta cada vez más habitual, por el sinnúmero de beneficios que ofrece frente a artefactos no automatizados.

El principio de estos sistemas, está basado en el uso de varios artefactos de seguridad ya existentes; centralizar su monitoreo y manipulación a través de placas de desarrollo de bajo costo y con un alto nivel de accesibilidad para el manejo de la información; brinda un mayor grado de confianza al usuario, quien podrá de manera sencilla tener acceso a cada uno de los componentes configurados en el sistema de seguridad.

“Rio Solar Smart Energy” es una empresa en expansión dedicada a la venta e instalación de calentadores solares en la ciudad de Riobamba, con proyección a futuro de comercializar sistemas domóticos de seguridad de alta calidad y de bajo costo. Es por esto que resulta necesario el diseño de un sistema domótico para la seguridad de una vivienda mediante pasarela activada por voz y video.

### **Justificación**

La realización de este proyecto está enfocado al diseño un sistema de seguridad para una futura comercialización con la empresa “Rio Solar Smart Energy”. Mediante la aplicación del conocimiento adquirido a lo largo de la carrera.

Después de analizar las falencias que presentan los sistemas de seguridad actuales, se conoce que: los elevados costos de comercialización de estos sistemas presentan un gran limitante para la adquisición del ciudadano promedio, la gran limitación de su funcionamiento exclusivamente a periféricos de la misma marca, obliga al consumidor final a adquirir accesorios que elevan el costo por obtener un sistema con mayor confiabilidad, no todos los proveedores de sistemas de seguridad realizan las instalaciones correctamente, reduciendo así la confiabilidad del mismo.

El presente sistema, está enfocado en la optimización de recursos económicos empleados por el usuario y la maximización del rendimiento entregado por el sistema, brindando, seguridad y centralización en el manejo de cada uno de los periféricos empleados en las diferentes áreas de la vivienda.

Actualmente la empresa Rio Solar Smart Energy tiene como plan prioritario la inversión de recursos, humanos y económicos, en el desarrollo de un dispositivo que dé solución a los problemas anteriormente planteados, para una futura comercialización de un sistema compacto de seguridad domótico que cumpla con las expectativas de los usuarios y de la empresa.

Este sistema domótico pretende trabajar conjuntamente con el cercado eléctrico, cámaras ip, y sensores, todo conectado a una central, que, ante cualquier tipo de anomalía, permitirá al usuario monitorear su hogar desde las cámaras, permitiendo así conocer en tiempo real, que está sucediendo y de esta manera evitar posibles incidentes que pudieren suceder.

El diseño de este dispositivo resultara beneficioso para la empresa y su futura línea de productos de seguridad domótica, así como para los futuros usuarios, puesto que ofrecerá mayor confiabilidad a los hogares, y de la misma forma una línea domótica sin duda posicionará a la empresa en un nivel alto de competitividad.

## **Objetivos**

### **Objetivo General:**

Diseñar un sistema para seguridad a una vivienda mediante pasarela activada por voz y video, para la empresa “Rio Solar Smart Energy” de la ciudad de Riobamba.

### **Objetivos Específicos:**

- Implementar un cercado eléctrico que trabaje conjuntamente con sensores y cámaras, para la detección de anomalías en el sistema de seguridad.
- Implementar un sistema de comandos por voz, capaz de dar acceso al sistema de seguridad.
- Realizar la captura y transmisión de datos en tiempo real a un dispositivo móvil que permita monitorear los acontecimientos en la vivienda.
- Validar el correcto funcionamiento del sistema a través de pruebas de tiempos de respuesta.

# CAPÍTULO I

## 1. MARCO TEÓRICO REFERENCIAL

### 1.1. Sistema de seguridad

#### 1.1.1. Definición

Un sistema de seguridad es el conjunto de equipos, software, normas y/o la combinación de estos, reunidos en un determinado ambiente con el fin de resguardar a personas, software informáticos y bienes materiales que existen en un entorno, de trabajo, familiar, educativo, etc. Facilitando el manejo de situaciones de riesgo y posibles siniestros, presentes en dichos ambientes.



**Figura 1-1:** Sistema de seguridad mediante huellas dactilares

Fuente: <https://goo.gl/GHHPDi>

#### 1.1.2. Sistema de seguridad en el hogar

Son sistemas comprendidos por aparatos electrónicos y mecánicos que trabajan en conjunto, generalmente asistidos algún tipo de software informático que brindan acceso al usuario mediante pasarela, destinados a la protección de personas y bienes dentro del perímetro que comprende la propiedad del usuario y sus alrededores.

### 1.1.3. *Historia*

Hace más de 160 años cuando las personas confiaban su seguridad a animales domésticos, Russel Pope de Somerville plantea la primera instalación de alarma electromagnética, (Ver Figura 2-1) de sencillo funcionamiento pero de una enorme eficacia, pese a esto, tras haber comprado la patente de Pope, es Edwin Holmes el primer hombre en realizar instalaciones de alarmas a nivel industrial, junto a una campaña publicitaria que aprovechaba el auge del telégrafo para publicitar su alarma.



**Figura 2-1:** Prototipo de Pope alarma electromagnética

**Fuente:** <https://goo.gl/reya8d>

Tras la ingeniosa campaña publicitaria, la empresa llega a su mayor punto de éxito con Edwin T. Holmes, quien ingeniosamente utilizó el cableado de las redes telefónicas para lograr un sistema centralizado de sus alarmas.

Pronto el sistema telefónico, permite interconectar las casas de diferentes distritos, de esta manera si en una casa se dispara una alarma, los demás sabrían que en aquel lugar posiblemente está ocurriendo un robo; pero no fue suficiente para mantener a las personas tranquilas, hacía falta un sistema de llamadas de emergencia para poder proporcionar asistencia al usuario, es así que cada distrito contaba con una central de llamadas de emergencia, que servía para poder asistir a los usuarios en caso de algún siniestro en las viviendas.

En la actualidad existen dispositivos basados en los sistemas mecánicos ideados hace más de 160 años, que combinados con la electrónica y la centralización de la información crean sistemas

capaces de cubrir prácticamente cualquier área, complementándose día tras día con los avances tecnológicos, especialmente con el llamado internet de las cosas (IoT) (Terrance Huges, 2012), permitiendo una monitorización de los eventos que ocurren en el área protegida por el sistema, desde cualquier parte del mundo.



**Figura 3-1:** Internet de las Cosas (IoT)

Fuente: <https://goo.gl/SJ2sQ3>

#### ***1.1.4. Elementos en un sistema de seguridad***

Con el desarrollo tecnológico los sistemas de seguridad han pasado de ser pequeños sistemas de control simples a convertirse en sistemas constituidos por una gran variedad de elementos, que trabajan conjuntamente y de forma coordinada. Actualmente los sistemas de seguridad están compuestos por elementos activos y pasivos.

##### ***1.1.4.1. Elementos Activos***

En los sistemas de seguridad un elemento activo se entiende como aquel que pretende evitar un daño o intrusión en el área protegida por el sistema de seguridad por medio de un comportamiento de acción-reacción, como alarmas, electroválvulas de agua/gas, etc.

##### ***1.1.4.2. Elementos Pasivos***

Son elementos capaces de dar aviso a un sistema de seguridad acerca de una intrusión o daño, pero no son capaces de realizar ninguna acción para evitar que la acción en proceso se cumpla, generalmente en esta categoría están la gran variedad de sensores que podemos encontrar en el mercado dirigidos a la seguridad, como sensores PIR, magnéticos, voltímetros, amperímetros, etc.

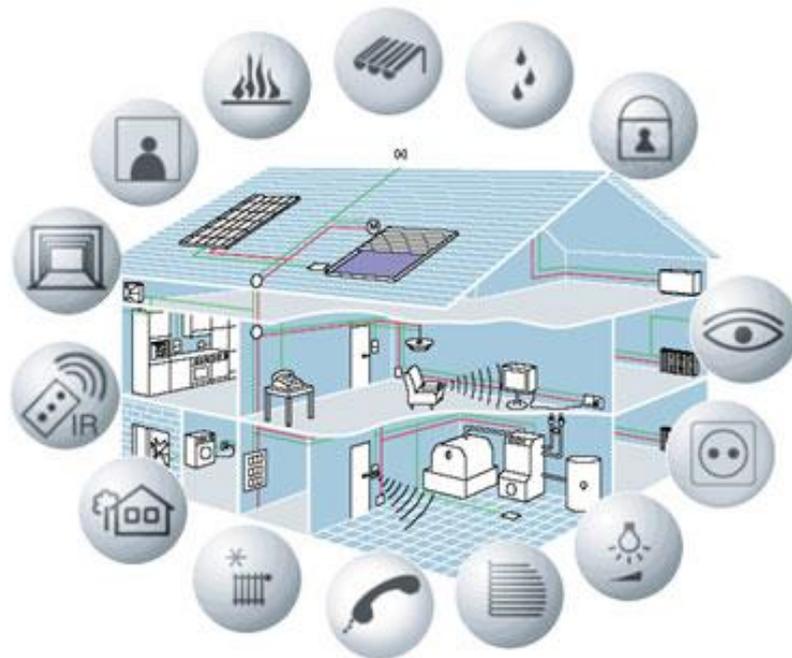
## 1.2. Domótica

### 1.2.1. Definición

Es un conjunto de disciplinas como la electricidad, informática, electrónica, telecomunicaciones, etc. Dirigidas al manejo y gestión de elementos de seguridad pasivos y activos existentes en un entorno familiar u oficina con la finalidad de brindar seguridad, confort y bien estar al usuario (Luzuriaga Godoy, 2016).

### 1.2.2. Aplicación de la domótica

Existe un gran campo aplicativo en la cual se puede implementar proyectos domóticos, en la actualidad se encuentra en auge el uso de estos sistemas para convertir espacios de uso común en sitios autónomos que brinden seguridad y confort, sea a través de la monitorización de sensores y el control de actuadores, realizando tareas como climatización de ambientes o detectando posibles siniestros como fugas de gas o incendios, etc.



**Figura 4-1:** Domótica: de un capricho para ricos a una herramienta de ahorro

Fuente: <https://goo.gl/ao9LN5>

El mejorar la calidad de vida de los usuarios es una prioridad en la domótica, es por esto, que, es común observar estos sistemas en áreas como el hogar, la oficina, ambientes de trabajo, en el

sector industrial y edificios, para estos últimos se emplea la disciplina Inmótica, que es la misma domótica enfocada a grandes estructuras o áreas.(Maldonado Pérez & Chacha Meléndrez, 2017)

### ***1.2.3. Ventajas de la Domótica***

Existe un gran número de ventajas que la domótica brinda a sus usuarios, dependiendo el tipo de instalación y uso de la misma, entre las principales tenemos:

- Ahorro económico en cuanto se refiere al control del consumo de la energía.
- Minimización significativa de las posibilidades de ser víctima de situaciones de riesgo en el hogar y sus alrededores.
- Control y acceso remoto las 24 horas del día a dispositivos conectados a la red domótica del hogar.
- Autonomía del hogar para brindar mayor confort al usuario.
- Autonomía del hogar para la detección de posibles fallos en el sistema alertando tempranamente al usuario y evitando daños mayores en el sistema.

### ***1.2.4. Desventajas de la Domótica***

A pesar de los años que la domótica lleva en uso dentro de la vida cotidiana de las personas, aun es una disciplina en desarrollo que presenta desventajas a la hora de su uso e implementación, como:

- Al ser una disciplina relativamente joven, no todas las personas son capaces de manejar con total facilidad los diferentes dispositivos y programas que componen una red domótica.
- No todos los sistemas domóticos están diseñados con un entorno amigable con el usuario, en caso de posibles fallos, la mayoría de sistemas actualmente existentes requieren de atención técnica especializada de parte del fabricante del sistema, debido a la limitada información acerca de los productos.
- Los costos de instalación de una red de dispositivos domóticos pueden ser medianamente elevados.
- La gran limitación que existe por parte de los fabricantes de dispositivos domóticos, obliga al usuario a consumir productos de una sola marca.

### 1.2.5. Domótica y Pasarela Residencial

El gran número de facilidades que brinda la domótica para el manejo y gestión de dispositivos en los diferentes entornos no es suficiente para un completo desarrollo de un sistema de seguridad, pues, adicionalmente de la gran cantidad de dispositivos presentes en una red domótica, es necesario un sistema que permita la comunicación multidireccional entre todos los dispositivos que conforman la red domótica, y el usuario final, independientemente del lugar en el que se encuentre.

La pasarela residencial es un conjunto de hardware / software que trabaja con normas y protocolos establecidos por el usuario para la conexión entre dispositivos, permitiendo la interacción de datos, aportando con un nivel mayor de accesibilidad en el sistema en el cual está trabajando. Dentro de las redes domóticas, la pasarela residencial permite la comunicación, el acceso y control de datos del sistema desde cualquier parte.



**Figura 5-1:** Pasarela residencial

Fuente: <https://goo.gl/1ZPxqt>

#### 1.2.5.1. Características de la Pasarela residencial

Una pasarela residencial aparte de aportar la fácil accesibilidad a los diferentes dispositivos que componen una red, debe estar sujeta a ciertas condiciones como:

## **Fácil instalación**

En lo posible el dispositivo debería ser plug&play, o en su defecto evitar grandes configuraciones que confundan o entorpezcan el trabajo del usuario, durante la instalación y funcionamiento del sistema.

## **Asistencia y/o configuración remota**

En caso que el dispositivo requiera de configuraciones adicionales en beneficio del usuario estas deben ser realizables remotamente sin interrumpir el trabajo del consumidor, también debe existir actualizaciones constantes del software, para una mejora del rendimiento y compatibilidad con nuevos dispositivos.

## **Compatibilidad con redes**

Debe ser compatible con las redes para transmisión de datos más importantes en el mercado, (Ethernet, HomePlug, HomePNA, HomeRF) o en su defecto dar las facilidades para el dispositivo pueda comunicarse con estas redes indirectamente, sea mediante equipos, software de compatibilidad o por accesorios incluidos con el dispositivo principal.

## **Confiabilidad de Software**

Las pasarelas residenciales deben poseer un sistema robusto capaz de hacer frente a posibles ataques informáticos, que pretendan vulnerar la seguridad e interferir o interrumpir el proceso de comunicación entre los dispositivos conectados a la red y el usuario o a su vez capturando datos importantes del sistema como claves de acceso o simplemente inhabilitándolo a través de instrucción en la pasarela.

## **Confiabilidad de Hardware**

Debe poseer la capacidad necesaria para el manejo de todas las solicitudes de envío y recepción de datos del sistema, garantizando suficiente memoria o velocidad de procesamiento para lograr un desempeño óptimo, sufriendo el desgaste mínimo posible de sus componentes y garantizando mayor tiempo de vida útil.

### 1.2.6. Elementos de un sistema domótico

Un sistema domótico está compuesto por 5 elementos principales, los cuales se encargan de la gestión y el funcionamiento correcto de todo el sistema a través del trabajo en conjunto con periféricos y accesorios.

#### 1.2.6.1. Central de Gestión

Son equipos encargados de la gestión de cada uno de los dispositivos conectados al sistema domótico, brinda también un fácil acceso al usuario para la visualización de los eventos ocurridos en las áreas protegidas.

Existen diferentes tipos de centrales de gestión, que pueden ser instaladas en paredes, cuartos eléctricos o como es el caso de los sistemas actuales no son en sí un dispositivo físico, sino un software que el proveedor proporciona al usuario el cual puede ser instalado en un computador o cualquier dispositivo móvil.

A rasgos generales podemos distinguir entre dos grandes centrales de gestión.



**Figura 6-1:** Central de gestión no programable

Fuente: <https://goo.gl/vC11aS>

Las centrales diseñadas por los mismos fabricantes de productos domóticos, que incorporan un software específico y una programación determinada, la cual se limita a realizar operaciones dentro de la programación preinstalada en el dispositivo y permite la poca o nula modificación de la programación para la gestión del sistema de seguridad, además este tipo de centrales tienen una compatibilidad limitada a los productos del mismo fabricante, a pesar de la amplia gama de

productos que ofrece cada marca, el problema para la mayoría de consumidores radica en el costo de dichos accesorios que día a día son más imprescindibles en un sistema domótico.

Y las centrales autómatas programables, que no están dirigidas solo al control domótico , sino a la gestión de un sin número de tareas a nivel familiar, empresarial o industrial, se caracterizan por el uso de protocolos estandarizados para la comunicación con otros dispositivos, por lo cual no se limita a una sola marca en cuanto a periféricos , permitiendo la libre elección del usuario, sin embargo este tipo de centrales requieren de conocimientos acerca de programación por parte del usuario, para poder desarrollar un sistema de gestión óptimo.

#### 1.2.6.2. *Nodos*

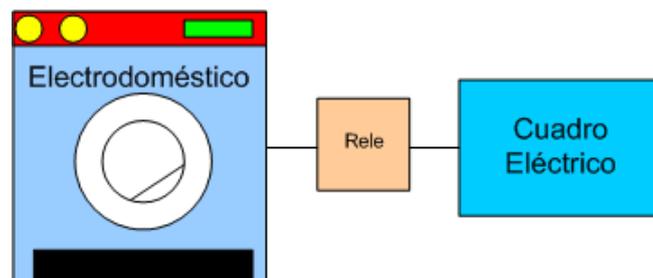
Es un punto en el cual dos objetos se encuentran ya sea para cambiar información o procesarla, dentro de la domótica un nodo es aquel punto en el cual se realiza la comunicación de los dispositivos existentes en el sistema, un nodo realiza una función de pasarela primitiva.

#### 1.2.6.3. *Actuadores*

Son dispositivos eléctricos, mecánicos y/o electrónicos también conocidos como elementos de seguridad activos, comúnmente en un sistema domótico se puede hallar:

#### **Contactores o relés.**

Son de uso común dentro de la gestión domótica de dispositivos en instalaciones eléctricas convencionales, como circuitos de iluminación y fuerza, su principal uso es el aislamiento y protección de los circuitos de control.



**Figura 7-1:** Control de electrodoméstico por relé

Fuente: <https://goo.gl/giJgLL>

Evitando daños por sobre carga en cualquiera de los elementos asociados a la central de gestión. Poseen una vida útil determinada por número de activaciones, y generalmente son de tipo modulares para facilitar la identificación y reemplazo en caso de daño.

La elección del tipo de relé depende del tipo de carga que vaya a manejar el mismo, tomando en cuenta que el encargado de realizar el corte o paso de energía eléctrica hacia el dispositivo final es una bobina que se encuentra en la parte interna del relé que tiene una capacidad de carga específica. Normalmente deberíamos calcular la corriente que va a circular por el relé, sin embargo ya existen modelos estandarizados de relés clasificados por el voltaje a manejar emparejada con una corriente máxima.

**Tabla 2-1:** Valores nominales relés de bajo voltaje

<b>Voltaje</b>	<b>Amperaje</b>
5V	10 A
5V	20 A
5V	40 A
12V	10 A
12V	20 A
12V	40 A
24V	10 A
24V	22 A
24V	40 A

Realizado por: Guaranga, J. 2018

#### 1.2.6.4. Sirenas

Son dispositivos o programas en su mayoría conectados a sensores u otro dispositivo de seguridad, capaces de emitir un aviso de alerta al usuario, por medio de sonidos, notificaciones o encendido de todo tipo de actuadores en caso vulnerabilidad del sistema de seguridad, normalmente son de uso común sirenas sonoras, que emiten ruido a una intensidad entre 110 - 118 dB, que ayudan a alertar tanto al usuario como a los alrededores cercanos al siniestro.



**Figura 8-1:** Sirena de alerta a 12V DC

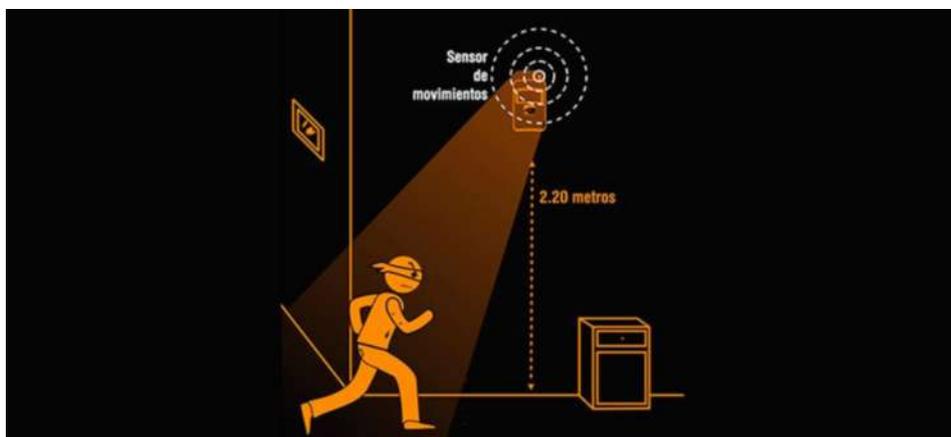
Fuente: <https://goo.gl/W9pMUc>

#### 1.2.6.5. Sensores

Son dispositivos electrónicos que según su construcción son capaces de detectar variaciones de magnitudes físicas que ocurren en el ambiente en el cual se encuentran, dentro de una instalación domótica actúan como dispositivos pasivos de entrada, encargados de la recopilación de información acerca de los acontecimientos que ocurren en la vivienda. Entre los sensores más comunes en una instalación domótica domiciliaria tenemos.

#### **Sensor de Presencia (PIR)**

Los sensores de presencia son dispositivos destinados a la detección de intrusos en una vivienda, por medio de variación en el volumen, o a través de luz infrarroja, existe gran variedad de estos sensores, con diferentes ángulos y distancias de detección, por lo que se requiere de un estudio previo para su ubicación, de esta manera lograr una eficiencia máxima y evitar posibles falsas alarmas (Ver Figura 9-1).



**Figura 9-1:** Sensor de Presencia

Fuente: <https://goo.gl/6UKbxo>

Generalmente en interiores se encuentran ubicados en la parte superior de una esquina de una habitación y en exteriores, depende del estudio de eficiencia para determinar su ubicación.

El componente principal de estos sensores es un piro eléctrico, que básicamente es un elemento que detecta variación de señal infrarroja recibida, el cual envía esta señal a un transistor de efecto de capo encargado de la amplificar la señal recibida.

Al detectar un cambio en su señal, esta es procesada por un circuito de control encargado de activar un relé, siempre que la señal cumpla con determinadas características como frecuencia, amplitud, duración, etc.



**Figura 10-1:** Tarjeta de control sensor PIR

Fuente: <https://goo.gl/sDsv2E>

Al encender por primera vez el sensor, este empieza a recibir una señal infrarroja emitida por todos los elementos presentes en ese momento en el ambiente donde fue instalado, sean estas paredes, plantas, adornos, etc. Al llegar a un punto de estabilidad y si detecta una variación en la emisión de señales infrarrojas detectara como un intruso, dando aviso a la central de gestión.

### **1.3. Cercas eléctricas domesticas**

#### **1.3.1. Definición**

Una cerca eléctrica es un elemento de seguridad activo/pasivo compuesto por (alambres, postes, sujetadores, tensores) que entregan energía eléctrica con voltajes entre 8000 – 17000V para cercas de protección domiciliarias y con voltajes superiores a los 30000 V para cercas enfocadas al control animal.

Una cerca eléctrica puede ser clasificada como un elemento activo/pasivo cuya finalidad es liberar una descarga sobre un intruso, provocando un estado de shock temporal, al detectar una unión entre su línea de corriente y cualquier punto a tierra. Generalmente están conectados a un sistema de alarma o central de monitorización que da aviso al propietario del sitio en el cual fue instalado, a la vez que ahuyenta al individuo que intentó vulnerar el sistema (Luzuriaga Godoy, 2016).

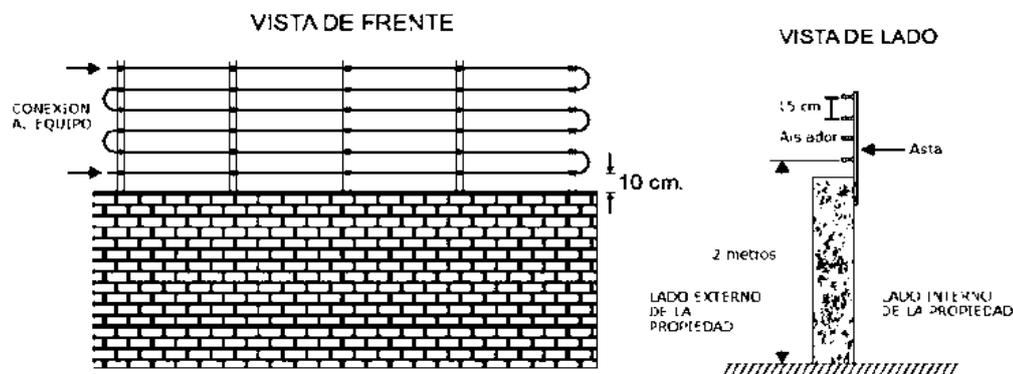
### 1.3.2. Composición y Funcionamiento

El cerco eléctrico reacciona ante una variación del comportamiento de sus alrededores, detectando el corte o toque de los alambres, gracias a una conexión a tierra que posee, es capaz de cerrar el circuito en estas circunstancias, enviando un pulso de alta tensión sobre el sujeto que pretende entrar en el área protegida.

Para su funcionamiento depende de un electrificador que se encarga de proporcionar la energía a los alambres del sistema, dependiendo del tipo y calidad del electrificador, se puede proporcionar suministro eléctrico hasta 5km en los electrificadores de gama media, sin que existan pérdidas significativas de energía. Un sistema de cercado eléctrico está compuesto por:

#### 1.3.2.1. Alambre con doble galvanización

Un conjunto de alambres con doble capa de galvanización, brinda una mayor capacidad de conducción del voltaje a lo largo del recorrido de la cerca, garantizando un óptimo funcionamiento, de la línea de conducción, como de la línea a tierra.



**Figura 11-1:** Esquema cerca eléctrica

Fuente: <https://goo.gl/Qwg6Ri>

El número de alambres en un cerco pueden variar de 3, 5, 7 hilos, con voltajes entre 8000 y 17000V dependiendo el tipo de electrificador a ser utilizado y de las necesidades del usuario.

### 1.3.2.2. Postes

Los postes utilizados en una cerca eléctrica son tubos elaborados de un material no conductor en los cuales van anclados los tensores. El tamaño del poste se elige de acuerdo al número de líneas que tendrá el cercado eléctrico, los postes van empotrados en la pared con una sujeción lo suficientemente fuerte para soportar la tensión de todo el cercado eléctrico.



**Figura 12-1:** Postes de 5 líneas

Fuente: <https://goo.gl/gcy7U5>

### 1.3.2.3. Tensores

Son instrumentos que van anclados a los postes que a su vez están conectados a los alambres con doble galvanización que recorren todo el perímetro a ser cercado.



**Figura 13-1:** Tensor metálico de cerca

Fuente: <https://goo.gl/aFqWi4>

La principal función de los tensores es mantener los alambres lo más horizontalmente posibles evitando que ocurran choques entre las líneas o tierra.

#### 1.3.2.4. Aros de supervisión

Los aros de supervisión cumplen la función de supervisión de la posición de las líneas del cercado eléctrico, se ubican a la entrada y salida de cada poste del alambre con doble galvanización, en caso de existir contacto entre la línea y el aro de supervisión, automáticamente la cerca eléctrica dará aviso para encender la alarma.



**Figura 14-1:** Aro de supervisión de cerca

Fuente: <https://goo.gl/M8SfFE>

#### 1.3.2.5. Electrificador

El electrificador cuenta con dos transformadores para su funcionamiento, un reductor y un elevador, el primero se encarga de disminuir de 110v a 12v el suministro de energía, para que posteriormente el transformador elevador de alta tensión, lo lleve por voltajes superiores a los 5000v.

También cuenta con una batería en caso de fallas energéticas que garantiza al menos por 48 horas la protección del sistema hasta que el fluido energético sea reestablecido. El pulso de alto voltaje emitido por la cerca eléctrica, no representa un peligro significativo para el intruso, debido a la baja corriente que utiliza en orden de los “mA”.

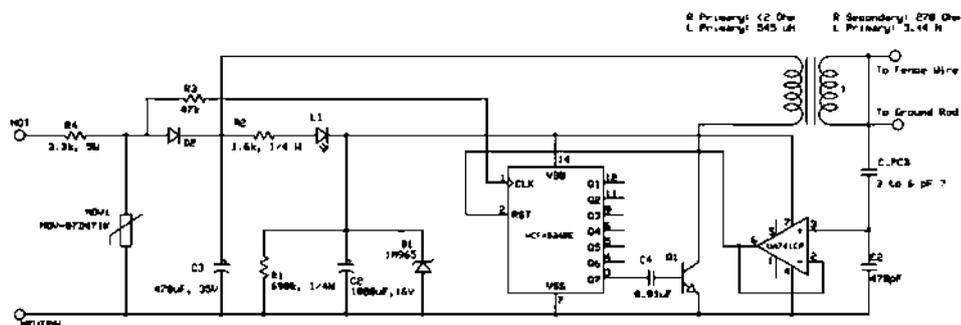
Este electrificador está diseñado para cortas distancias en comparación a otros electrificadores comerciales que proporciona un alcance de hasta 20km.



**Figura 15-1:** Electrificador Hagroy i8

Fuente: <https://goo.gl/CRUVN3>

Aproximadamente cada 1.3 segundos se produce un pulso eléctrico en la unidad de vallas, con funcionamiento similar a todos los dispositivos de alto voltaje, este circuito posee un transformador de aspiras que recibe en su entrada pulsos de voltaje a altas velocidades, este voltaje es almacenado previamente en un capacitor que descarga su energía en el primario del transformador generando un pulso de corta duración y alto voltaje.



**Figura 16-1:** Esquema circuito electrificador

Fuente: <https://goo.gl/wy9uyb>

El diodo D2 recibe la entrada de 12V. Debido a que solo se usa un solo diodo, la potencia solo se entrega en un medio ciclo de la entrada de CA. El condensador C2 es un condensador de filtro para la fuente de alimentación de los circuitos integrados. Filtra la ondulación del voltaje de entrada mediado por la onda.

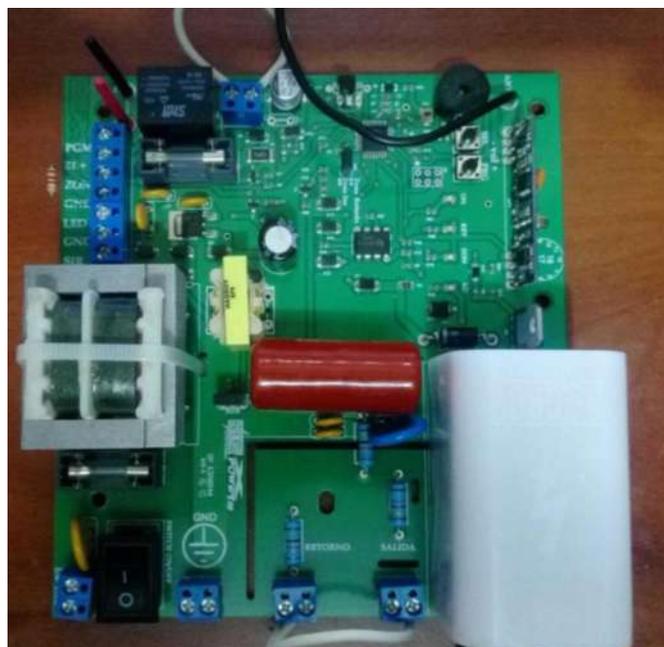
El voltaje se regula a alrededor de 12 voltios mediante el diodo Zener D1. R1 es un valor grande, 680k, y lo más probable es que se use para descargar C2 cuando la unidad no está encendida.

El transformador es la parte que aumenta el voltaje del pulso desde el lado primario al lado secundario. El lado secundario es donde está conectada la cerca. De acuerdo con las especificaciones del fabricante, el pulso de voltaje de salida es de aproximadamente 7000 voltios en nivel bajo y 13000 en nivel alto.

Los pulsos de salida de alto voltaje ocurren aproximadamente una vez cada 1.3 segundos. La temporización del circuito es producida por un contador de rizado binario ya que cuenta los pulsos de reloj. Los pulsos de reloj para el contador se derivan de la forma de onda de la línea de potencia 60Hz. La mitad positiva de la onda sinodal de entrada produce la porción lógica alta del pulso de reloj.

La corriente en el pin de entrada del reloj está limitada por R4 y R3. Esas dos resistencias tienen una resistencia combinada de aproximadamente 50k ohmios, lo que limitaría la corriente en el pin del reloj a solo alrededor de 3 mA.

Una vez que el contador ha contado hasta 64 pulsos, la salida del contador Q6 en el pin 3, el bit más significativo, irá alto. Esta es la señal que se utiliza para iniciar la descarga del condensador de almacenamiento a través del bobinado primario del transformador.



**Figura 17-1:** Circuito electrificador

**Realizado por:** Guaranga, J. 2018

La salida del contador Q6 se utiliza para pulsar la puerta del transistor Q1 para encenderse brevemente, lo que completa la ruta desde el condensador de almacenamiento de energía C3 a través del primario del transformador a tierra, para producir el impulso de salida en el secundario.

Una cosa interesante es, mientras que uno esperaría que dado que el contador se dispara después de 64 ciclos de reloj, tomaría 64/60 o 1.07 segundos entre los pulsos de salida. Sin embargo, se observó que la tasa de repetición era de aproximadamente 1,3 segundos.

La razón de este intervalo de tiempo algo más largo entre los impulsos de salida se debe a que hasta que el condensador C3 se haya cargado a un voltaje suficientemente alto, los pulsos aplicados a la entrada del reloj no tendrán un voltaje suficientemente alto como para ser reconocidos por el contador.(LargeMouthBass, 2013)

### ***1.3.3. Uso de las cercas eléctricas***

Por su versatilidad y sencillo manejo las cercas eléctricas son fácilmente instalables en ambientes de hogar u oficina, granjas, bodegas zonas francas, centros comerciales, sobre muros, verjas, balcones, etc. tomando en cuenta que por norma el alambre electrificado más bajo debe estar a una altura mínima de dos metros desde el suelo, pues de otra manera podrían surgir descargas accidentales, por descuido de los peatones que no tienen intención de ingresar al área protegida.

## **1.4. Video vigilancia**

### ***1.4.1. Introducción***

Video vigilancia es un sistema que permite la monitorización de eventos dentro de un perímetro resguardado por el sistema, desde cualquier parte. Siendo dichos eventos transmitidos a través de un servidor con acceso a internet conectado al sistema, sea inalámbricamente o a través de un grabador digital que sirve como central de adquisición de datos.

### ***1.4.2. Utilidad de video vigilancia***

Existe un inmenso mundo de posibilidades en los cuales los sistemas de video vigilancia son utilizados a día de hoy, comúnmente su uso varía según el tipo de usuario y la finalidad que este

le quiera dar a este sistema, como dueños de negocios que requieren tener una monitorización de sus empleados, su comportamiento ante los clientes, el despacho correcto de sus mercancías, etc.

Con el paso del tiempo y aumento de las jornadas de trabajo, cada vez los trabajadores pasan menos tiempo en su casa, lo cual incluye un aumento en la vulnerabilidad del hogar. Hoy es de uso común sistemas de video vigilancia en viviendas y departamentos, que permiten al usuario tener un mayor control y tranquilidad acerca de la seguridad de su vivienda.



**Figura 18-1:** Cámara tipo tubo Hikvision

Fuente: <https://goo.gl/URxg69>

Y el uso más común que se le ha dado es en empresas, oficinas, escuelas, colegios y lugares a los cuales acuden grandes cantidades de personas, pues es en estos lugares donde se requiere un mayor control de los acontecimientos que ocurren, pues el nivel de riesgos aumenta conforme aumenta la densidad de presencia de personas a un solo lugar.

Gracias a los sistemas de video vigilancia se han podido evitar siniestros como robos, a personas y/o locales comerciales, también estos sistemas han servido como evidencia en un sin número de casos que han requerido intervención de la justicia, permitiendo así anular posibles argumentos irreales que entorpezcan la labor de los jueces.

#### **1.4.3. Componentes de un sistema de video vigilancia**

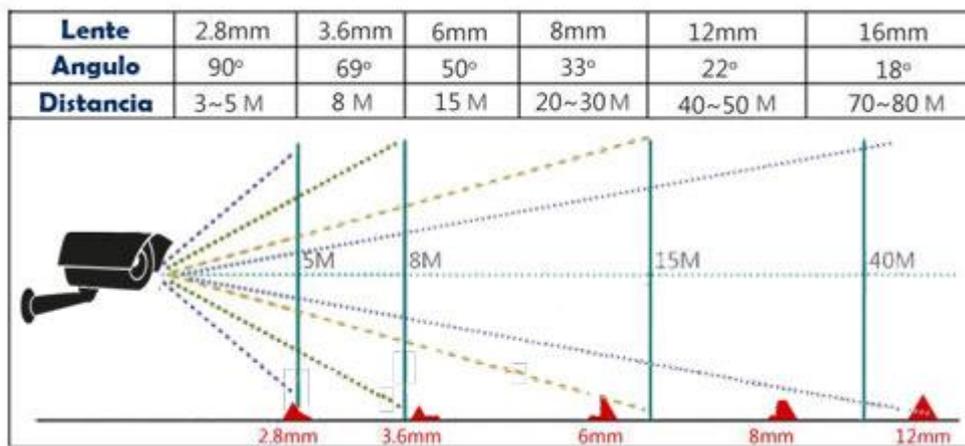
Generalmente los sistemas de video vigilancia están compuestos por cámaras de seguridad, central de monitoreo y cableado, aunque poco a poco se va dejando de lado este último, pues, existen cámaras que no necesitan un cableado para la transmisión de datos, simplemente lo realizan



## LENTE CON ABERTURA ADECUADA PARA CAPTAR IMAGEN

En el mercado existen varios tipos de lentes que incluyen las cámaras de seguridad, que vienen dadas por la capacidad en grados de apertura de su lente para mayor o menor áreas de cobertura, normalmente las cámaras vienen con apertura fija, pero también existen cámaras vari focales, es decir que pueden variar el ángulo de apertura de su lente, pero son mucho más costosas.

La apertura viene dado en mm, básicamente a medida que disminuye la apertura del lente de la cámara menor es el área que puede cubrir pero a la vez es mayor la distancia a la cual puede enfocar objetos con una calidad aceptable.



**Figura 20-1:** Relación entre apertura de lente, área y longitud

Fuente: <http://bits.com.ec/bits/angulo-de-vision-de-las-camaras-de-seguridad/>

## INSTALACIÓN INTERNA O EXTERNA

Otro de los puntos más importantes a la hora de elegir una cámara es saber el lugar en donde se va instalar, pues no es lo mismo una cámara que se encuentra dentro de una oficina la cual en el peor de los casos estará expuesta a bajos niveles de polvo, en comparación a una cámara instalada para controlar el tráfico a las afueras de un edificio, la cual debe soportar las diferentes afectaciones climáticas.

Por esta razón existen cámaras fabricadas especialmente para cada tipo de ambiente, con características especiales como protección anti polvo y/o lluvia en el caso de las externas o con lentes de mayor apertura que permiten una captación mayor de luz en el caso de las internas.

La mala elección entre una cámara para interior o exterior no solo afecta la vida útil de la misma, sino la economía del usuario, debido a los constantes mantenimientos, que requerirá o en casos más severos el remplazo de la misma por una nueva.



**Figura 21-1:** Cámara ip para exteriores

Fuente: <https://goo.gl/y2YnQU>

## **CONTROL SOBRE EL MOVIMIENTO DE LAS CÁMARAS**

La mayoría de las personas piensa que para una mejor monitorización, adquirir una cámara que permita al usuario tener el control sobre el movimiento de la misma (PTZ) es la mejor opción, (no siempre es la mejor elección), pues no, si bien es cierto tener una cámara que permita al usuario tener el control de movimiento es una experiencia totalmente nueva y novedosa, esto no quiere decir que es la mejor opción, al igual que los puntos anteriores esto depende enormemente de las necesidades del usuario y sobre todo el espacio físico en el cual se vaya a instalar el sistema de video vigilancia.

La mala elección entre una cámara estática y una con control de movimiento podrían incurrir en una enorme diferencia de precios para el usuario, pues las cámaras (PTZ) tienen costos elevados en comparación a las normales, la elección entre una y otra principalmente depende de:

Si existe o no el espacio físico para que la cámara pueda mover sus ejes, una vez seguros que tengo el espacio necesario, asegurar que los lugares a los cuales se va a acceder mediante el movimiento de la cámara tienen una calidad de visibilidad óptima, pues de nada sirve rotar la cámara hacia un lugar que no le da la luz y de todas maneras no se recibirá ninguna imagen y por último asegurarse que el costo de 1 cámara PTZ que cubra determinada área no sea mayor que el de varias cámaras normales haciendo la misma tarea.



**Figura 22-1:** Cámara PTZ para Exterior

**Fuente:** <https://goo.gl/4cPxtn>

#### 1.4.3.2. Medio de transmisión (Central de monitoreo)

Como fue mencionado anteriormente la necesidad de una central de monitoreo ha dejado de tener gran importancia dentro de los sistemas de video vigilancia a nivel de hogares, pues además de ya existir cámaras que transmiten sus datos inalámbricamente y solo requieren una dirección ip con acceso a internet, poseer una central de transmisión incurre en gastos adicionales para el usuario.

Sin embargo los sistemas no centralizados también tienen desventajas, como:

- Necesidad de una Ip para cada cámara de seguridad.
- Configuraciones adicionales en caso de conflictos en asignación DHCP.
- Consumo de ancho de banda por cada cámara instalada.
- Área de cobertura limitada por la señal Wi-Fi emitida por las cámaras y por el router.
- No poseen una central exclusiva para la grabación de eventos.

Es por esto que aún existe un debate entre el uso o no de una central de monitorización, pues por ahora la decisión de usar una central o no depende del usuario y sus necesidades.

La elección de una central de monitorización está ligada principalmente al número de cámaras empleadas en el sistema, pues actualmente todas incluyen transmisión de datos a través de internet además de las salidas convencionales VGA y HDMI.



**Figura 23-1:** Central de monitoreo

Fuente: <https://goo.gl/vceYNp>

## 1.5. Arduino

### 1.5.1. Introducción

Arduino es una plataforma de código abierto conformada por software y hardware de fácil manejo inicialmente ideado para estudiantes y personas sin conocimientos específicos acerca de electrónica o lenguaje de programación, caracterizado por su manejo y aprendizaje intuitivo y fácil.

Al ser una plataforma de código abierto, está constantemente en evolución y mejora; conjuntamente con una comunidad mundial que día tras día aporta nuevas ideas, con miles de proyectos disponibles para cualquier persona interesada en desarrollar el suyo propio, y la gran variedad de placas y periféricos de desarrollo que ofrece para este fin.

Arduino tiene como base para su programación lenguaje tipo C, por excelencia el lenguaje asociado a Unix, es un lenguaje de nivel medio con gran portabilidad de datos a través del uso de objetos sencillos como letras o números. A menudo es confundido este lenguaje, como uno propio de arduino por la facilidad en el manejo de pines de entradas y salidas gracias a librerías desarrolladas por la comunidad del código libre.



**Figura 24-1:** Arduino Started Kit

Fuente: <https://goo.gl/bkzLN7>

### 1.5.2. Elementos básicos de programación C++

Arduino hace uso de las mismas estructuras y sintaxis que el lenguaje C++, entre sus elementos de uso común y más básico tenemos según la página de Arduino.

- ***{}* entre llaves.** Las llaves sirven para definir el principio y el final de un bloque de instrucciones. Se utilizan para los bloques de programación *setup()*, *loop()*, *if..*, etc.
- ***;* punto y coma.** El punto y coma “*;*” se utiliza para separar instrucciones en el lenguaje de programación de Arduino.
- ***/\*... \*/* bloque de comentarios.** Los bloques de comentarios, o comentarios multi-línea son áreas de texto ignorados por el programa que se utilizan para las descripciones del código o comentarios que ayudan a comprender el programa.
- ***//* línea de comentarios.** Una línea de comentario empieza con *//* y terminan con la siguiente línea de código. Al igual que los comentarios de bloque, los de línea son ignoradas por el programa y no ocupan espacio en la memoria. (“Lenguaje de programación Arduino | Aprendiendo Arduino,” 2016)

### ***1.5.3. Placa de desarrollo arduino mega 2560 R3***

Es una tarjeta de desarrollo de código abierto que cuenta con 54 pines funcionales como entradas/salidas, entre los cuales existen 16 pines analógicos de las cuales 14 cuentan con la función PWM, y 4 UARTs, listos para su funcionamiento inmediato, basta con conectarlo al PC, a través de un puerto USB o alimentarlo con una fuente de voltaje de 5V.

Arduino mega es una de las placas de desarrollo más completas, además de sus entradas y salidas, y el lenguaje Processing/Writing, es la tarjeta que tiene más compatibilidad con programas de terceros, como Matlab, Octave, etc. también cuenta con una amplia gama de shields y módulos de expansión disponibles en el mercado, permitiendo de esta manera la creación casi sin límites de cualquier proyecto que el usuario pueda imaginar.

### ***1.5.4. Placas y módulos de Expansión Arduino***

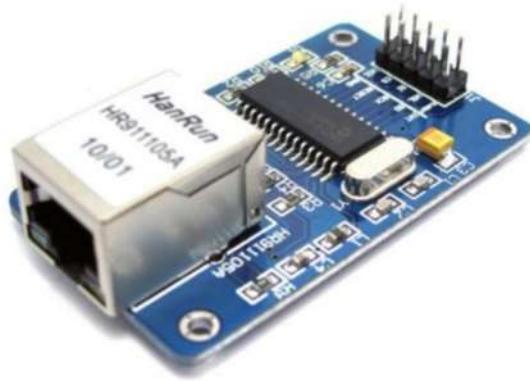
Son dispositivos desarrollados para complementar o agregar funciones a las placas principales, como entradas Ethernet, manejo de servomotores, manejo de relés, etc.

Estos dispositivos complementarios se caracterizan por proporcionar al usuario una mayor facilidad para el manejo de actuadores, sensores o la transmisión de datos, generalmente estas placas o módulos trabajan conjuntamente con librerías específicas que ayudan a que la programación de estos dispositivos no se torne compleja.

Actualmente existen placas de expansión de inimaginables tipos, haciendo prácticamente ilimitadas las opciones para el desarrollador.

#### ***1.5.4.1. Módulo de expansión ethernet ENC28J60***

El módulo ethernet ENC28J60 es un popular componente gracias a su relación costo – beneficio para el desarrollo de proyectos que requieran el uso de pasarela, generalmente se lo encuentra también en proyectos doméstico que no requieran grandes demandas de recursos, pues este módulo trabaja a 3.3v de alimentación.



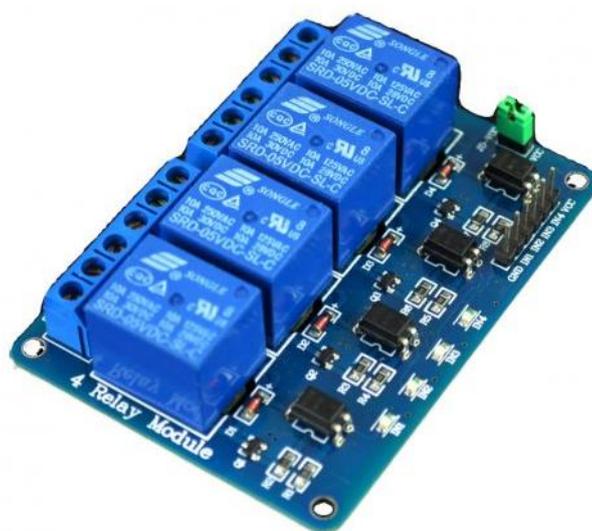
**Figura 25-1:** Módulo de expansión Ethernet ENC28J60

**Fuente:** <https://goo.gl/jfHoXf>

Este módulo una vez conectado a la placa principal funciona como una tarjeta de red normal, la cual posee una dirección física Mac a la cual será asignada una dirección IP por parte del Router, o cualquier servidor DHCP.

#### 1.5.4.2. Módulo de relés

Es una tarjeta de relés opto acoplados, cuya principal función es la de proporcionar al usuario la capacidad de controlar etapas de alto voltaje 110V-220V a través de las placas de desarrollo arduino, mediante el envío de pulsos de alto o bajo a la etapa de control que gobierna cada uno de los relés que componen el modulo.



**Figura 26-1:** Módulo de relés

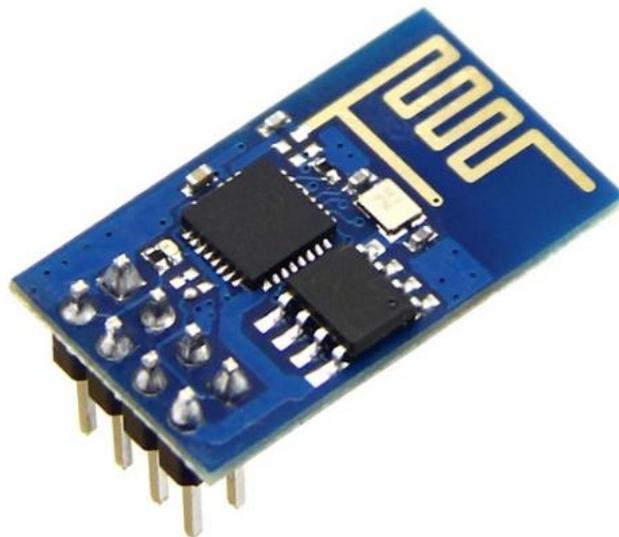
**Fuente:** <https://goo.gl/tLHdGB>

Posee dos modos de alimentación, ya sea a través de los pines conectados directamente a la placa arduino, que desempeña el papel de controladora, en el cual la energía proporcionado por la placa arduino se distribuirá en la etapa de control y la etapa de bobinado de los relés o también es posible conectar en modo de fuentes independientes, por lo general es recomendable este método pues así aseguramos la integridad de nuestra placa arduino, en caso de sobre carga de las bobinas o incapacidad de la placa arduino controladora de todo el sistema para suministrar la corriente necesaria para el correcto funcionamiento de nuestro proyecto.

#### 1.5.4.3. *Chip IoT (Esp8266)*

Dentro de los nuevos módulos que han ganado espacio en el mundo del desarrollo de proyectos el módulo esp8266 es el que por mucho ha destacado por su gran poder y mejor relación costo beneficio en el mundo del internet de las cosas (IoT), esp8266 es un chip con integración nativa del protocolo TCP/IP y conexión Wi-Fi.

Su principal función es dotar de acceso a los enormes beneficios que el acceso a una red ya sea con conexión a internet o acceso local pueda dar a un dispositivo, entendiéndose esto como el ahorro en cableado que supone una comunicación entre dispositivos. Su principal característica a parte de su conectividad es sin duda su bajo consumo, lo cual lo hace ideal como complemento que aporta grandes beneficios sin incurrir en gastos adicionales por consumo energético.



**Figura 27-1:** Esp8266 E01

**Fuente:** <https://goo.gl/NbH9qh>

#### 1.5.4.4. Módulo reconocimiento de voz Elechouse

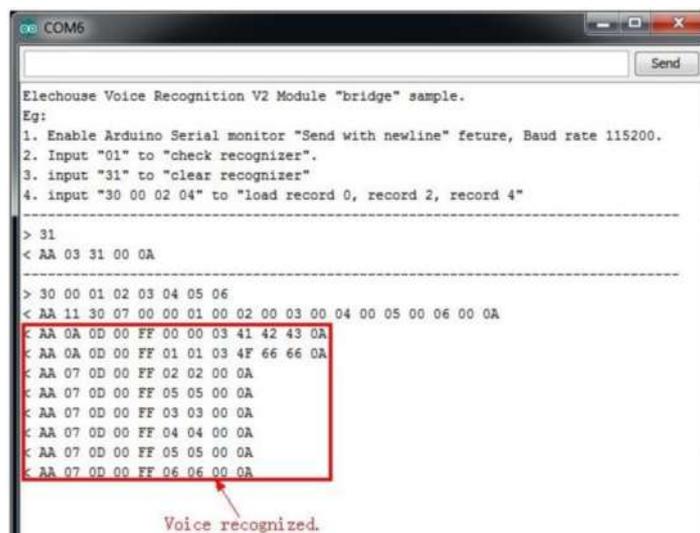
El un módulo de reconocimiento de voz es un placa capaz de reconocer alrededor de 80 comandos por voz, 7 al mismo tiempo, necesita una etapa de aprendizaje para reconocer el comando a ser utilizado, se extrae de la biblioteca de almacenamiento de comandos (máximo 80) un máximo de 7 que funcionaran al mismo tiempo para la operación de las entradas y salidas.



**Figura 28-1:** Módulo reconocimiento de voz Elechouse

Fuente: <https://goo.gl/sSpTvD>

Utiliza una entrada de voltaje de 4,5-5,5V con un consumo menor a 40mA, y una entrada analógica de micrófono monocanal de 3,5 mm adicionalmente una interfaz de pin para el micrófono.



**Figura 29-1:** Comparación tramas de datos

Fuente: <https://goo.gl/buErX8>

Su funcionamiento está basado en el almacenamiento de tramas hexadecimales en registros de la memoria interna del módulo, al iniciar el proceso de entrenamiento, el modulo transforma el audio en código hexadecimal y entra en espera para esperar una coincidencia de código antes de dar el aviso de grabación exitosa.

Si es que las tramas coinciden el módulo guarda el comando y se informa al usuario que el proceso fue realizado con éxito.

De manera similar se procede mientras se ejecuta un programa que utilice los comandos guardados; Una vez iniciado el proceso el modulo entra en estado de espera hasta recibir una señal de audio a través del micrófono, transforma el audio en código hexadecimal y compara con los almacenados, si coincide continuara con la acción programada, caso contrario no ocurrirá nada.

## **1.6. Fuentes de alimentación Arduino**

A medida que incrementa el uso de placas de desarrollo como arduino, tarde temprano surge la necesidad de alimentar nuestras placas con una fuente portable, recargable. Bien sea por proyectos que requieren permanecer en lugares sin fuentes de alimentación continua o proyectos como robots móviles y entre otros, que, a fuerza necesitan de una fuente de alimentación compacta y de larga autonomía.

### ***1.6.1. Regulador de voltaje***

Todas las placas de arduino disponen de un regulador de voltaje, el cual supone una pequeña caída de tensión, por lo que debemos proporcionar una tensión de almenos 5V. Por debajo de esa tensión lo más probable es que Arduino se apague. Por otro lado, cuanto mayor sea el voltaje, mayor será el calor que el regulador debe disipar. No es recomendado aplicar más de 12V al regular porque supone un esfuerzo excesivo. Proporcionar más de 20V al regulador lo dañará inevitablemente.

### ***1.6.2. Voltaje regulado***

También podemos aplicar la tensión nominal (5V o 3.3V) según el modelo directamente a la placa sin emplear regulador. Es decir podemos alimentar suministrando 5V al pin 5V de arduino. Por ejemplo, es lo que estamos haciendo cuando alimentamos Arduino desde USB.

En el caso de aplicar voltaje directamente la fuente de alimentación que empleemos tendrá que está regulada a la tensión nominal con un alto grado de precisión. Una variación o un pico de tensión dañarán el arduino, ya que no estamos empleando el regulador de voltaje.

### **1.6.3. Opciones de alimentación**

#### **1.6.3.1. Pila de 9V**

Usar una pila de 9V es una de las opciones más extendidas, especialmente para usuarios que se están iniciando y en proyectos de pequeño tamaño. El voltaje de 9V es apropiado para alimentar Arduino. Tienen la ventaja de ser fáciles de encontrar y usar. Además hay disponibles cables y porta pilas, que incluso incorporan un conector jack tipo Arduino, lo que hace que sean fáciles de usar.



**Figura 30-1:** Pila de 9V

**Fuente:** <https://www.luisllamas.es/alimentar-arduino-baterias/>

Como desventajas, las pilas de 9V disponen de baja densidad energética. Una pila tiene una capacidad típica de 500-600mAh. Además, son proporcionan una intensidad de corriente máxima muy baja, en torno a 300mA, útil solo para proyectos pequeños. Por otro lado, 9V es una tensión inadecuada para la mayoría de actuadores. Resulta excesiva para la mayoría de motores DC y servos, mientras que no es suficiente para grandes motores brushless y paso a paso, que funcionan con 12V y, además, requieren mucho más intensidad.

#### **1.6.3.2. Baterías recargables AA**

Similar al caso anterior, pero esta vez con baterías recargables. Podemos usar baterías de NiCd (en desuso) o NiMh. El voltaje es un poco inferior, 1.2V por batería, por lo que necesitaremos 5 baterías para conseguir los 6V. De forma similar a las opciones anteriores, podemos encontrar

fácilmente porta baterías para 5 baterías AA, así como todo tipo accesorios y cables para conectar de forma sencilla a Arduino.



**Figura 31-1:** Baterías recargables AA

Fuente: <https://www.luisllamas.es/alimentar-arduino-baterias/>

Las pilas recargables AA son fáciles de encontrar, pero son más caras que las pilas no recargables. A esto debemos añadir la necesidad de un cargador. Pese a ello, estos sobre costes se compensan rápidamente, resultando más económico que comprar pilas constantemente.

La carga de las pilas recargables también es algo inferior a las no recargables. Las baterías de NiCd tienen una carga de entre 500-1000 mAh. Las baterías de NiMh tienen capacidades de entre 600-2500 mAh. La intensidad máxima es similar, aproximadamente 1A de forma sostenible, lo cual es suficiente para proyectos y robots pequeños, requieren mucho más intensidad.

#### 1.6.3.3. *Baterías de litio*

Las baterías de litio 18650 son conocidas por alimentar linternas y láser de mano. Aunque no son tan conocidas como solución de alimentación para Arduino, en realidad son una gran opción a tener en cuenta. Dos baterías de litio 18650 puestas en serie proporcionan 7.4-8.2V, que es un voltaje perfecto para alimentar Arduino. También encontramos porta baterías que permiten incorporar con facilidad estas baterías a nuestros proyectos.

Las baterías de litio 18650 tienen la ventaja de proporcionar una alta capacidad de carga. Las baterías de primeras marcas proporcionan hasta 4800 mAh. (Las baterías “chinas” ofrecen capacidades muy superiores, aunque en realidad entregan entre 1500-2500). Proporcionan una capacidad de descarga de entre 1C – 2C, según modelos. Esto supone una intensidad máxima de

hasta 10A, aunque por seguridad no resulta aconsejable drenar más de 2-4 A sin estar muy seguros de la calidad y características de vuestra batería.

Por supuesto, las baterías de litio 18650 son recargables. Aunque se requiere un cargador especial para baterías 18650, son baratos y se amortizan rápidamente. Como desventajas, son una opción algo más cara que las anteriores, especialmente si compramos baterías buenas (algo que es recomendable). Además, su manipulación debe ser más cuidadosa, dado que el uso incorrecto de baterías de litio puede ser peligroso, causando sobrecalentamientos o incluso incendios.

Por otro lado el voltaje es algo elevado para motores DC y servos, aunque podemos reducirlo insertando un par de diodos, que de paso servirán como protección adicional a las baterías.



**Figura 32-1:** Batería de litio

**Fuente:** <https://www.luisllamas.es/alimentar-arduino-baterias/>

#### 1.6.3.4. *Batería de polímero litio (LiPo)*

Las baterías de Polímero Litio (Lipo) son la opción más avanzada para alimentar nuestros proyectos. Las baterías LiPo se presentan en varios voltajes, en función del número de celdas. Así las LiPo de 2 celdas (2S) proporcionan 7.4V-8.4, y las de 3 celdas (3S) proporcionan 11.1-12.6V. Ambos voltajes son adecuados para alimentar un Arduino.

Las baterías de 2S (7.4-8.4V) pueden emplearse para alimentar servos y motores DC, aunque es una tensión algo excesiva y conviene reducirla hasta los 6-7V. Las baterías de 3S (11.1-12.6V) son adecuadas para accionar grandes motores paso a paso, y motores brushless.

Las baterías LiPo presentan la densidad de energía más alta de las opciones planteadas. Podemos encontrar baterías con capacidades desde 500 a 5000mAh. Además, las baterías LiPo tienen la ventaja de proporcionar enormes intensidades. Es posible encontrar baterías 20-25C, lo que se traduce en intensidades de descarga de 50 a 100A, requerida por los motores más grandes.

Las baterías LiPo también son la opción más cara, aunque al ser recargables resultan económicas a largo plazo, más aún si tenemos en cuenta sus características eléctricas. La mayor desventaja de estas baterías es la dificultad y cuidado que hay que tener en su uso. Manipular de forma indebida una batería LiPo puede ser extremadamente peligroso, por la gran cantidad de energía que almacenan.

La carga de este tipo de baterías tiene que realizarse mediante cargadores especiales, y no deberían dejarse sin supervisión durante el proceso. Hasta el almacenamiento de estas baterías debe realizarse en condiciones controladas.



**Figura 33-1:** Batería LiPo

Fuente: <https://www.luisllamas.es/alimentar-arduino-baterias/>

## 1.7. Plataforma Blynk

### 1.7.1. Introducción

Blynk es una plataforma pensada como soporte en proyectos de desarrollo en el campo del internet de las cosas, que combina software y hardware, dedicada al control de placas de código abierto, Arduino, Raspberry, etc. Que permite la creación de aplicaciones móviles de manera sencilla, y la comunicación de esta aplicación con la placa a través de un servidor proporcionado por esta plataforma.



**Figura 34-1:** Interfaz aplicación Blynk

Fuente: <https://www.blynk.cc/>

### 1.7.2. *Funcionamiento*

Blynk es una plataforma que trabaja conjuntamente con placas de desarrollo y emplea su propio software para enlazar una aplicación móvil con el hardware a ser controlado, de manera sencilla y rápida.

El primer paso hacia un inmenso mundo en la creación de proyectos que empleen IoT es descargar la aplicación Blynk dependiendo el sistema operativo en el cual deseemos trabajar Android o IOS, dentro de la aplicación nos proporciona un número de serie único que servirá para enlazar nuestra aplicación con nuestro dispositivo, a través del servidor proporcionado por la plataforma de Blynk que nos asegura una conexión estable y segura.



**Figura 35-1:** Aplicación desarrollada con Blynk

Fuente: <https://goo.gl/RKMyyB>

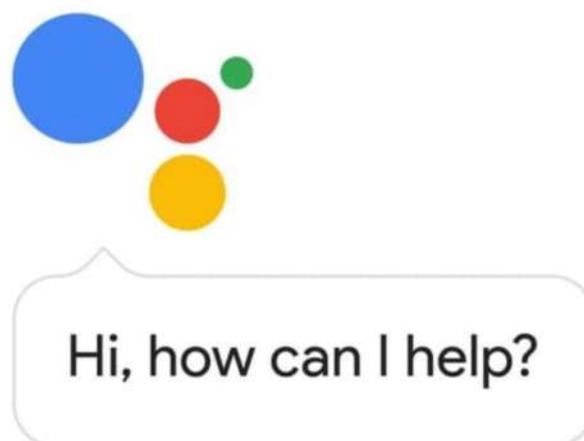
Blynk utiliza un diseño de programación interactivo, con variables virtuales y cajas (widgets) que facilita enormemente la creación de aplicaciones, en proyectos básicos no necesita de programación adicional en la IDE de la placa controladora, basta con un amplio catálogo de widgets con funciones como lectura de entradas análogas, lectura de entradas digitales, pulsadores, displays, control de los periféricos de nuestro teléfono, etc.

## 1.8. Google Assistant e IFTTT

### 1.8.1. Google Assistant

El asistente de google es un software desarrollado por google presente en una inmensidad de componentes electrónicos, principalmente en dispositivos móviles como smartphones, tabletas, y determinadas laptops. El asistente de google se encarga de la gestión de multitareas que el usuario le encomiende, dentro de su dispositivo móvil, cada día la compatibilidad del asistente de google se ha expandido llegando a incluir automóviles, electrodomésticos inteligentes y últimamente se ha incluido la capacidad de ser mejorada la experiencia del asistente de google con desarrolladores externos.

La interacción con el asistente de google es a través de la voz del usuario pudiendo entablar conversaciones totalmente coherentes con el mismo, recibiendo sugerencias o consejos acerca de una inmensa cantidad de temas.



**Figura 36-1:** Asistente de Google

**Fuente:** <https://goo.gl/y3SM2k>



aplicación móvil o con la simple lectura de un sensor en un lado del mundo creando una reacción a dicho evento en el otro lado.

A pesar de que IFTTT es relativamente nuevo no es para nada complicado de utilizar, lo único necesario es registrarse en su plataforma y empezar a crear, día a día se suman más servicios y aplicaciones a los beneficios que ofrece IFTT, gracias a esto las posibilidades de integración de los usuarios crece cada día.

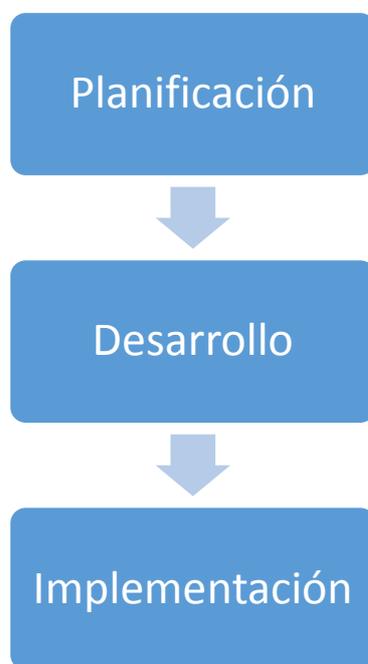
## CAPÍTULO II

### 2. MARCO METODOLÓGICO

#### 2.1. Metodología para el diseño de un sistema de seguridad para una vivienda

El presente proyecto aplica las directrices de la metodología heurística y experimental, en cada una de las etapas que componen el proyecto, como la adquisición de la información tanto para la planificación y desarrollo del sistema de seguridad, de la misma manera para su comportamiento y errores durante el proceso de implementación; identificado y definiendo el problema a ser resuelto, a través del análisis de registros previos en sistemas de seguridad implementados en viviendas.

Para el diseño de un sistema de seguridad que cumpla con los requisitos y necesidades de un usuario promedio el presente proyecto está dividido en tres etapas; etapa de planificación, etapa de desarrollo y etapa de implementación.



**Gráfico 1-2:** Etapas del diseño de un sistema de seguridad

Realizado por: Guaranga, J. 2018

## 2.2. Etapa de planificación

Antes de referir el proyecto a una planificación correcta es importante realizar una breve pre-planificación que permita aportar al diseño un enfoque de asesoramiento a la vez que se cubre las necesidades del usuario.

### 2.2.1. Requerimientos del usuario

Preliminarmente para poder determinar los dispositivos, materiales y equipos a utilizar en el sistema se realiza una breve encuesta al usuario acerca del beneficio que se espera obtener del sistema de seguridad. Es así que se obtiene los datos iniciales, procurando cubrir cada una de las necesidades expuestas por el usuario (Ver tabla 1-2).

**Tabla 1-2:** Requerimientos del usuario para el sistema de seguridad

Requerimientos	Equipos/Dispositivo/Software
Seguridad del hogar	Cercado eléctrico
Control de actuadores	Central domótica
Monitoreo	Sistema de video vigilancia
Acceso desde cualquier parte	Pasarela residencial/ Aplicación Móvil
Flexibilidad de expansión	Central domótica

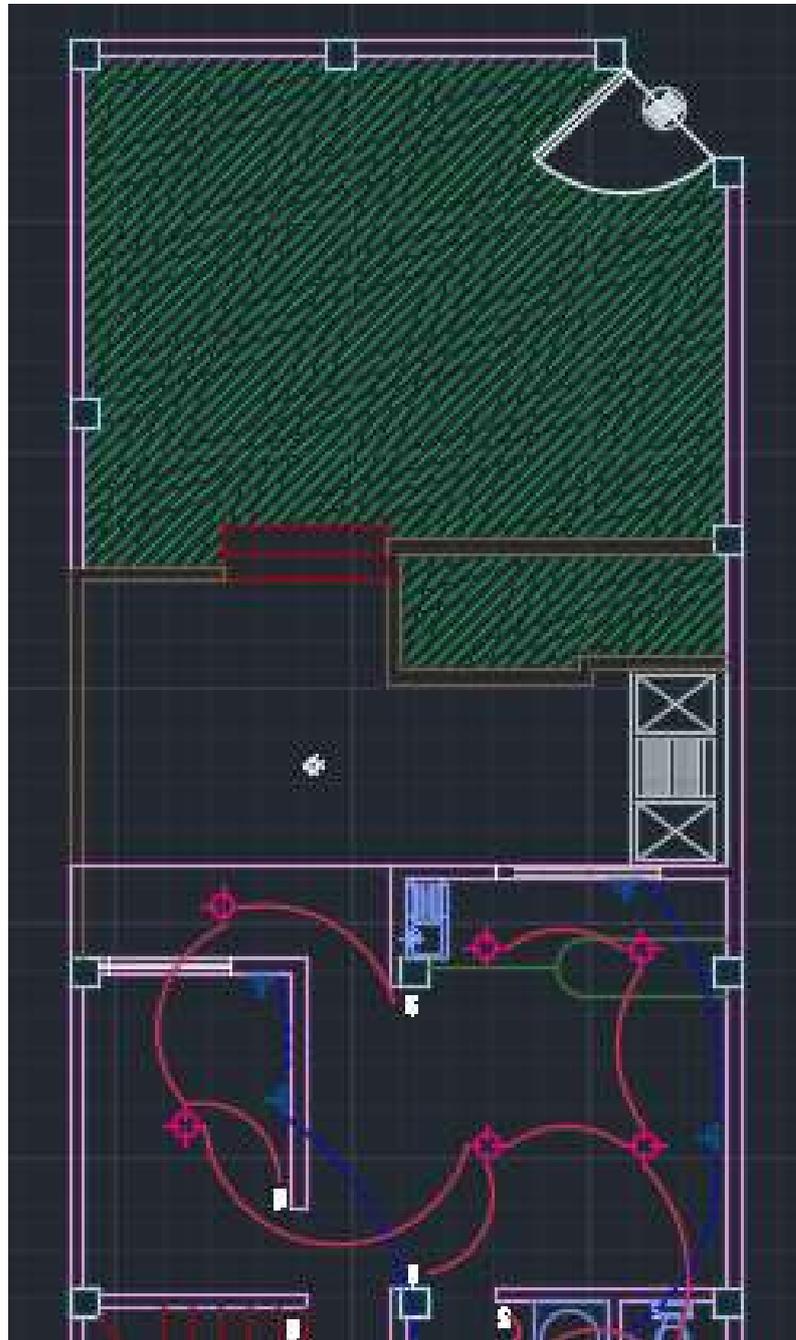
Realizado por: Guaranga, J. 2018

**Tabla 2-2:** Identificación planos vivienda/ áreas de riesgo / fuentes de alimentación

Áreas de riesgo	Determinante
Patio posterior (40m <sup>2</sup> )	Limitante con vecinos y vía publica
Patio lateral (180m <sup>2</sup> )	Patio compartido entre vecinos
Puerta posterior	Acceso desde calle secundaria con poca iluminación
Perímetro patio posterior (15m)	Perímetro protegido con paredes de baja altura
<b>Fuente de alimentación de energía</b>	<b>Carga Max</b>
Caja termina 1	Breaker 10 A 2000W Max
Toma corriente bodega	200W Max
Toma corriente estudio	200W Max

Realizado por: Guaranga, J. 2018

También se obtienen los recursos físicos preliminares del área en la cual se pretende desarrollar el sistema de seguridad a través del estudio de los planos arquitectónicos y eléctricos de la vivienda, para reconocer las áreas de riesgo de la vivienda, también las posibles fuentes de energía que alimentan a los componentes del mismo (Ver tabla 2-2).



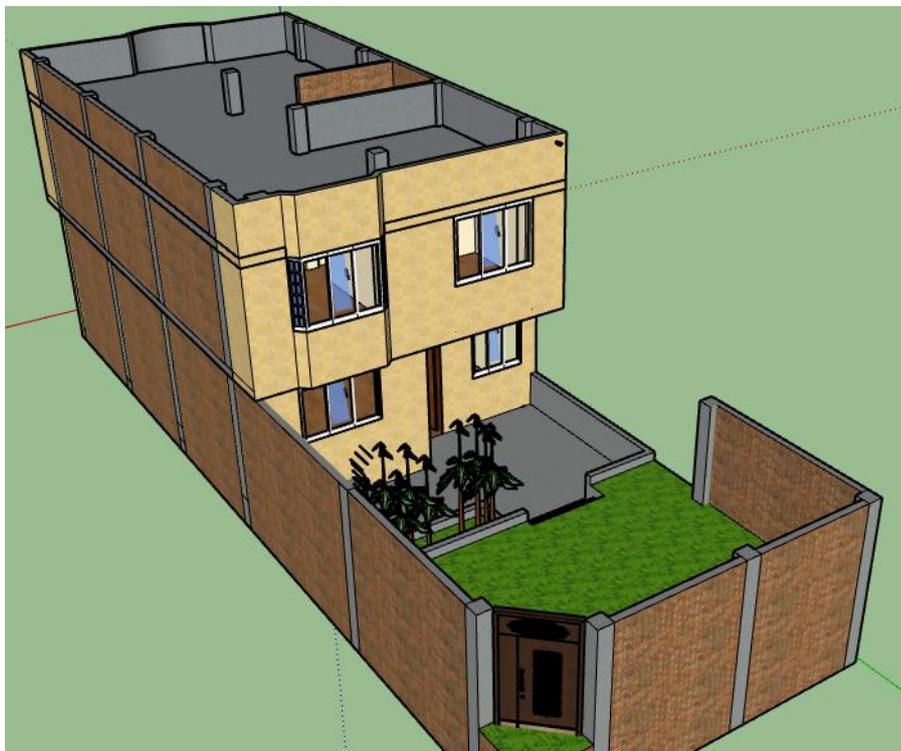
**Figura 1-2:** Plano parcial vivienda

Realizado por: Guaranga, J. 2018

Ahora con los parámetros preliminares bien establecidos se realiza la planificación del sistema, a través de la aplicación de conocimientos técnicos que cubran de la mejor manera las necesidades del usuario.

### 2.2.2. *Planificación*

En esta etapa es empleado el método heurístico y experimental a través de la definición del problema a ser cubierto, es necesario establecer una lista de materiales y equipos típicos en una instalación de seguridad sencilla, para una vivienda de dos plantas con un patio trasero de 40 m<sup>2</sup>.



**Figura 2-2:** Vista isométrica patio posterior

Realizado por: Guaranga, J. 2018

### 2.2.3. *Selección y ubicación de equipos*

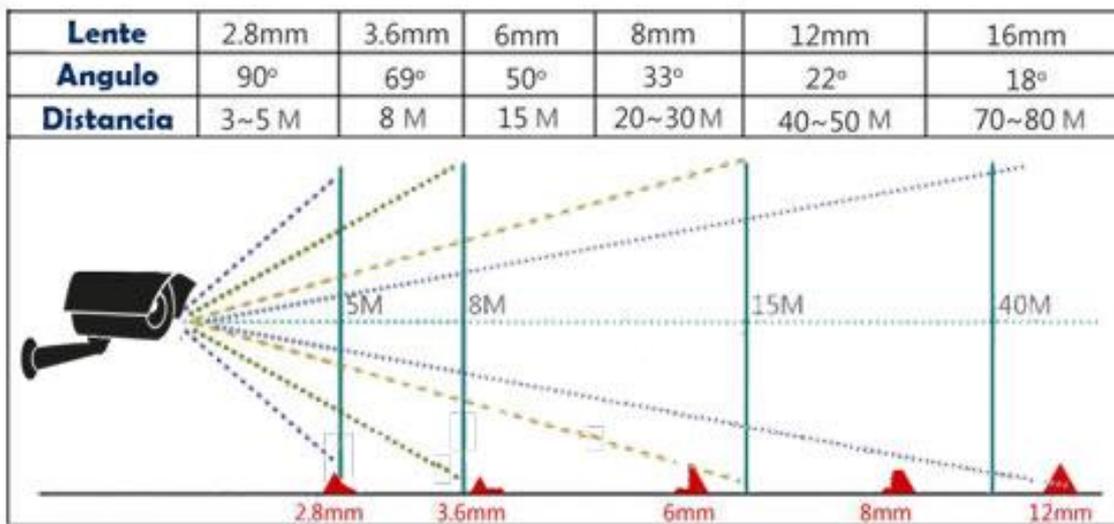
La selección y ubicación de equipos se realiza a partir de la información que se obtiene de usuario, se discrimina la información no relevante y se aplica conceptos teórico – prácticos de cada uno de los componentes del sistema.

### 2.2.3.1. Selección de equipos

#### SELECCIÓN CÁMARAS

Para la selección de cámaras se toma en cuenta el área a ser cubierta por el sistema, en este caso se tiene un patio posterior de 40m<sup>2</sup> aproximadamente (con límite a la calle y casas vecinas) y un patio lateral de 180m<sup>2</sup>. El primero dividido en dos áreas de 25 y 15 m<sup>2</sup> separados por un desnivel de 40cm.

Tomando en cuenta estos parámetros y la cobertura de las diferentes cámaras descritas en la (Figura 3-2) se procede a la selección de dos cámaras tipo tubo de 2.8 mm de lente fijo (uno para cada patio) con ángulo 90 grados y sensor de movimiento incluido en cada una.



**Figura 3-2:** Ángulo de visión cámaras de seguridad

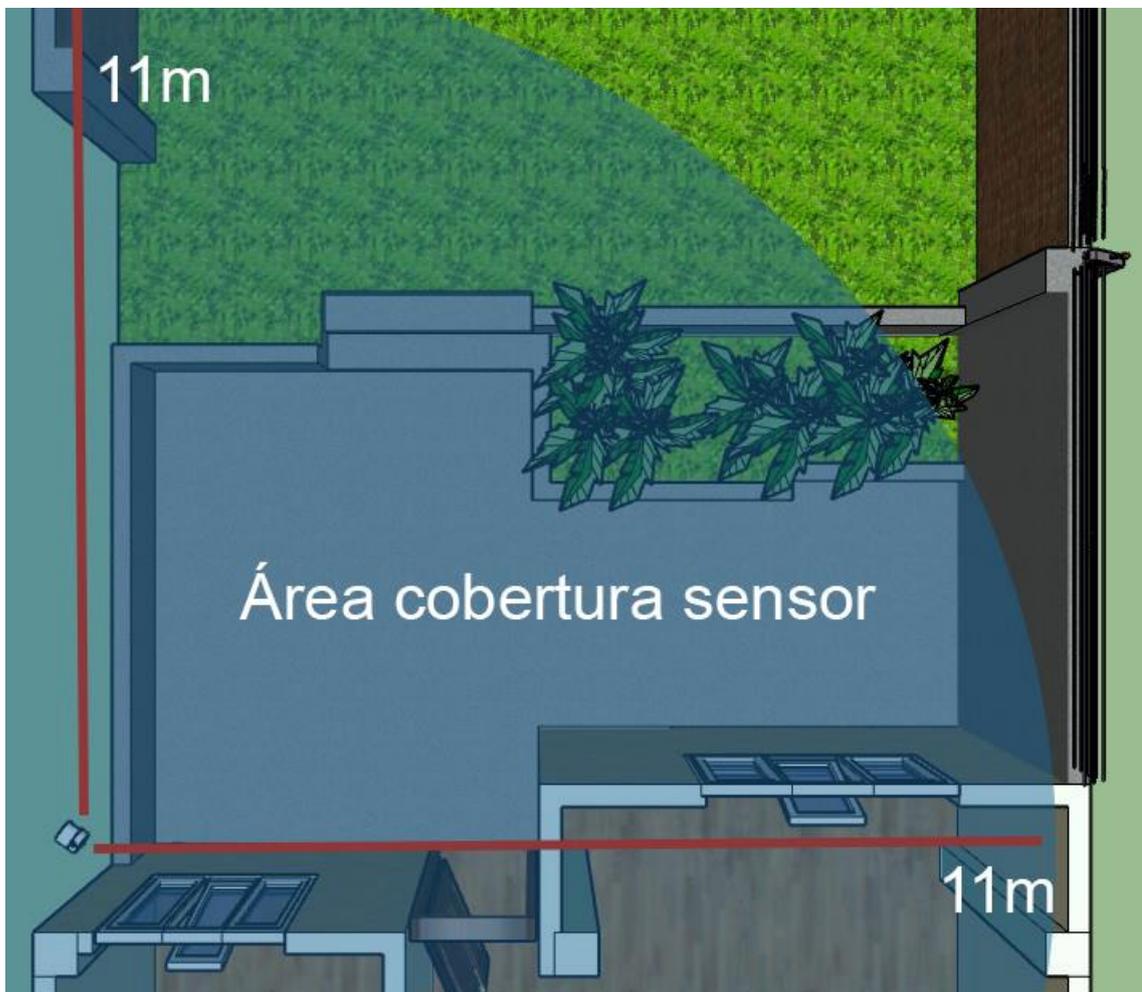
Fuente: <https://goo.gl/BXM1s9>

#### SELECCIÓN SENSOR PIR

Para la cobertura del área con desnivel es necesario proceder con los 4 pasos de diseño de acuerdo a la norma NTE INEN-ISO/IEC 29182-6 vigente en Ecuador para redes de sensores, que son: conocer las características del ambiente, la tecnología del sensor, el área de cobertura, y la modalidad de funcionamiento del mismo. Siguiendo esta normativa se determina las siguientes características para la selección:

- Características del ambiente: Temperaturas de 6 a 20 grados, poca humedad y lluvias recurrentes.
- Tecnología del Sensor: Sensor PIR.
- Área de Cobertura: Patio de 15m<sup>2</sup>.
- Modalidad de funcionamiento: Detección de intrusos mayormente en la noche (ambiente con poca luz).

De esta manera se selecciona un sensor PIR con rango de 11m y ángulo de detección de 180 grados, que cubre satisfactoriamente el área que pretendemos cubrir, cabe recalcar que a pesar de tener un ángulo de detección de 180°, la posición en la cual es colocado el sensor influye de manera directa, el ángulo que por defecto indica el fabricante.



**Figura 4-2:** Área cobertura sensor

Realizado por: Guaranga, J. 2018

## SELECCIÓN NVR

Para la selección del Router y NVR solo es necesario regirse a las necesidades del sistema. Teniendo en cuenta que, se emplean 2 cámaras de seguridad, cualquier NVR disponible en el mercado sirve para nuestro propósito, pues todo NVR tiene como disponibilidad mínima 4 entradas de cámaras, un aspecto importante a no olvidar es la disponibilidad de las cámaras o el NVR para transmitir sus datos a un servidor para su monitoreo.

Es así que el NVR HIKVISION DS-7108HGHI es seleccionado, pues cumple con todos los requerimientos necesarios para cubrir las necesidades del sistema, así también cuenta con una plataforma propia para la visualización de sus cámaras desde cualquier dispositivo móvil con acceso a internet.



**Figura 5-2:** NVR HIKVISION DS 7108

Fuente: <https://goo.gl/8xAXzg>

## SELECCIÓN ROUTER

Un router tiene varios aspectos importantes para su selección como, área a cubrir, número de puertos RJ-45, nivel de seguridad, protocolos de internet a ser utilizados, niveles de seguridad y cifrado de datos. Para el caso de nuestro sistema tomando en cuenta que solo requerimos dos conexiones RJ-45 con acceso a internet, que manejen protocolos IEEE 802.11n/g/b, una para la central de gestión y otra para el NVR que se encargará de la transmisión de datos de las cámaras, se emplea el Router D-Link TL-WR841HP con las siguientes características:

**Tabla 3-2:** Características Router D-Link

<b>Características de Hardware</b>	
Interfaz	4 Puertos Lan 10/100Mbps
	1 Puerto Wan 10/100Mbps
Botón	Botón WPS/Reset
Antena	2 Antenas desmontables omnidireccionales de 9dBi(RP-SMA)
Fuente de alimentación externa	12VDC/1A
Estándares Inalámbricos	IEEE 802.11n,IEEE802.11g,IEEE802.11b
Dimensiones (W X D X H)	6.6in.x5.1i.x1.2pulgadas.(168.5mmx130mmx31.5mm)

Fuente: <https://goo.gl/Lx7rTD>

## **SELECCIÓN ELECTRIFICADOR Y CERCA ELÉCTRICA**

Debido a que los cercos eléctricos y electrificadores están regidos bajo la norma IEC 60335-2-76 y vienen acondicionados en diferentes presentaciones de acuerdo al sistema en el cual se va a utilizar, tomando en cuenta que el perímetro que deseamos cubrir son 15m lineales (Ver tabla 2-2)., el fabricante con el cual se escoge trabajar “Hagroy” recomienda el electrificador HAGROY I8, con cobertura máxima de 1500m (HAGROY ELECTRONIC S.A.C., 2009).

### *2.2.3.2. Ubicación de equipos*

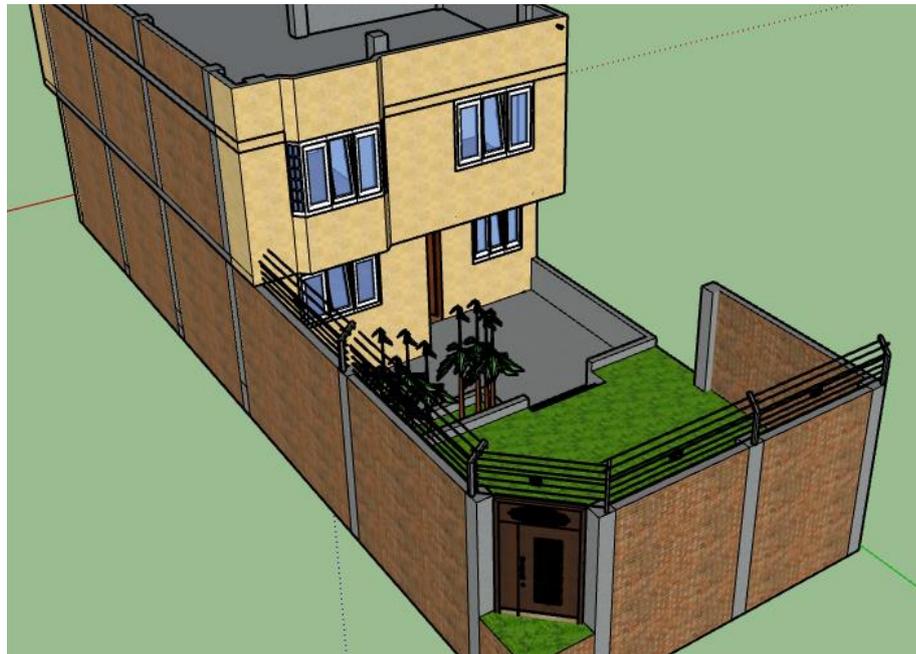
La ubicación de cada uno de los componentes, como la cobertura de los sensores a ser utilizados en la monitorización del sistema y los componentes a ser activados remotamente a través de la aplicación, son implementados mediante la identificación de las áreas vulnerables y a través de pruebas de funcionalidad y análisis de resultados, logrando así un sistema optimo que cubra las necesidades del cliente.

En base a lo expuesto anteriormente, se determina la ubicación de cada uno de los equipos que se emplean en el sistema, tomando en cuenta las características por los cuales fueron seleccionados cada uno, como, el área de cobertura, en el caso del sensores, capacidad de enfoque de acuerdo a la distancia de ubicación de las cámaras, tipo de conectividad para brindar acceso a pasarela en el NVR, y capacidad de carga máxima que soporta la fuente que alimenta a cada uno de los componentes del sistema.

**Tabla 4-2:** Ubicación de equipos

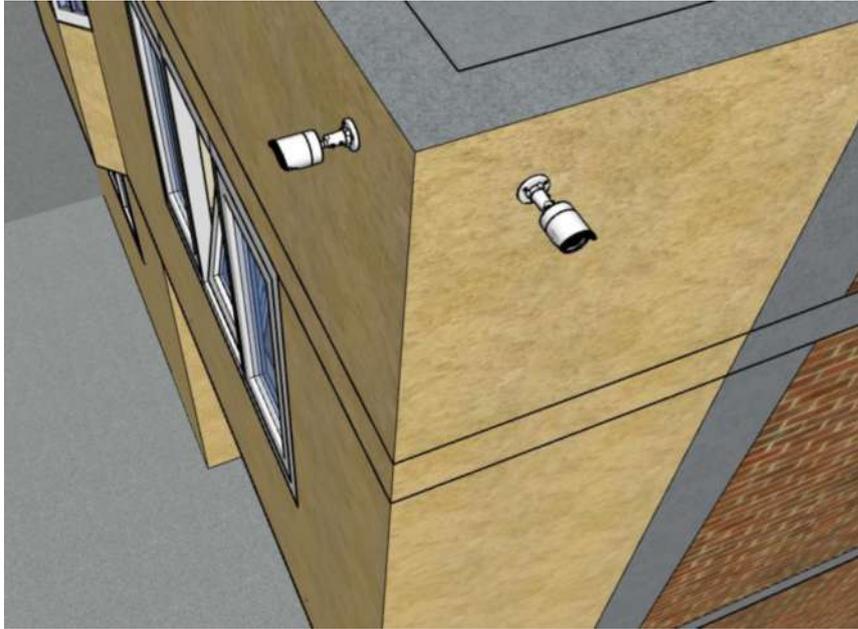
<b>Material o Equipos</b>	<b>Ubicación</b>
Cerca eléctrica	Perímetro posterior vivienda
Cámara de video vigilancia 1	Pared con vista al patio posterior
Cámara de video vigilancia 2	Pared vista al patio lateral
Electrificador	Pared bodega
Sensor de movimiento	Volado lateral izquierdo de la primera losa
Chapa eléctrica	Puerta principal
Trasformador chapa eléctrica	Pared bodega
NVR	Cuarto estudio
Router	Cuarto estudio
Central de control	Pared bodega
Central de gestión de sensores	Pared bodega

**Realizado por:** Guaranga, J. 2018



**Figura 6-2:** Ubicación tentativa cerca eléctrica

**Realizado por:** Guaranga, J. 2018



**Figura 7-2:** Ubicación tentativa cámaras de seguridad

Realizado por: Guaranga, J. 2018

Además de lo expuesto anteriormente, en base a pruebas con cada uno de los equipos, se toma en cuenta precautelar la seguridad del usuario y la seguridad de los equipos frente a posibles siniestros como sabotaje del sistema, efectos climáticos o robo, por lo cual se debe poner atención los siguientes aspectos para su ubicación (Ver Tabla 2-2).

**Tabla 5-2:** Aspectos a tomar en cuenta

<b>Equipo</b>	<b>Aspecto a tomar en cuenta</b>
Cámaras de video vigilancia	Posición que brinde un monitoreo con amplia cobertura de visión.
Sensor de movimiento	Máxima cobertura de censado, evitar objetos como ramas o platas que den falsas alarmas.
Cerca eléctrica	Altura mínima 2,00 m, evitar contactos involuntarios de peatones.
Electrificador	Preferentemente al ras del techo, evitar contacto con lugares húmedos, cocinas o baños.
NVR	Procurar el uso de transeptores activos o pasivos, y el tipo de cable utilizado, así evitar la pérdida de datos.

Realizado por: Guaranga, J. 2018



**Figura 8-2:** Ubicación tentativa sensor de movimiento

Realizado por: Guaranga, J. 2018



**Figura 9-2:** Ubicación tentativa electrificador y central de control domótico

Realizado por: Guaranga, J. 2018

#### 2.2.4. Selección de placas de desarrollo, módulos y circuitos

Una vez determinada la ubicación y equipos, es necesario realizar un proceso de selección de placas de desarrollo módulos y circuitos en base al número de entradas y salidas que maneja el sistema, el tipo de conexión requerida para dar cabida al acceso por pasarela, la alimentación de los circuitos que aseguren un correcto funcionamiento de cada uno de los componentes, y precautelar evitar un consumo innecesario de energía.

Para la selección de la placa principal se realiza una selección inversa de cada componente, es decir primero se selecciona los módulos que se utilizan, y finalmente se selecciona la placa principal en base a los requerimientos de cada módulo.

Cómo parámetro inicial se toma en cuenta el número de salidas necesarias para el control de los actuadores del sistema, tomando en cuenta el número de pines y tipo de alimentación que requiere el módulo, como se detalla la (Tabla 6-2), se selecciona el módulo de relés (4 relés), distribuidos de la siguiente forma:

**Tabla 6-2:** Requerimientos en placa módulo de relés

Actuador	Asignación de relé	Número de pines en placa	Consumo
Puerta eléctrica	Relé 1	1	5V – 500mA
Luz Patio	Relé 1	1	
Luz Estudio	Relé 1	1	
Sensor PIR	Relé 1	1	
Total Pines			6

Realizado por: Guaranga, J. 2018

Para la conexión a través de cable Ethernet se necesita un componente capaz de brindar confiabilidad en conexión, bajo consumo de energía y que ocupe un espacio reducido dentro del dispositivo final, por lo cual se selecciona el modulo módulo Ethernet enc28j60. (Ver. Tabla 7-2).

Para la función de reconocimiento de comandos de voz, es necesario un módulo capaz de brindar confiabilidad en la captación de audio y conversión de trama para la tarea de ejecución de acciones vinculadas a los distintos comandos, sin embargo la reducida capacidad de

procesamiento de cualquier placa de desarrollo arduino limita la efectividad que se pudiere tener en esta tarea.

Por lo cual la selección se enfoca en características como, mejores prestaciones y bajos consumos de energía, pues las diferentes marcas de módulos de reconocimiento de comandos de voz existentes en el mercado presentan mínimas diferencias en cuanto a rendimiento, por lo cual se selecciona el módulo de reconocimiento de voz elechouse. (Ver. Tabla 7-2).

**Tabla 7-2:** Requerimientos módulos: Ethernet enc28j60 – Reconocimiento de voz elechouse

Módulo	Número de pines en placa	Consumo
Módulo Ethernet enc28j60	7	138mA a 3,3V 145mA a 5,0V
Módulo reconocimiento de voz elechouse	4	5V - 40mA
<b>Total</b>	11	185 mA

Realizado por: Guaranga, J. 2018

Ahora que se tiene los módulos necesarios que requieren una conexión directa a la placa de desarrollo principal, se procede a seleccionar dicha placa en base a todos los requerimientos anteriormente mencionados, sin olvidar que se requieren 5 entradas adicionales para el manejo de señales de voltaje que aportan al funcionamiento del sistema.

De esta manera de las placas de desarrollo presentes en el mercado preliminarmente se selecciona 3 de las más comúnmente usadas en Ecuador, las cuales pueden satisfacer los requerimientos por parte de los módulos.

**Tabla 8-2:** Características placas de desarrollo

Placa	Consumo Max	# Pines Analógico-Digitales	Costo \$
Arduino uno	1A	6-14	10
Arduino mega	1A	16-54	20
Arduino due	2A	12-54	30

Realizado por: Guaranga, J. 2018

De acuerdo a las tablas anteriormente mostradas se requiere de un total de 11 pines digitales para manejo de módulos, más 4 entradas digitales de sensores, 1 entrada analógica para el censado de la cerca eléctrica, y 6 pines adicionales para alineación de los módulos.

**Tabla 9-2:** Requerimientos totales

# Pines analógicos	# Pines digitales	Pines alimentación	Amperaje total
1	15	6	685mA

Realizado por: Guaranga, J. 2018

Referido a la (Tabla 9-2), y de acuerdo a la (Tabla 8-2) solamente el arduino mega y arduino due satisfacen con los requerimientos del sistema, sin embargo el costo de la placa arduino due supera al de arduino mega por un 50% más, por lo cual se selecciona a arduino mega como la placa de desarrollo a utilizar en el sistema.

En base al consumo máximo capaz de suministrar la placa de desarrollo Arduino MEGA 2650 y el consumo de los diferentes módulos 685 mA de carga, se establece que la placa principal seleccionada nos asegura un funcionamiento correcto del sistema.

Como parte final en la selección de módulos y placas, para cubrir un futuro incremento en el número de entradas, como es típico en una instalación domiciliaria es necesario agregar una central de recolección de datos que crea una red Wi-Fi para lo cual es necesario el uso del chip IoT esp8266, por su ventaja en costo-beneficio en el uso de aplicaciones IoT en el mercado actual, además de la facilidad que ofrece para su programación e infinidad de aplicaciones posibles.

En total se usan 3, uno que funciona en modo de servidor y dos en forma de cliente.

### **2.2.5. Características fuentes de alimentación**

Gracias a la selección de placas se conoce el consumo de corriente del dispositivo lo cual es de utilidad para saber qué características debe tener la fuente de alimentación que se usa para el desarrollo del proyecto.

### *2.2.5.1. Fuente de alimentación DC*

Debido a los 685mA que consume el total de placas que se encargan del funcionamiento de este proyecto, y los 5V que recomienda el fabricante, basta con una fuente de alimentación DC que se conecta a la red eléctrica domiciliaria de 1A a 5V, sin embargo debido a que se desconoce las cargas que soporta las bobinas de los relés (con un máximo de 400mA de sobrecarga), se escoge una fuente de 1.5A de esta manera asegurar el correcto funcionamiento hasta en los casos de mayor exigencia.

### *2.2.5.2. Batería LiPo*

De acuerdo a las características para la selección de fuente DC y tomando en cuenta que el proyecto es un prototipo que busca estudiar el comportamiento con una fuente de energía portable, se escoge una batería lipo de 1.5Ah, que entra en funcionamiento cuando la fuente de energía DC conectada a la red eléctrica presente fallos, a través de la conmutación de un relé. (Ver pág. 65-66, para mayor detalle e información del circuito de conmutación, y duración de la batería en casos de fallas o corte de suministro eléctrico en el sistema de seguridad).

### *2.2.6. Características dispositivo móvil*

El proyecto está enfocado para ser gestionado desde un dispositivo móvil, tal como tabletas digitales, teléfonos inteligentes y otros gadgets con acceso a las tiendas de aplicaciones de Android (google play store) o Apple (IOS) debido a la disponibilidad de las aplicaciones que se utilizan en el desarrollo del dispositivo.

Es importante que el dispositivo cumpla con características básicas para la gestión y manejo de aplicaciones como memoria de almacenamiento suficiente, pantalla táctil, entre otros.

Además se requiere de características puntuales como:

- Acceso a redes de comunicación 3G (como mínimo) y 4G de preferencia.
- Sistema operativo Android 4.5.0 o IOS 9.3.1.
- Acceso a google assistant (ya sea de forma nativa o a través de app).
- 2Gb de memoria ram para el correcto funcionamiento de la aplicación y manejo de tráfico de datos.

- Pantalla de 4’’ como mínimo, de otro modo la visualización de aplicación no será la as óptima.

Todas estas especificaciones son tomadas de acuerdo a las exigencias de la plataforma Blynk para el correcto funcionamiento de la aplicación que se desarrolla.

### **2.2.7. Características de software para el desarrollo de la aplicación**

Después de la investigación previamente realizada, la plataforma de desarrollo de aplicaciones Blynk es elegida por las facilidades que brinda al usuario tanto para el desarrollo de la aplicación como para el alojamiento y transmisión de datos, proporcionando un servidor externo que permite de manera sencilla vincular placas de código abierto como arduino directamente con el proyecto.

Adicionalmente el sistema estará asistido por plataformas de integración de servicios y aplicaciones como es IFTTT para el manejo del sistema por comandos de voz.

## **2.3. Etapa de desarrollo**

Después de realizar la planificación y selección de los componentes a ser utilizados, para la etapa de desarrollo del sistema es dividido dos, en etapa de desarrollo y disposición de hardware, conexión de equipos y desarrollo de software.

### **2.3.1. Desarrollo y disposición de Hardware**

Durante esta etapa es necesario tomar en cuenta voltajes de entrada y salida de los diferentes sensores y actuadores que intervienen en el sistema de seguridad, siendo necesario la elaboración un circuito reductor de voltaje, de 15V a 5V implementado en la salida de alarma del electrificador, que es activado al entrar en contacto con cualquier objeto que cierre el circuito de la cerca eléctrica con tierra, de esta manera obtenemos un voltaje de 15V en la salida como alto, y 7V como bajo.

Sin embargo la entrada de voltaje admisible como alto en la placa arduino es 5V, por lo cual el circuito reductor entra en funcionamiento, disminuyendo el voltaje de 15V a 5V en alto y de 7V a 2,3V aproximadamente, garantizando que no se producirán falsas alarmas a causa del oscilamiento (entre 7V – 10V en bajo y de 14V – 15V en alto) de voltaje que existe en la bornera

de aviso del electrificador a causa de los capacitores y transformadores elevadores del circuito elevador perteneciente al electrificador.

Por lo cual es necesario el cálculo de un divisor de voltaje que garantice como máximo en la salida del electrificador 5V.

Cálculo división de voltaje:

$$V_{out} = \frac{V_{in} * R2}{R1 + R2}$$

$$5V = \frac{15V * R1}{R1 + R2}$$

Donde asumimos  $R1 = 1k\Omega$

$$5V = \frac{15V * 1k\Omega}{1k\Omega + R2}$$

$$15V(1k\Omega) = 5V(R2) + 5V(1k\Omega)$$

$$10V(1k\Omega) = 5V(R2)$$

$$R2 = \frac{10V(1k\Omega)}{5V}$$

$$R2 = 2k\Omega$$

Valores de salida con oscilaciones de voltaje entre 7V a 14V.

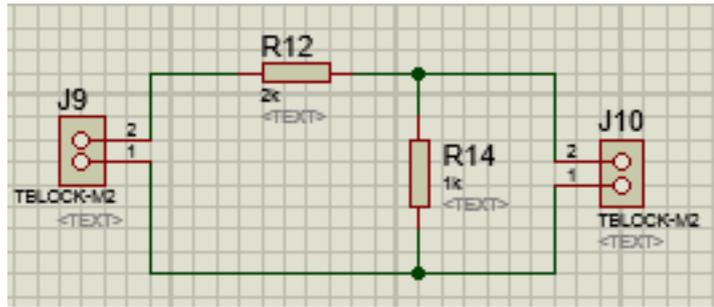
$$R2 = 2K\Omega$$

$$V_{out} = \frac{V_{in} * 1k\Omega}{1k\Omega + 2k\Omega}$$

$$V_{out} = \frac{14V * 1k\Omega}{1k\Omega + 2k\Omega} = 4,66V$$

$$V_{out} = \frac{10V * 1k\Omega}{1k\Omega + 2k\Omega} = 3.33V$$

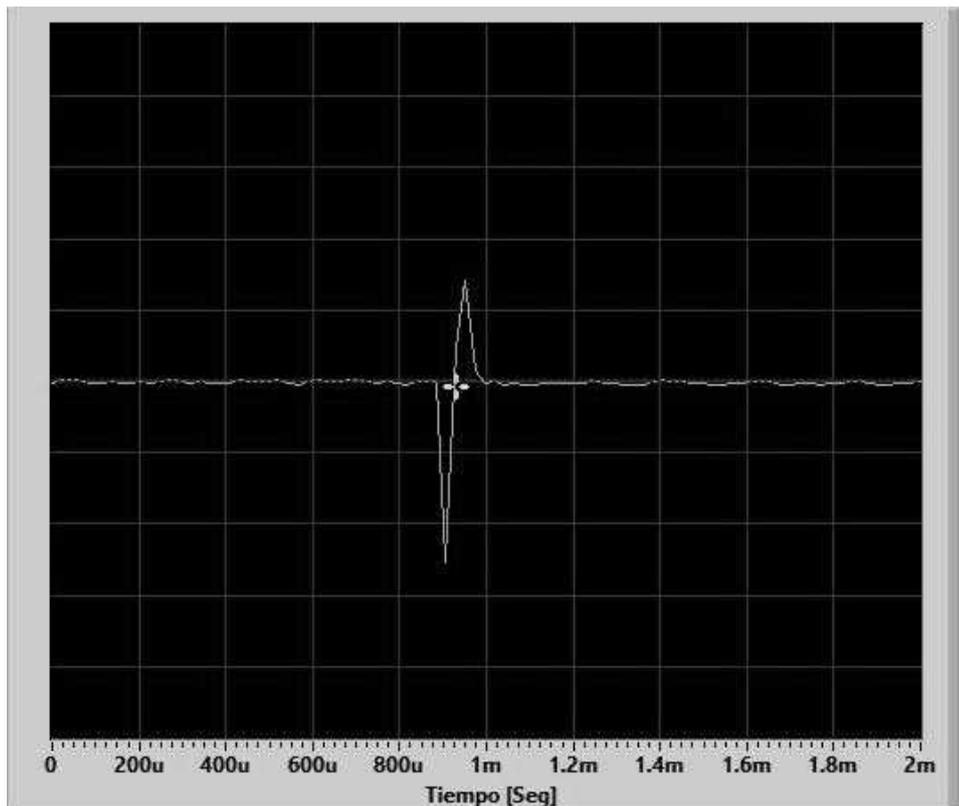
$$V_{out} = \frac{7V * 1k\Omega}{1k\Omega + 2k\Omega} = 2.33V$$



**Figura 10-2:** Circuito divisor de voltaje

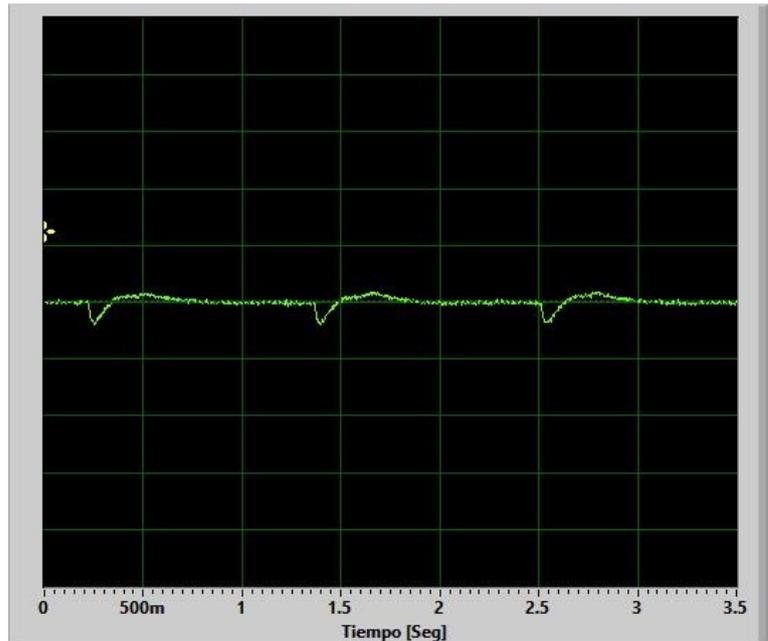
Realizado por: Guaranga, J. 2018

Adicionalmente para el censado del voltaje de la cerca es necesario el desarrollo de un circuito que emplee un cruce por cero que sincronice los disparos de alto voltaje de la cerca con los tiempos de medición del arduino, a la vez que reduzca dicho voltaje a rangos admisibles por las placas de desarrollo.



**Figura 11-2:** Pulso de alto voltaje cerca eléctrica

Realizado por: Guaranga, J. 2018



**Figura 12-2:** Pulsos tiempo de cesado

Realizado por: Guaranga, J. 2018

Un circuito de sincronización de cruce por cero emplea un opto acoplador que recibe un voltaje de entrada, el cual excita el fotodiodo que activa un scr que permite el paso de voltaje a la salida del opto acoplador, para este circuito se toma en cuenta que el fotodiodo para ser activado necesita una corriente de 50mA y un voltaje de 5V por lo cual a través de la ley de Ohm se calcula las resistencias necesarias para lograr dicho objetivo.

#### **Cálculo voltaje entrada:**

$$V_{out} = \frac{V_{in} * R_2}{R_1 + R_2}$$

$$5V = \frac{13000V * R_2}{2M\Omega + R_2}$$

Donde asumimos  $R_1 = 2M\Omega$

Donde:  $R_2$  es la resistencia en paralelo con el voltaje de entrada

Para el cálculo de la resistencia necesaria se asume  $R_1$

$$R_2 * 5V + 5V * 2M\Omega = 13000V * R_2$$

$$R_2(13000V - 5V) = 5V * 2M\Omega$$

$$R_2 = \frac{5V * 2M\Omega}{13000V - 5V}$$

$$R2 = 784.76 \Omega$$

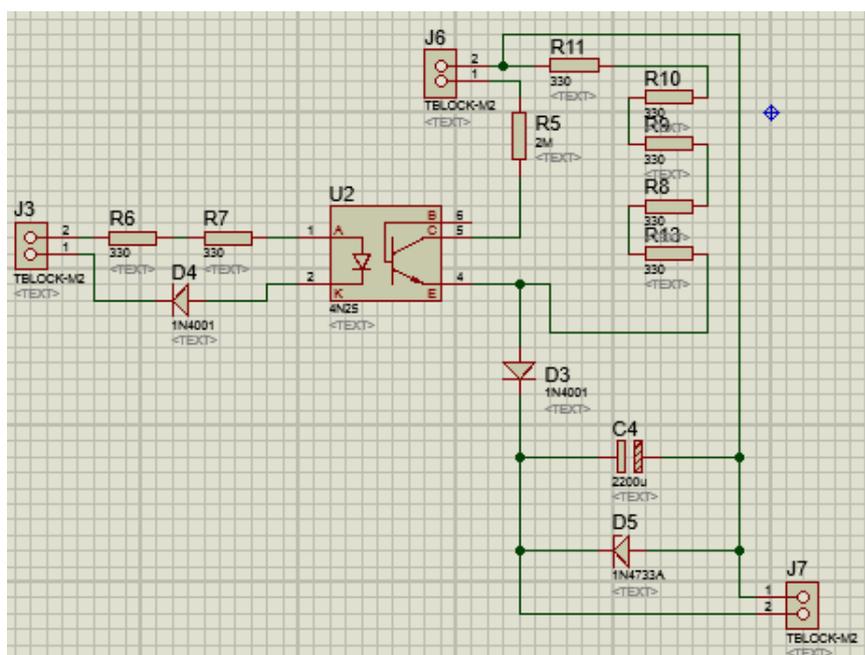
En cuanto a la corriente de entrada, debido a que la cerca eléctrica proporciona un alto voltaje a una corriente sumamente reducida, por debajo de los 10mA no necesitamos realizar cálculos para reducción de corriente.

**MAXIMUM RATINGS** ( $T_A = 25^\circ\text{C}$  unless otherwise noted)

Rating	Symbol	Value	Unit
<b>INPUT LED</b>			
Reverse Voltage	$V_R$	3	Volts
Forward Current — Continuous	$I_F$	60	mA
LED Power Dissipation @ $T_A = 25^\circ\text{C}$ with Negligible Power in Output Detector Derate above $25^\circ\text{C}$	$P_D$	120	mW
		1.41	mW/ $^\circ\text{C}$

**Figura 13-2:** Especificaciones opto acoplador 4N25

Fuente: (Motorola Inc., 1995)

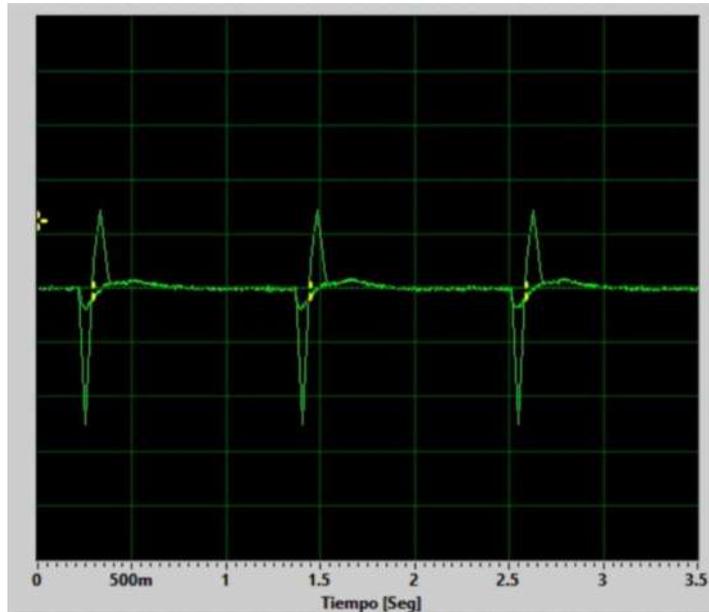


**Figura 14-2:** Circuito sincronización y cruce por cero de la cerca eléctrica

Realizado por: Guaranga, J. 2018

El circuito reductor está formado por un diodo zener que garantiza que el voltaje máximo a la salida de 5.1 V, un capacitor de 24V a 1200uF que actúa como filtro en la reducción de las oscilaciones causadas por el electrificador, y resistencias en serie/paralelo que están en modo

divisor de voltaje, todo esto sobre una baquelita perforada, ubicada antes de la entrada a la placa de desarrollo arduino mega.



**Figura 15-2:** Sincronización: Disparo alto voltaje - Tiempo de censado

Realizado por: Guaranga, J. 2018

La corriente máxima de entrada del opto acoplador empleado es de 60mA según su hoja de datos, y debido a que la corriente suministrada por el electrificador es menor a esta no es necesario un circuito adicional para reducción de corriente suministrada al opto acoplador, sin embargo debido a los pulsos de alto voltaje, se producen oscilaciones y retornos de corriente en las salidas del transformador de alto voltaje por lo cual es necesario la implementación de un diodo rectificador en la salida a tierra del opto acoplador, el cual en caso de retorno de corriente impida el paso de la misma hacia el circuito.

Para la entrada del sensor de movimiento es necesario el diseño de una placa que adapte el voltaje de salida del sensor en un rango proporcional de 110 a 5V. Debido a que el sensor empleado como señal de salida proporciona 110V AC es necesario convertir este voltaje a una señal admisible por el microcontrolador de la placa madre del sistema.

Como primer paso se rectifica la señal de entrada a través de un puente de diodos, basándonos en las hojas de datos elegimos un puente de diodos que soporte dicha tensión 110V AC.

Se opta por el puente rectificador 2W10 que soporta un máximo de 1000V pico de forma repetitiva. A continuación se calcula el voltaje de salida que me proporciona el puente de diodos

con la señal rectificada, para lo cual primero calcularemos el voltaje pico en un tomacorriente normal.

Cálculos voltaje en CD entregada por el puente de diodos.

$$Vp = (V_{rm} * \sqrt{2})$$

$$Vp = 155.56 V$$

$$V_{cd} = \frac{2 * Vp}{\pi}$$

$$V_{dc} = 99.032 V$$

*Vcd es el voltaje al cual se agrega un capacitor para reducir el ruido de la señal*

$$V_{out} = \frac{V_{in} * R2}{R1 + R2}$$

$$5V = \frac{99.032V * R1}{R1 + R2}$$

Donde asumimos  $R1 = 1k\Omega$

$$5V = \frac{99.032V * 1k\Omega}{1k\Omega + R2}$$

$$99.032V(1k\Omega) = 5V(R2) + 5V(1k\Omega)$$

$$94.032V(1k\Omega) = 5V(R2)$$

$$R2 = \frac{94.032V(1k\Omega)}{5V}$$

$$R2 = 18.8064 k\Omega \cong 20k$$

**Maximum Ratings and Electrical Characteristics** @ $T_A=25^\circ C$  unless otherwise specified

Single Phase, half wave, 60Hz, resistive or inductive load.  
For capacitive load, derate current by 20%.

Characteristic	Symbol	2W005	2W01	2W02	2W04	2W06	2W08	2W10	Unit
Peak Repetitive Reverse Voltage	$V_{RRM}$								
Working Peak Reverse Voltage	$V_{RRM}$	50	100	200	400	600	800	1000	V
DC Blocking Voltage	$V_R$								
RMS Reverse Voltage	$V_{R(RMS)}$	35	70	140	280	420	560	700	V
Average Rectified Output Current (Note 1)	$I_O$	2.0							A

**Figura 16-2:** Hoja de datos 2W10 puente de diodos

Fuente: (Won-Top Electronics, 2002)

Se debe garantizar el funcionamiento del diodo zener que actúa como regulador de voltaje, permitiendo el paso de hasta 5.1 V dc. Por lo cual requerimos una entrada máxima de corriente de 49mA, admisible por el diodo Zener (información descrita en la hoja de datos del mismo diodo), entonces se calcula las resistencias que proporcionan la disminución de esta corriente al nivel deseado.

TYPE No.	WORKING VOLTAGE $V_Z$ (V) <sup>(1)</sup> at $I_{Ztest}$	TEST CURRENT $I_{Ztest}$ (mA)	DIFFERENTIAL RESISTANCE			REVERSE CURRENT at REVERSE VOLTAGE		WORKING CURRENT $I_{zm}$ (mA)	NON-REPETITIVE PEAK REVERSE CURRENT $I_{zsm}$ (mA) <sup>(2)</sup>
			$r_{dif}(\Omega)$ at $I_{Ztest}$	$r_{dif}(\Omega)$ at $I_Z$	$I_Z$ (mA)	$I_R$ ( $\mu$ A)	$V_R$ (V)		
	NOM.	MAX.	MAX.	MAX.	MAX.	MAX.	MAX.		
1N4728A	3.3	76	10	400	1	100	1	276	1380
1N4729A	3.6	69	10	400	1	100	1	252	1260
1N4730A	3.9	64	9	400	1	50	1	234	1190
1N4731A	4.3	58	9	400	1	10	1	217	1070
1N4732A	4.7	53	8	500	1	10	1	193	970
1N4733A	5.1	49	7	550	1	10	1	178	890

**Figura 17-2:** Hoja de datos diodos Zener

Fuente: (Philips Semiconductors, 2007)

$$I_{cd} = \frac{V_{cd}}{Rl} = \frac{99,032 V}{R1 + R2}$$

$$\frac{99.032V}{1k\Omega + 2k\Omega} = 33.01mA$$

$$P_{cd} = 33.01mA * 99.032Vdc$$

Pero se necesita las corrientes que circula por el diodo Zener y no el de las resistencias, así que el procedimiento debería seguir con mallas.

$$malla 1: 99.03 = I1 * 2K + I1 * 1K - I2 * 1K$$

$$malla 3; 0.7 = I2 * 1K - I1 * 1K$$

$$I2 * 1k = 0.7 + 1K * I1$$

$$I2 = \frac{0.7 + 1K * I1}{1K} \text{ Ecuación 1}$$

donde:

$$99.03 * 1K + 0.7 = I1 * 2K * 1K + I1 * 1K * 1K - 1KI1$$

$$I1 = 33.021 mA \text{ Remplazando en 1}$$

$$I2 = \frac{0.7 + 1K * 33.021mA}{1K}$$

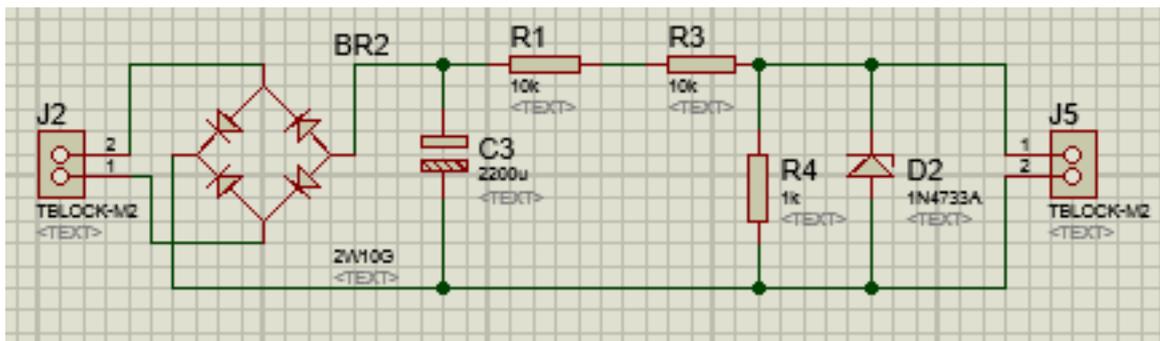
$$I2 = 33.721mA$$

### Cálculo de potencia de disipación del circuito

$$P_{cd} = V_{dc} * I_{dc}$$

$$P_{cd} = 3.26W$$

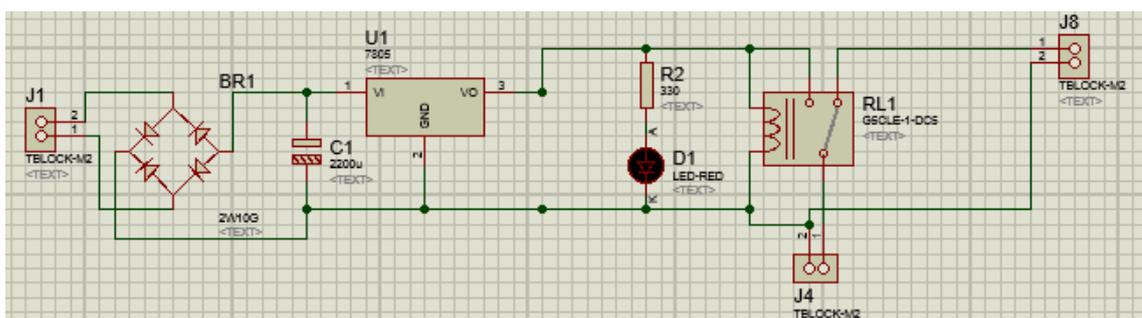
Como se puede observar las resistencias calculadas para nuestro circuito de acondicionamiento de señal cumplen con todas os requerimientos del sistema y cada uno de sus componentes sin sobre cargarlos o saturarlos.



**Figura 18-2:** Circuito Reductor Sensor de Movimiento

Realizado por: Guaranga, J. 2018

En cuanto a los voltajes requeridos y proporcionados por los demás componentes del sistema de seguridad, es necesario implementar un circuito adicional de alimentación para su funcionamiento. De acuerdo a la Tabla 9-2, podemos decir que necesitamos una fuente entre 685mA a 1 A de demanda de corriente por parte de los circuitos.



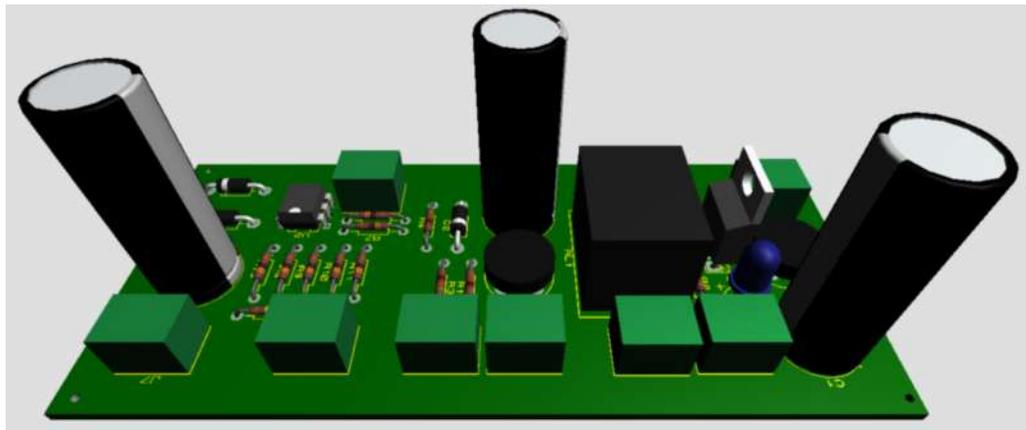
**Figura 19-2:** Fuente 1.5A, alimentación placas y módulos

Realizado por: Guaranga, J. 2018

Adicionalmente se debe tomar en cuenta un consumo extra de energía por parte de los relés que componen el modulo que está conectado al arduino 0,4 A más de consumo. Comenzamos con un circuito reductor de 110v a 12v el cual ingresa a un puente rectificado de diodo, después de la

atrapada de rectificación el circuito sigue a un regulador de voltaje 7805 el cual asegurara una salida de 5V con un amperaje de consumo máximo de 1.4 A energía suficiente para suministrar cada uno de los componentes de la central de gestión.

Adicionalmente en la salida del voltaje se encuentra un relé a 5V el cual actúa como cambiador de fuente de alimentación en caso de ausencia de energía eléctrica, momento en el cual entra en acción una batería de 1.5 A capaz de alimentar al circuito por unas 3 horas aproximadamente.



**Figura 20-2:** Fuente 1.5 A - Circuito lectura y cruce por cero cerca eléctrica

**Realizado por:** Guaranga, J. 2018

Por último se implementa un circuito de distribución de energía para todas las diferentes placas de desarrollo y módulos que componen el sistema, este circuito es capaz de suministrar los 1.5A necesarios para el correcto funcionamiento.

De esta manera este circuito tiene una entrada con regulador de 5 V para evitar sobre voltajes sobre las placas, también cuenta con borneras de distribución de 3 patas capaces de aguantar 2 A de corriente y unas borneras de distribución de tierra.



**Figura 21-2:** Circuito de distribución de energía

**Realizado por:** Guaranga, J. 2018

### 2.3.1.1. Arduino mega 2650 y módulos de desarrollo

Una vez resuelto el problema de voltajes de alimentación a las entradas y salidas; basado en la lista de sensores y actuadores a ser utilizados, es elaborada una lista de distribución de pines y asignación de funciones para cada uno de los módulos, entrada y salidas a ser conectadas al arduino mega 2650.

**Tabla 10-2:** Asignación de funciones pines Arduino Mega 2650

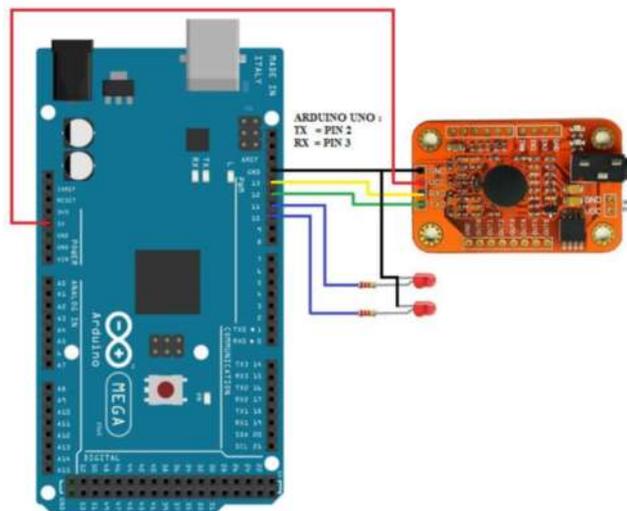
Arduino Mega 2650	
A01	Entrada monitoreo voltaje cerca eléctrica
D20	Entrada aviso de alarma cerca eléctrica
D25	Entrada sensor de movimiento

Realizado por: Guaranga, J. 2018

**Tabla 11-2:** Conexión Arduino Mega 2650 – Módulo reconocimiento de voz

Arduino Mega 2650	Módulo reconocimiento de voz Elechouse
Pin 12	RXD
Pin 13	TXD
GND	GND
VCC	VCC

Realizado por: Guaranga, J. 2018



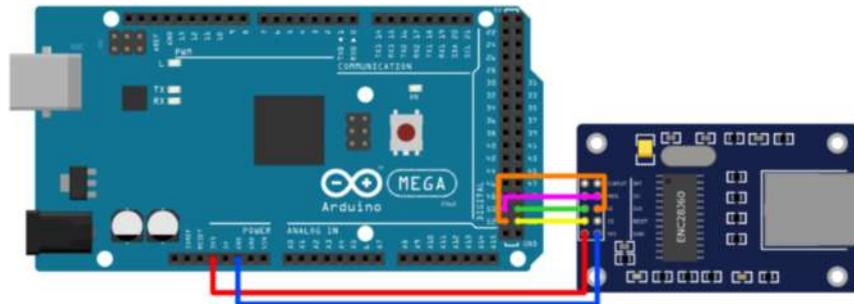
**Figura 22-2:** Conexión módulo reconocimiento de voz elechouse

Fuente: <https://goo.gl/2KEC1C>

**Tabla 12-2:** Distribución Pines Arduino Mega 2650 – Módulo Enc28j60

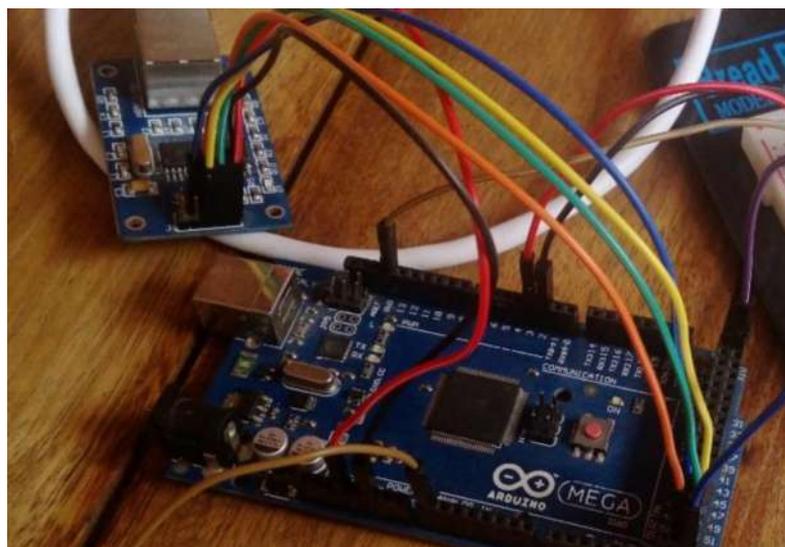
Arduino Mega 2650	Módulo Enc28j60
D53	CS
D51	SI
D50	SO
D52	SCK
RESET	RESET
D2	INT
3V3	VCC
GND	GND

Realizado por: Guaranga, J. 2018



**Figura 23-2:** Conexión Arduino Mega 2650 – Modulo Enc28j60

Fuente: <https://goo.gl/4iLurA>



**Figura 24-2:** Prueba conexión módulo Enc28j60

Realizado Por: Juan Guaranga

**Tabla 13-2:** Distribución pines arduino mega 2650 – Módulo relés

Arduino Mega 2650	Módulo relés
D21	In1
D22	In2
D23	In3
D24	In4
VCC	VCC
GND	GND

Realizado por: Guaranga, J. 2018

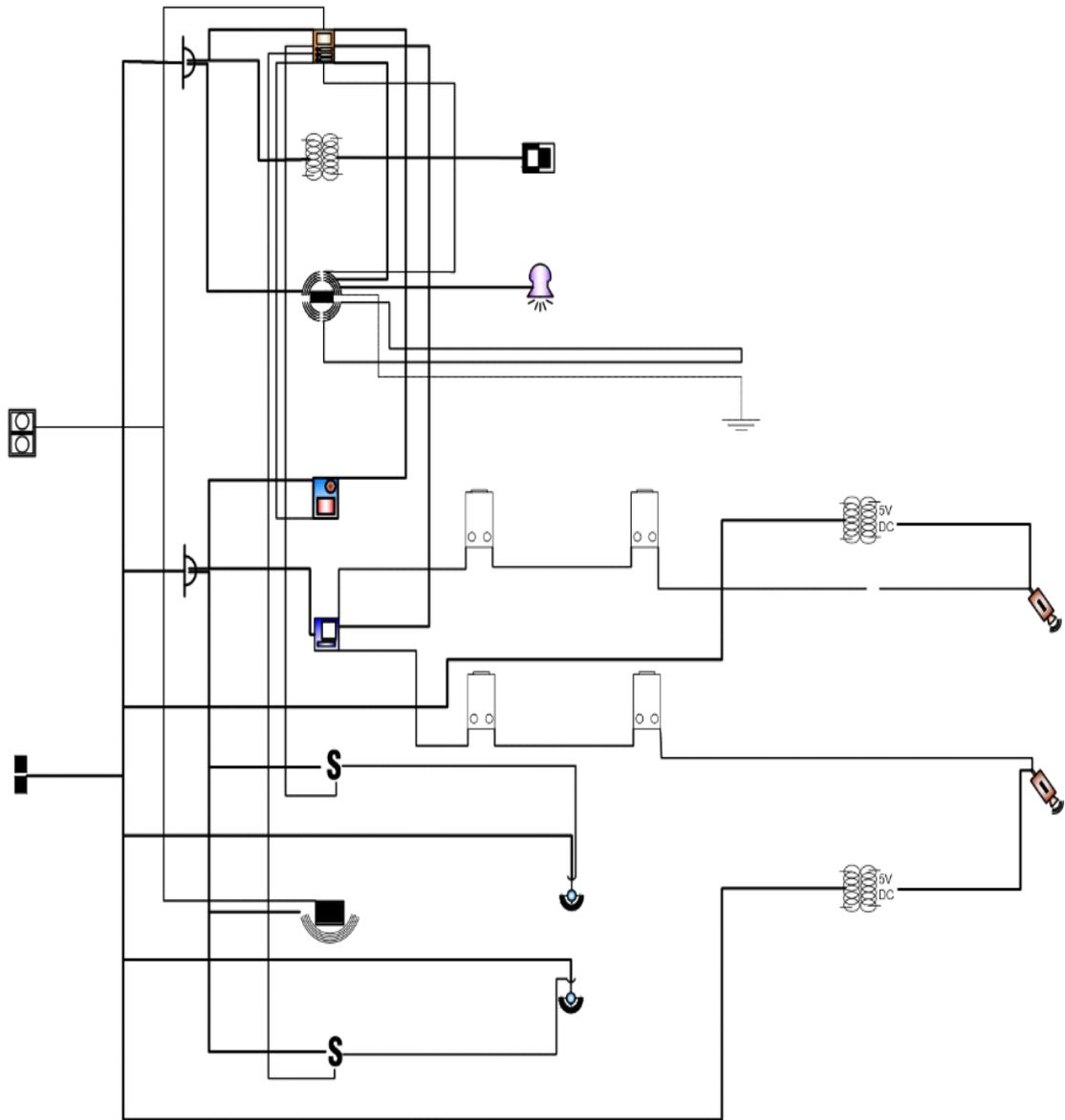


**Figura 25-2:** Conexión arduino mega 2650 – Módulo relés

Realizado por: Guaranga, J. 2018

### 2.3.2. *Conexión de equipos*

Antes de realizar la parte del desarrollo del software que gobierna el funcionamiento los diferentes equipos y módulos del sistema empleados, es necesario realizar la conexión adecuada de cada uno de los componentes externos (activos y pasivos) como cámaras, router, NVR, electrificador y conexión a tierra.



**Grafico 2-2:** Esquema conexión equipos y Simbología domótica empleada

Realizado por: Guaranga, J. 2018

Esquema completo en Anexo 2 (Planos arquitectónicos y domótico de la vivienda para el diseño del sistema de seguridad).



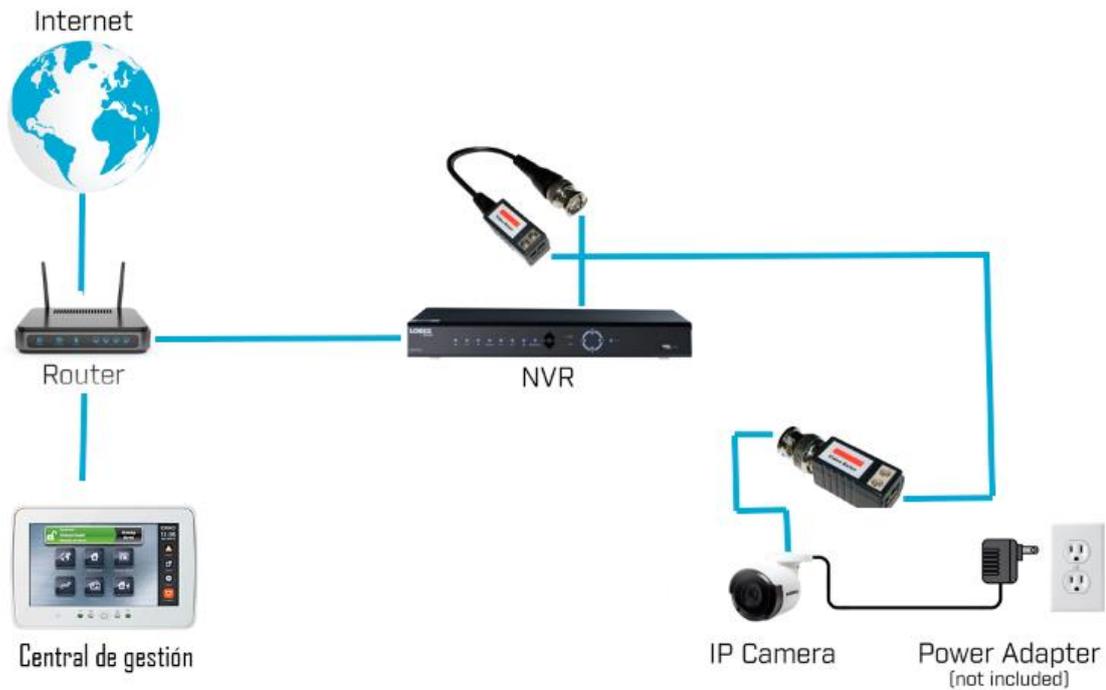
**Grafico 3-2:** Simbología domótica empleada

Realizado por: Guaranga, J. 2018

### 2.3.2.1. Conexión cámaras, NVR, Router y Central de Gestión

Las cámaras de seguridad Hikvision Ds-2CE16C0T-IRPF 2.8mm, son cámaras de tipo tubular que funcionan con un lente fijo de 2.8mm a una distancia de 5M máximo, para garantizar niveles de enfoque y nitidez óptimos, estas cámaras necesitan de un NVR al cual poder enviar la señal que captan, mediante cable UTP conectado a video balunes (uno a la salida de la cámara y otra a la entrada del NVR) con una capacidad de transmisión máxima de 100m.

Necesitan una alimentación de 12v dc suministrado por una extensión ubicada en el tercer piso de la vivienda. EL NVR es conectado mediante un cable UTP categoría 5E al router para tener acceso a internet y de la misma manera la central de gestión doméstico, se enlaza a la red mediante un cable de conexión UTP.



**Figura 26-22:** Conexión cámaras, NVR, Router y central de gestión

**Realizado por:** Guaranga, J. 2018

### 2.3.2.2. Conexión Cerca eléctrica, electrificador

Basado en el esquema proporcionado por el fabricante, se debe tomar en cuenta tres aspectos importantes en la conexión del electrificador con la cerca eléctrica instalada.

La cerca eléctrica debe constar con todos su elementos sin excepción; postes suficientemente bien sujetos para soportar las tensiones de los alambres con doble galvanizado, aros de supervisión en todas las entradas y salidas del alambre, etc.

Identificar la bornera de salida y entrada de alto voltaje en el electrificador, basando en el esquema proporcionado por el fabricante. Unimos a los cables doblemente galvanizados que recorren el cercado eléctrico, esto serán los dos cables superiores, pues el último se trata de tierra.

Conectar la tierra del sistema, pero no lo conectaremos igual que los dos cables anteriores. La tierra en un sistema por cercado eléctrica juega un papel muy importante tanto para su correcto funcionamiento como para el aumento de la vida útil del circuito elevador de voltaje.



**Figura 27-2:** Circuito Electrificador

**Realizado por:** Guaranga, J. 2018

Esto debido a los picos de alto voltaje que produce, libera energía por medio de la puesta a tierra, y si no se tiene una puesta a tierra correctamente instalada dichas pulsaciones afectaran principalmente al transformador reductor, pues debido a su bobinado atraerá los picos de alto voltaje hacia él, y con el tiempo quedara disfuncional.

Para implementar un correcto sistema a tierra debemos realizar los siguientes cálculos basados en la norma NOM-001-SEDE-1999 vigente actualmente en el Ecuador, que establece que *“Para establecimientos, comerciales industriales y residenciales la resistividad de la puesta a tierra debe ser menor o igual a 25 ohm”* también establece que en instalaciones domiciliarias *“El*

*conductor de puesta a tierra de un equipo debe ser de tamaño nominal no menor a 13,3mm<sup>2</sup> (6AGW)*”(Secretaría de Energía. & MEXICANA, 1999)

Con estas premisas procedemos a el cálculo de la corriente que libera el circuito en cada pulso, para lo cual nos basamos en las especificaciones del fabricante del electrificador, el cual muestra una potencia máxima de 4.32 W

$$I_n = \frac{\text{Potencia}}{V_{\text{max Cerca Eléctrica}}}$$

$$I_n = \frac{4.32}{13000}$$

$$I_n = 0.3323\text{mA}$$

Determinada la corriente disipada en cada pulso, y observando las hojas de datos de los conductores, podemos ver que el conductor 6AGW recomendado según la norma, sirve perfectamente para este fin, pues este soporta una corriente de 115A.

En cuanto a la resistividad de 25ohms recomendados, se debe conocer la resistividad del terreno el cual es de 200Ω-m (Química, Azulejos, & Ecuatoriana, 2010) aproximadamente, y se calcula la longitud necesaria para alcanzar la resistividad adecuada.

$$R = \frac{\rho}{L} \text{ Donde:}$$

*R = resistencia de la varilla puesta a tierra*

*ρ = Resistividad del terreno*

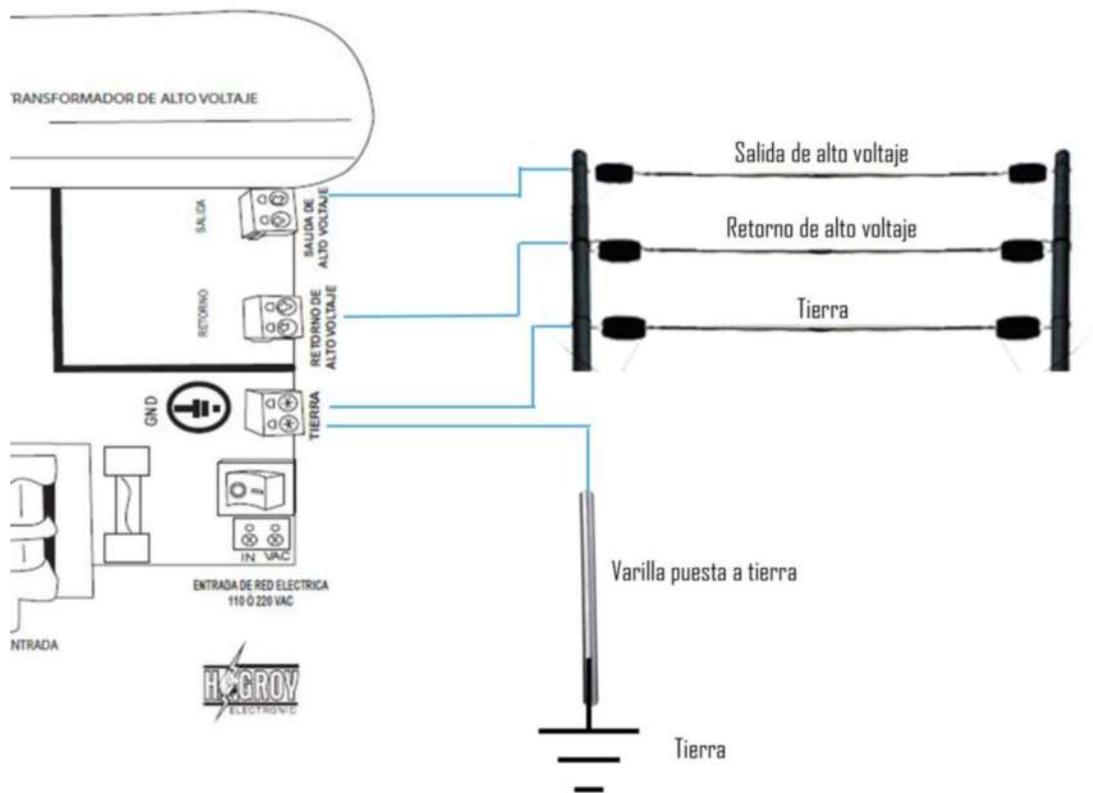
*L = Longitud de la varilla*

$$25 \Omega = \frac{200 \Omega - m}{L}$$

$$L = \frac{200 \Omega - m}{25 \Omega}$$

$$L = 8m$$

Dimensionado la puesta a tierra para un correcto funcionamiento del sistema, se procede a la conexión de la bornera de puesta a tierra, con el alambre doblemente galvanizado y la varilla calculada como indica el siguiente esquema:



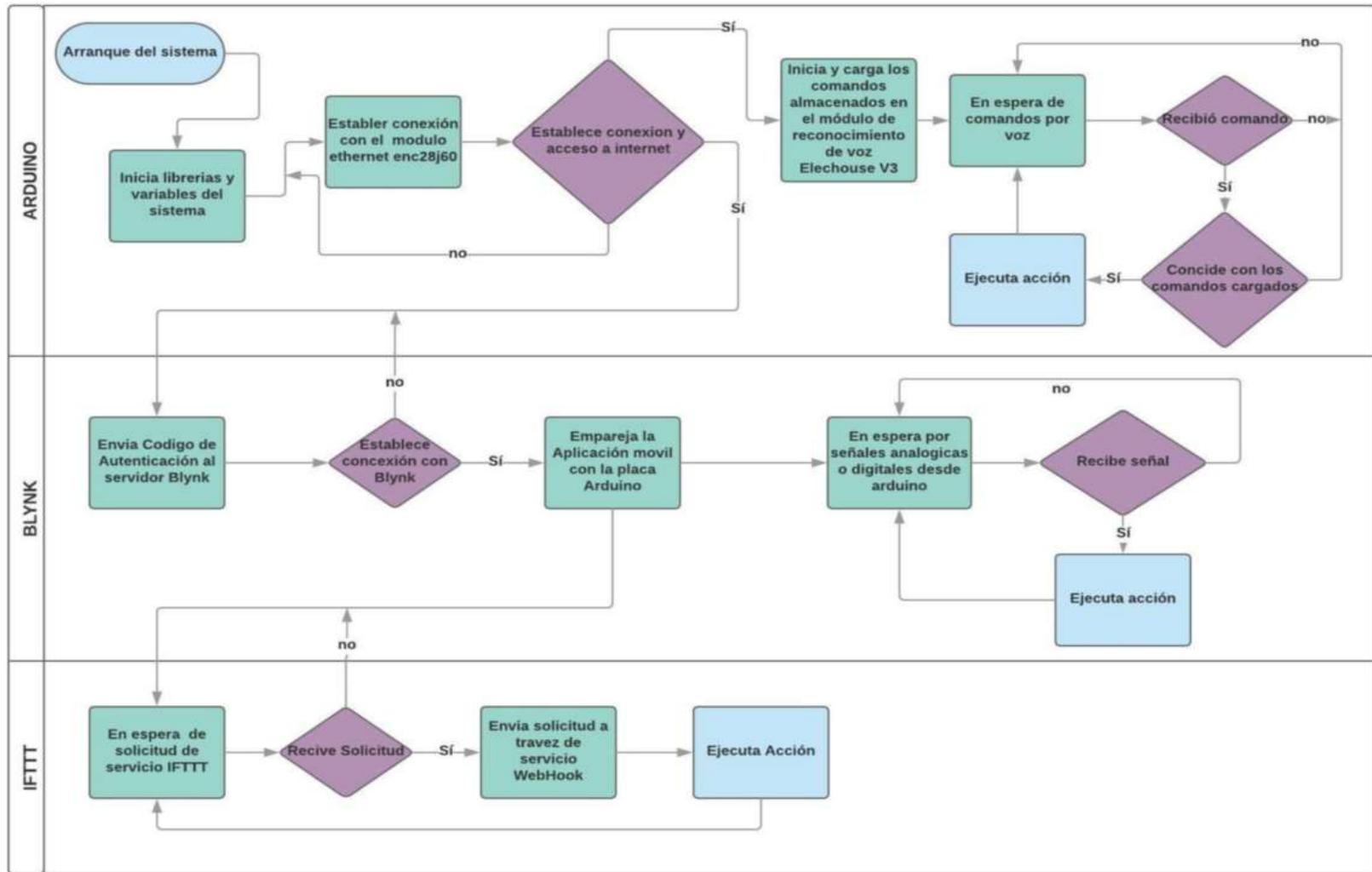
**Figura 28-2:** Conexión electrificador y puesta a tierra

Realizado por: Guaranga, J. 2018

### 2.3.3. *Desarrollo de Software*

Realizadas las conexiones preliminares de las placas de desarrollo para el control del sistema de seguridad proseguimos con la programación necesaria para la configuración y control de cada uno de los componentes dependientes de la placa central (Arduino Mega 2650).

El proceso de programación inicia con una representación esquemática del programa dando así una idea mucho más clara de que es lo que se pretende hacer.



**Figura 29-2:** Desarrollo de Software

Realizado por: Guaranga, J. 2018

Después de revisar el esquema presentado y con una clara idea del funcionamiento del programa se incluye las diferentes librerías que permiten la comunicación de Arduino con cada uno de los periféricos utilizados, también se incluyen librerías adicionales necesarias para la transmisión de datos entre la aplicación en desarrollo y la plataforma de programación Blynk con la placa de control central.

**Tabla 14-2:** Librerías incluidas en la IDE de programación arduino

Módulo	Librería
Módulo reconocimiento de voz Elechouse	VoiceRecognitionV3.h
Módulo Enc28j60	UIPEthernet.h
Conexión Servidor Blynk	BlynkSimpleUIPEthernet.h

Realizado por: Guaranga, J. 2018

### 2.3.3.1. Vinculación placa arduino mega 2650 con el servidor Blynk

Para la conexión con el servidor Blynk es necesario la inclusión de librerías pertenecientes a esta plataforma de desarrollo, las cuales se pueden conseguir en la misma página de Blynk que también incluyen ejemplos y tutoriales para el inicio del proceso de vinculación.



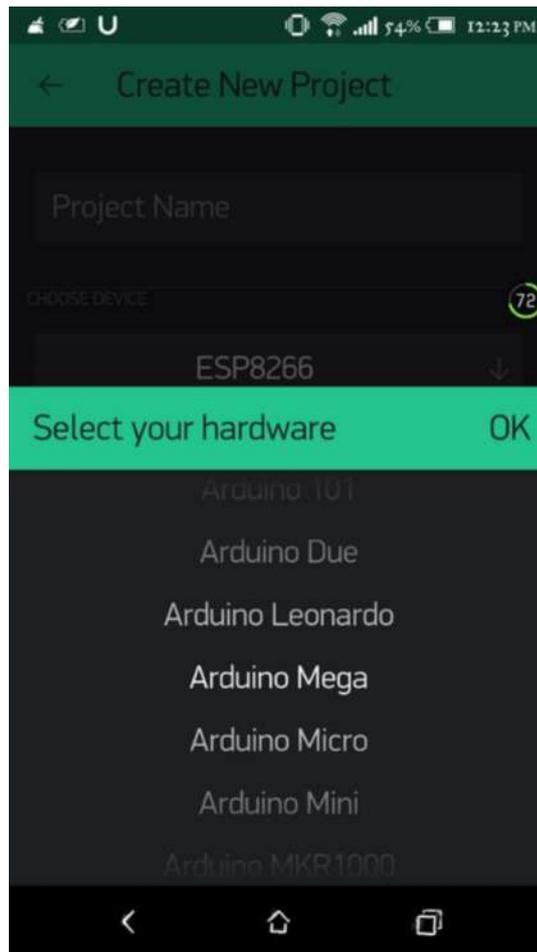
**Figura 30-2:** Librería Blynk

Fuente: <https://goo.gl/ybCjko>

Configurar en primer lugar el modulo Ethernet ENC28j60 el cual nos permite en primera instancia tener un acceso a internet para el envío y recepción de datos. Puesto que las conexiones necesarias están descritas en la etapas de desarrollo de hardware solo se debe incluir las librerías

<UIPEthernet.h> para su funcionamiento y probar si el router asigna una dirección IP, lo cual es probado conjuntamente con la conexión al servidor Blynk descrita a continuación.

Para la conexión con el servidor es necesario obtener el “Auth Token” proporcionado por la aplicación Blynk que se debe descargar desde las tiendas de aplicaciones play store (Android) o app store (IOS), una vez instalada la aplicación se obtiene el número de “Auth Token”.



**Figura 31-2:** Creación nuevo proyecto Blynk

**Realizado por:** Guaranga, J. 2018

Creamos un nuevo proyecto en la aplicación Blynk descargada, se escoge la placa con la cual se vinculará la aplicación y el servidor, el tipo de conexión e inmediatamente después de crear el proyecto la aplicación proporciona un código “número de Auth Token” que es el número con el cual se identifica a nuestra placa y a través del cual se envían y reciben los datos desde la aplicación a la placa y viceversa; el número de Auth Token también es enviado por correo electrónico.



necesitan una placa adicional para acondicionar la señal de acuerdo a las necesidades del usuario, siendo estas tipos analógicas, o digitales.

De acuerdo al esquema de programación y conexión de periféricos, podemos tener una idea acerca del número de entradas y salidas que se utilizarán en el proyecto, siendo estas de tipo entero, flotante o char.

```
int inputPin = A0; // Entrada señal cerca eletrica
int inputSensor = 3; //Entrada Sensor PIR
int inputCamaras = 4; // Entrada Alerta Cámaras
int inputSirena =5; // Entrada Sirena
char auth[] = "a8e7e71c64da414cbeabe97e012ebe98";
```

**Figura 34-2:** Declaración entradas directas

Realizado por: Guaranga, J. 2018

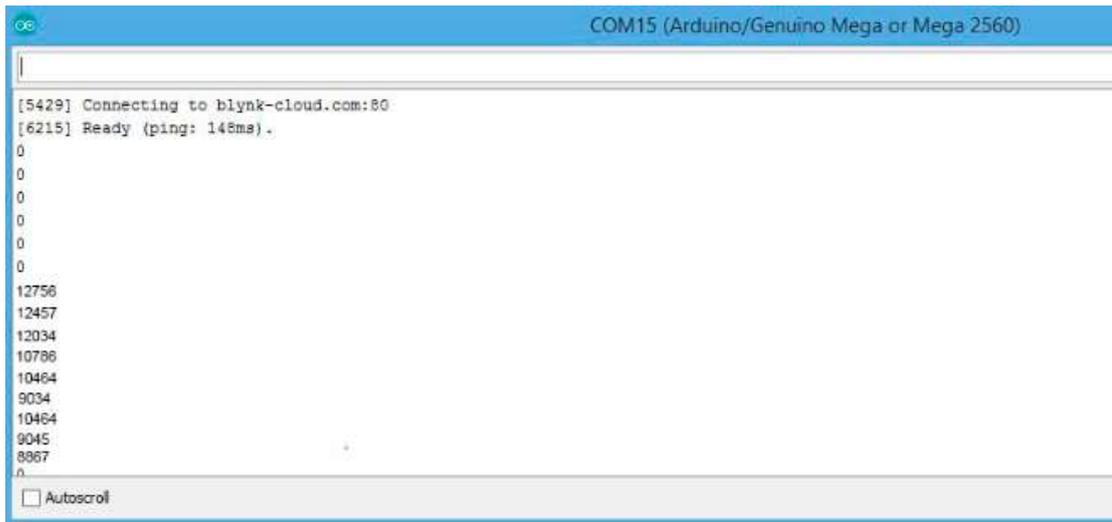
Se declara el estado inicial de los pines que controlan el módulo de relés para evitar posibles saltos o retornos de corriente de las bobinas, o en otros casos la inversión del sentido de alto o bajo que el código programado tiene como estado inicial a los relés, ocasionando así fallos graves en el funcionamiento del sistema, en caso de reinicio del sistema o fallos de energía, al recibir energización directamente activan sus bobinas de control, las cuales representan actuadores finales

```
//Asegurar nivel ALTO en cada entrada de rele
digitalWrite (2, RELAY_OFF);
digitalWrite (3, RELAY_OFF);
digitalWrite (4, RELAY_OFF);
digitalWrite (5, RELAY_OFF);
```

**Figura 35-2:** Declaración salidas módulo relés

Realizado por: Guaranga, J. 2018

Después de declarar todas las variables y tener vinculado nuestro dispositivo con el servidor de Blynk podemos programar las diferentes acciones que necesitamos que ejecute nuestro programa como el caso de recibir una entrada analógica o digital o recibir comandos o alertas desde el servidor.



**Figura 36-2:** Prueba lectura de sensores y voltaje de cerca

**Realizado por:** Guaranga, J. 2018

Dentro de los problemas encontrados al trabajar, Blynk no permite la utilización de comandos nativos de arduino dentro de la función Void Loop, lo cual representa un problema de gran consideración, pues todo sistema de monitoreo requiere de una ejecución continua, sin embargo Blynk proporciona herramientas adicionales para la ejecución de funciones repetitivas, como la función Blynk timer.

Esta función permite ejecutar sentencias en ciclos de tiempo controlados, a la vez que no saturan al servidor en caso de envío de paquetes u órdenes.

### SimpleTimer()

The constructor. You usually need only one SimpleTimer object in a sketch.

```
SimpleTimer timer;
```

### int setInterval(long d, timer\_callback f)

Call function f every d milliseconds. The callback function must be declared as `void f()`.

```
void repeatMe() {
  // do something
}

timerId = timer.setInterval(1000, repeatMe);
```

### int setTimeout(long d, timer\_callback f)

Call function f once after d milliseconds. The callback function must be declared as `void f()`. After f has been called, the interval is deleted, therefore the value timerId is no longer valid.

**Figura 37-2:** Función Timer Blynk

**Fuente:** Guaranga, J. 2018

Un aspecto importante a tener en cuenta para la etapa de programación son los eventos que el sistema debe tomar como anomalías, de esta manera ejecutar acciones que alerten al usuario acerca de dichas anomalías. Para el desarrollo de este proyecto se toman en cuenta los eventos (Ver. Tabla 15-2) que representan un posible riesgo para la seguridad de las personas, de la vivienda o de los equipos.

**Tabla 15-2:** Anomalías a tomar en cuenta en el diseño

Anomalía	Posible riesgo
Caída de tensión en cerca	Fallo de equipos, intento de intrusión
Detección movimiento cámaras	Intento de intrusión
Detección de movimiento sensor PIR	Intento de intrusión
Desconexión del sistema	Fallo de equipos, intento de intrusión
Activación Sirena	Intento de intrusión

Realizado por: Guaranga, J. 2018

Estos eventos se toman en cuenta para la programación de acciones en la etapa de desarrollo de aplicación, a través del envío de alertas al servidor, que a su vez se encarga de notificar al usuario acerca de la anomalía, permitiendo que pueda inmediatamente visualizar desde su dispositivo móvil lo que ocurre en la vivienda.

### 2.3.3.3. Programación módulo de reconocimiento de voz

```

~/
#include <SoftwareSerial.h>
#include "VoiceRecognitionV3.h"

/**
 * Connection
 * Arduino      VoiceRecognitionModule
 * 2  ----->   TX
 * 3  ----->   RX
 */
VR myVR(2,3);    // 2:RX 3:TX, you can choose your favourite pins.

/*****
** declare print functions */
void printSeparator();
void printSignature(uint8_t *buf, int len);
void printVR(uint8_t *buf);
void printLoad(uint8_t *buf, uint8_t len);
void printTrain(uint8_t *buf, uint8_t len);
void printCheckRecognizer(uint8_t *buf);
void printUserGroup(uint8_t *buf, int len);
void printCheckRecord(uint8_t *buf, int num);
void printCheckRecordAll(uint8_t *buf, int num);
void printSigTrain(uint8_t *buf, uint8_t len);
void printSystemSettings(uint8_t *buf, int len);
void printHelp(void);

```

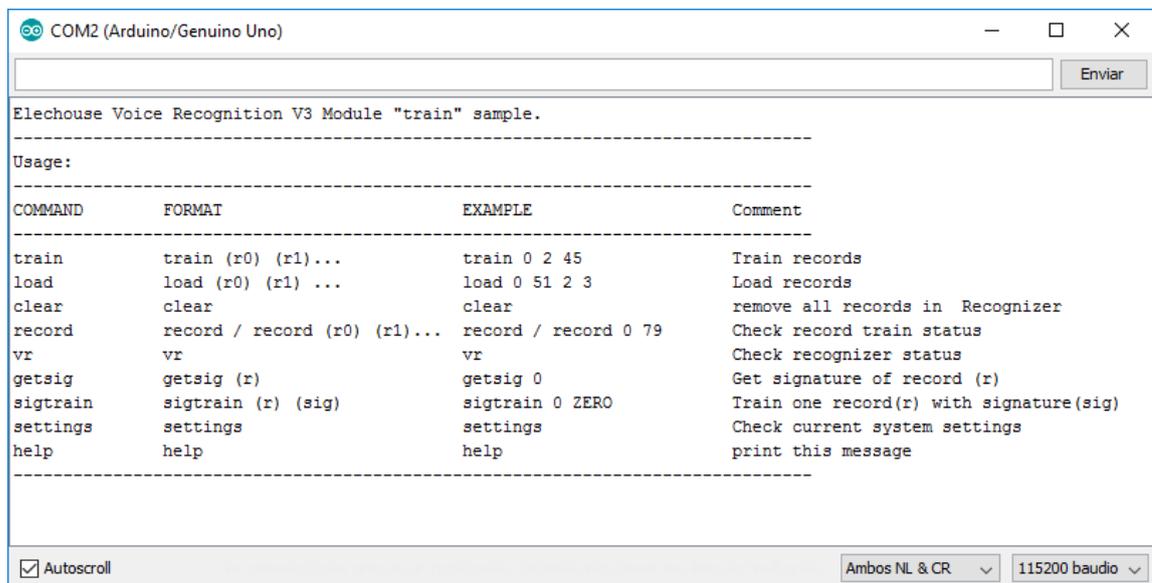
**Figura 38-2:** Código entrenamiento reconocimiento de voz

Realizado por: Guaranga, J. 2018

Para el reconocimiento de voz la placa Elechouse V3 trabaja conjuntamente con un micrófono monofónico que es el encargado de recoger la señal emitida por una persona.

Se incluye la librería para el manejo de la placa, el primer paso es el aprendizaje del reconocimiento de voz, que cumple con la tarea de entrenar la placa para que responda a los diferentes comandos de voz, el funcionamiento es muy sencillo, simplemente transfiere el programa a la placa he inmediatamente inicia el proceso de entrenamiento a través de comandos encargados de realizar el grabación de cada comando.

Comprobado que la placa responde a los comandos de voz establecidos, se realiza pruebas sencillas controlando un led a través de los comandos de voz, este mismo procedimiento fue utilizado para agregar comandos según nuestras necesidades.



```
COM2 (Arduino/Genuino Uno)
Elechouse Voice Recognition V3 Module "train" sample.
-----
Usage:
-----
COMMAND      FORMAT          EXAMPLE          Comment
-----
train        train (r0) (r1)...  train 0 2 45     Train records
load         load (r0) (r1) ...  load 0 51 2 3    Load records
clear        clear            clear            remove all records in Recognizer
record       record / record (r0) (r1)...  record / record 0 79  Check record train status
vr           vr              vr              Check recognizer status
getsig       getsig (r)       getsig 0         Get signature of record (r)
sigtrain     sigtrain (r) (sig)  sigtrain 0 ZERO  Train one record(r) with signature(sig)
settings     settings         settings         Check current system settings
help         help            help            print this message
-----
 Autoscroll
Ambos NL & CR
115200 baudio
```

**Figura 39-2:** Entrenamiento reconocimiento de voz

**Realizado por:** Guaranga, J. 2018

La etapa de entrenamiento es la etapa en la cual grabamos nuestra voz a través del micrófono, mediante el monitor serial se puede observar el progreso, por el cual el mismo programa nos guía paso a paso.

```
Elechouse Voice Recognition V2 Module "train" sample.
-----
settings
-----
Baud rate: 9600
Outpu IO Mode: Pulse
Pulse Width: 10ms
Auto Load: disable
Group control by external IO: disabled
-----
sigtrain 0 On
-----
Record: 0      Speak now
Record: 0      Speak again
Record: 0      Cann't matched
Record: 0      Speak now
Record: 0      Speak again
Record: 0      Success
Success: 1 ←
Record 0      Trained ←
SIG: On ←
```

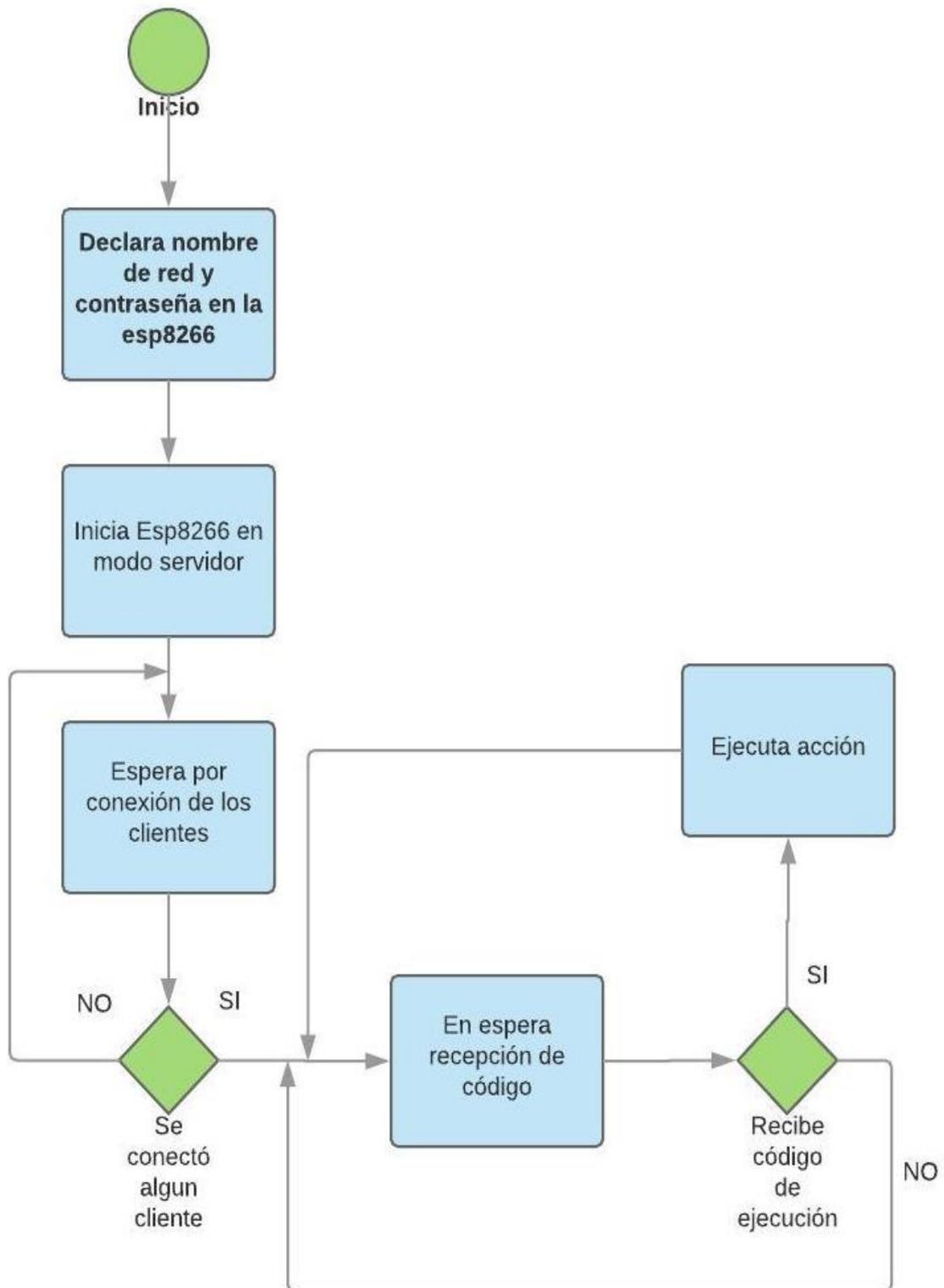
**Figura 40-2:** Entrenamiento: Grabación de comandos

**Realizado por:** Guaranga, J. 2018

Después de culminar la etapa de entrenamiento, y carga de los comandos guardados podremos integrar este código al programa principal con toda seguridad, y asignar funciones para cada comando de voz.

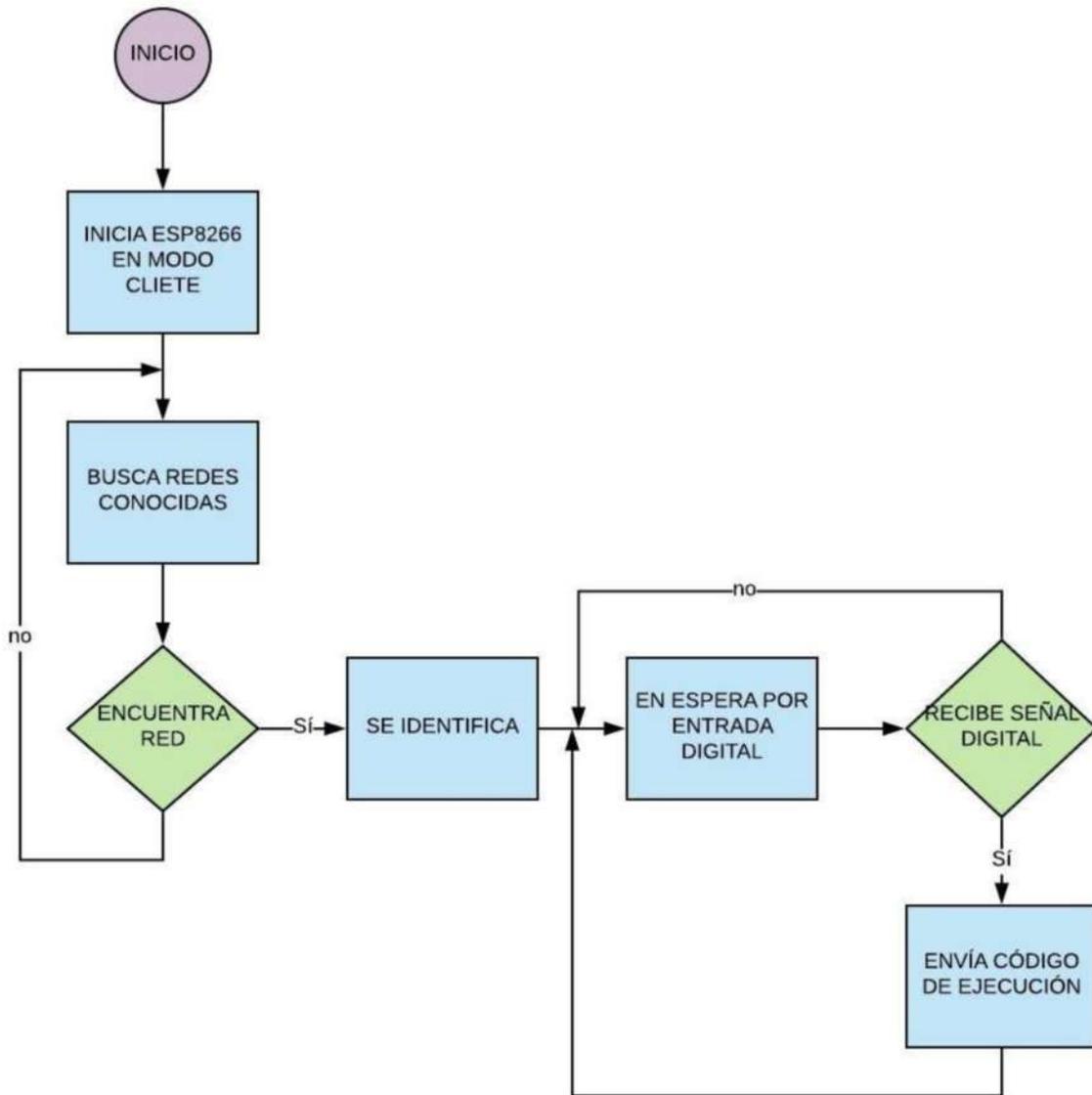
#### 2.3.3.4. Algoritmo central de sensores

Para la ejecución de la central de gestión de sensores es necesario la programación de un esp8266 en modo servidor y uno en modo cliente, de esta manera se logra la comunicación entre los diferentes dispositivos inalámbricos y el sistema principal.



**Figura 41-2:** Diagrama de flujo esp8266 servidor

Realizado por: Juan Guaranga



**Figura 42-2:** Diagrama de flujo esp8266 cliente

**Realizado por:** Juan Guaranga

### 2.3.3.5. Desarrollo aplicación móvil

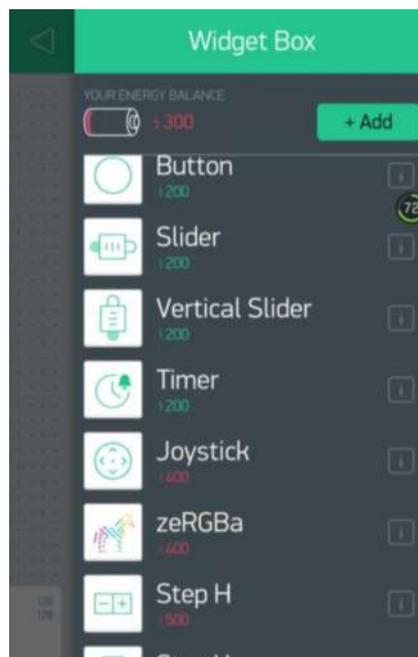
Ahora que las conexiones preliminares físicas, las configuraciones para el manejo de las placas a través de librerías y la declaración del vínculo para la comunicación entre la aplicación con el hardware a través del Auth Token están listas procedemos a desarrollar la aplicación con la cual gestionamos el sistema de seguridad.



**Figura 43-2:** Espacio de trabajo Blynk

**Realizado por:** Guaranga, J. 2018

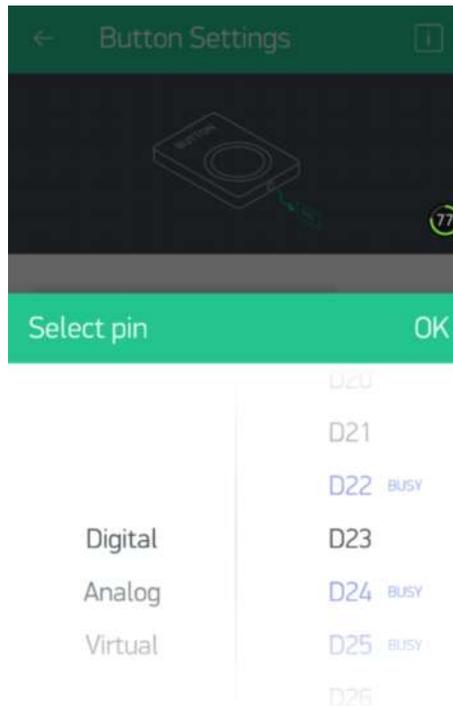
Blynk a través del código Auth Token proporcionado se encarga de gestionar todas las entradas y salidas declaradas en el código arduino, de la misma forma emplea variables virtuales en caso de tener la necesidad de conversión de datos o almacenar un historial temporal. Al poseer un modo de programación modular, basta con arrastrar y soltar los widgets dentro del espacio de trabajo, en operaciones sencillas no requiere añadir código adicional en la IDE de arduino.



**Figura 44-2:** Lista de widgets disponibles

**Realizado por:** Guaranga, J. 2018

Para declarar las variables que se utilizan dentro de la IDE Arduino se debe seleccionar en la pantalla el número de variable a utilizar e inmediatamente esta estará vinculada con el código fuente de Arduino.



**Figura 45-2:** Selección variables a utilizar

**Realizado por:** Guaranga, J. 2018

Una de las mayores ventajas que nos ofrece Blynk es el uso de variables virtuales, las cuales podemos utilizar como para envío y recepción de datos desde Arduino a Blynk y viceversa, básicamente son canales de comunicación entre la placa de desarrollo y la plataforma, además que las variables virtuales pueden ser usadas con librerías de arduino para el control de servo motores, paneles LCD, etc. Gracias a esto la facilidad para el comando de los sensores y actuadores es mucho más fácil de implementar en el sistema.

Al finalizar el trabajo se exporta el trabajo como una aplicación independiente lista para ser publicada en cualquier plataforma de aplicaciones disponibles en el mercado.

#### 2.3.3.6. Integración de servicios a través de IFTTT

Como parte final de la etapa de desarrollo se realiza una integración de hardware y software a través de IFTTT, logrando de esta manera controlar todo el sistema con comandos de voz desde el teléfono inteligente del usuario y no solo a través del módulo de reconocimiento de voz que obligatoriamente requiere que el usuario esté presente en su hogar, sino desde cualquier lugar donde se encuentre el usuario con su smartphone.

Para lo cual debemos seguir unos sencillos pasos en la plataforma de IFTTT que se detallan a continuación.

- Iniciar sesión en la página de IFTTT o acceder mediante su aplicación móvil.
- Crear una nueva Applet.
- Se despliega una pantalla con la frase “If this then that”, damos click en “this” y e inmediatamente se muestra una ventana con todos los servicios que se pueden integrar.
- Elegir el servicio que se desea integrar (en nuestro caso elegiremos google assistant como aplicación para enviar los comandos de voz).
- Elegimos el tipo de comando que deseamos ejecutar con google assistant.
- Completamos los campos con el comando deseado.
- Ahora damos click en la opción “that” de la frase “If this, then that”.
- Elegimos “Webhooks” como servicio de integración y envío de datos.
- Llenamos el formulario con los datos del servidor de Blynk y la variable que deseamos controlar en arduino.
- Damos click en aceptar y listo, google assistant está integrado con nuestro proyecto.



**Make a web request**

This action will make a web request to a publicly accessible URL. NOTE: Requests may be rate limited.

**URL**

`http://188.166.206.43/a8e7e71c64da414cbeabe97e012ebe98/uptime/D22`

Surround any text with "<<<" and ">>>" to escape the content

**Add ingredient**

**Method**

GET

The method of the request e.g. GET, POST, DELETE

**Content Type**

application/json

**Figura 46-2:** Configuración Webhook

Realizado por: Guaranga, J. 2018

## 2.4. Etapa de implementación

El procedimiento para la implementación del proyecto dentro del perímetro de la vivienda se lleva a cabo siguiendo la planificación previamente revisada, una vez comprobados los circuitos de manejo del sistema, se reemplaza los componentes provisionales de prueba como leds, pulsadores focos, por componentes reales, de la siguiente manera:

### 2.4.1. Cerca eléctrica y perímetro de la vivienda

La cerca eléctrica se implementa a una altura de 2,5m desde el suelo para asegurar la integridad de los habitantes del hogar y de peatones, en una longitud de 15m que rodea el patio trasero de la vivienda, cuenta con 3 hilos templados con postes tensores cada 2,5m, alimentados por un electrificador de 13000V regulables (7000V – 13000V) colocado en la parte superior de la pared de la bodega.



**Figura 47-2:** Cerca eléctrica en el perímetro de la vivienda

Realizado por: Guaranga, J. 2018

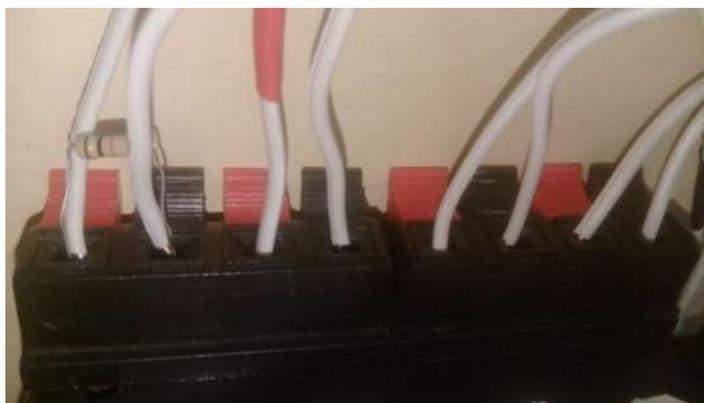
La conexión a tierra de la cerca eléctrica es realizada con un cable AWG #6 conectado a una varilla enterrada a lo largo de 8 metros en el patio trasero. Las conexiones para el electrificador son conectadas mediante la identificación de cada uno de los cables doblemente galvanizados en la bodega de la vivienda.



**Figura 48-2:** Varilla de puesta a tierra

Realizado por: Juan Guaranga

Después de realizar la conexión de la cerca eléctrica se procede a la identificación de las salidas del electrificador necesarias que servirán para el control y censado de la cerca desde la central de gestión domótica.



**Figura 49-2:** Entradas central de gestión

Realizado por: Juan Guaranga

#### **2.4.2. Sistema de video vigilancia**

La central de video vigilancia es implementada en el cuarto de estudio de la vivienda, junto al router con una conexión directa mediante cable UTP.

Las cámaras se ubican en el tercer piso de la vivienda enfocadas en sentido sur-este y nor-este, para una cobertura máxima del patio posterior y lateral. Para la conexión de transmisión de video y alimentación eléctrica es necesario adaptar una conexión tipo PoE debido a la distancia que existe entre el NVR y las cámaras.



**Figura 50-2:** Conexión cable PoE – cámaras de video vigilancia

**Realizado por:** Juan Guaranga

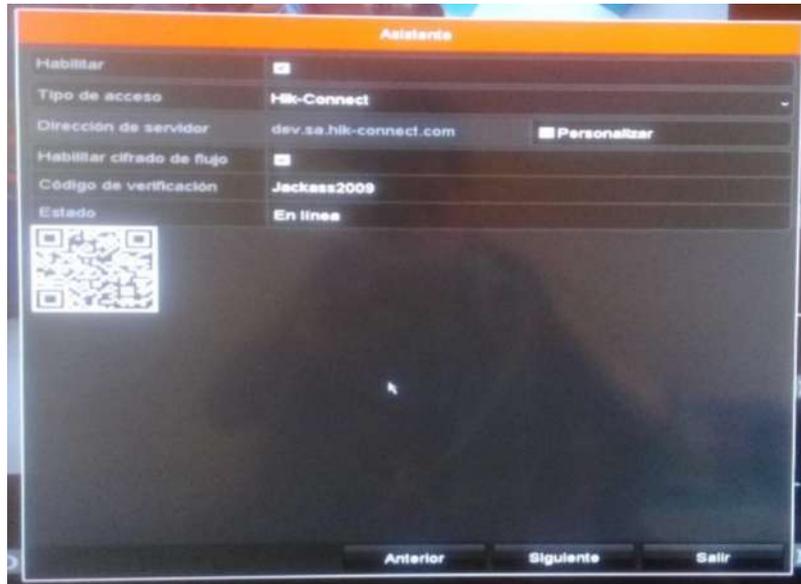
De esta manera mediante la conexión de video balunes y la conexión adaptada de PoE por cada cable UTP categoría 5E se puede transmitir y alimentar dos cámaras, cada una con su caja de revisión con protección contra lluvia y polvo.



**Figura 51-2:** Cámaras de video vigilancia con cajas de revisión

**Realizado por:** Juan Guaranga

Se concluye el proceso de conexión de las cámaras con la central de monitoreo, se configura el NVR para que permita el acceso a los datos de video mediante su aplicación móvil Hik-Connect, desde cualquier parte del mundo, para lo cual es necesario (por seguridad) asignar una IP fija y registrar al NVR en la página del fabricante, la cual proporciona manuales con las configuraciones necesarias.



**Figura 52-2:** Configuración NVR – Hik-Connect

Realizado por: Juan Guaranga

Después de comprobar la correcta conexión del NVR con la plataforma de visualización Hik-Connect, se realiza la configuración de la detección de movimiento, controlando sensibilidad, alerta de alarma y horarios de detección.



**Figura 53-2:** Detección de movimiento cámaras de video vigilancia

Realizado por: Juan Guaranga

Como parte final del sistema de video vigilancia se toma una salida de voltaje de la placa del NVR cuando la alarma de detección de movimiento sea activada, dado que el fabricante no proporciona una salida nativa, a la vez que demostramos que este sistema es perfectamente funcional con cualquier tipo de NVR, aun cuando este no presente salidas nativas para alarmas.



**Figura 54-2:** Señal de salida de alerta NVR

Realizado por: Juan Guaranga

### 2.4.3. Centrales de gestión domótica

La central de gestión domótica principal es implementada dentro de “una caja de proyectos” que se puede adquirir en cualquier tienda de electrónicos y adaptada posteriormente de acuerdo las necesidades del usuario.

Como punto de partida se implementa dentro de la caja, las entradas y salidas que gobiernan el funcionamiento de todo el sistema, compuesto por 4 entradas y 4 salidas, distribuidos de la siguiente manera.

**Tabla 16-2:** Entradas y salidas del sistema

Salidas	Tipo	Entradas	Voltaje
Iluminación patio trasero	Relé	Alto voltaje electrificador	13000
Iluminación estudio	Relé	Alarma electrificador	10 - 14
Trasformador chapa eléctrica	Relé	Alarma NVR (detección de movimiento)	5
Sirena	Relé	Central de sensores	3.3

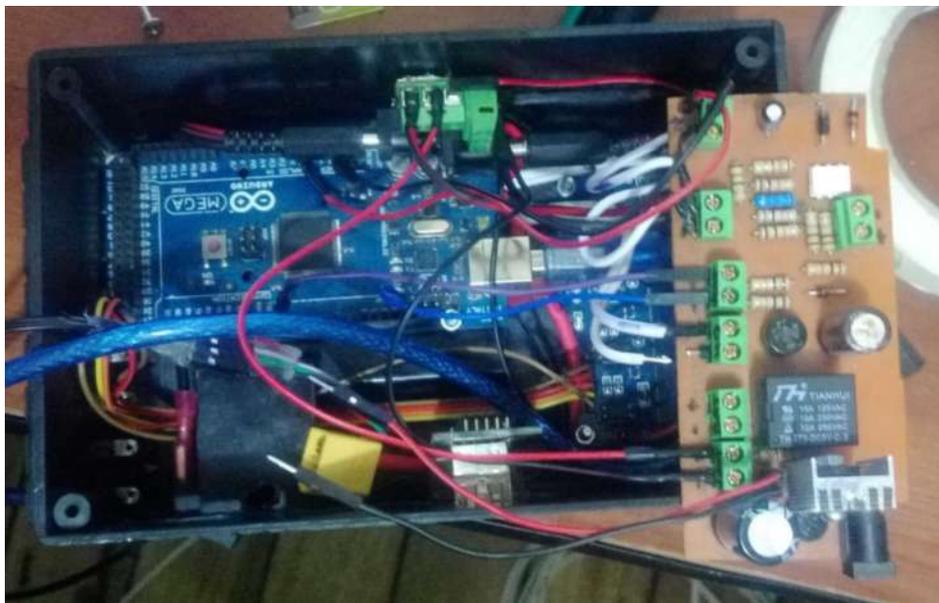
Realizado por: Juan Guaranga



**Figura 55-2:** Entradas central de gestión

Realizado por: Juan Guaranga

Seguido de esto se colocan los módulos que permiten la conexión a internet y reconocimiento de voz junto a la placa de desarrollo Arduino Mega 2650 dentro de la caja de una manera que facilita su conexión y/o mantenimiento. Adicionalmente se incorpora la placa de control, tratamiento y alimentación, la cual permite un correcto funcionamiento del sistema.



**Figura 56-2:** Placa de manejo de señales de voltaje – fuente de alimentación

Realizado por: Juan Guaranga

También se implementa una placa de distribución de energía para los módulos que conforma la central de gestión, junto a una fuente de energía alterna, que, en caso de falla de energía garantiza el funcionamiento del sistema entre 3 y 4 horas.

La central de gestión de sensores funciona dentro de una caja de proyectos de menor tamaño, pues solo requiere el chip IoT esp8266 en su interior y una salida por cable que está conectada a la central de gestión principal. Esta central posee una antena omnidireccional adaptada, soldada a la antena principal provista en la placa del módulo, pues de esta manera se garantiza una mejor cobertura del área evitando pérdidas de datos en la comunicación con los periféricos.



**Figura 57-2:** Central de gestión de sensores

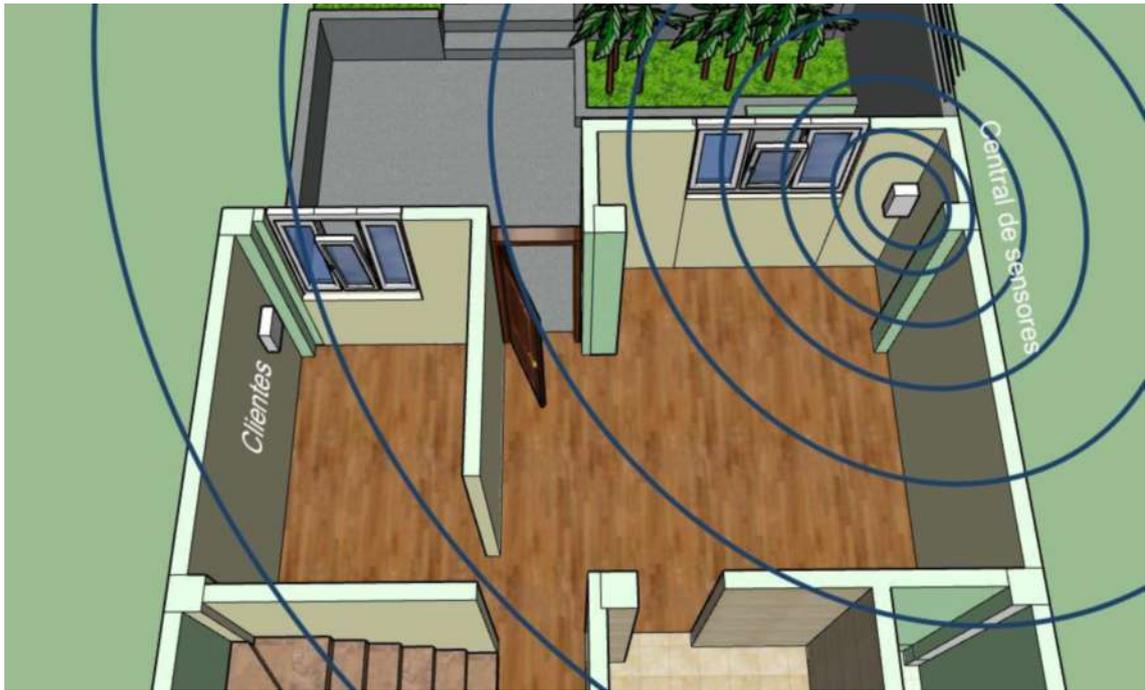
Realizado por: Juan Guaranga

Finalmente como complemento de la central de gestión de sensores se maneja dos ESP8266 en modo servidor, para su uso en caso de ampliación de la de red de sensores.



**Figura 58-2:** Clientes IoT para sensores

Realizado por: Juan Guaranga



**Figura 59-2:** Ubicación y cobertura central de sensores.

**Realizado por:** Juan Guaranga

## CAPÍTULO III

### 3. ANÁLISIS Y RESULTADOS

#### 3.1. Determinación de indicadores

En base a las necesidades que se esperaba cubrir y los resultados esperados en cuanto se refiere al rendimiento del sistema de seguridad diseñado, se procedió a establecer indicadores a tomar en cuenta para la realización de pruebas estadísticas.

**Tabla 2-3:** Desempeño esperado en un sistema de seguridad

Objetivo Principal	Limitaciones
Rapidez	Precisión
Flexibilidad	Dos o más medios de transmisión de datos
Confiabilidad	Alto rango de cobertura inalámbrica

Fuente: (Pereira Rama, 2014)

En base a la (Tabla 1-3), del desempeño esperado en un sistema de seguridad, para el presente proyecto se tomó en consideración como indicadores de rendimiento a la confiabilidad, la rapidez y la precisión siendo definidos cada uno de estos como:

- Confiabilidad.- Característica de un dispositivo de mostrar en cantidad mínima o nula errores en la medición o transmisión de datos.
- Rapidez.- Tiempo promedio que tarda un determinado dispositivo en capturar datos o recibir órdenes, procesar la información y ejecutar una acción de acuerdo a los protocolos establecidos para su funcionamiento, los tiempo de respuesta se tomaron en segundos.
- Precisión.- Característica de un dispositivo a sufrir cierta variación respecto a una medida real de un mismo objeto o magnitud, las variaciones de medidas se tomaron en voltios.

##### 3.1.1. Número de Pruebas

Tomando en cuenta que se desconocía el tamaño de la población pues en un experimento en el cual la población no está delimitada a un sector específico, sino que depende de factores externos

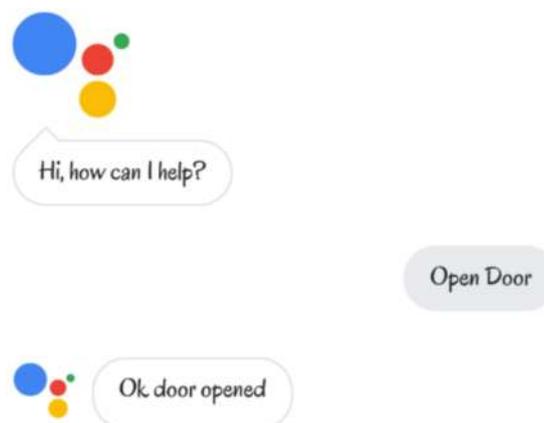
como eventos que provoquen la ejecución de acciones en el sistema y por consiguiente el aumento de la población de muestras sin depender del tiempo, no se puede establecer una población determinada, por lo cual debe realizar un muestreo probabilístico, basándonos en el principio de equi probabilidad y enfocándonos en los objetivos del proyecto se toma la muestra por juicio y conveniencia, debido al tiempo que requiere la toma de cada uno de los datos.

Los datos fueron tomados, teniendo en cuenta las condiciones en las cuales se desempeñan los sistemas de seguridad compuestos por cercas eléctricas y sistemas de video vigilancia enfatizando en errores comunes de instalación de equipos cometidos por personal no capacitado, que, ignora la importancia de conceptos como una correcta acometida de puesta a tierra, entre otros.

Debido a lo expuesto, se estimó una media poblacional de 30 muestras para el proceso de reconocimiento de voz para la confiabilidad, tiempo de respuesta del sistema desde que se produce una alerta hasta que el usuario es notificado, y valores de voltaje medidos con un correcto sistema de puesta a tierra, que fueron analizadas en base a las pruebas chi-cuadrado y t-student para medias independientes.

### 3.2. Prueba de confiabilidad del sistema

Para las pruebas de confiabilidad del sistema fueron tomados, de reconocimiento de comandos de voz con dos sistemas, el primero con el módulo de reconocimiento de voz elechouse y el segundo con el asistente de google, 30 datos con cada uno. Con el fin de comprobar la confiabilidad de cada uno de los métodos de reconocimiento implementados en el sistema. Por lo cual fue realizado un análisis estadístico tipo chi cuadrado.



**Figura 1-3:** Reconocimiento comandos de voz google assistant

**Realizado por:** Guaranga, J. 2018

**Tabla 2-3:** Muestra prueba de confiabilidad

Detección comando de voz “encender”			
Comando	Trama arduino	Módulo elechouse	Google Assistant
Encender	45-6E-63-65-6E-00-00-05-4F-65-45-0A	NO	SI
Encender	45-6E-63-65-6E-00-00-05-4F-65-65-72	NO	SI
Encender	45-6E-63-65-6E-64-65-72	SI	SI
Encender	45-6E-63-65-6E-43-05-4F-65-65-72-A0	SI	SI
Encender	45-6E-63-65-20-6E-65-72	NO	SI
Encender	45-6E-63-65-6E-64-65-72	SI	SI
Encender	AA-07-OD-OO-FF-06-06-OO-OA	NO	SI
Encender	AA-07-OD-OO-FF-06-06-OO-OA	NO	SI
Encender	45-6E-63-65-20-6E-65-72	NO	SI
Encender	45-6E-63-65-6E-00-00-05-4F-65-65-72	NO	SI
Encender	45-6E-63-65-6E-00-00-05-4F-65-65-72	NO	SI
Encender	45-6E-63-65-6E-64-65-72	SI	SI
Encender	45-6E-63-65-6E-64-65-72	SI	SI
Encender	45-6E-63-65-6E-64-65-72	SI	SI
Encender	45-6E-63-65-6E-00-00-05-4F-65-65-72	NO	SI
Encender	45-6E-63-65-6E-00-00-05-4F-65-65-72	NO	SI
Encender	45-6E-63-65-6E-64-65-72	SI	SI
Encender	45-6E-63-65-6E-00-00-05-4F-65-45-0A	NO	SI
Encender	45-6E-63-65-6E-00-00-05-4F-65-45-0A	NO	SI
Encender	45-6E-63-65-6E-00-00-05-4F-65-45-0A	NO	SI
Encender	45-6E-63-65-6E-64-65-72	SI	SI
Encender	45-6E-63-65-6E-00-00-05-4F-65-45-0A	SI	SI
Encender	45-6E-63-65-6E-00-00-05-4F-65-45-0A	NO	SI
Encender	45-6E-63-65-6E-64-65-72	SI	SI
Encender	45-6E-63-65-6E-00-00-05-4F-65-45-0A	NO	SI
Encender	45-6E-63-65-6E-64-65-72	SI	SI
Encender	45-6E-63-65-6E-64-65-72	SI	SI
Encender	45-6E-63-65-20-6E-64-70	NO	SI
Encender	45-6F-63-65-20-6E-65-72	NO	SI
Encender	45-6E-64-65-20-6E-65-72	NO	NO
Encender	4A-6E-63-65-20-6E-65-72	NO	SI

Realizado por: Guaranga, J. 2018

Por lo cual establecemos las hipótesis a probar:

- Ho: Existe igual confiabilidad en el reconocimiento de comandos de voz por el módulo elechouse y google assistant

- Ha: No existe igual confiabilidad en el reconocimiento de comandos de voz por el módulo elechouse y google assistant.

**Tabla 3-3:** Contingencia: Recepción de datos

			Recepción de datos entre: módulo de reconocimiento elechouse y google assistant		Total
			Módulo elechouse	Google assistant	
Se detectó el comando de voz	NO	Recuento	19	4	23
		% dentro de Se detectó el comando de voz	82,6%	17,4%	100,0%
		% dentro de Recepción de datos entre: módulo de reconocimiento elechouse y google assistant	61,3%	12,9%	37,1%
		% del total	30,6%	6,5%	37,1%
	Si	Recuento	12	27	39
		% dentro de Se detectó el comando de voz	30,8%	69,2%	100,0%
		% dentro de Recepción de datos entre: módulo de reconocimiento elechouse y google assistant	38,7%	87,1%	62,9%
		% del total	19,4%	43,5%	62,9%
Total	Recuento	31	31	62	
	% dentro de Se detectó el comando de voz	50,0%	50,0%	100,0%	
	% dentro de Recepción de datos entre: módulo de reconocimiento elechouse y google assistant	100,0%	100,0%	100,0%	
	% del total	50,0%	50,0%	100,0%	

Realizado por: Guaranga, J. 2018

Para la prueba de hipótesis se asume un nivel de significancia del 5%.  $\alpha = 0,05$  dando como valor esperado de éxito un 95% por cada 1000 muestras, en la toma de datos del muestreo probabilístico

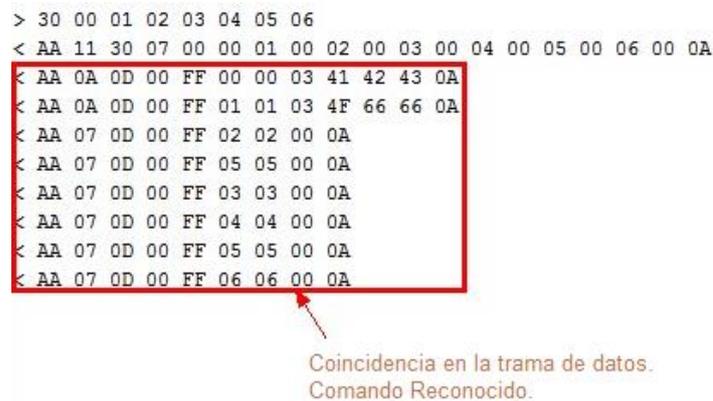
**Tabla 4-3:** Pruebas de chi-cuadrado

	Valor	gl	Sig. asintótica (bilateral)	Sig. exacta (bilateral)	Sig. exacta (unilateral)
Chi-cuadrado de Pearson	15,552 <sup>a</sup>	1	,000		
Corrección por continuidad	13,547	1	,000		
Razón de verosimilitudes	16,552	1	,000		
Estadístico exacto de Fisher				,000	,000
Asociación lineal por lineal	15,301	1	,000		
N de casos válidos	62				

a. 0 casillas (0,0%) tienen una frecuencia esperada inferior a 5. La frecuencia mínima esperada es 11,50.  
 b. Calculado sólo para una tabla de 2x2.

Realizado por: Guaranga, J. 2018

Dado que la probabilidad de la prueba chi cuadrado 0,00 es decir es menor al nivel de significancia se rechaza la hipótesis nula y se acepta la alternativa, por lo cual se concluye que los diferentes métodos de reconocimiento de comandos de voz presentan confiabilidades significativamente diferentes.



**Figura 2-3:** Reconocimiento comandos de voz módulo elechouse

Realizado por: Guaranga, J. 2018

De acuerdo a las tablas mostradas se puede decir que Google assistant es la mejor opción para el procesamiento de comandos de voz para el sistema pues posee una efectividad en el reconocimiento de comandos de voz de más del 80%.

### 3.3. Prueba de rapidez del sistema

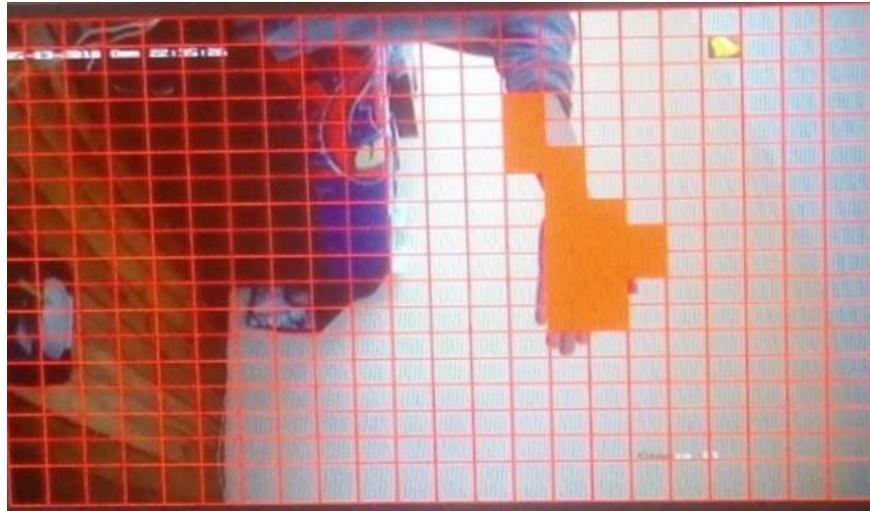
Para las pruebas de rapidez del sistema fueron tomados 30 datos, con el sistema conectado a un adaptador de voltaje 110V AC a 5V DC y 30 datos conectado a una batería lipo de 1.5A a 7.5 V.

**Tabla 5-3: Muestra rapidez del sistema**

Rapidez	
Tiempo de Respuesta Alerta	
Adaptador DC (seg)	Batería Lipo (seg)
2,75	12,30
4,01	14,76
2,16	14,20
2,81	13,82
2,64	13,54
4,40	13,13
4,35	13,70
4,74	14,35
4,37	13,43
3,48	14,40
2,75	12,76
2,72	13,32
2,41	12,82
2,66	14,00
2,64	13,84
3,40	13,80
3,99	13,97
4,97	15,00
4,63	13,29
4,67	14,17
4,28	12,74
3,18	14,74
4,71	13,14
4,84	13,47
4,98	12,44
3,31	13,34
2,41	12,56
4,72	12,19
5,00	14,88
4,41	14,86
4,15	14,27

Realizado por: Guaranga, J. 2018

Con el fin de comparar si existe una diferencia significativa en los tiempos de respuesta entre estos dos casos. Por lo cual fue realizado un análisis estadístico en base a la prueba T para muestras independientes.



**Figura 3-3:** Detección de movimiento cámara

**Realizado por:** Guaranga, J. 2018

### 3.3.1.1. Prueba de normalidad a las distribuciones

A continuación establecemos una prueba de normalidad a las distribuciones de los datos obtenidos, de esta forma sabremos si es posible realizar una prueba t-student.

- $H_0$ : Los datos obtenidos en el experimento tiempos de respuesta siguen una distribución normal.
- $H_a$ : Los datos obtenidos en el experimento de tiempos de respuesta no siguen una distribución normal.

**Tabla 6-3:** Pruebas de normalidad

Tiempo de respuesta del sistema	Alimentación al circuito	Kolmogorov-Smirnov <sup>a</sup>			Shapiro-Wilk		
		Estadístico	gl	Sig.	Estadístico	gl	Sig.
	Batería lipo	,110	31	,200*	,958	31	,263
	Adaptador DC	,147	31	,084	,930	31	,044

\*. Este es un límite inferior de la significación verdadera.

a. Corrección de la significación de Lilliefors

**Realizado por:** Guaranga, J. 2018

Según la prueba de normalidad de Shapiro-Wilk la probabilidad en el tipo de fuente de alimentación en el experimento de tiempos de respuesta es mayor al nivel de significancia, por lo cual, acepta la hipótesis nula, por lo que se concluye que los datos siguen una distribución normal.

### 3.3.1.2. Prueba de homogeneidad de varianzas

- $H_0: \sigma_A^2 = \sigma_B^2$  (Las varianzas en el experimento de tiempos de respuesta de acuerdo al tipo de alimentación son iguales.)
- $H_a: \sigma_A^2 \neq \sigma_B^2$  (Las varianzas en el experimento de tiempos de respuesta de acuerdo al tipo de alimentación no son iguales.)

Establecemos el nivel de significancia en  $\alpha = 0,05$  y calculamos el estadístico de prueba

**Tabla 7-3:** Prueba de homogeneidad de varianzas

Estadístico de Levene	gl1	gl2	Sig.
,772	1	60	,383

Realizado por: Guaranga, J. 2018

Según la prueba de Levene, los datos los tiempos de respuesta de los dos tipos de fuente de alimentación tienen las mismas varianzas.

Por lo tanto concluimos que es viable realizar un análisis por medio de T-student para muestras independiente.

### 3.3.1.3. Prueba T-para muestras independientes

#### Definir la hipótesis

- $H_0$ : Los tiempos de respuesta del sistema de acuerdo a cada fuente de alimentación son iguales.
- $H_a$ : Los tiempos de respuesta del sistema de acuerdo a cada fuente de alimentación no son iguales.

Para la prueba de hipótesis se utilizara un nivel de significancia del 5%.  $\alpha = 0,05$

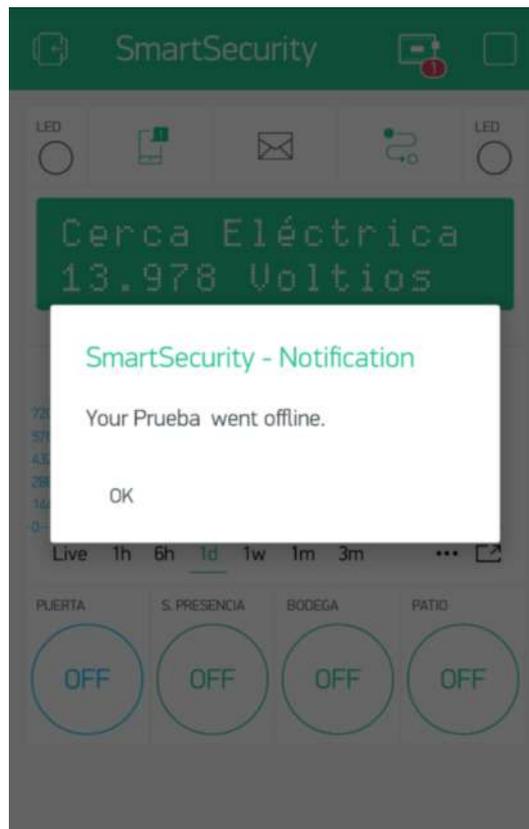
## Calculamos el estadístico de prueba

**Tabla 8-3:** Prueba T para la igualdad de medias

Tiempo de Respuesta del sistema. Varianzas iguales	Prueba T para la igualdad de medias						
	t	gl	Sig. (bilateral)	Diferencia de medias	Error típ. de la diferencia	95% Intervalo de confianza para la diferencia	
						Inferior	Superior
	42,751	60	,000	10,00387	,23400	9,53580	10,47195

**Realizado por:** Guaranga, J. 2018

Dado que la probabilidad de la prueba t para muestras independientes es de 0,000 es decir es menor al nivel de significancia se rechaza la hipótesis nula y se acepta la alternativa, por lo cual se concluye que el tiempo de respuesta del sistema con la fuente de alimentación (adaptador DC) es diferente significativamente diferente (mejor) a los tiempos de respuesta del sistema con una batería lipo.

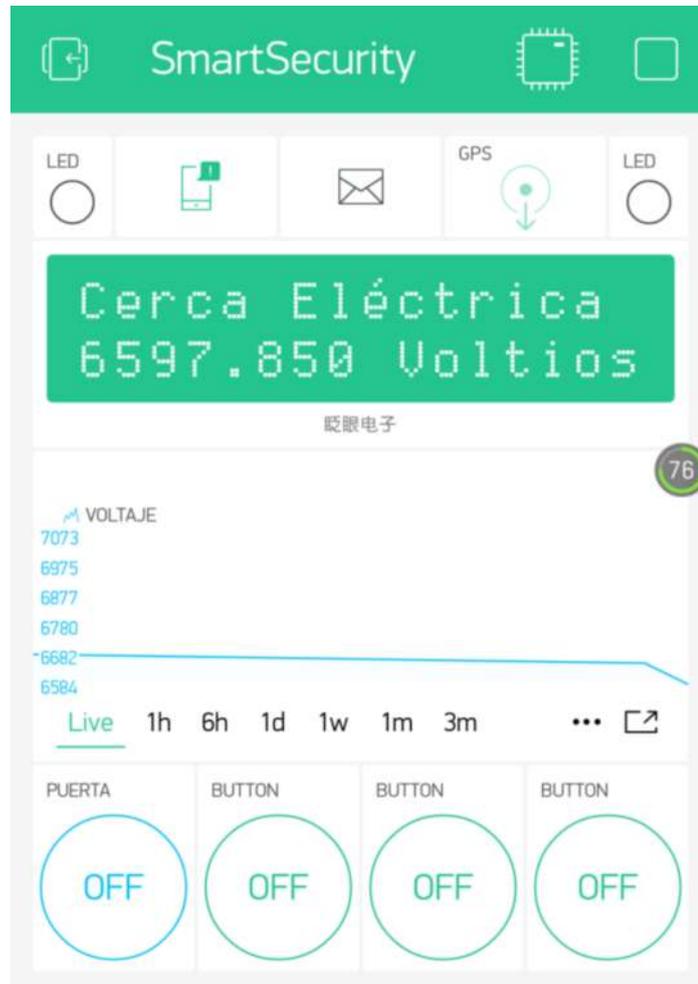


**Figura 4-3:** Recepción notificación de alerta

**Realizado por:** Guaranga, J. 2018

### 3.4. Prueba de precisión del sistema

Para las pruebas de precisión del sistema fueron tomados 30 datos, con el sistema conectado solamente a tierra proporcionado por la placa del electrificador y 30 datos con el sistema conectado a tierra física.



**Figura 5-3:** Medida en voltios, tensión de la cerca eléctrica

**Realizado por:** Guaranga, J. 2018

Con el fin de comprobar si existe mayor o menor dispersión en la oscilación de voltaje entregado por el electrificador en cada pulso. Por lo cual fue realizado un análisis estadístico en base a la prueba T para muestras independientes.

**Tabla 9-3: Muestra precisión**

Precisión	
Medición de oscilación del voltaje cerca eléctrica	
Puesta a tierra (V)	Sin puesta a tierra (V)
12662	12045
12030	12927
12269	11556
12730	12700
13509	11638
12329	12926
13311	11509
12219	11752
13033	11537
12039	12647
12716	12533
12348	12354
13198	12547
12855	12137
13229	12316
12666	12227
13265	12272
13007	11924
12845	12962
12193	12414
12440	11980
12232	12113
12042	11730
12428	11619
12276	12662
12196	11634
13271	12281
12729	12965
13356	11637
12480	12873
12071	12536

Realizado por: Guaranga, J. 2018

### Por lo cual establecemos las hipótesis a probar

- $H_0$ : Existe igual dispersión en la oscilación de voltajes entregado por el electrificador conectado a tierra física y el electrificador conectado a tierra proporcionado por la placa.
- $H_a$ : Existe diferente dispersión en la oscilación de voltaje entregado por el electrificador conectado a tierra física y el electrificador conectado a tierra proporcionado por la placa.

#### 3.4.1.1. Prueba de normalidad de distribución

- $H_0$ : Los datos obtenidos en el experimento de oscilación de voltajes siguen una distribución normal.
- $H_a$ : Los datos obtenidos en el experimento de oscilación de voltajes no siguen una distribución normal.

**Tabla 10-3:** Pruebas de normalidad

Oscilación de voltaje	Tipo de respuesta	Kolmogorov-Smirnov <sup>a</sup>			Shapiro-Wilk		
		Estadístico	gl	Sig.	Estadístico	gl	Sig.
voltaje	Puesta a tierra sin varilla	,128	31	,200*	,926	31	,035
	Puesta a tierra con varilla	,127	31	,200*	,930	31	,045

\*. Este es un límite inferior de la significación verdadera.

a. Corrección de la significación de Lilliefors

Realizado por: Guaranga, J. 2018

Según la prueba de normalidad de Shapiro-Wilk la probabilidad en el tipo de conexión a tierra del sistema es mayor al nivel de significancia, por lo cual, acepta la hipótesis nula, por lo que se concluye que los datos siguen una distribución normal.

#### 3.4.1.2. Prueba de homogeneidad de varianzas

- $H_0: \sigma_A^2 = \sigma_B^2$  (Las varianzas en el experimento de oscilación de voltajes de acuerdo al tipo de alimentación son iguales.)
- $H_a: \sigma_A^2 \neq \sigma_B^2$  (Las varianzas en el experimento de oscilación de voltajes de acuerdo al tipo de alimentación no son iguales.)

Establecemos el nivel de significancia en  $\alpha = 0,05$  y calculamos el estadístico de prueba.

**Tabla 11-3:** Prueba de homogeneidad de varianzas

Estadístico de Levene	gl1	gl2	Sig.
,077	1	60	,783

Realizado por: Guaranga, J. 2018

Según la prueba de Levene, los datos de oscilación de voltajes de los dos tipos de conexión a tierra tienen las mismas varianzas.

### 3.4.1.3. Prueba T-para muestras independientes

#### Definir la hipótesis

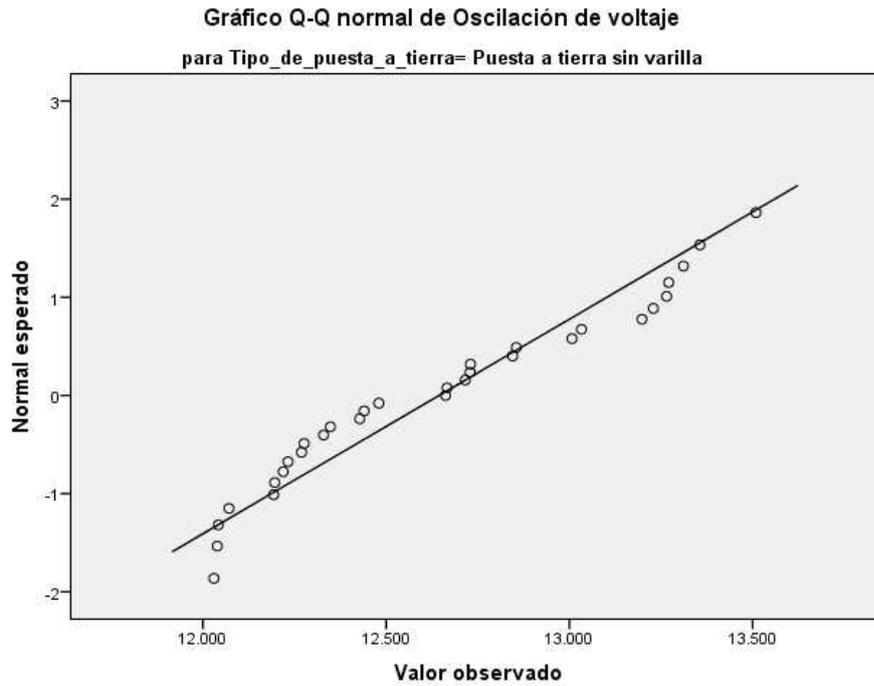
- Ho: Existe igual dispersión en la oscilación de voltajes entregado por el electrificador conectado a tierra física y el electrificador conectado a tierra proporcionado por la placa.
- Ha: Existe diferente dispersión en la oscilación de voltaje entregado por el electrificador conectado a tierra física y el electrificador conectado a tierra proporcionado por la placa.

Para la prueba de hipótesis se utilizara un nivel de significancia del 5%.  $\alpha = 0,05$  y calculamos el estadístico de prueba

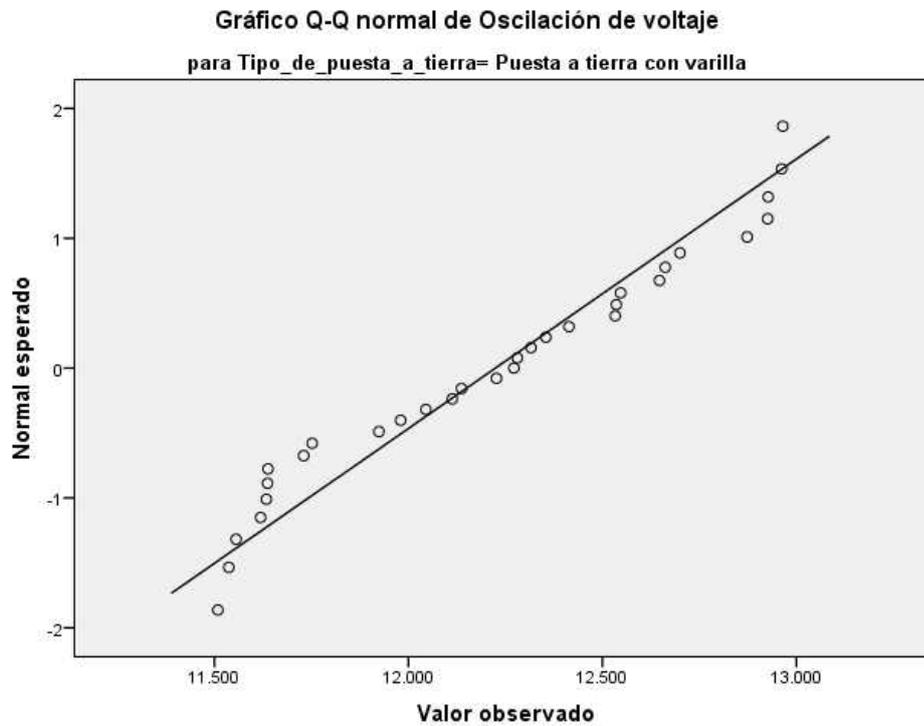
**Tabla 12-3:** Prueba de muestras independientes

Oscilación de voltaje. Varianzas iguales	Prueba T para la igualdad de medias						
	t	gl	Sig. (bilateral)	Diferencia de medias	Error típ. de la diferencia	95% Intervalo de confianza para la diferencia	
						Inferior	Superior
	3,519	60	,001	420,03226	119,34470	181,30731	658,75721

Realizado por: Guaranga, J. 2018



**Gráfico 1-3:** Dispersión de voltajes respecto a lo esperado. Sistema de puesta a tierra sin varilla  
Realizado por: Guaranga, J. 2018



**Gráfico 2-3:** Dispersión de voltajes respecto a lo esperado. Sistema de puesta a tierra con varilla  
Realizado por: Guaranga, J. 2018

Se concluye que el existe una dispersión de datos diferente entre el sistema de puesta a tierra de placa y puesta a tierra física. Según los datos y graficas proporcionadas por la prueba T para datos independientes muestra que un sistema de puesta a tierra, permite una menor dispersión de datos (Voltajes) y por ende una mayor precisión de la lectura de los mismos.

#### 3.4.1.4. Comparación valores de tensión

**Tabla 13-3:** Comparación voltajes cerca eléctrica

Voltaje cerca eléctrica		
Prueba	Sistema diseñado	Voltímetro de alta tensión
1	12662	13000
2	12030	13000
3	12269	13000
4	12730	13000
5	13509	13000
6	12329	13000
7	13311	13000
8	12219	13000
9	13033	13000
10	12039	13000
11	12716	13000
12	12348	13000
13	13198	13000
14	12855	13000
15	13229	13000
16	12666	13000
17	13265	13000
18	13007	13000
19	12845	13000
20	12193	13000
21	12440	13000
22	12232	13000
23	12042	13000
24	12428	13000
25	12276	13000
26	12196	13000
27	13271	13000
28	12729	13000
29	13356	13000
30	12480	13000

Realizado por: Guaranga, J. 2018

Para lograr una mejor comprensión de la precisión, procedemos a realizar una comparación entre los valores obtenidos en el censado de la tensión de la cerca eléctrica a través de nuestro sistema diseñado comparado con los valores obtenidos con un multímetro de alta tensión.

**Tabla 14-3:** Resumen estadístico sistema diseñado

<i>Sistema Diseñado</i>	
<b>Media</b>	12644,32258
<b>Error típico</b>	82,19356105
<b>Mediana</b>	12662
<b>Moda</b>	#N/A
<b>Desviación estándar</b>	457,6343801
<b>Varianza de la muestra</b>	209429,2258
<b>Mínimo</b>	12030
<b>Máximo</b>	13509
<b>Suma</b>	391974
<b>Cuenta</b>	31

Realizado por: Guaranga, J. 2018

**Tabla 15-3:** Resumen estadístico voltímetro alta tensión

<i>Voltímetro alta tensión</i>	
<b>Media</b>	13000
<b>Error típico</b>	0
<b>Mediana</b>	13000
<b>Moda</b>	13000
<b>Desviación estándar</b>	0
<b>Varianza de la muestra</b>	0
<b>Mínimo</b>	13000
<b>Máximo</b>	13000
<b>Suma</b>	403000
<b>Cuenta</b>	31

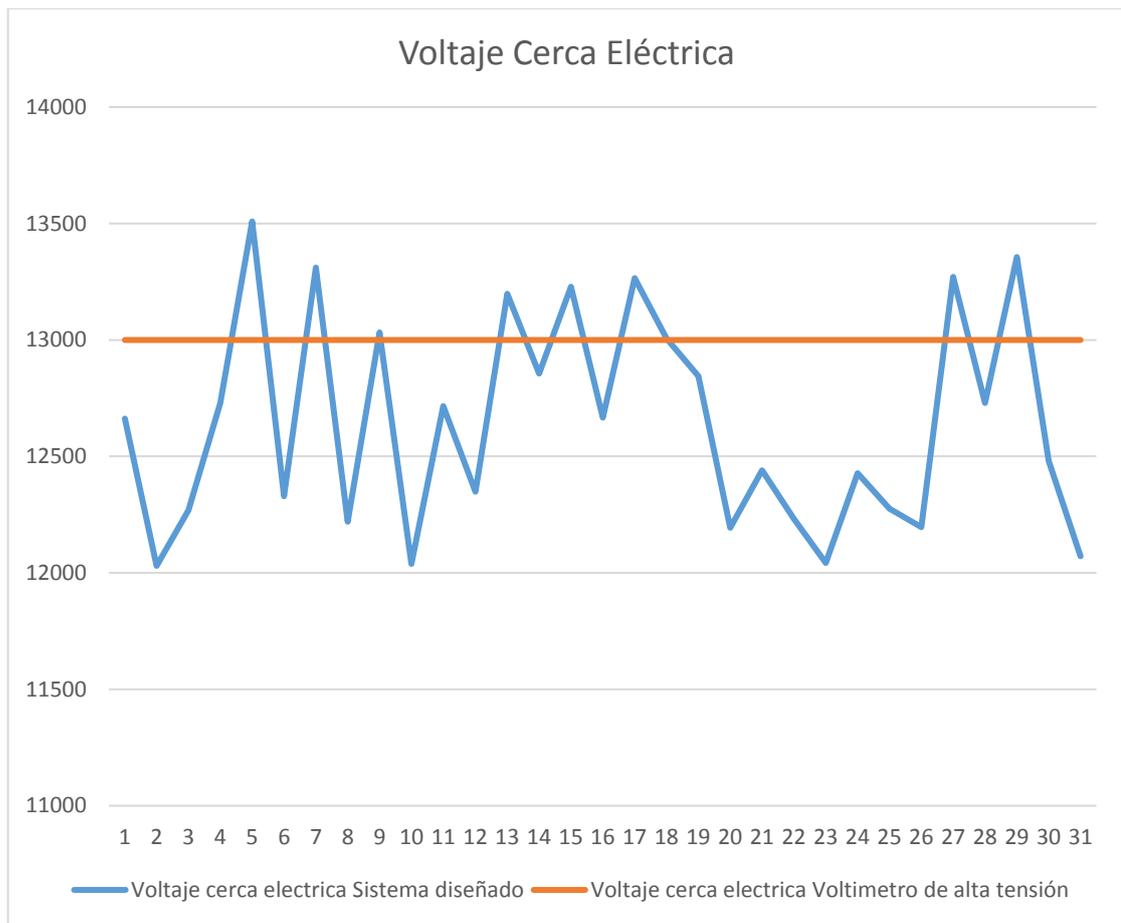
Realizado por: Guaranga, J. 2018

Como resultado de los datos obtenidos de las mediciones de voltaje en nuestro sistema diseñado y los datos obtenidos por el multímetro de alta tensión se tiene que:

- El sistema diseñado realiza una medición de voltaje con una desviación estándar de 457V.
- Mientras que los datos obtenidos mediante el voltímetro de alta tensión presenta una desviación estándar de 0V.

Cabe recalcar que las mediciones obtenidas mediante el voltímetro de alta tensión se limitan a dos cifras elevadas  $\times 10^3$  lo cual reduce su precisión, dando como resultado datos ideales no 100% reales.

El siguiente grafico de dispersión muestra la distribución de las medidas de las los dos sistemas en cada uno de las pruebas realizadas.



**Gráfico 3-3:** Dispersión de medidas de voltaje

**Realizado por:** Guaranga, J. 2018

#### 3.4.1.5. Cálculo error absoluto

La siguiente tabla contiene las diferencias absolutas entre los datos medidos por el sistema y los datos esperados.

**Tabla 16-3:** Diferencia sistema y voltímetro

Prueba	Error entre Sistema y Voltímetro
1	-338
2	-970
3	-731
4	-270
5	509
6	-671
7	311
8	-781
9	33
10	-961
11	-284
12	-652
13	198
14	-145
15	229
16	-334
17	265
18	7
19	-155
20	-807
21	-560
22	-768
23	-958
24	-572
25	-724
26	-804
27	271
28	-271
29	356
30	-520

Realizado por: Guaranga, J. 2018

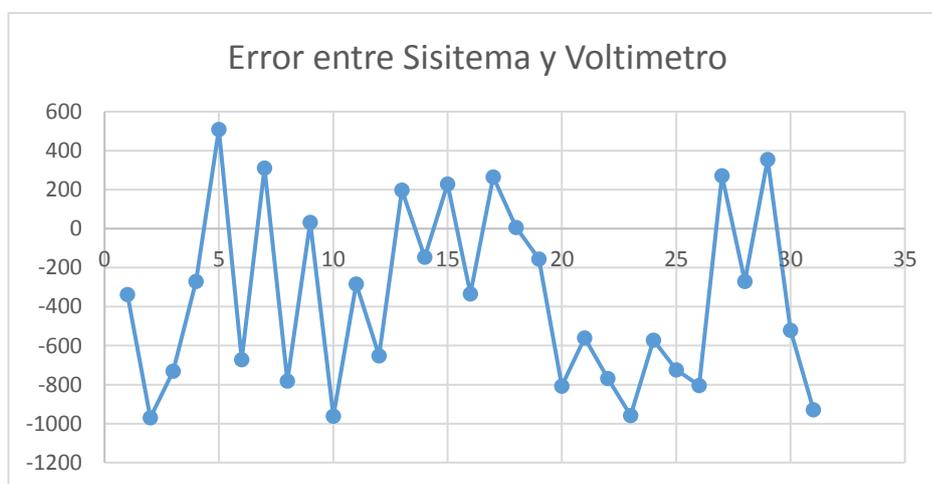
Las medidas obtenidas para esta comparación son las muestras descritas para la prueba de precisión del sistema comparándolo con un multímetro de alto voltaje proporcionado por el fabricante del electrificador Hagroy, por lo tanto los valores obtenidos por el multímetro poseen un nivel alto de confiabilidad.

**Tabla 17-3:** Resumen estadístico diferencia sistema-voltímetro

<i>Error entre Sistema y Voltímetro</i>	
<b>Media</b>	-355,6774
<b>Error típico</b>	82,193561
<b>Mediana</b>	-338
<b>Moda</b>	#N/A
<b>Desviación estándar</b>	457,63438
<b>Varianza de la muestra</b>	209429,23
<b>Rango</b>	1479
<b>Mínimo</b>	-970
<b>Máximo</b>	509
<b>Suma</b>	-11026
<b>Cuenta</b>	31

Realizado por: Guaranga, J. 2018

Al realizar el análisis de los datos tabulados podemos afirmar que las oscilaciones de voltaje en la toma de muestras para nuestro sistema diseñado presentan a desviación estándar de 457,63. En la siguiente gráfica se presenta los errores dispersos en cada una de las pruebas respecto a los valores esperados.



**Gráfico 4-3:** Error entre sistema diseñado y voltímetro de alta voltaje

Realizado por: Guaranga, J. 2018

### 3.5. Análisis de costos

El sistema diseñado pretende ser un modelo funcional, que cumpla con la mayoría de las características de los sistemas domóticos actuales, sin incurrir en costos elevados.

### 3.5.1. Costo de materiales y herramientas

Los materiales y herramientas utilizados en la implementación del proyecto se especifican a continuación.

**Tabla 18-3:** Costos materiales y herramientas

Cantidad	Detalle	Valor unitario	Total
8	Postes acero	\$ 2,50	\$ 20,00
1	Rollo de alambre galvanizado	\$ 2,50	\$ 2,50
30	Tensores	\$ 0,50	\$ 15,00
30	Aros de vigilancia	\$ 0,30	\$ 9,00
1	Varilla de 6mm de grosor	\$ 8,00	\$ 8,00
2	Metro de cable AWG #6	\$ 0,70	\$ 1,40
20	Metro de cable AWG gemelo #14	\$ 0,30	\$ 6,00
20	Metro de cable UTP CAT 5E	\$ 0,25	\$ 5,00
2	Par de video balunes	\$ 3,00	\$ 6,00
1	Materiales electrónicos	\$ 20,00	\$ 20,00
1	Caja de proyectos grande	\$ 8,00	\$ 8,00
1	Caja de proyectos pequeña	\$ 3,00	\$ 3,00
<b>Total</b>			\$ 103,90

Realizado por: Guaranga, J. 2018

### 3.5.2. Costo de equipos

**Tabla 19-3:** Costos de equipos

Cantidad	Detalle	Valor unitario	Total
1	Arduino Mega	\$ 18,00	\$ 18,00
1	Módulo de 4 relés	\$ 8,00	\$ 8,00
1	Modulo Ethernet enc28j60	\$ 7,00	\$ 7,00
1	Modulo reconocimiento de voz Elechouse	\$ 45,00	\$ 45,00
1	NodeMCU	\$ 8,00	\$ 8,00
2	esp8266	\$ 4,00	\$ 8,00
2	Cámara tipo tubo Hikvision	\$ 35,00	\$ 70,00
1	NVR Hikvision	\$ 80,00	\$ 80,00
2	Regulador DC 5V	\$ 3,00	\$ 6,00
1	Electrificador	\$ 65,00	\$ 65,00
<b>Total</b>			\$ 315,00

Realizado por: Guaranga, J. 2018

Los equipos utilizados en el diseño de este sistema pueden variar de acuerdo a cada caso, dependiendo el área a ser protegida, la cantidad de actuadores, la cantidad de sensores y sobre

todo el presupuesto y requerimientos del usuario. La siguiente tabla muestra los equipos necesarios para una instalación en una casa promedio de dos planta, un patio posterior.

### 3.5.3. Costo de implementación

Los costos de la implementación engloban variables como el diseño y desarrollo de quipos y software no comerciales.

**Tabla 19-3:** Costos de implementación

Cantidad (Personal)	Descripción	Tiempo	Precio	Total
1	Desarrollo de Software	80,00	\$ 5,00	\$ 400,00
1	Instalación de equipos	48,00	\$ 2,00	\$ 96,00
1	Ensamblaje Centrales	24,00	\$ 2,00	\$ 48,00
<b>Total</b>				\$ 544,00

Realizado por: Guaranga, J. 2018

### 3.5.4. Costo total del proyecto

El costo total del proyecto toma en cuenta los sub costos descritos en las tablas anteriores.

**Tabla 20-3:** Costo total del proyecto

Descripción	Costo
<b>Costo de materiales y herramientas</b>	\$ 103,90
<b>Costo equipos</b>	\$ 315,00
<b>Costo de implementación</b>	\$ 544,00
<b>Total</b>	\$ 962,90

Realizado por: Guaranga, J. 2018

Cabe recalcar que el costo total del proyecto, incluye el desarrollo total del proyecto, este precio no recae directamente sobre un usuario final, ya que los costos de desarrollo de software, así mismo como diseño de las centrales son asumidos por la empresa “Rio Solar Smart Energy”, la cual realizará a futuro un estudio de mercado para la determinación de un precio parcial por software que se le cobra al usuario incluido en la venta de los equipos.

## CONCLUSIONES

- La técnica empleada para la planificación del desarrollo del sistema de seguridad, por medio de un estudio de planos arquitectónicos del espacio a trabajar, permiten brindar una asesoría técnica adecuada y diseñar un sistema que cumpla con las necesidades del usuario.
- Se logró implementar un sistema de comandos por voz, que brinda acceso al sistema mediante google assistant, en base a las pruebas realizadas, se puede decir que el sistema implementado tiene una efectividad del 95%.
- Se logró implementar un sistema de seguridad con tiempos de respuesta ante anomalías detectadas, entre 1.3 y 6 segundos, lo cual brinda al usuario un alto grado de confiabilidad.
- Se logró implementar una central basada en software y hardware libre, para el manejo e integración de los principales componentes de un sistema de seguridad domiciliario, lo cual permite al usuario una libre elección entre las diferentes marcas, calidades de sensores y actuadores disponibles en el mercado.
- El tiempo de respuesta del sistema de seguridad desarrollado puede variar de acuerdo a varios factores ajenos al diseño del mismo, por ejemplo la velocidad de conexión a internet que disponga el usuario en su domicilio o el tipo de fuente de alimentación.
- El software desarrollado al no poseer una interfaz gráfica nativa en el hardware que controla el sistema, permite reducir el coste computacional, empleando exclusivamente sus recursos a el monitoreo y control del sistema de seguridad.
- La implementación de una circuito de sincronización para medir la tensión de la cerca eléctrica, también sirve como circuito de protección para toda la central de gestión, permitiendo el paso de energía solo en los instantes de disparo del electrificador, evitando el paso de retornos de corriente dañinas.
- El desarrollo de una central de gestión secundaria para el manejo de sensores, da al sistema y al usuario la posibilidad de ampliar casi sin límite el alcance de cobertura del mismo, sin la necesidad de renovar la central de gestión primaria.

## RECOMENDACIONES

- Se debe delimitar el área de detección de movimiento en las cámaras de seguridad, para reducir errores de falsas alarmas cuando el sistema está activado.
- Para garantizar los tiempos de respuesta del sistema expuestos en este trabajo de titulación es recomendado utilizar una fuente de alimentación de 5V a 1.5A para la central de gestión primaria y una fuente de 5V a 500mA para la central de gestión de sensores.
- Para mejorar la sensibilidad de la cerca eléctrica para la detección de intrusiones dentro del área protegida es recomendable el uso de un sistema de puesta a tierra física, tomando en cuenta las cargas eléctricas que ocupen la misma línea de fuerza que el electrificador.
- El uso de software y hardware libre hicieron posible el diseño y construcción de un sistema capaz de integrar los principales componentes de un sistema de seguridad a bajo costo, pese a esto, el desempeño del mismo puede mejorar con el uso de tarjetas de desarrollo de mayor costo.
- Para garantizar el correcto funcionamiento de las salidas de la central de gestión, se debe tomar en cuenta la carga máxima soportada por los relés que gobiernan los actuadores, en este caso la carga máxima es de 10A por cada relé.
- Se debe evitar colocar la central de gestión de sensores cerca de techos, espejos u otros dispositivos que emitan o reciban señales de radio frecuencia, pues esto reduce el área de cobertura de la antena de la central.
- Para futuros sistemas de seguridad desarrollados se recomienda la integración de la central de gestión primaria con la central de gestión de sensores dentro de un mismo dispositivo, también el uso de componentes electrónicos SMD para la reducción del tamaño de placas electrónicas.

## **BIBLIOGRAFÍA**

**CADENA Alvarado & CORONEL Arevalo.** *Diseño e Implementación de un Sistema Domótico para Control y Seguridad en Tiempo Real Vía Teléfono Celular* (Tesis). (Pre grado) Escuela Superior Politécnica de Chimborazo Riobamba, Ecuador. 2010.

**HAGROY ELECTRONIC S.A.C.** *Manual de instalación* [En línea]. (2009). Ate-Lima-Perú: 2009. [Consulta: 2018-03-12]. Disponible en: [www.Hagroy.com](http://www.Hagroy.com).

**LARGEMOUTHBASS.** *Reverse Engineering an Electric Fence Charger* [En línea]. (2013). [Consulta: 2018-05-04]. Disponible en: <http://www.instructables.com/id/Reverse-Engineering-an-Electric-Fence-Charger/>.

**CRESCO Enrique.** *Lenguaje de programación Arduino Aprendiendo Arduino* [En línea]. (2016). [Consulta: 2018-02-07]. Disponible en: <https://aprendiendoarduino.wordpress.com/2016/03/31/lenguaje-de-programacion-arduino/>.

**LUZURIAGA Godoy,** *Desarrollo de un sistema de cerca eléctrica para el control de ganado en la Universidad Nacional de Loja* (Tesis). (Pre grado) Universidad Nacional de Loja, Loja, Ecuador. 2016.

**PÉREZ Hugo, & CHACHA Edgar.** *Implementación de dos módulos técnicos para prácticas de domótica e inmótica mediante protocolos de comunicación X10 y HDL Buspro* (Tesis). (Pre grado) Escuela Superior Politécnica de Chimbroazo, Riobamba, Ecuador. 2017.

**MOTOROLA INC. S. T.** *6-Pin DIP Optoisolators Transistor Output* [En línea]. (1995). [Consulta: 2018-04-18]. Disponible en: <https://www.noisebridge.net/images/1/17/4N26.pdf>.

**MAYNÉ, Jordi.** *Estado actual de las Comunicaciones por Radio Frecuencia*. s.l.:Silica, 2009, pp.12-19.

**ROMERO MORALES, Cristobal; et al.** *Domótica e inmótica. Viviendas y edificios inteligentes*. 2ª Edición, México: Alfaomega, 2007.

**PEREIRA Antonio.** *Propuesta de un modelo de sistema de información interorganizacional: aplicación en el sector de la domótica para la gestión de servicios del hogar digital* (Tesis). (Doctoral) Universidad Politécnica de Madrid, Madrid, España. 2014.

**PHILIPS SEMICONDUCTORS.** *Data sheet, 81(3)* [En línea]. (2007). [Consulta: 2018-04-12]. Disponible en: [https://doi.org/10.1007/978-3-211-89836-9\\_355](https://doi.org/10.1007/978-3-211-89836-9_355).

**POMATOCA Mauro.** *Sistema puesta a tierra para el laboratorio de máquinas eléctricas de la escuela de ingeniería electrónica* (Tesis). (Pre grado) Escuela Superior Politécnica de Chimborazo, Riobamba, Ecuador. 2010.

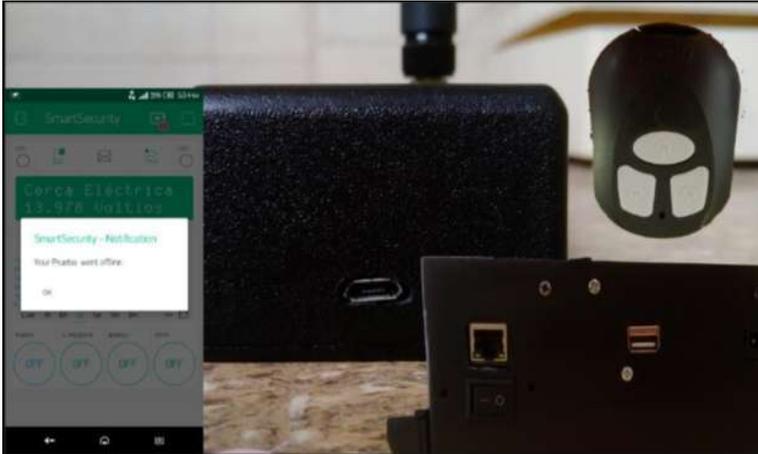
**SECRETARÍA DE ENERGÍA MEXICANA,** NOM-001-SEDE-1999 Instalaciones eléctricas (utilización) [En línea]. (1999). [Consulta: 2018-05-15]. Disponible en: [www.cucba.udg.mx/sites/default/files/proteccioncivil/NOM-001-SEDE-1999.pdf](http://www.cucba.udg.mx/sites/default/files/proteccioncivil/NOM-001-SEDE-1999.pdf).

**URRUTIA Carrasco.** *Sistema de video vigilancia mediante cámaras ip para mejorar la seguridad ciudadana en zona central del cantón baños de agua santa* (Tesis). (Pre grado) Universidad Técnica de Ambato, Ambato, Ecuador. (2015).

**WON-TOP ELECTRONICS.** *2w005 – 2w10* [En línea]. (2002). [Consulta: 2018-05.11]. Disponible en: <http://pdf.datasheetcatalog.com/datasheet/wte/2W08.pdf>.

## **ANEXOS**

**Anexo A:** Manual de Usuario



# SMARTSECURITY

## MANUAL DE USUARIO

Manual de usuario, del sistema de seguridad para vivienda con exceso mediante voz y video.

#### 4. PRESENTACIÓN

-

#### 5. DISPOSITIVOS INCLUIDOS

-

#### 6. PARTES DE LOS DISPOSITIVO

-

#### 7. CONEXIÓN

-

#### 8. MANEJO APLICACIÓN MOVIL

#### 9. OK GOOGLE

-

-

Escuela Superior  
Politécnica de  
Chimborazo

Diseñado por Juan Erique Guaraga A.

# PRESENTACIÓN

SMARTSECURITY ES UN CONJUNTO DE DISPOSITIVOS ENFOCADOS A BRINDAR SEGURIDAD EN EL HOGAR A TRAVÉS DE LA GESTIÓN Y MANEJO DE SENSORES Y ACTUADORES TÍPICOS EN LOS SISTEMAS DE SEGURIDAD CONVENCIONALES, ADEMÁS INCLUYE LA POSIBILIDAD DEL MANEJO DE ENCENDIDO Y APAGADO DE LUCES, ABERTURA PUERTAS, VENTANAS, INALAMBRICAMENTE DE TODO TIPO DE DISPOSITIVO DE USO COMÚN EN EL HOGAR QUE PERMITA UNA MANIFULACIÓN ON OFF



SmartSecurity forma parte de los dispositivos desarrollados por la empresa Rio Solar Smart Energy enfocados a brindar comodidad a un bajo costo al usuario.

## DISPOSITIVOS INLCUIDOS

### CENTRAL DE GESTIÓN PRINCIPAL



Se encarga del manejo de los principales dispositivos de seguridad como sensores de gas, humo, sensores PIR, puertas de acceso principales, etc.

### CENTRAL DE GESTIÓN INALÁMBRICA



Se encarga tanto de la recepción y envío de datos hacia los diferentes dispositivos conectados a la red inalámbrica que crea la central.

### CONTROL REMOTO



Se encarga del encendido y apagado de todo el sistema

## CLIENTES INALÁMBRICOS



Se encargan de la recepción y/o envío de comandos para el manejo de los diferentes sensores y/o actuadores que el usuario desee manejar mediante la central de gestión inalámbrica.

## PARTES DE LOS DISPOSITIVOS

### CENTRAL DE GESTIÓN PRINCIPAL

#### PARTE FRONTAL



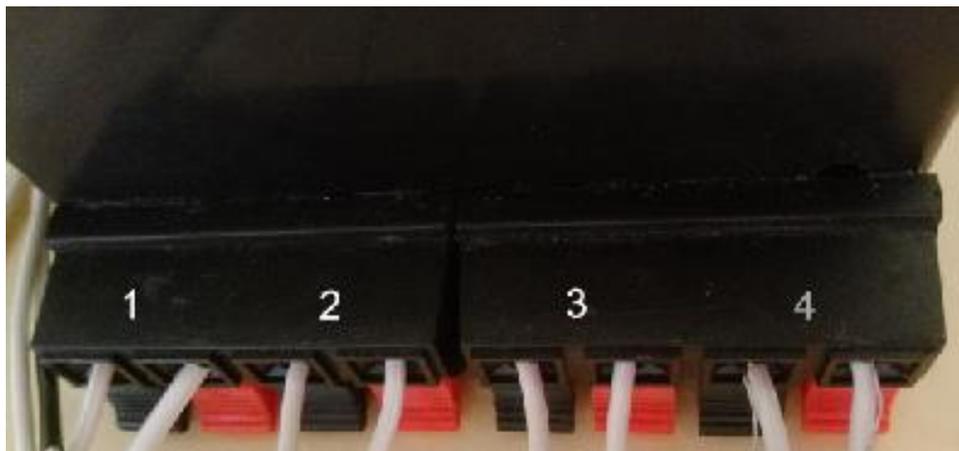
1. Switch paso de energía de la batería
2. Puerto RJ45
3. Puerto USB (Solo modo programación)
4. Jack hembra alimentación 5V
5. Led indicador alimentación red de energía eléctrica

## PARTE LATERAL DERECHA



4 Salidas por relé (hasta 10A)

## PARTE LATERAL IZQUIERDA



1. Entrada de alto voltaje cerca eléctrica
2. Entrada central de gestión inalámbrica
3. Entrada (12V)
4. Entrada auxiliar 5V

## CENTRAL DE GESTIÓN INALÁMBRICA



1. Salida de aviso a central de gestión principal
2. Antena Wi-Fi 5dB
3. Led indicador encendido
4. Jack entrada micro USB alimentación 5V

## CONTROL REMOTO



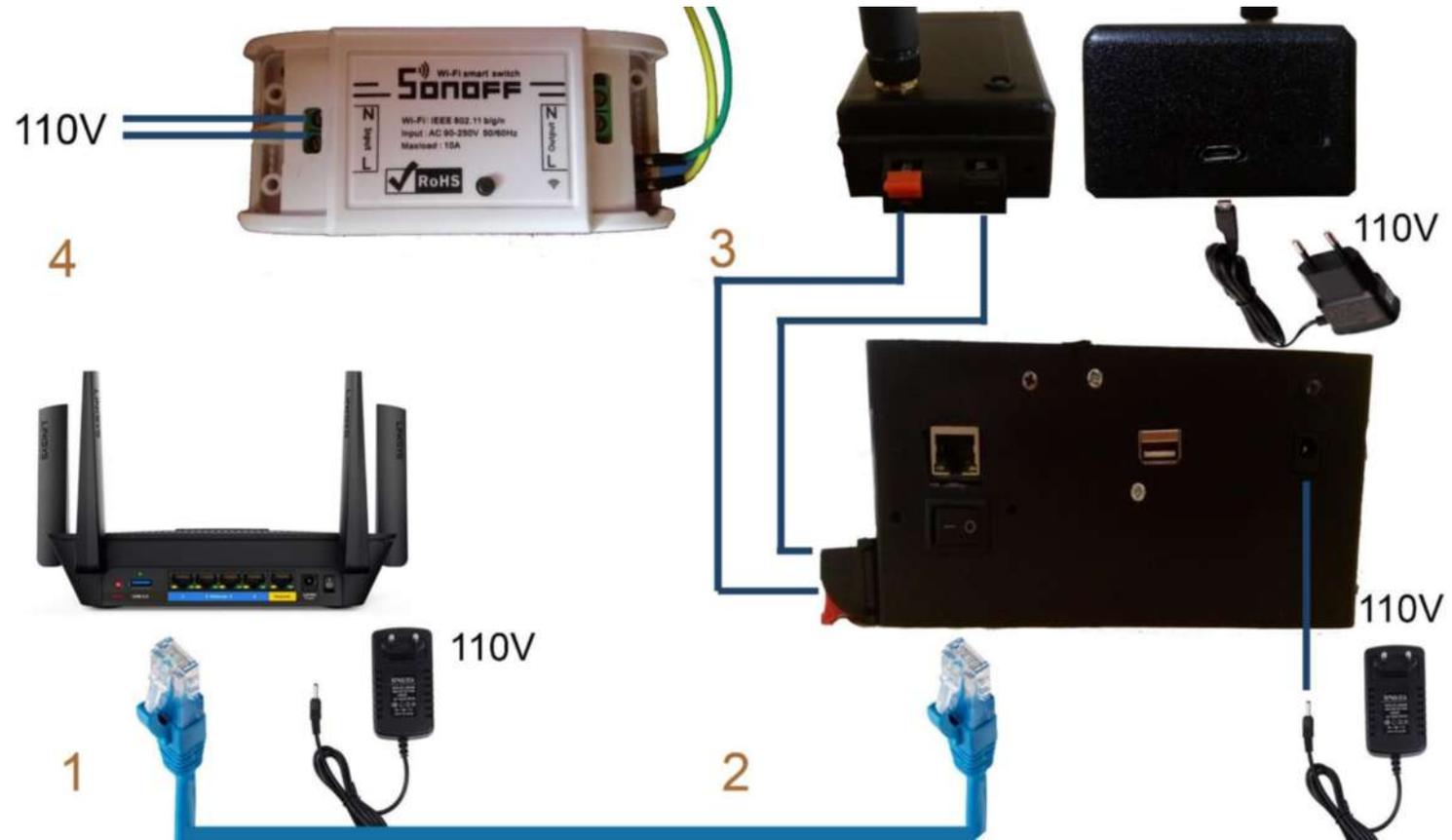
1. Encendido Apagado del Sistema
2. Pánico/Control alarma
3. Sin uso

## CLIENTES INALÁMBRICOS



1. Entrada de alimentación 110 V
2. Entrada de sensor 3.3 V
3. Salida con relé (hasta 10A)

# CONEXIÓN



# 1

1. Identifique el router proporcionado por su proveedor de servicio de internet
2. Asegúrese que su router este encendido
3. Con un cable UTP categoría 5E conecte a una de las salidas rj45 de su router

Nota: La conexión de entradas y salidas de la central de gestión principal queda a criterio y necesidades del usuario, preferentemente deberán ser manejadas por personal técnico.

# 2

1. Con la otra punta del cable UTP que previamente conectó a su router, conecte en la entrada rj45 de la central de gestión principal.
2. Encienda el Switch de alimentación por batería (la batería ya viene instalada en el interior del dispositivo)
3. Conecte el adaptador a la toma de corriente eléctrica de 110V de su hogar

NOTA IMPORTANTE: La entrada USB de la central de gestión es solo para uso de programación, en caso de fallos y deberá ser manejada solo por personal autorizado.

# 3

1. Conecte la salida de la central de gestión inalámbrica a la entrada de central de gestión principal.
2. Asegúrese que la antena Wi-Fi está bien conectada a la central.
3. Conecte el adaptador de alimentación 5V micro USB a la toma de corriente eléctrica de su vivienda.

**AUTOMATICAMENTE SE INICIA LA RED PARA LA GESTION INALAMBRICA LLAMADA "SENSORES"**

# 4

1. Conecte sus clientes inalámbricos a los sensores o actuadores que desee gestionar, (solo una entrada y una salida por cliente)
2. Conecte la alimentación a la toma eléctrica de la vivienda directamente.
3. Automáticamente se conectara con la central de gestión inalámbrica.

## USO DE LA APLICACIÓN

Se proporciona la aplicación de gestión del dispositivo el cual debe ser instalado por personal autorizado para el emparejamiento de la aplicación con el sistema.

Una vez empareja el usuario puede hacer uso de la aplicación para manejar su sistema.



Se muestra la pantalla de inicio de la aplicación en la cual bien por defecto configurado 4 salidas para cada uno de los relés de la central de gestión.

También se muestra el nivel de tensión de la cerca eléctrica (en caso de tenerla) de la vivienda, y una gráfica que muestra el historial de medición de la misma.

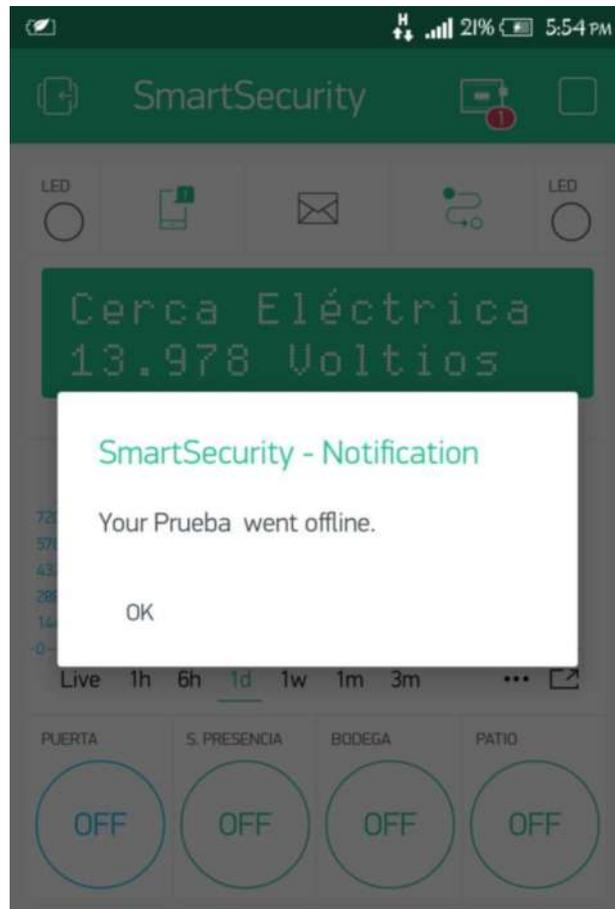


Basta con presionar sobre el botón deseado para accionar uno de los relés ya sea en modo switch o modo pulsador. Lo cual queda a criterio del usuario.



Para cambiar entre modo switch y modo pulsado se debe pulsar dos veces sobre el botón para configurarlo.

Cuando ocurra alguna anomalía o el sistema este fuera de línea recibirá una notificación en su dispositivo móvil.



En caso de cualquier intento de instrucción o anomalía la sirena del sistema de encenderá y usted podrá monitorear su sistema desde las cámaras de seguridad (en caso de poseerlas). Y dar por terminado el proceso de pánico de sistema si es que la situación lo amerita.

## OK GOOGLE

Si se encuentra ocupado ya sea realizando algún deporte o conduciendo su vehículo y no le es posible abrir la app proporcionada, puede hacer uso de comandos por voz a través de google assistant integrado en el sistema.

Basta con decir “Ok google”

E inmediatamente el asistente de google estará listo para recibir sus órdenes.

NOTA: DADO QUE LOS COMANDOS DE VOZ PERSONALIZADOS PARA EL MANEJO DEL SISTEMA ESTÁ AÚN EN FASE DE PRUEBA SE REQUIERE DE LA CONFIGURACIÓN DE CADA COMANDO A TRAVÉS DE PERSONAL TÉCNICO AUTORIZADO.



Por ahora dispone de comandos básicos pero muy útiles para el manejo del sistema como, encender y apagar luces, abrir puertas, y funciones que estén bajo el mando de los relés instalados.

NOTA: EL SISTEMA INCLUYE UN SISTEMA DE COMANDOS POR VOZ MEDIANTE ARDUINO PERO DEBIDO A SU ESCASA PRECISIÓN NO ES RECOMENDABLE USARLO.

**Anexo B:** Plano del proyecto

## Anexo C: Código del programa

### SERVIDOR

```
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266WebServer.h>

/* Set these to your desired credentials. */
const char *ssid = "Sensores";
const char *password = "12345678";

ESP8266WebServer server(80);

/*
  waiting for the link 192.168.4.1/pir
*/
void handleRoot() {
  server.send(200, "text/html", "<h1>You are connected</h1>");
}

void Alerta()
{
  Serial.println("Link Requested from one of the device");

  digitalWrite(5, HIGH);
  delay(7000);
  digitalWrite(5, LOW);
  delay(1000);
}

}

void setup() {
  delay(1000);
  Serial.begin(115200);
  Serial.println();
  Serial.print("Configuring access point...");
  /* You can remove the password parameter if you want the AP to be open. */
  WiFi.softAP(ssid, password);
  pinMode(5, OUTPUT); //GPIO 5
  IPAddress myIP = WiFi.softAPIP();
  Serial.print("AP IP address: ");
  Serial.println(myIP);
  server.on("/", handleRoot);
  server.on("/pir", Alerta);
  server.begin();
  Serial.println("HTTP server started");
}

void loop() {
  Serial.println("Waiting for the link");
  server.handleClient();
}
```

### CLIENTE

```
#include <Arduino.h>
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>

#define pir 3 //GPIO 3
const char* ssid = "Sensores";
const char* password = "12345678";
void setup() {

  Serial.begin(115200);
  Serial.setDebugOutput(true);

  Serial.println();
  Serial.println();
  Serial.println();
  for(uint8_t t = 4; t > 0; t--) {
    Serial.printf("[SETUP] WAIT %d...\n", t);
```

```

    Serial.flush();
    delay(1000);
}
WiFi.mode(WIFI_STA);
WiFi.disconnect(true);
WiFi.begin(ssid,password);
pinMode(pir,INPUT);
pinMode(13, OUTPUT);
while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
    delay(15000);
    Serial.println("Connected");
}
    pinMode(pir,INPUT);
    pinMode(13, OUTPUT);
for (int i=0; i <= 6; i++){
    digitalWrite(13, HIGH);
    delay(50);
    digitalWrite(13, LOW);
    delay(50);
    digitalWrite(13, HIGH);
    delay(50);
}
}
}

void loop() {
    // wait for WiFi connection
    if(WiFi.status() == WL_CONNECTED) {

        HTTPClient http;

        if(digitalRead(pir) == HIGH)
        {
            digitalWrite(13, LOW);
            delay(500);
            digitalWrite(13, HIGH);
            delay(500);

            Serial.print("[HTTP] begin...\n");
            // requesting required link
            http.begin("http://192.168.4.1/pir");//HTTP

            Serial.print("[HTTP] GET...\n");
            // start connection and send HTTP header
            int httpCode = http.GET();a

            http.end();
            delay(7000);
        }

    }
}
}

```

## PRINCIPAL

```

#define BLYNK_PRINT Serial

#include <SPI.h>
#include <UIPEthernet.h>
#include <BlynkSimpleUIPEthernet.h>

/*****Comandos de voz*****/
#include <SoftwareSerial.h>
#include "VoiceRecognitionV3.h"
VR myVR(12,13); // 2:RX 3:TX, you can choose your favourite pins.
uint8_t records[7]; // save record
uint8_t buf[64];
int led = 22;
#define onRecord (0)
#define offRecord (1)

void printSignature(uint8_t *buf, int len)

```

```

{
  int i;
  for(i=0; i<len; i++){
    if(buf[i]>0x19 && buf[i]<0x7F){
      Serial1.write(buf[i]);
    }
    else{
      Serial1.print("[");
      Serial1.print(buf[i], HEX);
      Serial1.print("]");
    }
  }
}
}
void printVR(uint8_t *buf)
{
  Serial1.println("VR Index\tGroup\tRecordNum\tSignature");

  Serial1.print(buf[2], DEC);
  Serial1.print("\t\t");

  if(buf[0] == 0xFF){
    Serial1.print("NONE");
  }
  else if(buf[0]&0x80){
    Serial1.print("UG ");
    Serial1.print(buf[0]&(~0x80), DEC);
  }
  else{
    Serial1.print("SG ");
    Serial1.print(buf[0], DEC);
  }
  Serial1.print("\t");

  Serial1.print(buf[1], DEC);
  Serial1.print("\t\t");
  if(buf[3]>0){
    printSignature(buf+4, buf[3]);
  }
  else{
    Serial1.print("NONE");
  }
  Serial1.println("\r\n");
}
/*****/

int inputPin = A0; // Entrada señal cerca eletrica
int inputAlerta = 8; //Entrada
int inputSensores = 5; // Entrada Alerta Sensores
int inputEncendido =7; // Entrada
char auth[] = "a8e7e71c64da414cbeabe97e012ebe98";

/*****/

/*****/

void myTimerEvent()
{
  // vamos a leer la entrada por el pin analógico 0:
  int valorMedido = analogRead(inputPin);
  /*****/
  int valorEncendido = digitalRead(inputEncendido);
  /*****/
  Serial.print("Cerca Eléctrica:");
  Serial.println(valorMedido);
  // ahora convertimos la lectura analógica (que va de 0 a 1023) a voltaje (8000 - 13000V):
  if (valorEncendido == 1) {
    int voltajeCerca=random(900,1000);
    float voltaje = voltajeCerca * (14300.0 / 1023.0);
    Blynk.virtualWrite(V0,voltaje);
  }
  else
  {
    Blynk.virtualWrite(V0,0);
  }
}
}

```

```

/*****
unsigned long inicio, fin, transcurrido;

void myTimerSeguridadEvent()
{
  int valorMedido = analogRead(inputPin);
  int valorSensores;
  int valorEncendido;
  int valorAlerta;
  /*****
  inicio=millis(); //guarda el tiempo actual como referencia
  // vamos a leer las entradas de Alerta:
  valorSensores = digitalRead(inputSensores);
  valorEncendido = digitalRead(inputEncendido);
  valorAlerta = digitalRead(inputAlerta);
  if (valorEncendido == 1) {
    Serial.println("*****SISTEMA ENCENDIDO*****");
    Serial.print("Encendido:");
    Serial.println(valorEncendido);
    Serial.println("Valores Leidos:");
    Serial.print("Cerca Eléctrica:");
    Serial.println(valorMedido);
    Serial.print("Sensores:");
    Serial.println(valorSensores);
    Serial.print("Alerta Cerca:");
    Serial.println(valorAlerta);

    if (valorSensores == 1 || valorAlerta == 1 ) {
      Blynk.virtualWrite(V10,1);
      Serial.println("*****Alerta Seguridad*****");

      valorEncendido = 0;
      valorSensores = 0;
      valorAlerta = 0;

    }
  }
  else
  {
    Serial.println("*****Sin Alertas*****");
    Serial.print("Encendido:");
    Serial.println(valorEncendido);
    Serial.println("Valores Leidos:");
    Serial.print("Cerca Eléctrica:");
    Serial.println(valorMedido);
    Serial.print("Sensores:");
    Serial.println(valorSensores);
    Serial.print("Alerta Cerca:");
    Serial.println(valorAlerta);
  }
  Blynk.virtualWrite(V10,0);
  fin=millis();
  transcurrido=fin-inicio; //Calcula el tiempo desde la última lectura
  Serial.print("Tiempo de respuesta:"); //Imprime el valor en el monitor serial
  Serial.println(transcurrido); //Imprime el valor en el monitor serial
}

/*****Comandos voz*****/
void myTimerComandosEvent()
{
  int ret;
  ret = myVR.recognize(buf, 50);
  if(ret>0){
    switch(buf[1]){
      case onRecord:
        /*** turn on LED */
        digitalWrite(led, LOW);
        //break;
      //case offRecord:
        /*** turn off LED*/
        delay (500);
        digitalWrite(led, HIGH);
        break;
      default:
        Serial1.println("Record function undefined");
    }
  }
}

```

```

        break;
    }
    /** voice recognized */
    printVR(buf);
}
}

/*****

BlynkTimer timer; // Create a Timer object called "timer"!
BlynkTimer timerSeguridad; // Create a Timer object called "timer"!
BlynkTimer timerComandosVoz; // Create a timer object called "timer"!

void setup()
{
    // Debug console
    myVR.begin(9600);
    Serial1.begin(115200);
    Serial.begin(9600);
    Blynk.begin(auth);
    randomSeed(1024);
    //Asegurar nivel ALTO en cada entrada de rele
    pinMode(inputSensores,INPUT);
    pinMode(inputEncendido,INPUT);
    pinMode(22, OUTPUT);
    pinMode(23, OUTPUT);
    pinMode(24, OUTPUT);
    pinMode(25, OUTPUT);
    digitalWrite (22, HIGH);
    digitalWrite (23, HIGH);
    digitalWrite (24, HIGH);
    digitalWrite (25, HIGH);
    Blynk.virtualWrite(V10,0);
    timer.setInterval(1000L, myTimerEvent);
    timerSeguridad.setInterval(7000L, myTimerSeguridadEvent);
    timerComandosVoz.setInterval(1000L, myTimerComandosEvent);
    /*****Comandos de voz*****/

    Serial1.println("Elechouse Voice Recognition V3 Module\r\nControl LED sample");

    pinMode(led, OUTPUT);

    if(myVR.clear() == 0){
        Serial1.println("Recognizer cleared.");
    }else{
        Serial1.println("Not find VoiceRecognitionModule.");
        Serial1.println("Please check connection and restart Arduino.");
        while(1);
    }

    if(myVR.load((uint8_t)onRecord) >= 0){
        Serial1.println("onRecord loaded");
    }

    /* if(myVR.load((uint8_t)offRecord) >= 0){
        Serial1.println("offRecord loaded");
    }*/
    /*****
}

BLYNK_WRITE(V5) // Button Widget writes to Virtual Pin V5
{
    int pinValue = param.asInt();
    Serial.println(pinValue);
    if(pinValue == HIGH) { // if Button sends 1
        digitalWrite (22, LOW); // start the function
        //delay(500);
        digitalWrite (22, HIGH);
    }
}

BLYNK_WRITE(V1) // Button Widget writes to Virtual Pin V5
{
    int pinValueSeguridad = param.asInt();
    Serial.println(pinValueSeguridad);
}

```

```

if(pinValueSeguridad == HIGH) { // if Button sends 1
  digitalWrite (23, LOW);      // start the function
}
}
BLYNK_WRITE(V2) // Button Widget writes to Virtual Pin V5
{
  int pinValueAlerta = param.asInt();
  Serial.println(pinValueAlerta);
  if(pinValueAlerta == LOW) { // if Button sends 1
    digitalWrite (23, HIGH);   // start the function
  }
}

void loop()
{
  Blynk.run();
  timer.run();// Initiates BlynkTimer
  timerSeguridad.run();// Initiates BlynkTimer
  timerComandosVoz.run();// Initiates BlynkTimer
}

#include <SoftwareSerial.h>
#include "VoiceRecognitionV3.h"

/**
 * Connection
 * Arduino  VoiceRecognitionModule
 * 2 -----> TX
 * 3 -----> RX
 */
VR myVR(2,3); // 2:RX 3:TX, you can choose your favourite pins.

/*****
/** declare print functions */
void printSeperator();
void printSignature(uint8_t *buf, int len);
void printVR(uint8_t *buf);
void printLoad(uint8_t *buf, uint8_t len);
void printTrain(uint8_t *buf, uint8_t len);
void printCheckRecognizer(uint8_t *buf);
void printUserGroup(uint8_t *buf, int len);
void printCheckRecord(uint8_t *buf, int num);
void printCheckRecordAll(uint8_t *buf, int num);
void printSigTrain(uint8_t *buf, uint8_t len);
void printSystemSettings(uint8_t *buf, int len);
void printHelp(void);

/*****
// command analyze part
#define CMD_BUF_LEN 64+1
#define CMD_NUM 10
typedef int (*cmd_function_t)(int, int);
uint8_t cmd[CMD_BUF_LEN];
uint8_t cmd_cnt;
uint8_t *paraAddr;
int receiveCMD();
int checkCMD(int len);
int checkParaNum(int len);
int findPara(int len, int paraNum, uint8_t **addr);
int compareCMD(uint8_t *para1 , uint8_t *para2, int len);

int cmdTrain(int len, int paraNum);
int cmdLoad(int len, int paraNum);
int cmdTest(int len, int paraNum);
int cmdVR(int len, int paraNum);
int cmdClear(int len, int paraNum);
int cmdRecord(int len, int paraNum);
int cmdSigTrain(int len, int paraNum);
int cmdGetSig(int len, int paraNum);
int cmdSettings(int len, int paraNum);
int cmdHelp(int len, int paraNum);
/** cmdList, cmdLen, cmdFunction has correspondence */
const char cmdList[CMD_NUM][10] = { // command list table
  {
    "train" }
  ,

```

```

    {
    "load" }
    ,
    {
    "clear" }
    ,
    {
    "vr" }
    ,
    {
    "record" }
    ,
    {
    "sigtrain" }
    ,
    {
    "getsig" }
    ,
    {
    "Settings" }
    ,
    {
    "test" }
    ,
    {
    "help" }
    ,
};
const char cmdLen[CMD_NUM]= { // command length
5, // {"train"},
4, // {"load"},
5, // {"clear"},
2, // {"vr"},
6, // {"record"},
8, // {"sigtrain"},
6, // {"getsig"},
8, // {"Settings"},
4, // {"test"},
4, // {"help"}
};
cmd_function_t cmdFunction[CMD_NUM]={ // command handle fuction(function pointer table)
cmdTrain,
cmdLoad,
cmdClear,
cmdVR,
cmdRecord,
cmdSigTrain,
cmdGetSig,
cmdSettings,
cmdTest,
cmdHelp,
};

/***** temporary data */
uint8_t buf[255];
uint8_t records[7]; // save record

void setup(void)
{
    myVR.begin(9600);

    /** initialize */
    Serial.begin(115200);
    Serial.println(F("Elechouse Voice Recognition V3 Module \"train\" sample.));

    printSeperator();
    Serial.println(F("Usage:"));
    printSeperator();
    printHelp();
    printSeperator();
    cmd_cnt = 0;
}

void loop(void)
{

```

```

int len, paraNum, paraLen, i;

/** receive Serial command */
len = receiveCMD();
if(len>0){
  /** check if the received command is valid */
  if(!checkCMD(len)){

    /** check parameter number of the received command */
    paraNum = checkParaNum(len);

    /** display the received command back */
    Serial.write(cmd, len);

    /** find the first parameter */
    paraLen = findPara(len, 1, &paraAddr);

    /** compare the received command with command in the list */
    for(i=0; i<CMD_NUM; i++){
      /** compare command length */
      if(paraLen == cmdLen[i]){
        /** compare command content */
        if( compareCMD(paraAddr, (uint8_t *)cmdList[i], paraLen) == 0 ){
          /** call command function */
          if( cmdFunction[i](len, paraNum) != 0){
            printSeperator();
            Serial.println(F("Command Format Error!"));
            printSeperator();
          }
          break;
        }
      }
    }

    /** command is not supported*/
    if(i == CMD_NUM){
      printSeperator();
      Serial.println(F("Unkonwn command"));
      printSeperator();
    }
  }
  else{
    /** received command is invalid */
    printSeperator();
    Serial.println(F("Command format error"));
    printSeperator();
  }
}

/** try to receive recognize result */
int ret;
ret = myVR.recognize(buf, 50);
if(ret>0){
  /** voice recognized, print result */
  printVR(buf);
}
}

/**
 * @brief receive command from Serial.
 * @param NONE.
 * @retval command length, if no command receive return -1.
 */
int receiveCMD()
{
  int ret;
  int len;
  unsigned long start_millis;
  start_millis = millis();
  while(1){
    ret = Serial.read();
    if(ret>0){
      start_millis = millis();
      cmd[cmd_cnt] = ret;
      if(cmd[cmd_cnt] == '\n'){
        len = cmd_cnt+1;

```

```

    cmd_cnt = 0;
    return len;
}
cmd_cnt++;
if(cmd_cnt == CMD_BUF_LEN){
    cmd_cnt = 0;
    return -1;
}
}

if(millis() - start_millis > 100){
    cmd_cnt = 0;
    return -1;
}
}
}

/**
 * @brief compare two commands, case insensitive.
 * @param para1 --> command buffer 1
 * para2 --> command buffer 2
 * len --> buffer length
 * @retval 0 --> equal
 * -1 --> unequal
 */
int compareCMD(uint8_t *para1 , uint8_t *para2, int len)
{
    int i;
    uint8_t res;
    for(i=0; i<len; i++){
        res = para2[i] - para1[i];
        if(res != 0 && res != 0x20){
            res = para1[i] - para2[i];
            if(res != 0 && res != 0x20){
                return -1;
            }
        }
    }
    return 0;
}

/**
 * @brief Check command format.
 * @param len --> command length
 * @retval 0 --> command is valid
 * -1 --> command is invalid
 */
int checkCMD(int len)
{
    int i;
    for(i=0; i<len; i++){
        if(cmd[i] > 0x1F && cmd[i] < 0x7F){

        }
        else if(cmd[i] == '\t' || cmd[i] == ' ' || cmd[i] == '\r' || cmd[i] == '\n'){

        }
        else{
            return -1;
        }
    }
    return 0;
}

/**
 * @brief Check the number of parameters in the command
 * @param len --> command length
 * @retval number of parameters
 */
int checkParaNum(int len)
{
    int cnt=0, i;
    for(i=0; i<len; ){
        if(cmd[i]!='\t' && cmd[i]!=' ' && cmd[i] != '\r' && cmd[i] != '\n'){
            cnt++;
            while(cmd[i] != '\t' && cmd[i] != ' ' && cmd[i] != '\r' && cmd[i] != '\n'){

```

```

        i++;
    }
}
i++;
}
return cnt;
}

/**
 * @brief Find the specified parameter.
 * @param len --> command length
 * @param paraIndex --> parameter index
 * @param addr --> return value. position of the parameter
 * @param retval length of specified parameter
 */
int findPara(int len, int paraIndex, uint8_t **addr)
{
    int cnt=0, i, paraLen;
    uint8_t dt;
    for(i=0; i<len; ){
        dt = cmd[i];
        if(dt!='\t' && dt!=' '){
            cnt++;
            if(paraIndex == cnt){
                *addr = cmd+i;
                paraLen = 0;
                while(cmd[i] != '\t' && cmd[i] != ' ' && cmd[i] != '\r' && cmd[i] != '\n'){
                    i++;
                    paraLen++;
                }
                return paraLen;
            }
            else{
                while(cmd[i] != '\t' && cmd[i] != ' ' && cmd[i] != '\r' && cmd[i] != '\n'){
                    i++;
                }
            }
        }
        else{
            i++;
        }
    }
    return -1;
}

int cmdHelp(int len, int paraNum)
{
    if(paraNum != 1){
        return -1;
    }
    printSeperator();
    printHelp();
    printSeperator();
    return 0;
}

/**
 * @brief Handle "train" command
 * @param len --> command length
 * @param paraNum --> number of parameters
 * @param retval 0 --> success
 * @param -1 --> Command format error
 */
int cmdTrain(int len, int paraNum)
{
    int i, ret;
    if(paraNum < 2 || paraNum > 8){
        return -1;
    }

    for(i=2; i<=paraNum; i++){
        findPara(len, i, &paraAddr);
        records[i-2] = atoi((char *)paraAddr);
        if(records[i-2] == 0 && *paraAddr != '0'){
            return -1;
        }
    }
}

```

```

    }
    printSeperator();
    ret = myVR.train(records, paraNum-1, buf);
    // ret = myVR.train(records, paraNum-1);
    if(ret >= 0){
        printTrain(buf, ret);
    }
    else if(ret == -1){
        Serial.println(F("Train failed."));
    }
    else if(ret == -2){
        Serial.println(F("Train Timeout."));
    }
    printSeperator();
    return 0;
}

/**
 * @brief Handle "load" command
 * @param len --> command length
 * paraNum --> number of parameters
 * @retval 0 --> success
 * -1 --> Command format error
 */
int cmdLoad(int len, int paraNum)
{
    int i, ret;
    if(paraNum < 2 || paraNum > 8){
        return -1;
    }

    for(i=2; i<=paraNum; i++){
        findPara(len, i, &paraAddr);
        records[i-2] = atoi((char *)paraAddr);
        if(records[i-2] == 0 && *paraAddr != '0'){
            return -1;
        }
    }
    // myVR.writehex(records, paraNum-1);
    ret = myVR.load(records, paraNum-1, buf);
    printSeperator();
    if(ret >= 0){
        printLoad(buf, ret);
    }
    else{
        Serial.println(F("Load failed or timeout."));
    }
    printSeperator();
    return 0;
}

/**
 * @brief Handle "clear" command
 * @param len --> command length
 * paraNum --> number of parameters
 * @retval 0 --> success
 * -1 --> Command format error
 */
int cmdClear(int len, int paraNum)
{
    if(paraNum != 1){
        return -1;
    }
    if(myVR.clear() == 0){
        printSeperator();
        Serial.println(F("Recognizer cleared."));
        printSeperator();
    }
    else{
        printSeperator();
        Serial.println(F("Clear recognizer failed or timeout."));
        printSeperator();
    }
    return 0;
}

```

```

/**
 * @brief Handle "vr" command
 * @param len --> command length
 * paraNum --> number of parameters
 * @retval 0 --> success
 * -1 --> Command format error
 */
int cmdVR(int len, int paraNum)
{
    int ret;
    if(paraNum != 1){
        return -1;
    }
    ret = myVR.checkRecognizer(buf);
    if(ret<=0){
        printSeperator();
        Serial.println(F("Check recognizer failed or timeout."));
        printSeperator();
        return 0;
    }
    printSeperator();
    printCheckRecognizer(buf);
    printSeperator();
    return 0;
}

/**
 * @brief Handle "record" command
 * @param len --> command length
 * paraNum --> number of parameters
 * @retval 0 --> success
 * -1 --> Command format error
 */
int cmdRecord(int len, int paraNum)
{
    int ret;
    if(paraNum == 1){
        ret = myVR.checkRecord(buf);
        printSeperator();
        if(ret>=0){
            printCheckRecordAll(buf, ret);
        }
        else{
            Serial.println(F("Check record failed or timeout."));
        }
        printSeperator();
    }
    else if(paraNum < 9){
        for(int i=2; i<=paraNum; i++){
            findPara(len, i, &paraAddr);
            records[i-2] = atoi((char *)paraAddr);
            if(records[i-2] == 0 && *paraAddr != '0'){
                return -1;
            }
        }
    }

    ret = myVR.checkRecord(buf, records, paraNum-1); // auto clean duplicate records
    printSeperator();
    if(ret>=0){
        printCheckRecord(buf, ret);
    }
    else{
        Serial.println(F("Check record failed or timeout."));
    }
    printSeperator();
}
else{
    return -1;
}
return 0;
}

/**
 * @brief Handle "sigtrain" command
 * @param len --> command length
 * paraNum --> number of parameters

```

```

* @retval 0 --> success
* -1 --> Command format error
*/
int cmdSigTrain(int len, int paraNum)
{
    int ret, sig_len;
    uint8_t *lastAddr;
    if(paraNum < 2){
        return -1;
    }

    findPara(len, 2, &paraAddr);
    records[0] = atoi((char *)paraAddr);
    if(records[0] == 0 && *paraAddr != '0'){
        return -1;
    }

    findPara(len, 3, &paraAddr);
    sig_len = findPara(len, paraNum, &lastAddr);
    sig_len +=( (unsigned int)lastAddr - (unsigned int)paraAddr );

    printSeperator();
    ret = myVR.trainWithSignature(records[0], paraAddr, sig_len, buf);
    // ret = myVR.trainWithSignature(records, paraNum-1);
    if(ret >= 0){
        printSigTrain(buf, ret);
    }
    else{
        Serial.println(F("Train with signature failed or timeout."));
    }
    printSeperator();

    return 0;
}

/**
* @brief Handle "getsig" command
* @param len --> command length
* paraNum --> number of parameters
* @retval 0 --> success
* -1 --> Command format error
*/
int cmdGetSig(int len, int paraNum)
{
    int ret;
    if(paraNum != 2){
        return -1;
    }

    findPara(len, 2, &paraAddr);
    records[0] = atoi((char *)paraAddr);
    if(records[0] == 0 && *paraAddr != '0'){
        return -1;
    }

    ret = myVR.checkSignature(records[0], buf);

    printSeperator();
    if(ret == 0){
        Serial.println(F("Signature isn't set."));
    }
    else if(ret > 0){
        Serial.print(F("Signature:"));
        printSignature(buf, ret);
        Serial.println();
    }
    else{
        Serial.println(F("Get sig error or timeout."));
    }
    printSeperator();

    return 0;
}

/**
* @brief Handle "test" command

```

```

* @param len --> command length
* paraNum --> number of parameters
* @retval 0 --> success
* -1 --> Command format error
*/
int cmdTest(int len, int paraNum)
{
    printSeperator();
    Serial.println(F("TEST is not supported."));
    printSeperator();
    return 0;
}

int cmdSettings(int len, int paraNum)
{
    int ret;
    if(paraNum != 1){
        return -1;
    }
    ret = myVR.checkSystemSettings(buf);
    if( ret > 0){
        printSeperator();
        printSystemSettings(buf, ret);
        printSeperator();
    }
    else{
        printSeperator();
        Serial.println(F("Check system settings error or timeout"));
        printSeperator();
    }
    return 0;
}

/*****
/**
* @brief Print signature, if the character is invisible,
* print hexible value instead.
* @param buf --> command length
* len --> number of parameters
*/
void printSignature(uint8_t *buf, int len)
{
    int i;
    for(i=0; i<len; i++){
        if(buf[i]>0x19 && buf[i]<0x7F){
            Serial.write(buf[i]);
        }
        else{
            Serial.print(F("["));
            Serial.print(buf[i], HEX);
            Serial.print(F("]"));
        }
    }
}

/**
* @brief Print signature, if the character is invisible,
* print hexible value instead.
* @param buf --> VR module return value when voice is recognized.
* buf[0] --> Group mode(FF: None Group, 0x8n: User, 0x0n: System
* buf[1] --> number of record which is recognized.
* buf[2] --> Recognizer index(position) value of the recognized record.
* buf[3] --> Signature length
* buf[4]~buf[n] --> Signature
*/
void printVR(uint8_t *buf)
{
    Serial.println(F("VR Index\tGroup\tRecordNum\tSignature"));

    Serial.print(buf[2], DEC);
    Serial.print(F("\t\t"));

    if(buf[0] == 0xFF){
        Serial.print(F("NONE"));
    }
    else if(buf[0]&0x80){

```



```

        Serial.print(F("Untrained"));
    }
}

Serial.println();
}
}

/**
 * @brief Print record train status.
 * @param buf --> Check record command return value
 * buf[0] --> Number of checked records
 * buf[2i+1] --> Record number.
 * buf[2i+2] --> Record train status. (00: untrained, 01: trained, FF: record value out of range)
 * (i = 0 ~ buf[0]-1 )
 * num --> Number of trained records
 */
void printCheckRecord(uint8_t *buf, int num)
{
    Serial.print(F("Check "));
    Serial.print(buf[0], DEC);
    Serial.println(F(" records.));

    Serial.print(num, DEC);
    if(num>1){
        Serial.println(F(" records trained.));
    }
    else{
        Serial.println(F(" record trained.));
    }
}

for(int i=0; i<buf[0]*2; i += 2){
    Serial.print(buf[i+1], DEC);
    Serial.print(F("\t-->\t"));
    switch(buf[i+2]){
        case 0x01:
            Serial.print(F("Trained"));
            break;
        case 0x00:
            Serial.print(F("Untrained"));
            break;
        case 0xFF:
            Serial.print(F("Record value out of range"));
            break;
        default:
            Serial.print(F("Unknown Stauts"));
            break;
    }
    Serial.println();
}
}

/**
 * @brief Print record train status.
 * @param buf --> Check record command return value
 * buf[0] --> Number of checked records
 * buf[2i+1] --> Record number.
 * buf[2i+2] --> Record train status. (00: untrained, 01: trained, FF: record value out of range)
 * (i = 0 ~ buf[0]-1 )
 * num --> Number of trained records
 */
void printCheckRecordAll(uint8_t *buf, int num)
{
    Serial.print(F("Check 255"));
    Serial.println(F(" records.));

    Serial.print(num, DEC);
    if(num>1){
        Serial.println(F(" records trained.));
    }
    else{
        Serial.println(F(" record trained.));
    }
}
myVR.writehex(buf, 255);
for(int i=0; i<255; i++){
    if(buf[i] == 0xF0){

```

```

        continue;
    }
    Serial.print(i, DEC);
    Serial.print(F("\t-->\t"));
    switch(buf[i]){
    case 0x01:
        Serial.print(F("Trained"));
        break;
    case 0x00:
        Serial.print(F("Untrained"));
        break;
    case 0xFF:
        Serial.print(F("Record value out of range"));
        break;
    default:
        Serial.print(F("Unknown Stauts"));
        break;
    }
    Serial.println();
}
}

/**
 * @brief Print check user group result.
 * @param buf --> Check record command return value
 * buf[8i] --> group number.
 * buf[8i+1] --> group position 0 status.
 * buf[8i+2] --> group position 1 status.
 * ...
 * buf[8i+6] --> group position 5 status.
 * buf[8i+7] --> group position 6 status.
 * (i = 0 ~ len)
 * len --> number of checked groups
 */
void printUserGroup(uint8_t *buf, int len)
{
    int i, j;
    Serial.println(F("Check User Group:"));
    for(i=0; i<len; i++){
        Serial.print(F("Group:"));
        Serial.println(buf[8*i]);
        for(j=0; j<7; j++){
            if(buf[8*i+1+j] == 0xFF){
                Serial.print(F("NONE\t"));
            }
            else{
                Serial.print(buf[8*i+1+j], DEC);
                Serial.print(F("\t"));
            }
        }
        Serial.println();
    }
}

/**
 * @brief Print "load" command return value.
 * @param buf --> "load" command return value
 * buf[0] --> number of records which are load successfully.
 * buf[2i+1] --> record number
 * buf[2i+2] --> record load status.
 * 00 --> Loaded
 * FC --> Record already in recognizer
 * FD --> Recognizer full
 * FE --> Record untrained
 * FF --> Value out of range"
 * (i = 0 ~ (len-1)/2 )
 * len --> length of buf
 */
void printLoad(uint8_t *buf, uint8_t len)
{
    if(len == 0){
        Serial.println(F("Load Successfully. "));
        return;
    }
    else{
        Serial.print(F("Load success: "));
    }
}

```

```

    Serial.println(buf[0], DEC);
}
for(int i=0; i<len-1; i += 2){
    Serial.print(F("Record "));
    Serial.print(buf[i+1], DEC);
    Serial.print(F("t"));
    switch(buf[i+2]){
    case 0:
        Serial.println(F("Loaded"));
        break;
    case 0xFC:
        Serial.println(F("Record already in recognizer"));
        break;
    case 0xFD:
        Serial.println(F("Recognizer full"));
        break;
    case 0xFE:
        Serial.println(F("Record untrained"));
        break;
    case 0xFF:
        Serial.println(F("Value out of range"));
        break;
    default:
        Serial.println(F("Unknown status"));
        break;
    }
}
}

/**
 * @brief Print "train" command return value.
 * @param buf --> "train" command return value
 * buf[0] --> number of records which are trained successfully.
 * buf[2i+1] --> record number
 * buf[2i+2] --> record train status.
 * 00 --> Trained
 * FE --> Train Time Out
 * FF --> Value out of range"
 * (i = 0 ~ len-1 )
 * len --> length of buf
 */
void printTrain(uint8_t *buf, uint8_t len)
{
    if(len == 0){
        Serial.println(F("Train Finish."));
        return;
    }
    else{
        Serial.print(F("Train success: "));
        Serial.println(buf[0], DEC);
    }
    for(int i=0; i<len-1; i += 2){
        Serial.print(F("Record "));
        Serial.print(buf[i+1], DEC);
        Serial.print(F("t"));
        switch(buf[i+2]){
        case 0:
            Serial.println(F("Trained"));
            break;
        case 0xFE:
            Serial.println(F("Train Time Out"));
            break;
        case 0xFF:
            Serial.println(F("Value out of range"));
            break;
        default:
            Serial.print(F("Unknown status "));
            Serial.println(buf[i+2], HEX);
            break;
        }
    }
}

/**
 * @brief Print "sigtrain" command return value.
 * @param buf --> "sigtrain" command return value

```

```

* buf[0] --> number of records which are trained successfully.
* buf[1] --> record number
* buf[2] --> record train status.
* 00 --> Trained
* F0 --> Trained, signature truncate
* FE --> Train Time Out
* FF --> Value out of range"
* buf[3] ~ buf[len-1] --> Signature.
* len --> length of buf
*/
void printSigTrain(uint8_t *buf, uint8_t len)
{
    if(len == 0){
        Serial.println(F("Train With Signature Finish."));
        return;
    }
    else{
        Serial.print(F("Success: "));
        Serial.println(buf[0], DEC);
    }
    Serial.print(F("Record "));
    Serial.print(buf[1], DEC);
    Serial.print(F("\t"));
    switch(buf[2]){
    case 0:
        Serial.println(F("Trained"));
        break;
    case 0xF0:
        Serial.println(F("Trained, signature truncate"));
        break;
    case 0xFE:
        Serial.println(F("Train Time Out"));
        break;
    case 0xFF:
        Serial.println(F("Value out of range"));
        break;
    default:
        Serial.print(F("Unknown status "));
        Serial.println(buf[2], HEX);
        break;
    }
    Serial.print(F("SIG: "));
    Serial.write(buf+3, len-3);
    Serial.println();
}

/**
* @brief Print "settings" command return value.
* @param buf --> "settings" command return value
* buf[0] --> number of records which are trained successfully.
* buf[1] --> record number
* buf[2] --> record train status.
* 00 --> Trained
* F0 --> Trained, signature truncate
* FE --> Train Time Out
* FF --> Value out of range"
* buf[3] ~ buf[len-1] --> Signature.
* len --> length of buf
*/

const unsigned int io_pw_tab[16]={
    10, 15, 20, 25, 30, 35, 40, 45,
    50, 75, 100, 200, 300, 400, 500, 1000
};

void printSystemSettings(uint8_t *buf, int len)
{
    switch(buf[0]){
    case 0:
    case 3:
        Serial.println(F("Baud rate: 9600"));
        break;
    case 1:
        Serial.println(F("Baud rate: 2400"));
        break;
    }
}

```

```

case 2:
    Serial.println(F("Baud rate: 4800"));
    break;
case 4:
    Serial.println(F("Baud rate: 19200"));
    break;
case 5:
    Serial.println(F("Baud rate: 38400"));
    break;
default:
    Serial.println(F("Baud rate: UNKONOWN"));
    break;
}

switch(buf[1]){
case 0:
case 0xFF:
    Serial.println(F("Output IO Mode: Pulse"));
    break;
case 1:
    Serial.println(F("Output IO Mode: Toggle"));
    break;
case 2:
    Serial.println(F("Output IO Mode: Clear(When recognized) "));
    break;
case 3:
    Serial.println(F("Output IO Mode: Set(When recognized)"));
    break;
default:
    Serial.println(F("Output IO Mode: UNKONOWN"));
    break;
}

if(buf[2] > 15){
    Serial.println(F("Pulse width: UNKONOWN"));
}
else{
    Serial.print(F("Pulse Width: "));
    Serial.print(io_pw_tab[buf[2]], DEC);
    Serial.println(F("ms"));
}

if(buf[3] == 0 || buf[3] == 0xFF){
    Serial.println(F("Auto Load: disable"));
}
else{
    Serial.println(F("Auto Load: enable"));
}

switch(buf[4]){
case 0:
case 0xFF:
    Serial.println(F("Group control by external IO: disabled"));
    break;
case 1:
    Serial.println(F("Group control by external IO: system group selected"));
    break;
case 2:
    Serial.println(F("Group control by external IO: user group selected"));
    break;
default:
    Serial.println(F("Group control by external IO: UNKNOWN STATUS"));
    break;
}
}

void printHelp(void)
{
    Serial.println(F("COMMAND    FORMAT          EXAMPLE          Comment"));
    printSeperator();
    // Serial.println(F("-----"));
    Serial.println(F("train    train (r0) (r1)...    train 0 2 45    Train records"));
    Serial.println(F("load     load (r0) (r1) ...    load 0 51 2 3    Load records"));
    Serial.println(F("clear    clear                clear            remove all records in Recognizer"));
    Serial.println(F("record / record (r0) (r1)... record / record 0 79    Check record train status"));
    Serial.println(F("vr       vr                    vr                Check recognizer status"));
}

```

```
Serial.println(F("getsig      getsig (r)      getsig 0      Get signature of record (r)");
Serial.println(F("sigtrain  sigtrain (r) (sig)  sigtrain 0 ZERO  Train one record(r) with signature(sig)");
Serial.println(F("settings  settings          settings      Check current system settings");
Serial.println(F("help      help            help          print this message"); }
```