

UNIVERSIDAD RICARDO PALMA
FACULTAD DE INGENIERÍA
ESCUELA PROFESIONAL DE INGENIERÍA MECATRÓNICA



**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA
DOMÓTICO PARA APLICACIONES EN PACIENTES
PARAPLÉJICOS MEDIANTE CONTROL REMOTO A
TRAVES DE INTERNET Y RECONOCIMIENTO DE VOZ**

TESIS
PARA OPTAR EL TÍTULO PROFESIONAL DE
INGENIERO MECATRÓNICO

PRESENTADA POR

Bach. DÍAZ ALARCÓN, PATRICIA MARYLIN

Bach. YABAR BERROCAL, JOSE DANIEL

Asesor: Dr. SOTELO VALER, FREEDY

LIMA-PERÚ

2017

AGRADECIMIENTOS

De José Daniel Yabar Berrocal:

En el presente trabajo de tesis quiero agradecer primeramente a mis padres: Wilber Yabar y Catalina Berrocal, ya que ellos son los que me apoyaron a llegar hasta este día, su apoyo fue primordial para lograr cada objetivo que me he trazado. Segundo quiero agradecer a los profesores que a lo largo de mi carrera me han apoyado brindándome el conocimiento necesario para llegar hasta aquí, en especial al Dr. Ing. Elmer Córdova, que siempre demostró estar dispuesto a ayudarme en todo lo que él podía, de todo corazón, gracias.

De Patricia Marylin Díaz Alarcón:

Me gustaría que estas líneas sirvieran para expresar mi más profundo y sincero agradecimiento a todas aquellas personas que con su ayuda han colaborado en la realización del presente trabajo, en especial a mi madre Patricia Alarcón Morales, por haberme forjado como la persona que soy en la actualidad.

Un agradecimiento muy especial a mi asesor el Dr. Ing. Freedy Sotelo Valer por su asesoría siempre dispuesta.

Finalmente a mis amigos, que nos apoyamos mutuamente en nuestra formación profesional, en especial a Rony Cabrera y a todas aquellas maravillosas personas que aportaron con sus ideas para realización de este trabajo.

INDICE GENERAL

RESUMEN	xii
ABSTRACT	xiii
INTRODUCCION	1
CAPITULO I: PLANTEAMIENTO DEL PROBLEMA	3
1.1 Problema general y específico.	3
1.2 Objetivos.....	4
1.3 Justificación	5
1.4 Alcances y limitaciones de la investigación.	6
1.5 Estructura de la tesis.	7
CAPITULO II: MARCO TEORICO.....	11
2.1 Antecedentes de la investigación.	11
2.2 Óptica de la Investigación.....	15
2.3 Selección de Variables.	16
2.4 Bases Teóricas.....	17
2.4.1 Las personas con paraplejia.....	17
2.4.1.1 Definición de la paraplejia.....	17
2.4.1.2 Tipos de paraplejia.	18
2.4.1.3 Causa de la paraplejia.	18
2.4.1.4 Calidad de vida de las personas con paraplejia.	20
2.4.1.5 Tecnologías que ayudan a las personas con paraplejia.....	21
2.4.2 Sistema Domótico.....	28
2.4.2.1 Definición.....	28
2.4.2.2 Historia de la domótica.	29
2.4.2.3 De la instalación convencional a la Domótica.....	31
2.4.2.4 Especificaciones de la Gestión Domótica.....	33
2.4.2.5 Sistemas de Control Domóticos.	34
2.4.2.5.1 Sistemas Centralizados.....	34
2.4.2.5.2 Sistemas Descentralizados.	35
2.4.2.5.3 Sistemas Distribuidos.	37
2.4.2.5.4 Sistemas Híbrida/Mixta.....	37
2.4.2.5.5 Conexiones entre dispositivos.....	38

2.4.2.6 Componentes del Sistema Domótico.....	40
2.4.2.6.1 Sensores en Domótica.	40
2.4.2.6.2 Actuadores.	45
2.4.2.6.3 Controladores.	59
2.4.2.6.4 Video Vigilancia.	70
2.4.2.6.5 Medios de comunicación.	74
2.4.3 Interfaces de usuario.	77
2.4.3.1 Definición.....	77
2.4.3.2 Tipos de interfaces de usuario.....	78
2.4.3.3 Reconocimiento de voz.	79
2.4.3.3.1 Definición de voz.	79
2.4.3.3.2 Cualidades de voz.	80
2.4.3.3.3 Técnicas de reconocimiento de voz.....	82
2.4.3.4 Aplicación web.....	83
2.4.3.4.1 Definición.....	83
2.4.3.4.2 Página web estática.....	84
2.4.3.4.3 Página web dinámica.....	86
2.4.3.4.4 Servidor web.....	88
2.4.3.4.5 Lenguajes de programación.	90
CAPITULO III: DISEÑO E IMPLEMENTACION DEL SISTEMA	92
3.1 Características generales.	92
3.2 Diseño del Sistema Domótico.....	93
3.2.1 Puerta de ingreso.	94
3.2.1.1 Consideraciones previas.	94
3.2.1.2 Diseño del Mecanismo.	95
3.2.1.3 Diseño Electrónica de Potencia.....	107
3.2.2 Cortina automática.....	108
3.2.2.1 Consideraciones previas.	108
3.2.2.2 Selección del Motor.	110
3.2.2.3 Diseño Electrónica de Potencia.....	114
3.2.3 Ventanas.	116
3.2.3.1 Consideraciones previas.	116
3.2.3.2 Selección del motor-reductor.....	118

3.2.3.3 Diseño Electrónica de Potencia.....	121
3.2.4 Iluminación.	122
3.2.4.1 Consideraciones previas.	122
3.2.4.2 Selección.....	123
3.2.4.3 Diseño Electrónica de Potencia.....	124
3.2.5 Calefacción.....	125
3.2.5.1 Consideraciones previas.	125
3.2.5.2 Selección.....	126
3.2.5.3 Diseño Electrónica de Potencia.....	128
3.2.6 Alarma.	129
3.2.6.1 Consideraciones previas.	129
3.2.6.2 Selección.....	129
3.2.6.3 Diseño Electrónica de Potencia.....	130
3.2.7 Video Vigilancia.	131
3.2.7.1 Consideraciones previas.	131
3.2.7.2 Selección.....	132
3.3 Diseño de las Interfaces de Usuario.....	134
3.3.1 Sistema de Control Central.....	136
3.3.1.1 Consideraciones previas.	136
3.3.1.2 Controlador.....	137
3.3.1.3 Algoritmo.	139
3.3.2 Sistema de Control por Internet.....	145
3.3.2.1 Consideraciones previas.	145
3.3.2.2 Selección del Sistema Embebido	146
3.3.2.3 Configuración de la red con IP estática.	148
3.3.2.4 Programación del Servidor Web.....	149
3.3.2.5 Librerías/scripts en Python para manejo del SCC	155
3.3.3 Sistema de Control por Voz.....	165
3.3.3.1 Consideraciones previas.	165
3.3.3.1 Selección del módulo de comunicación.....	166
3.3.3.2 Selección del módulo de voz	169
3.3.3.3 Configuración de comandos de voz.....	170
3.4 Implementación del Sistema Domótico.....	175

3.4.1 Implementación de la puerta de ingreso y cortina.	175
3.4.2 Implementación de la ventana.	177
3.4.2 Implementación de la iluminación, alarma y calefacción.	178
3.4.3 Implementación de la cámara de videovigilancia.	181
3.5 Implementación de las Interfaces de usuario.	181
CAPITULO IV: PRUEBAS Y RESULTADOS	184
4.1 Pruebas sin usuario con paraplejía.	184
4.2 Pruebas con el usuario parapléjico.....	186
4.2.1 Pruebas 1: Control por Internet de la habitación domótica.	188
4.2.2 Pruebas 2: Control por Voz de la habitación domótica.	190
4.2.3 Pruebas 3: Cambio de grupo de los comandos de voz.	191
4.2.4 Entrevista a la voluntaria.	192
4.2.5 Resultados de las pruebas con control por internet.....	193
4.2.6 Resultados de las pruebas con control por voz.	194
CONCLUSIONES	195
RECOMENDACIONES	196
REFERENCIAS BIBLIOGRAFICAS.....	197
ANEXOS	204

INDICE DE FIGURAS

Figura 1. Estructura de la tesis.....	9
Figura 2. Robot asistencial Jaco.	23
Figura 3. Sistema Lokomat.	25
Figura 4. Exoesqueleto Rewalk.....	26
Figura 5. Robot Asibot.	27
Figura 6. Robot Nuka.	28
Figura 7. Tecnologías en un Sistema Domótico.....	29
Figura 8. Sistema Centralizado.....	35
Figura 9. Sistema Descentralizado.	36
Figura 10. Sistema Distribuido.	37
Figura 11. Sistema Híbrido.	38
Figura 12. Sensor de movimiento activo.	44
Figura 13. Sensor de movimiento pasivo.	44
Figura 14. Motor paso a paso.	46
Figura 15. Focos ahorradores.....	49
Figura 16. Lámpara fluorescente.	50
Figura 17. Persianas motorizadas.....	54
Figura 18. Calefactor.....	57
Figura 19. Alarma.....	59
Figura 20. Microcontrolador PIC.	61
Figura 21. Placa Arduino.....	66
Figura 22. Raspberry Pi.	68
Figura 23. Sistema de video vigilancia analógico.....	71
Figura 24. Sistema de vigilancia IP.	73
Figura 25. Estructura página web estatica.	85
Figura 26. Página web dinámica.	87
Figura 27. Diagrama representativo de un servidor web.....	89
Figura 28. Habitación de la persona con paraplejía.	93
Figura 29. Arquitectura del Sistema Domótico.	93
Figura 30. Puerta de ingreso.....	95
Figura 31. Mecanismo tornillo-tuerca.....	97

Figura 32. Diagrama de posición de los eslabones de la puerta.....	98
Figura 33. Tornillo seleccionado.	99
Figura 34. Motor-reductor marca TSUKASA.....	101
Figura 35. Mecanismo de dos barras en posición inicial.....	101
Figura 36. Mecanismo de dos barras en posición final.	102
Figura 37. Mecanismo de barras cuando está a 80° del eje X.....	103
Figura 38. Velocidad del motor-reductor de la puerta según norma.	106
Figura 39. Final de Carrera en Posición Inicial.....	106
Figura 40. Puente H para motor-reductor de la puerta.....	108
Figura 41. Cortina roller.	109
Figura 42. Cortina roller enrollada.....	110
Figura 43. Motor-reductor SWF VALEO.....	111
Figura 44. Cortina roller desenrollada.	112
Figura 45. Finales de carrera de cortinas.....	114
Figura 46. Puente H para motor-reductor de la cortina.	116
Figura 47. Ventana de la habitación.....	117
Figura 48. Ventana cerrada.....	118
Figura 49. Motor paso a paso MINEBEA-MATSUSHITA.....	120
Figura 50. Driver A499.....	122
Figura 51. Foco ahorrador marca Phelix.....	124
Figura 52. Circuito de potencia para iluminación.	125
Figura 53. Estufa marca Imaco.....	127
Figura 54. Circuito de potencia para calefacción.	128
Figura 55. Transductor electroacustico para alarma.....	130
Figura 56. Circuito de potencia para alarma.	131
Figura 57. Descripción de los sistemas de control.	135
Figura 58. Comunicación serial de los sistemas de control.	137
Figura 59. Controlador del Sistema de Control General.	138
Figura 60. Algoritmo de Sistema de Control Central.....	140
Figura 61. Código de programación Comunicación SCC.....	141
Figura 62. Código de programación Puerta SCC.....	142
Figura 63. Código de programación Persiana SCC.	143
Figura 64. Código de programación Ventana SCC.....	144

Figura 65. Código de programación Iluminación/Calefacción/Alarma.....	144
Figura 66. Red Local para el SCI.	146
Figura 67. Puertos de comunicación del Raspberry Pi 3.	148
Figura 68. Configuración de IP estática en Raspberry Pi.....	149
Figura 69. Instalación del Apache en Raspberry Pi 3	150
Figura 70. Configuración Web Cámara D-Link.....	151
Figura 71. Página Principal del SCI.	155
Figura 72. Librerías Python para SCI.....	156
Figura 73. Código Python de librería client.py.....	156
Figura 74. Inicio de Código Python de librería webserver.py	157
Figura 75. Estructura de Control If-else I de código Python.....	157
Figura 76. Estructura de Control If-else II de código Python.....	158
Figura 77. Estructura de Control If-else III de código Python.....	158
Figura 78. Estructura de Control If-else IV de código Python.	159
Figura 79.Estructura de Control If-else V de código Python.	159
Figura 80. Estructura de Control If-else VI de código Python.	159
Figura 81. Estructura de Control If-else VII de código Python.	160
Figura 82. Estructura de Control If-else VIII de código Python.....	160
Figura 83.Estructura de Control If-else IX de código Python.	160
Figura 84.Estructura de Control If-else X de código Python.	161
Figura 85.Estructura de Control If-else XI de código Python.	161
Figura 86. Estructura de Control If-else XII de código Python.	161
Figura 87. Estructura de Control If-else XIII de código Python.....	162
Figura 88. Programa WinSCP.....	163
Figura 89. Configuración de PuTTY.....	164
Figura 90. Servidor Web y Arduino conectado.....	164
Figura 91. Diagrama de bloques del SCV.....	166
Figura 92. Modulo Receptor Bluetooth HC-05.	168
Figura 93. Conexión EasyVR – ArduinoUno – Emisor HC-05.....	170
Figura 94. Comandos entrenados en EasyVR Commander.	172
Figura 95. Código de Programación del ArduinoUno.....	173
Figura 96. Integración del SCV.	174
Figura 97. Instalación del motor paso a paso a la puerta.....	175
Figura 98. Instalación motor-reductor y cortina.....	176

Figura 99. Circuito PCB de los motor-reductores de la puerta y cortina.	176
Figura 100. Mecanismo para el movimiento de la ventana.	177
Figura 101. Implementación del Driver A999.	178
Figura 102. Instalación de iluminación.	179
Figura 103. Instalación de alarma.	179
Figura 104. Instalación de calefacción.	180
Figura 105. Módulo de ocho relés para calefacción.	180
Figura 106. Cámara IP DCS-930L.	181
Figura 107. Implementación del tablero electrico.	182
Figura 108. Implementación de la página web del SCI.	182
Figura 109. Implementación del módulo de voz SCV.	183
Figura 110. Pruebas del Sistema de Control por Internet.	185
Figura 111. Pruebas del Sistema de Control por Voz.	186
Figura 112. La Sra. Hermelinda Morales antes de las pruebas.	187
Figura 113. Presentación del sistema al voluntario.	187
Figura 114. Prueba 1: Conexión con el página web principal	188
Figura 115. Prueba 1: Control por Internet de la habitación domótica.	189
Figura 116. Prueba 2: Control por Voz de la habitación.	190
Figura 117. Prueba 3: Cambio de grupo en cambios de voz.	191

INDICE DE TABLAS

Tabla 1. Características del motor-reductor TSUKASA.	100
Tabla 2. Características técnicas del motor-reductor SWF VALEO.	111
Tabla 3. Características del motor paso a paso MINEBEA-MATSUSHITA....	120
Tabla 4. Características del foco ahorrador marca Phelix.....	123
Tabla 5. Características de la estufa marca Imaco.	127
Tabla 6. Características de la alarma sonora.....	130
Tabla 7. Características de la cámara de vigilancia.	134
Tabla 8. Entradas y Salidas Digitales del Sistema Domótico.	136
Tabla 9. Entradas y Salidas del Arduino y nombre de etiquetas.	138
Tabla 10. Comandos de actividad.....	142
Tabla 11. Páginas Webs del SCI.	152
Tabla 12. Comandos Webs del SCI.....	154
Tabla 13. Características del módulo Bluetooth HC-05 para SCV.....	168
Tabla 14. Comandos de voz para el SCV.....	171

RESUMEN

La presente tesis abarca el diseño e implementación de un sistema domótico para aplicaciones en pacientes parapléjicos mediante control remoto a través de internet y reconocimiento de voz. Las personas con discapacidad personas a menudo sufren de discriminación e inferioridad lo que origina la falta de oportunidades e inclusión a la sociedad, independiente del país de donde vivan. En este grupo se encuentra las personas con lesión medular traumática, más conocido como paraplejía cuya calidad de vida es muy crítica dependiendo de la severidad de la lesión, recursos disponibles, aceptación de la lesión y sobre todo del nivel de independencia. Es por ello que la investigación presenta una alternativa para mejorar el grado de independencia y, por consecuencia, la calidad de vida de una persona con paraplejía. Por último, la tesis es clasificada como una investigación aplicada por su carácter científico y tecnológico.

Palabras claves: domótica, reconocimiento de voz, control por internet, discapacidad y paraplejía.

ABSTRACT

This thesis covers the design and implementation of an automated system for applications in paraplegic patients by remote control via internet and voice recognition. These people often suffer from discrimination and inferiority which causes a lack of opportunity and inclusion into society, regardless of the country where they live. This group is people with traumatic spinal cord injury, better known as paraplegia whose quality of life is very critical depending on the severity of the injury, available resources, acceptance of the lesion and especially the level of independence. That is why the research presents an alternative to improve the degree of independence and, consequently, the quality of life of a person with paraplegia. Finally, the thesis is classified as applied research for scientific and technological nature.

Keywords: automation, voice recognition, internet control, disability and paraplegia.

INTRODUCCION

En el Perú el número de personas con discapacidad corresponde al 11% de la población, es decir 3.3 millones de personas que seguirán incrementándose en las próximas décadas. Si a esto, además, se le suma el número de personas que, producto de la edad, accidentes o enfermedades, presentan algún tipo de discapacidad transitoria, se puede concluir que se trata de un grupo significativo de la población peruana (Cabrera, Chávez, 2013, p.20).

Si bien, cada discapacidad plantea una serie de retos distintos en la vida cotidiana de la persona que la padece, una de las problemáticas más recurrentes es la de la interacción del individuo con su entorno. Esto se debe principalmente a la poca movilidad asociada con el reposo, fracturas, vejez o al uso de aparatos como sillas de ruedas.

Normalmente, el ambiente cerrado con el que más frecuentemente interactúan las personas con discapacidad es el de su vivienda. En la vida diaria, una persona sin problemas motrices interactúa con los objetos de su entorno de manera natural y simple, puesto que éste fue diseñado y construido para ser utilizado de esta forma. Sin embargo, para las personas con paraplejia, el acto de encender la luz o alcanzar el control remoto, puede tornarse en una tarea complicada (Acosta *et al.*, 2015, p.9).

Por ello, desde hace algunos años se ha estado desarrollando un área de la informática que en conjunto con la mecánica y electrónica se aboca a la automatización de residencias o también llamada Domótica. Si bien esta tecnología podría ayudar a que las personas con discapacidad puedan desenvolverse mejor en sus hogares, en la actualidad estos sistemas se centran básicamente en aumentar el confort de personas que no presentan dificultades serias de desplazamiento.

Por tal motivo, esta tesis de tesis tiene como finalidad ofrecer una solución al problema mediante el diseño e implementación de un sistema domótico, el cual podrá ser controlado por una persona con paraplejia para facilitarle la realización de sus actividades cotidianas y así poder mejorar su calidad de vida.

CAPITULO I: PLANTEAMIENTO DEL PROBLEMA

1.1 Problema general y específico.

En las últimas décadas, la población mundial de personas con discapacidad física viene incrementando según el Informe Mundial sobre la Discapacidad 2011 (Banco Mundial, 2011, p.7). Los problemas de estas personas son muy similares independientemente del país donde vivan. Su principal característica es la situación de discriminación y de inferioridad, en cuanto a oportunidades sociales, que estas personas sufren respecto al resto de individuos de su entorno.

Entre este grupo se encuentra las personas con lesión medular traumática, más conocido como paraplejia. La paraplejia es una enfermedad que ocasiona que la parte inferior del cuerpo quede paralizada, careciendo de funcionalidad. El problema de estas personas es que su calidad de vida es muy crítica dependiendo de la severidad de la lesión, recursos disponibles, aceptación de la lesión y sobre todo del nivel de independencia. La mayoría de estas personas depende de una silla de ruedas para movilizarse, lo cual tiene un gran impacto mental y físico que ocasiona muchos problemas secundarios. Además, las condiciones psicosociales pueden incluir depresión y a largo plazo problemas relacionados con la pérdida del empleo, la pareja y el bienestar. La calidad de vida cambia precipitadamente después de una lesión traumática de la médula espinal.

Es por esto, que este trabajo de investigación nace a la problemática de mejorar la calidad de vida e independencia de las personas con paraplejia, además de incluirlo socialmente. En este caso el paciente podrá controlar diferentes dispositivos de su hogar a través de un sistema domótico. De esta manera, el problema general y específicos se describen a continuación:

Problema General

¿Cómo desarrollar el diseño e implementación de un sistema domótico para aplicaciones en pacientes parapléjicos mediante control remoto a través de internet y reconocimiento de voz?

Problemas Específicos

¿Cómo desarrollar el diseño e implementación de la arquitectura domótico en la habitación de una persona con paraplejia?

¿Cómo desarrollar el diseño e implementación de la interfaz de usuario que usará la persona con paraplejía mediante reconocimiento de voz y control remoto a través de internet?

¿Cómo desarrollar la adaptación del paciente parapléjico con el sistema domótico?

1.2 Objetivos

Para la consecución de la idea principal de esta Tesis de Grado, se proponen los siguientes objetivos, que al ser alcanzados, constituirán las principales aportaciones de este trabajo:

Objetivos Generales:

- Diseño e implementación de un sistema domótico para aplicaciones en pacientes parapléjicos mediante control remoto a través de internet y reconocimiento de voz.

Objetivos Específicos:

- Diseñar e implementar la arquitectura domótica en la habitación de la persona con paraplejia (Tesisista: Patricia Díaz).
- Diseñar e implementar la interfaz de usuario del sistema domótico que usará la persona con paraplejia mediante reconocimiento de voz y control remoto a través de internet (Tesisista: Jose Yabar).
- Validar la adaptación del usuario parapléjico con el sistema domótico.

1.3 Justificación

La robótica ha venido evolucionando con la creación de máquinas cada vez más complejas, que puedan asistir a las personas en todas sus actividades diarias. Una de ellas es la asistencia a personas parapléjicas que son dependientes en la realización de sus tareas cotidianas como vestirse, comer, realizar su aseo personal, etc.

Por ello, la importancia de esta tesis radica en que mejora la calidad de vida e independencia de las personas parapléjicas diseñando un sistema domótico en su hogar.

Además, se le incluye socialmente evitando la depresión y problemas a largo plazo que perjudiquen su salud física y mental. También se le incluye tecnológicamente porque necesita estas herramientas para desenvolverse con normalidad en su vida (Cabrera, Chávez, 2013, p.21).

Por otro lado, la ejecución del presente trabajo es un aporte científico que se desarrolla en la Escuela Profesional de Ingeniería Mecatrónica de la Facultad de Ingeniería en la Universidad Ricardo Palma. El trabajo de tesis permitirá ampliar conocimientos de técnicas emergentes en diseño de sistemas domóticos.

1.4 Alcances y limitaciones de la investigación.

La tesis en desarrollo pretende diseñar e implementar un sistema domótico en la habitación de una persona con discapacidad parapléjica donde se le permita abrir/cerrar una puerta de ingreso de 20 kilogramos, enrollar/desenrollar una cortina roller de 1.7 metros, abrir/cerrar ventana de vidrio de 0.75 metros, prender/apagar la iluminación de 18W, prender/apagar la calefacción de 2000W, prender/apagar una alarma sonora 85dB y observar las imágenes proveídas de una cámara de video vigilancia web. Además, esta tesis tiene el alcance de diseñar e implementar las interfaces

de usuario que le permitirá al paciente parapléjico controlar la habitación. La primera se trata de una interfaz web escrita en PHP y la segunda de una interfaz de voz a través del módulo EasyVR 3.0. Ambas interfaces se comunican con un sistema de control central que utiliza un Arduino Mega 2560.

Entre las limitaciones de la tesis en desarrollo se tiene los siguientes:

- Se utiliza dispositivos electrónicos de código abierto (open source) con el objetivo de replicar y ampliar en futuras aplicaciones.
- No se realiza mediciones cuantitativas de interacción entre la persona y el sistema domótico. Más bien, se toma las sugerencias y comentarios de una persona con paraplejía.
- No se cuenta con el apoyo de un gran número de pacientes con paraplejía para las pruebas. Sin embargo, será posible adaptarlas a las necesidades especiales de cada una.

1.5 Estructura de la tesis.

La tesis se encuentra dividida en 4 capítulos, Conclusiones, Recomendaciones, referencias bibliográficas y anexos como se muestra esquemáticamente en la figura 1.

La estructura de la tesis responde a la resolución secuencial del problema de diseño e implementación de un sistema domótico con aplicaciones en

pacientes parapléjicos mediante control remoto a través de internet y reconocimiento de voz. A continuación, se describe la composición de los cuatro capítulos y un breve resumen de los mismos.

Introducción y Capítulo 1: Planteamiento del Problema

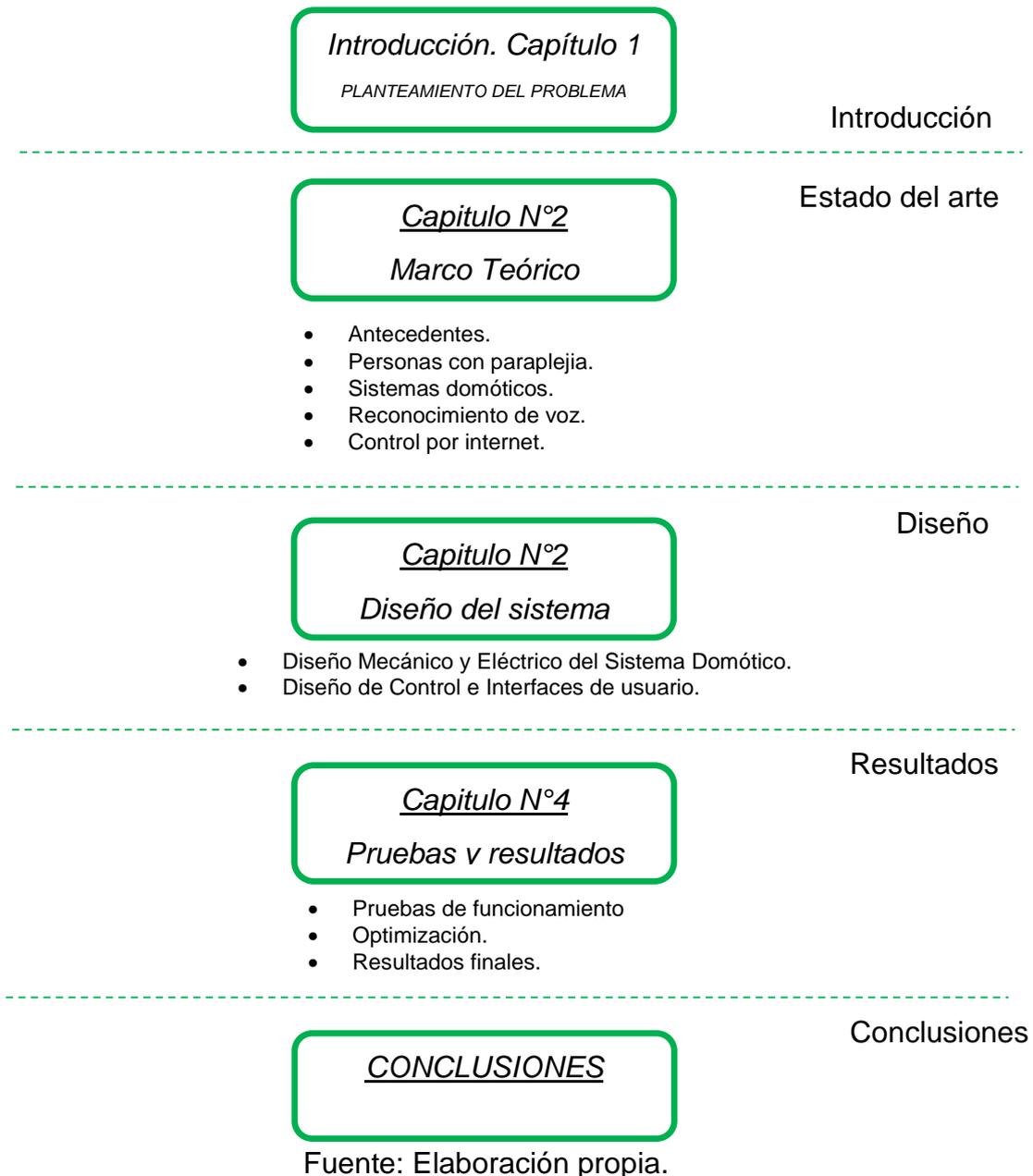
En este capítulo se expone una introducción a la tesis y el planteamiento del problema y los objetivos de la misma.

A pesar de la cantidad de actuadores, sensores y controladores, realizar un sistema de tales características es un reto apasionante con una gran cantidad de variables a considerar para su diseño e implementación.

Capítulo 2. Marco teórico

En este capítulo se describe los antecedentes y conceptos teóricos que enmarcan la investigación. Así como se hace referencia a las tecnologías que actualmente ayudan a las personas con paraplejia. Por otro lado, se estudia el marco de la domótica, que nos ayudará a entender las consideraciones que se deben tener antes de empezar a diseñar e implementar el sistema.

Figura 1. Estructura de la tesis.



Capítulo 3. Diseño e Implementación del sistema

En este capítulo de la tesis se describe el diseño e implementación de todo el sistema propuesto. Se inicia con la descripción del sistema domótico: sistema mecánico, eléctrico y de electrónica de potencia. Concluyendo con el diseño e implementación del control e interfaces de usuario.

Capítulo 4. Pruebas y Resultados

En este capítulo se describe las pruebas y los resultados obtenidos durante la investigación.

Conclusiones y Recomendaciones

Finalmente, se presenta las conclusiones, recomendaciones y las futuras líneas de investigación que se pueden generar a partir de la presente tesis.

CAPITULO II: MARCO TEORICO

2.1 Antecedentes de la investigación.

Múltiples instituciones científicas nacionales e internacionales vienen desarrollando proyectos e investigaciones afines al proyecto de tesis propuesto. Podemos citar los siguientes trabajos de investigación:

- Acosta, Christian; Tipán, Carlos (2015). En su tesis de pregrado para obtener el título profesional de Ingeniero en Electrónica y Control titulada, Implementación de una consola domótica residencial con control local y remoto a través del internet. Escuela Politécnica Nacional, Quito Ecuador, en la que concluye que: Se logró que el sistema se disponga de una pantalla táctil que permita manejar el interfaz, el mismo que fue desarrollado se ejecuta en la consola y se comunica con los elementos inalámbricos. Todo esto se alcanzó con el sistema embebido FriendlyARM Mini2440. El lenguaje de programación escogido en el lado servidor fue PHP ya que actualmente es el más utilizado, gratuito, con amplia documentación y de fácil aprendizaje. Además es multiplataforma y ofrece compatibilidad con la mayoría de sistemas gestores de bases de datos (SGBD) (p.116).
- Alvarez, Carlos; Holguin, Danilo; Serrano, Ernesto (2007). En su tesis de pregrado para obtener el título profesional de Ingeniero en Electrónica y Telecomunicaciones titulada, Diseño de una Instalación Domótica en un Condominio para el Control de Seguridad e Iluminación mediante la Tecnología LonWorks. Escuela Superior Politecnica del Litoral, Guayaquil

Ecuador, en la que concluye que este proyecto es una solución integral en cuanto a las necesidades básicas de ahorro energético y seguridad de un sistema domótico (p.231).

- Barbecho, José; Fernandez, Wilian (2013). En su tesis de pregrado para obtener el título profesional de Ingeniero Eléctrico titulada, Aplicaciones multimedia para control de un sistema domótico orientado a personas con discapacidad. Universidad de Cuenca, Cuenca Ecuador. en la que concluye que: Los comandos de voz permite que personas con discapacidad visual puedan hacer uso de su voz, a la hora de manipular una carga eléctrica, con lo cual logramos que tengas autonomía al momento de la activar y desactivar un dispositivo eléctrico a su vez los comandos de voz logran que la persona discapacitada o persona sin ningún grado de discapacidad puedan manipular de una manera sencilla, varias cargas que son comunes en una vivienda familiar (p. 164).
- Cabrera, Rony; Chavez, Paul (2013). En su tesis de pregrado para obtener el título profesional de Ingeniero Mecatrónico titulado, Diseño e implementación de un robot humanoide asistencial con aplicaciones en pacientes parapléjicos. Universidad Ricardo Palma, Lima Perú. En la que se concluye que es posible utilizar tecnologías de robot humanoides asistenciales para mejorar la calidad de vida de personas con paraplejia (p.305).
- Guerra, Felipe (2013). En su tesis de pregrado para obtener el título profesional de Ingeniero de Telecomunicaciones titulado, Diseño de un sistema de control domótico y video vigilancia supervisado por un teléfono móvil. Pontificia universidad católica del Perú, Lima Perú, en la concluye

que: El sistema, puede ser incrementado en tamaño y funcionalidades con relativa facilidad, debido a que ocupa una parte mínima de la capacidad de la red local, así como de la capacidad en la salida a internet de la vivienda, este incremento lo puede realizar la empresa que provee el servicio. Debido a las características de los componentes utilizados, el sistema no sería apto para su instalación en exteriores, sino únicamente dentro de la vivienda (p. 65).

- Heriquez, Mauricio (2005). En su tesis de pregrado para obtener el título profesional de Ingeniero en Comunicación titulado Investigación y desarrollo sistema prototipo de asistencia domótica para personas con movilidad limitada. Universidad Austral de Chile, Puerto Montt, Chile, en la que concluye que un primer prototipo de asistencia domótica con software abierto abre las puertas para futuras investigaciones interdisciplinarias para personas con movilidad limitada (p. 298).
- Miranda, Guido; Villacrés, Raul; Villamar, Franklin (2009). En su tesis de pregrado para obtener el título profesional de Ingeniero en Electrónica y Telecomunicaciones titulado, Diseño de un sistema domótico aplicado a una clínica de hemodiálisis. Escuela Superior Politécnica del Litoral, Guayaquil, en la que concluye que: para el desarrollo de la Domotización de su Clínica de Hemodiálisis, es posible realizarlo con los dispositivos X-10, ya que su costo en implementación es inferior comparada con otras tecnologías a fines, ya que consisten en módulos plug & play, es decir que son adaptables a la línea eléctrica ya definida en la clínica y no necesitan de configuraciones para su utilización (p. 116).

- Montalvo, Christian (2011). En su tesis de pregrado para obtener el Título de Ingeniero en Sistemas Computacionales titulado, Control y Monitoreo de Edificios para Personas con Discapacidad, Mediante El Uso De Tecnología Mobile, Web, Sms y Reconocimiento de Voz. Universidad Técnica del Norte, Ecuador, en la que concluye que: El SICMEI permitió controlar los dispositivos eléctricos y electrónicos conectados a una red, dentro de cualquier edificio. El acceso a estos dispositivos era dificultoso para las PcD, dependiendo de terceras personas para acceder a estos, a su vez los reportes emitidos son de gran importancia para los familiares de las PcD, ya que les permite controlar el funcionamiento a lo largo de la jornada, además permitiendo ver quien tuvo acceso al sistema (p. 280).
- Pérez, Virgilio (2010). En su tesis de maestría para obtener el grado de Maestro en ciencias titulado, Contribución al diseño de sistemas domóticos y de entretenimiento utilizando hardware libre y software de código abierto. Instituto Politécnico Nacional, Tijuana, México. En la que concluye que es posible realizar la implementación de un sistema domótico y de entretenimiento utilizando hardware libre y software de código abierto, reduciendo costos de implementación y de diseño (p. 52).
- Rodríguez, Mario (2014). En su tesis de pregrado para obtener el título de Ingeniero en Electrónica y Comunicaciones titulado, Sistema de control remoto para aplicaciones domóticas a través de internet. Universidad Autónoma de Madrid, Madrid. En la que concluye que la aplicación web tiene la ventaja de ser accesible desde cualquier dispositivo con navegador y permite con facilidad añadir nuevas opciones de control y supervisión, así como nuevas funcionalidades (p. 126).

- Rodriguez, Wally (2012). En su tesis de pregrado para obtener el título de Ingeniero Electrónico titulado, Sistema de control domótico utilizando una central ip pbx basado en software libre. Pontificia Universidad Católica del Perú, Lima Perú. En la que concluye que mediante las interfaces se pudo controlar y recibir notificaciones de dos zonas geográficas, comunicándose desde distintos lugares de una manera eficiente (p. 53).
- Roman, Roger (2011). En su tesis de pregrado para obtener el título de Diseñador Industrial titulado, Diseño de un sistema domótico para control de iluminación y monitoreo de consumo eléctrico. Universidad industrial de Santander, Bucaramanga. En la que concluye que se implementa la interfaz gráfica como un medio de control más intuitivo y fácil de reconocer por el usuario, proponiendo a la domótica como herramienta tecnológica para el diseño ecológico y sustentable de gran valor para la comunidad pero que aún no se ha implementado totalmente en nuestra sociedad (p. 150).

2.2 Óptica de la Investigación.

Hipótesis General

El diseño e implementación de un sistema domótico para aplicaciones en pacientes parapléjicos mediante control remoto a través de internet y reconocimiento de voz, es posible y de gran relevancia porque satisface necesidades de usuarios con paraplejia mejorando su calidad de vida e independencia.

Hipótesis específicas

- El Diseño e implementación de la arquitectura domótica en la habitación de la persona con paraplejía, es viable y de gran relevancia.
- El Diseño e implementación de la interfaz de usuario del sistema domótico que usará la persona con paraplejía mediante reconocimiento de voz y control remoto a través de internet, es de gran importancia y factible.
- La validación de la adaptación del usuario parapléjico con el sistema domótico, es de gran importancia para comprobar su satisfacción.

2.3 Selección de Variables.

El diseño e implementación de un sistema domótico para aplicaciones en pacientes parapléjicos mediante control remoto a través de internet y reconocimiento de voz.

a. Variable 1: Sistema domótico a través de internet.

Indicadores:

- Potencia Motores DC.
- Desplazamiento de la persiana, ventana y puerta.
- Consumo de corriente eléctrica.
- Velocidad de navegación.

b. Variable 2: Sistema domótico por voz.

Indicadores:

- Potencia Motores DC.
- Desplazamiento de la persiana, ventana y puerta.

- Consumo de corriente eléctrica.
- Velocidad de Procesamiento de los comandos de voz.

b. Variable 3: Adaptación del usuario con paraplejia y el sistema domótico.

Indicadores:

- Nivel de satisfacción con el sistema.
- Tiempo de adaptación.

2.4 Bases Teóricas.

2.4.1 Las personas con paraplejia.

2.4.1.1 Definición de la paraplejia.

Etimológicamente, la palabra paraplejia, proviene de dos raíces griegas y su significado es: “PARÁ” que significa “al lado de” y “PLEXIA” que significa “ataque paralizante”. Desde el punto de vista clínico, este término designa parálisis bilateral, de ambas piernas o brazos (Ríos, 1985, p.20).

La lesión de la médula a un nivel alto (en la parte de la columna espinal correspondiente a la nuca), tiene como resultado, la cuadriplejia, con parálisis de brazos, así como de piernas, pero cuando el daño afecta a la porción torácica o inferior de la columna, únicamente se paralizan las piernas.

La parálisis implica pérdida de la actividad muscular voluntaria, por lesión de un punto cualquiera de las vías motoras

piramidales, desde la corteza cerebral, hasta la fibra muscular, dependiendo del tipo de lesión medular, también se presentan asociados trastornos de sensibilidad y lesiones atróficas.

2.4.1.2 Tipos de paraplejia.

Según Pérez (2015, p.40), existen distintos tipos de paraplejia y cada una tiene una forma diferente de tratarse.

- Paraplejia completa: la persona pierde toda la función motora, quiere decir que los nervios no funcionan correctamente.
- Paraplejia incompleta: la persona puede llegar a tener una recuperación parcial o completa de la enfermedad.

2.4.1.3 Causa de la paraplejia.

Son múltiples y variadas las causas que originan un síndrome de paraplejia, siendo los traumatismos los que ocupan el primer lugar, ya sea directa o indirectamente.

La incidencia en la vida civil es el resultado, en primer lugar, de agresiones físicas, accidentes automovilísticos, accidentes industriales, accidentes en el deporte, accidentes en el hogar y en segundo lugar las secuelas que resultan de las diversas

enfermedades y tumores (Ríos, 1985, p.33). A continuación describiremos cada una de ellas:

- Agresiones físicas

Estos tipos de accidentes son, generalmente, producidos en pleitos callejeros, al tratar de robar o por simples equivocaciones, dejando un saldo trágico de heridos que gracias a la medicina actual logran salvar la vida pero no así todas sus funciones quedando con una secuela invalidante de por vida.

- Accidentes automovilísticos

Originados por atropellos, volcaduras y choques, los cuales son provocados por conductores irresponsables que manejan en estado de ebriedad y comúnmente por el exceso de velocidad.

- Accidentes Industriales

Originados por la falta de protección frente a las máquinas de trabajo en una empresa.

- Accidentes en el hogar

Provocados por resbalones o caídas, y aunque estos parecen insignificantes pueden dejar trastornos notables desde el punto de vista médico.

Todos los accidentes que se han señalado anteriormente son los que ocasionan con mayor frecuencia la invalidez llamada paraplejía pero gran parte de ellos pueden ser evitados con un

programa de seguridad social e industrial donde se establezca un sistema normativo para la seguridad de los individuos.

2.4.1.4 Calidad de vida de las personas con paraplejia.

La calidad de vida de los usuarios parapléjicos es compleja, interactuando la severidad de la lesión, nivel de independencia, recursos disponibles, aceptación de la lesión y actividades sociales para la promoción de la salud. El usuario se encuentra en el riesgo de desarrollar condiciones secundarias que incluyen: infecciones crónicas de vías urinarias, úlceras de decúbito, espasticidad, entre otras. Además las condiciones psicosociales pueden incluir depresión y a largo plazo problemas relacionados con la pérdida del empleo, la pareja y el bienestar. La calidad de vida cambia precipitadamente después de una lesión traumática de la medula espinal.

Una condición secundaria a la lesión como la espasticidad es un factor que contribuye importantemente a la presentación de úlceras de presión. Estas lesiones retardan el tratamiento de rehabilitación y su reintegración a la comunidad, llegando a ser inclusive causa de muerte. Otro cambio físico que puede tener un impacto en la calidad de vida inmediata y futura de los usuarios parapléjicos, es el deterioro de la función vesical e

intestinal. Esta condición predispone desde el punto de vista médico y social a disminuir la calidad de vida, por el riesgo de infecciones urinarias, el estreñimiento, impactación fecal, las úlceras de presión y otras.

Desde el punto de vista social, la satisfacción personal es perceptiblemente más baja entre aquellos pacientes dependientes del cuidado vesical e intestinal con respecto a aquellos usuarios que realizan estas funciones independientemente, con una ventaja en independencia física, movilidad y funcionamiento ocupacional. Los usuarios independientes refirieron una mejor salud física general que los usuarios dependientes.

2.4.1.5 Tecnologías que ayudan a las personas con paraplejia.

Actualmente, existen sistemas que restauran total o parcialmente la capacidad perdida por el usuario discapacitado. A continuación se muestran tales sistemas con la finalidad de conocer sus características, ventajas y desventajas (Jardón, 2008, p.26).

Silla de ruedas

La silla de rueda está diseñada para permitir el desplazamiento de personas con problemas de locomoción debido a una lesión,

enfermedad física o psicológica. Las sillas de ruedas mecánicas son las más usadas por la mayoría de personas debido a su costo relativamente bajo, fácil adquisición y transportación, menor tamaño y peso y además de proporcionar beneficios físicos. Pero estas pueden causar dolor en las extremidades superiores y espalda, por eso las sillas de ruedas eléctricas son una buena opción para las personas que no tienen mucha fuerza o no tienen movilidad en sus miembros superiores. Además de que una silla de ruedas eléctrica tiene la posibilidad de integrar tecnologías que mejoren el control y la interacción del discapacitado con su entorno. Las señales cerebrales, mioeléctricas, el movimiento de la lengua, la voz y el seguimiento de los ojos pueden ser procesados e interpretados por un algoritmo computacional para la movilización de la silla de ruedas hacia la derecha, izquierda, adelante o atrás facilitando enormemente el control de la misma (Alcobendas, 2011, p.111).

Por otro lado, la posibilidad de montar un brazo robot sobre la silla de ruedas le da al usuario la capacidad de manipular objetos de su entorno, además de desplazarse libremente en él. El robot más conocido es “Jaco” de la empresa canadiense Kinova que puede ser montado en una silla de ruedas y ser controlado a través de un joystick para permitirle a la persona discapacitada comer, beber, abrir puertas y armarios, afeitarse,

lavarse los dientes o rascarse. Está fabricado de fibra de carbono pesando 5kg y es capaz de coger objetos de hasta 1.5kg con una distancia de casi un metro. Tiene 6 grados de libertad y de efector tiene una mano con tres dedos móviles recubiertos de un material antideslizante que permite recoger objetos con gran precisión, tal como se muestra en la figura 2.

Figura 2. Robot asistencial Jaco.



Fuente: Cabrera, Chavez, (2013, p.37)

Sistemas robotizados

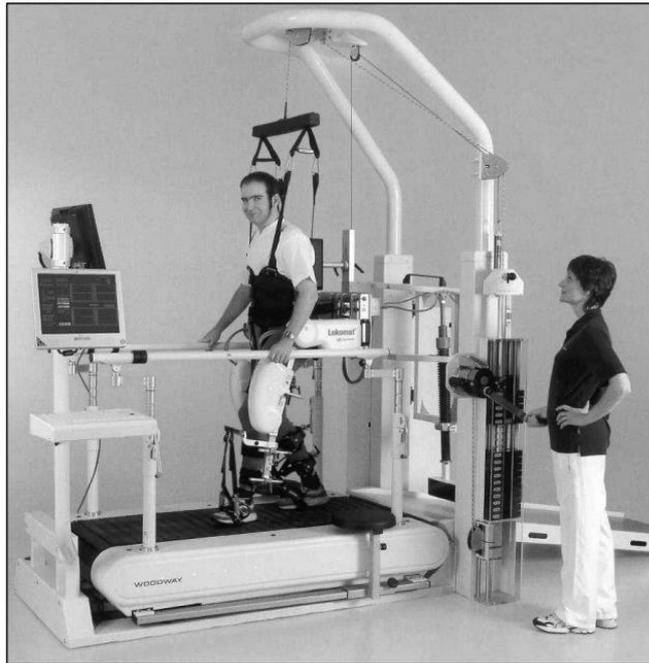
Actualmente, existen sistemas que ayudan a la rehabilitación del usuario parapléjico estimulando sus miembros inferiores. En esta sección hablaremos de los sistemas robotizados para rehabilitación “Lokomat” y “Rewalk” que son la mayor esperanza para las personas parapléjicas de poder volver a caminar. El objetivo del sistema lokomat es estimular, por repetición, los centros generadores de movimiento que existen en la medula espinal (Alcobendas, 2011, p.94). Lo que se hace es que al usuario por medio de una ortesis robótica se le

moviliza las caderas y las rodillas simulando una marcha “fisiológica”, es decir es un sistema de reeducación de la marcha. Este consta de una estructura anclada al techo, desde donde se suspende al usuario en un arnés para sostenerlo en pie sobre una cinta sin fin.

El control y la coordinación de la cinta con el movimiento de la ortesis robótica se llevan a cabo a través de un sistema informático que permite ajustar la longitud del paso, los arcos articulares de caderas y rodillas de forma simétrica o asimétrica y también de la velocidad de la marcha.

El equilibrio del usuario se consigue mediante un sistema de cargas dinámicas que permiten el ascenso y descenso del centro de masas durante la marcha. La ortesis robótica dirige el movimiento de los miembros inferiores, permitiendo la intervención de la musculatura del usuario, se puede ir permitiendo una mayor intervención del usuario según mejore la fuerza muscular, al disminuir la fuerza guía del sistema. En la figura 3 puede observarse el sistema Lokomat.

Figura 3. Sistema Lokomat.

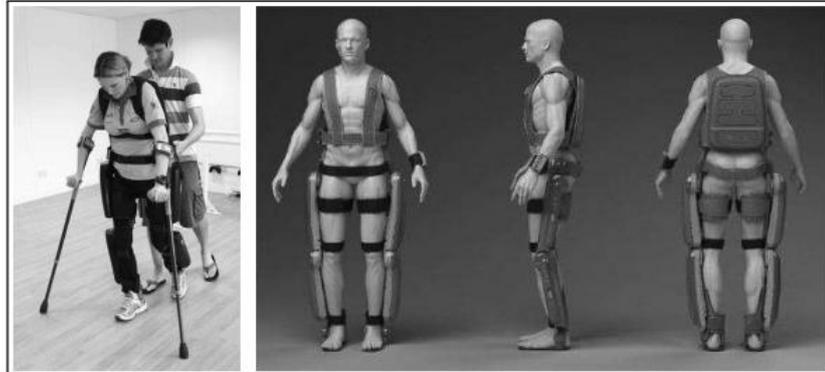


Fuente: Cabrera, Chavez, (2013, p.40)

Por otro lado, hacer que una persona parapléjica sea capaz de moverse por casa, estar de pie y hablar con la gente cara a cara son los objetivos del Sistema ReWalk de la empresa Argo Medical, como se muestra en la figura 4. Es un exoesqueleto que es controlado por un control remoto que lleva el usuario permitiendo seleccionar el modo de movimiento, caminar, subir escaleras, subir gradas, sentarse y pararse, etc. Contiene sensores que detectan el movimiento de la parte superior que son analizados y utilizados para activar y mantener patrones de caminata y otros medios de operación, por ejemplo: cuando el usuario activa la opción caminata en el control remoto, solo es necesario que se incline un poco hacia adelante para que la acción comience. Muletas y una mochila (donde se encuentra el sistema de control y sus baterías) son necesarias al usar el

exoesqueleto ReWalk, su peso es de 15kg y la duración de sus baterías son de 8 horas por día.

Figura 4. Exoesqueleto Rewalk.



Fuente: Cabrera, Chavez, (2013, p.43)

Robots Asistenciales

Uno de los robots asistenciales conocidos es el robot ASIBOT, que fue desarrollado por el RoboticsLab de la Universidad Carlos III de Madrid en España, cuyo objetivo principal es introducir elementos “inteligentes”, móviles y sobre todo modulares en la casa del discapacitado, tal como se muestra en la figura 5. Se trata de un brazo robot escalador de 5 GDL simétrico. Cada extremo tiene un mecanismo de anclaje para conectarse a la pared o a la silla de ruedas y una pinza. El robot tiene una estructura de aluminio en sus diferentes articulaciones y los eslabones son de fibra de carbono. De esta forma, el robot con tal solo 12kg, es autónomo salvo que precisa de conexión exterior para su alimentación.

Figura 5. Robot Asibot.



Fuente: Cabrera, Chávez, (2013, p.46)

Otro ejemplo de robots asistenciales es el robot Nuka. La terapia psicológica con animales es muy usada alrededor del mundo pero presenta sus riesgos, es por eso que el Instituto de Ciencia y Tecnología Industrial Avanzada de Japón (AIST) creó al robot Nuka que es una foca bebé con apariencia y textura muy amigable, siendo su mayor potencial el afectivo. Por tanto, está indicada para ser utilizada en tratamientos con personas con síntomas de demencia, personas mayores, niños hospitalizados o personas con alguna discapacidad y que tengan depresión como son las personas parapléjicas. En la figura 6 se muestra una imagen del robot. (Cabrera, Chávez, 2013, p 32).

Figura 6. Robot Nuka.



Fuente: Cabrera, Chávez, (2013, p.47)

2.4.2 Sistema Domótico.

2.4.2.1 Definición.

La domótica es el conjunto de tecnologías aplicadas al control y la automatización inteligente de una vivienda, tal como se muestra en la figura 7, que permite una gestión eficiente del uso de la energía, que aporta seguridad y confort, además de comunicación entre el usuario y el sistema. Domótica es la unión entre domus (casa en latín) y robótica (de robota, esclavo o servidor en checo). Sin embargo, la domótica no limita su campo de acción a las viviendas sino que existe otro tipo de edificaciones (Álvarez *et al.*, 2007, p.4).

Figura 7. Tecnologías en un Sistema Domótico.



Fuente: Álvarez *et al.*, (2007, p.13)

Además, hay que hacer notar la diferencia que existe entre la implantación de un sistema domótico en una vivienda unifamiliar, en un edificio o en una ciudad. Así, aparecen nuevos términos como inmótica edificios (para el sector terciario como oficinas, hoteles, etc.) y urbótica para las (ciudades o edificios inteligentes).

La domótica permite dar respuesta a los cambios sociales y nuevas tendencias de nuestra forma de vida, facilitando la flexibilidad y polifuncionalidad del diseño de casas y hogares más humanos, más personales.

2.4.2.2 Historia de la domótica.

Desde el modo de vida, hace tan solo una década, la incursión en nuestros hogares de las tecnologías de la información y la comunicación, las tecnologías vinculadas al entretenimiento, a

la búsqueda de la información, al trabajo y al ocio, han transformado nuestros hogares y la forma en que vivimos en ellos. La domótica forma parte de esta revolución tecnológica desde 1972, tras muchas investigaciones debido a la falta de combustible y tratar de ahorrar energía aparecieron los primeros equipos domotizados de edificios basados en la Tecnología de vanguardia y manteniéndose en constante desarrollo a un siendo uno más que lideran el mercado los llamados X-10 mientras que los años posteriores las grandes empresas encargadas de la fabricación de equipos eléctricos y electrónicos europeos y asiáticos manifestaron un creciente interés por la búsqueda de los dispositivos. En Norte América fueron los primeros en incursionar al mercado sistemas simples y específicos al control de temperatura ambiente de los edificios de oficinas.

A medida que iba acrecentando el uso de ordenadores, ya terminando la década de los ochenta e iniciando los de los noventa, se empezaron a incorporar en los edificios los sistemas de cableado estructurado para facilitar la conexión de todo tipo de terminales y periféricos entre sí, utilizando un cableado estándar y tomas repartidos por todo el edificio. Además de los datos, estos sistemas de cableado permitían el transporte de la voz y la conexión de algunos dispositivos de control y de seguridad (Atahualpa, 2014, p.10).

2.4.2.3 De la instalación convencional a la Domótica.

En una instalación convencional el usuario se conformaba solamente con iluminarse, calentarse y disponer de puntos de acceso para poder conectar los dispositivos eléctricos. En estas condiciones los requisitos eléctricos, normativos, se limitaban a proteger las líneas y a las personas contra riesgos eléctricos. Con el transcurso de los años, se han ido añadiendo nuevas necesidades al confort y la mejor gestión de la energía, entre otras.

Para dar respuestas a estas demandas, han ido apareciendo en el mercado aparatos que permiten aplicaciones específicas como programar la calefacción, regular la temperatura ambiental, regular la iluminación, conectar una alarma, etc.

Esta incorporación aumenta una mayor complejidad en la instalación (mayor densidad de cables, etc.), dado que cada aparato funcionaba independientemente del resto, lo cual se traduce en largas y costosas intervenciones por parte del instalador eléctrico, del albañil, etc. Este problema requería una respuesta tecnológica que fuese capaz de:

- Disminuir el cableado.

- Aumentar la flexibilidad que comparte una fácil aplicación y/o reconfiguración del sistema.
- Integrar los servicios y las instalaciones en un solo sistema, posibilitando la comunicación e interacción entre ellos.

Con los avances en los microprocesadores, la convergencia de la informática y las telecomunicaciones aparecen nuevas tecnologías en la automatización de las instalaciones eléctricas de viviendas y edificios. Estas soluciones tecnológicas han configurado las características esenciales que requiere un sistema domótico:

- Sencillez y fiabilidad.
- Flexibilidad y modularidad.
- Facilidad de aplicación.
- Estandarización.
- Sin mantenimiento.
- No alterar el estilo de vida del usuario.

Además, en contra de lo que en un principio puede parecer, los costos de una instalación domótica, a partir de cierta funcionalidad son inferiores a los que requiere una instalación convencional.

2.4.2.4 Especificaciones de la Gestión Domótica.

El significado propio de la domótica es lograr una mejor utilización, gestión y control de las redes que permiten seguridad, confort y ahorro energético en las viviendas y edificios. Cuando se habla de domótica nos referimos al control de instalaciones de manera presencial o bien de manera remota. Su aportación es un control y una gestión eficiente de todos los equipos y tecnologías que forman la red. Los objetivos que se quiere conseguir una vez se aplica esta combinación de tecnologías son:

- Control y gestión de las instalaciones que forman la vivienda o edificios por parte del usuario, para poder disfrutar de un grado elevado de comodidad.
- Las tecnologías aplicadas a un edificio han de permitir una fácil adaptación a los cambios que puedan suceder en la red que forman como ampliar los sensores de una habitación, mejorar los modelos de los actuadores de una sala, etc. y una optimización del consumo de los recursos energéticos de los que dispone el edificio.
- Conexión del edificio o vivienda con el exterior, ya sea para poder realizar la compra vía internet o acceder a visualizar algún espacio del edificio de manera remota.

La seguridad ha de ser efectiva, no solo limitarse a alarmas de entrada sino también a simuladores de presencia en el edificio.

Estos suben y bajan persianas, encienden y apagan luces y más acciones en diferentes horas del día para evidenciar que el edificio está habitado. Los tipos de instalaciones que hay en grandes edificios son muy variados: climatización, control de acceso, alumbrado, etc. Todas ellas utilizan dispositivos diferentes que permiten realizar sus aplicaciones. Algunos ejemplos de grandes edificios son: hoteles, centros comerciales, instalaciones deportivas, naves industriales, etc. Estos objetivos permiten el desarrollo de la domótica en distintas aplicaciones abarcando diferentes sectores, teniendo todos en común el control del hogar o edificio donde hay personas. (Atahualpa, 2014, p.13)

2.4.2.5 Sistemas de Control Domóticos.

Los sistemas de control son aquellos sistemas capaces de recoger información proveniente de unas entradas (sensores o mandos), procesarlas y emitir órdenes a unos actuadores o salidas, con el objeto de conseguir confort, gestión de la energía o la protección de personal o bienes.

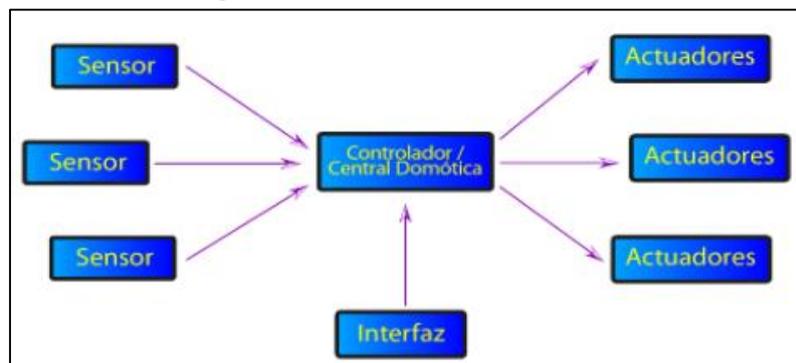
2.4.2.5.1 Sistemas Centralizados.

Son aquellos sistemas en los que los elementos para controlar y supervisar (sensores, luces, válvulas, etc.)

están conectados en un único punto, generalmente a la unidad de control central, que contiene la inteligencia de todo el sistema tal como se muestra en la figura 8. En los sistemas de configuración centralizada la comunicación entre elementos pasa, por tanto, por la unidad central.

Este sistema es ampliamente utilizado, por ejemplo, en gestión de la seguridad o en gestión de la energía. Los primeros sistemas domóticos de tipo centralizados, utilizados en automatización de viviendas, fueron los autómatas programables.

Figura 8. Sistema Centralizado.



Fuente: Atahualpa, (2014, p.15).

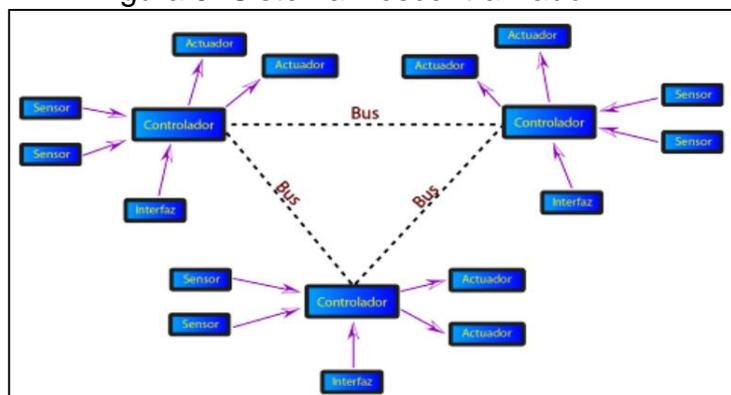
2.4.2.5.2 Sistemas Descentralizados.

En este tipo de sistemas existen diferentes elementos de control, cada uno de los cuales posee la capacidad de tratar la información que recibe y actuar en

consecuencia de forma autónoma tal como se muestra en la figura 9. En estos sistemas los elementos de control están lo más cerca posible de los elementos que se deben controlar.

Mediante este método se eliminan los dos problemas mencionados en el sistema centralizado. No existe una unidad de control central y por lo tanto, el usuario no depende de un solo fabricante; por otra parte, la avería de cualquier elemento no afecta al funcionamiento del resto. Además existen otras ventajas, como la facilidad de reconfiguración del sistema, lo que incide directamente en el grado de flexibilidad, y sobre todo en el ahorro de cableado en la instalación. Al tratarse de sistemas más caros, también son sistemas más potentes, que permiten implementar una gran cantidad de aplicaciones y servicios al usuario.

Figura 9. Sistema Descentralizado.

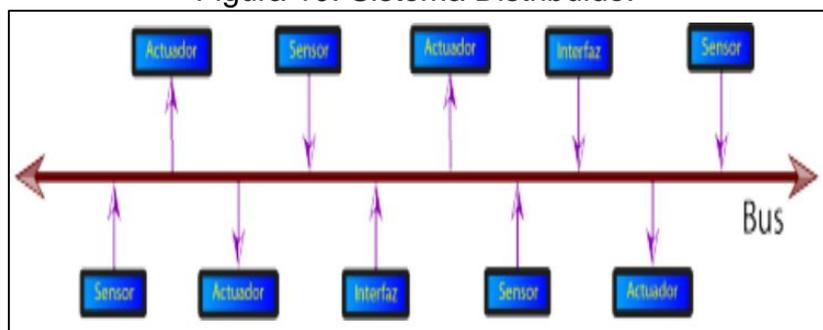


Fuente: Atahualpa, (2014, p.16).

2.4.2.5.3 Sistemas Distribuidos.

Este sistema de control se puede considerar como una combinación de los dos anteriores. Los sistemas de control distribuido disponen de varios controladores próximos a los que deben controlar, y entre ellos están unidos por algún medio físico de transmisión (por ejemplo BUS), como se muestra en la figura 10 (Atahualpa., 2014, p.16).

Figura 10. Sistema Distribuido.



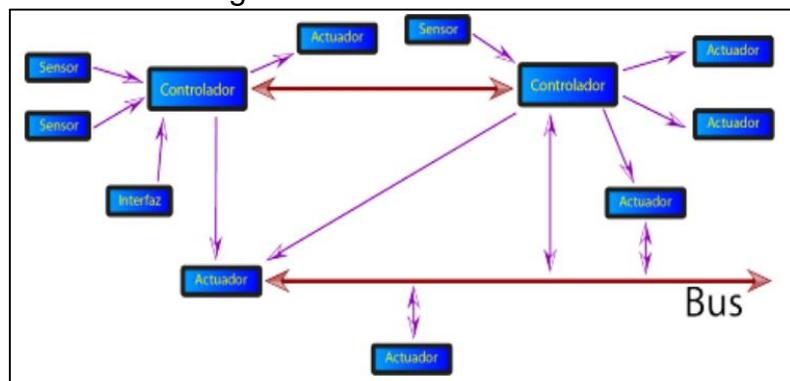
Fuente: Atahualpa, (2014, p.17).

2.4.2.5.4 Sistemas Híbrida/Mixta.

La complejidad de este sistema es debido a su combinación de sus arquitecturas como son los sistemas distribuidos, centralizados o descentralizados, tal como se muestra en la figura 11. Por lo que puede disponer de un mando o controlador central o varios mandos o controladores

descentralizados, en este tipo de sistemas mixtos las interfaces pueden tener una doble funcionalidad tanto los sensores y actuadores pueden también ser controladores y procesar la información según el programa, o la configuración, y pueden actuar de acuerdo a ella, estando en la misma red, sin que necesariamente pase por un controlador.

Figura 11. Sistema Híbrido.



Fuente: Atahualpa, (2014, p.17).

2.4.2.5.5 Conexiones entre dispositivos.

Todo sistema domótico tiene unos componentes básicos, los dispositivos, una vez formada la red a la cual pertenecen y donde actuarán, se han de conectar entre ellos y con las unidades de control.

La primera forma de conexión sería uniéndolos todos los elementos de la red mediante un par de cables para

interconectar toda la red, la llamada red de estrella. El problema de este tipo de conexión es su facilidad para dejar de funcionar, en cuanto un dispositivo deje de funcionar o un cable se rompa, la red deja de estar conectada y se perdería el control de toda la red.

El segundo modo de conexión es unir toda la red a partir de un bus de comunicación que comparten todos los dispositivos que forman dicha red. Los medios físicos más habituales en domótica para unir los dispositivos son:

- Red eléctrica.
- Corrientes portadoras.
- Cable par trenzado.
- Bus.
- Cable coaxial.
- Infrarrojos.
- Fibra óptica.
- Radiofrecuencia.

2.4.2.6 Componentes del Sistema Domótico.

2.4.2.6.1 Sensores en Domótica.

Un sensor es un dispositivo capaz de detectar magnitudes físicas o químicas, llamadas variables de instrumentación, y transformarlas en variables eléctricas (Reyes et al., 2013, p.26). Las variables de instrumentación pueden ser por ejemplo: temperatura, intensidad lumínica, distancia, aceleración, inclinación, desplazamiento, presión, fuerza, torsión, humedad, movimiento, pH, etc. Una magnitud eléctrica puede ser una resistencia eléctrica (como en una RTD), una capacidad eléctrica (como en un sensor de humedad), una tensión eléctrica (como en un termopar), una corriente eléctrica (como en un fototransistor), etc. Un sensor se diferencia de un transductor en que el sensor está siempre en contacto con la variable de instrumentación con lo que puede decirse también que es un dispositivo que aprovecha una de sus propiedades con el fin de adaptar la señal que mide para que la pueda interpretar otro dispositivo. Como por ejemplo el termómetro de mercurio que aprovecha la propiedad que posee el mercurio de dilatarse o contraerse por la acción de la temperatura. Un sensor también puede

decirse que es un dispositivo que convierte una forma de energía en otra.

Áreas de aplicación de los sensores: Industria automotriz, robótica, industria aeroespacial, medicina, industria de manufactura, etc. Los sensores pueden estar conectados a un computador para obtener ventajas como son el acceso a una base de datos, la toma de valores desde el sensor, etc.

Las características de un sensor son:

- Rango de medida: dominio en la magnitud medida en el que puede aplicarse el sensor.
- Precisión: es el error de medida máximo esperado.
- Offset o desviación de cero: valor de la variable de salida cuando la variable de entrada es nula. Si el rango de medida no llega a valores nulos de la variable de entrada, habitualmente se establece otro punto de referencia para definir el offset. (down)
- Linealidad o correlación lineal.

- Sensibilidad de un sensor: suponiendo que es de entrada a salida y la variación de la magnitud de entrada.
- Resolución: mínima variación de la magnitud de entrada que puede detectarse a la salida.
- Rapidez de respuesta: puede ser un tiempo fijo o depender de cuánto varíe la magnitud a medir. Depende de la capacidad del sistema para seguir las variaciones de la magnitud de entrada.
- Derivas: son otras magnitudes, aparte de la medida como magnitud de entrada, que influyen en la variable de salida. Por ejemplo, pueden ser condiciones ambientales, como la humedad, la temperatura u otras como el envejecimiento (oxidación, desgaste, etc.) del sensor.
- Repetitividad: error esperado al repetir varias veces la misma medida.

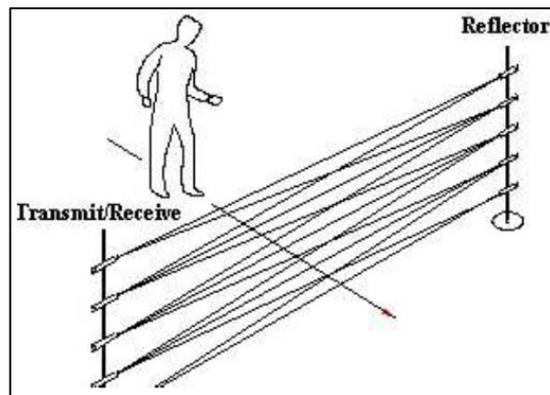
Sensor de movimiento

Un sensor de movimiento es un dispositivo electrónico equipado de sensores que responden a un movimiento físico. Se encuentran generalmente en sistemas de seguridad o en circuitos cerrados de televisión. El sistema puede estar compuesto, simplemente, por una cámara de vigilancia conectada

a un ordenador que se encarga de generar una señal de alarma o poner el sistema en estado de alerta cuando algo se mueve delante de la cámara. Aunque, para mejorar el sistema se suele utilizar más de una cámara, multiplexores y grabadores digitales. Además, se maximiza el espacio de grabación, grabando solamente cuando se detecta movimiento. Existen diferentes aplicaciones para un sensor de movimiento: seguridad, entretenimiento, iluminación, comodidad. Por ejemplo, en las tiendas se tienen sensores que detectan cuando una persona va a entrar y se abren las puertas automáticamente (Alta *et al.*, 2015, p.23). Los tipos de sensores de movimiento son:

- **Sensores activos:** este tipo de sensores inyectan luz, microondas o sonido en el medio ambiente y detectan si existe algún cambio en él, tal como se muestra en la figura 12.

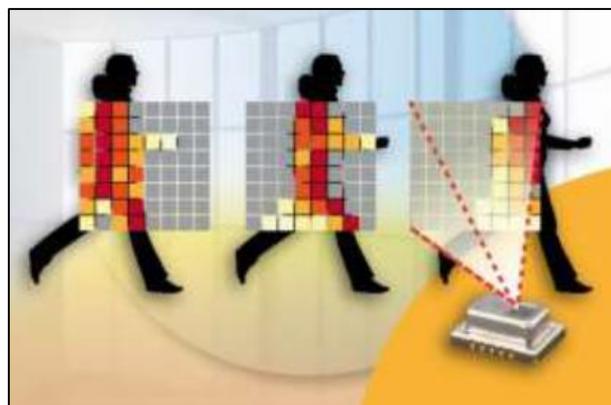
Figura 12. Sensor de movimiento activo.



Fuente: Alta *et al.*, (2015, p.23)

- Sensores pasivos: muchas alarmas y sensores utilizados usan la detección de ondas infrarrojas, como se muestra en la figura 13. Estos sensores son conocidos como PIR (pasivos infrarrojos). Para que uno de estos sensores detecte a los seres humanos se debe de ajustar la sensibilidad del sensor para que detecte la temperatura del cuerpo humano.

Figura 13. Sensor de movimiento pasivo.



Fuente: Alta *et al.*,(2015, p.23).

2.4.2.6.2 Actuadores.

Un actuador es aquel que suministra la potencia mecánica, la cual se puede convertir de fuentes eléctricas, hidráulicas o neumáticas (Cetinkunt, León, 2007, p.393). Este recibe la orden de un regulador o controlador y en función a ella genera la orden para activar un elemento final de control como, por ejemplo, una válvula. Existen varios tipos de actuadores como son:

- Electrónicos
- Hidráulicos
- Neumáticos
- Eléctricos

A continuación, se describe los actuadores utilizados en sistemas domóticos.

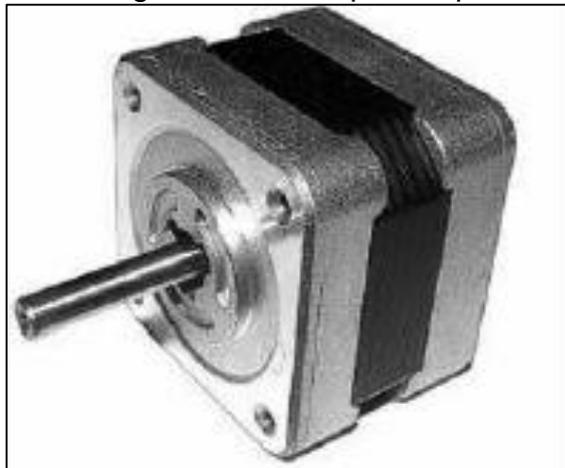
Motor paso a paso

El motor a paso es un dispositivo electromecánico que convierte una serie de impulsos eléctricos en desplazamientos angulares discretos, lo que significa que es capaz de avanzar una serie de grados (paso) dependiendo de sus entradas de control, que se observan en la figura 14.

El motor paso a paso se comporta de la misma manera que un conversor digital-analógico (D/A) y puede ser gobernado por impulsos procedentes de sistemas lógicos.

Este motor presenta las ventajas de tener precisión y repetitividad en cuanto al posicionamiento. Entre sus principales aplicaciones destacan como motor de frecuencia variable, motor de corriente continua sin escobillas, servomotores y motores controlados digitalmente.

Figura 14. Motor paso a paso.



Fuente: Cabrera et al., (2013, p.73).

Existen 3 tipos fundamentales de motores paso a paso:

- El motor de pasos de reluctancia variable (VR): tiene un rotor multipolar de hierro y un estator

devanado laminado, y rota cuando los dientes del rotor son atraídos a los dientes del estator electromagnéticamente energizados. La inercia del rotor de un motor de paso de reluctancia variable es pequeña y la respuesta es muy rápida, pero la inercia permitida de la carga es pequeña. Cuando los devanados no están energizados, el par estático de este tipo de motor es cero. Generalmente, el paso angular de este motor de paso de reluctancia variable es de 15° .

- El motor de pasos de rotor de imán permanente: permite mantener un par diferente de cero cuando el motor no está energizado. Dependiendo de la construcción del motor, es típicamente posible obtener pasos angulares de 7.5, 11.25, 15, 18, 45 o 90° . El ángulo de rotación se determina por el número de polos en el estator.
- El motor de pasos híbrido: consiste en un estator dentado y un rotor de tres partes (apilado simple). El rotor de apilado simple contiene dos piezas de polos separados por un magneto permanente magnetizado (Figura

II.15), con los dientes opuestos desplazados en una mitad de un salto de diente para permitir una alta resolución de pasos (Gallegos, Jarrín, 2012, p.35). Este tipo de motor tiene una alta precisión y alto par, se puede configurar para suministrar un paso angular tan pequeño como 1.8° .

Focos ahorradores

Los focos ahorradores son lámparas fluorescentes compactas autobalastadas que proporcionan un flujo luminoso igual al de los focos tradicionales pero con un menor consumo de energía, tal como se muestra en la figura 15. A diferencia que los focos incandescentes, los focos ahorradores funcionan por medio de un gas que ioniza y provoca la iluminación en conjunto con la pintura blanca especial que tienen las paredes interiores del tubo.

Estos focos consumen hasta un 80% menos energía, producen más luminosidad por watt y duran hasta 8 veces más que los focos tradicionales.

Figura 15. Focos ahorradores.



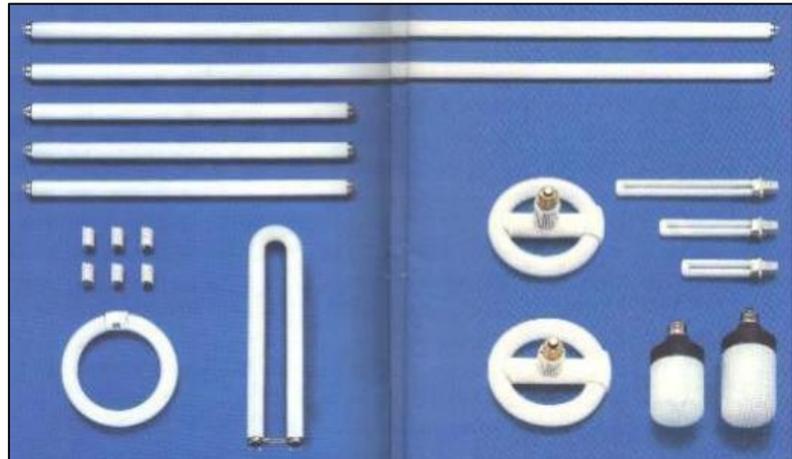
Fuente: Hernández et al. (2014, p.43).

Fluorescentes

Se conoce por luminaria fluorescente, al conjunto que forman una lámpara, denominada tubo fluorescente, y una armadura, que contiene los accesorios necesarios para el funcionamiento, tal como se ilustra en la figura 16. En ciertos lugares se conoce como luminaria solamente a la lámpara.

La lámpara es de descarga de vapor de mercurio a baja presión y se utiliza normalmente para la iluminación doméstica o industrial. Su ventaja frente a otro tipo de lámparas, como las incandescentes, es su eficiencia energética.

Figura 16. Lámpara fluorescente.



Fuente: González et al., (2013, p.60).

La lámpara consiste en un tubo de vidrio fino revestido interiormente con diversas sustancias químicas compuestas llamadas fósforos, aunque generalmente no contienen el elemento químico fósforo y no deben confundirse con él.

Esos compuestos químicos emiten luz visible al recibir una radiación ultravioleta. El tubo contiene además una pequeña cantidad de vapor de mercurio y un gas inerte, habitualmente argón o neón, a una presión más baja que la presión atmosférica. En cada extremo del tubo se encuentra un filamento hecho de tungsteno, que al calentarse al rojo contribuye a la ionización de los gases.

Las propiedades de este tipo de actuadores son:

- Luminosidad: las lámparas fluorescentes tienen un rendimiento luminoso que puede estimarse entre 50 y 90 lúmenes por vatio (lm/W). La luminosidad de la lámpara depende no solamente del revestimiento luminescente, sino de la superficie emisora, de modo que al variar la potencia varía el tamaño, por ejemplo, la de 18 W mide unos 60 cm, la de 36 W, 1,20 m y la de 54 W 1,80 m.
- Vida útil: es también mucho mayor que la de las lámparas de incandescencia, pudiendo variar con facilidad entre 5000 h y más de 75 000 h (entre 5 y 75 veces más que una bombilla), lo que depende de diversos factores, tales como el tipo de lámpara fluorescente o el equipo de la luminaria que se utilice con ella.
- Color: hay en el mercado distintos modelos con diferentes temperaturas de color. Esta está comprendida generalmente entre los 3000 K y los 6500 K (del blanco frío a luz día cálido). Sin embargo, en la actualidad se pueden conseguir tubos con una amplia gama de temperatura de color, lo que permite encontrar con relativa

facilidad modelos que van desde los 2700 K hasta los 10 000 K, recomendándose la elección en función del uso y de la iluminancia que vaya a instalarse. Las lámparas de temperatura de color alta (por ejemplo el color blanco 5000K) se recomiendan cuando se necesite una buena reproducción del color o con iluminancias altas; por el contrario, con iluminancias bajas o cuando se busquen coloraciones cálidas, se elegirá una temperatura de color baja.

Persianas motorizadas

Las persianas son elementos importantes en cualquier vivienda, esto se debe a que cumplen varias funciones: resguardarnos del sol, proteger nuestra vivienda y otorgarnos privacidad entre otras cosas; con el correr de los años estos accesorios fueron perfeccionándose y llegar a contar con lo que se conoce como “automaticidad”. Actualmente no es extraño observar más de una vivienda con persianas automáticas, éstas son más prácticas, funcionales y confortables para el usuario; que las persianas suban o bajen solas no es únicamente una cuestión de no cansarse haciendo esfuerzos físicos, sino una forma perfecta de proteger cualquier residencia.

Cuando se habla de protección se refiere tanto a los ladrones como al viento y a las lluvias; las persianas automáticas pueden centralizar su mando y controlarse mediante un control remoto o programas preestablecidos; estos últimos incluyen factores como horas de sol y oscuridad. Las persianas automáticas cuentan con un motor el cual debe elegirse de acuerdo al peso, eje y demás características de las mismas; es el motor el encargado de “hacer la fuerza” física necesaria para impulsar la persiana hacia arriba o hacia abajo; dicho motor se instala en un lado del eje sobre el cual se enrolla y desenrolla nuestro producto, tal como se muestra en la figura 17. Por otra parte, a través de cables, se podrá unir al accesorio para controlar la subida y bajada de la persiana, por ejemplo, ante fuertes vientos (Pueyo, 2012, p.71). Entre todas las ventajas que encontramos en este elemento decimos que aíslan frente los ruidos y que, según el material y el grosor pueden reducir hasta un 20% las pérdidas de calor consiguiendo así un gran ahorro en los sistemas de calefacción.

Respecto a su ubicación, por lo general se ubican en el exterior de las ventanas y por eso siempre están expuestas al sol, viento y al frío; de todas formas

pueden mantener el calor gracias al lacado o esmaltado de la lamas. Las persianas automáticas actuales se confeccionan en tres tipos de materiales: PVC, madera y aluminio; gracias a ellos se prolonga su vida útil ya que son inmunes a las deformaciones ante condiciones adversas.

Figura 17. Persianas motorizadas.



Fuente: Carranza et al., (2014, p.7).

Calefactor

Se denomina calefactores a los aparatos de funcionamiento generalmente eléctrico, cuyo mecanismo está basado en otorgar a un recipiente o contenedor, un flujo de aire caliente circulando de manera constante, generado por un radiador que motoriza la fuente calórica y un ventilador que transmite el aire caliente hacia el exterior. Los calefactores eléctricos producen energía calórica a partir de una fuente eléctrica. El modelo más

convencional de calefactor eléctrico es el de tipo “resistivo”, el cual produce calor a partir del “efecto Joule”. Este efecto se produce cuando por un conductor fluye corriente eléctrica, y debido a los choques que experimentan los átomos del material conductor con los electrones, estos transforman parte de su energía cinética en energía calórica, elevando la temperatura del conductor.

Esto se explica porque el movimiento de los electrones dentro de un cable es desordenado, provocando entre sí choques de manera continua, y esto es lo que provoca el aumento de la temperatura dentro del cable. Las aplicaciones que se le otorga a este efecto generalmente están relacionadas al funcionamiento de electrodomésticos, que, además de calefactores y estufas, incluye secadores de cabello, resistencia de planchas para indumentaria, hornallas para cocinas eléctricas, así como también para constituir las resistencias de hornos industriales, calentadores de agua, tostadores, desempañadores de vidrios para las ventanas traseras en los vehículos, alambres que se calientan evitando el efecto de acumulación de escarcha en los freezers, es decir, la tecnología “no frost”,

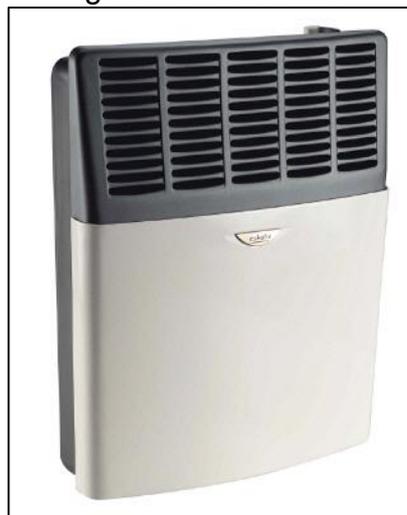
calefactores para aclimatar el agua de los acuarios y peceras, calefactores para invernaderos, así como también para infinidad de aplicaciones.

Otra clase de calefactores eléctricos funcionan a partir de un sistema que intercambia calor que se denomina “efecto Peltier”. Este es un mecanismo de funcionamiento más complejo que el anterior, que genera la diferencia de temperatura a partir del voltaje eléctrico. Se produce cuando una corriente eléctrica se hace circular por dos metales conectados a través de lo que se denomina “junturas de Peltier”, y esto produce que la corriente produzca una transferencia de calor que vaya de una junta de Peltier a la otra, provocando como efecto que una se enfríe y la otra se caliente. Esto se debe a que al traspasarse los electrones de una región de alta densidad a otra de densidad menor, estos se expanden, y de esta forma se enfría la región. Este tipo de funcionamiento es mucho menos utilizado en calefactores que el resistivo.

La resistividad resulta una propiedad desventajosa si lo que se pretende es transportar la energía, ya que

los materiales conductores no son conductores “perfectos”. La resistividad eléctrica al paso de la corriente produce que se pierda energía en la conducción, que se disipa en forma de calor. Por este motivo, resulta un método ineficiente si el propósito es que la energía se conduzca de un lado a otro, pero es un efecto deseable cuando el propósito que se persigue es la generación de calor (Heriquez, 2005, p.57). La resistividad de los conductores de material metálico tiende a aumentar con la temperatura, por lo cual para evitar que se funda el conductor el calor generado debe ser transferido. En la figura 18, se muestra la estructura de un calefactor.

Figura 18. Calefactor.



Fuente: Atahualpa, (2014, p.165).

Alarma

Se entiende por alarma la señal o aviso que advierte sobre la proximidad de un peligro, como se ilustra en la figura 19. El aviso de alarma informa a la comunidad en general o a una entidad específica (como ser los bomberos o la policía) que deben seguir ciertas instrucciones de emergencia dado que se ha presentado una amenaza.

Las alarmas que constituyen los organismos que deben responder ante una emergencia, suelen formar parte de un sistema que incluye diversos estados. El primero de ellos es el prealerta, que avisa a los responsables del organismo sobre un incidente que puede tener lugar. El segundo estado es el alerta, que exige tomar las medidas y los recaudos necesarios. Finalmente llega la alarma, que es el llamado a la acción. En las casas y los edificios, ya sean residenciales, comerciales o gubernamentales, el sistema de alarma es un elemento de seguridad pasiva (no puede evitar una situación de riesgo, pero advierte de ella y da aviso a las fuerzas de seguridad). Algunos sistemas pueden advertir de un peligro de forma automática (al detectar una intrusión por la noche a través de sensores de movimiento), mientras que otros

requieren de una acción por parte del usuario (cuando se inicia un robo y un empleado activa la alarma para llamar a la policía).

Figura 19. Alarma.



Fuente: Alvarez et al., (2007, p.143).

2.4.2.6.3 Controladores.

Un controlador o manejador de dispositivo es un programa informático que permite al sistema operativo interactuar con un periférico, haciendo una abstracción del hardware y proporcionando una interfaz (posiblemente estandarizada) para utilizar el dispositivo. Es una pieza esencial del software, sin la cual el hardware sería inutilizable.

Existen tantos tipos de controladores como tipos de periféricos, y es común encontrar más de un controlador posible para un mismo dispositivo, cada driver ofreciendo un nivel distinto de funcionalidades.

Los desarrollos recientes en la teoría de control están enfocados a sistemas de control complejos con capacidad de adaptación y aprendizaje, que consideran los procesos modelados por ecuaciones diferenciales no lineales. Al mismo tiempo se desarrolla intensamente la aplicación de sistemas de control basados en el conocimiento práctico de los operadores.

A continuación, se describen los controladores que pueden ser utilizados para proyectos domóticos:

Microcontroladores PIC

Dentro de la amplia gama de microcontroladores que se encuentran en el mercado, la familia de los PIC es la más solicitada por los diseñadores por las siguientes razones: sencillez de manejo, disponibilidad de información, precio y por su velocidad alta de funcionamiento (Dujmovic, 2006). En la figura 20 se muestra el microcontrolador 18F4520.

Figura 20. Microcontrolador PIC.



Fuente: Mendez, (2014, p.5).

El nombre actual no es un acrónimo. En realidad, el nombre completo es PICmicro, aunque generalmente se utiliza como Peripheral Interface Controller (controlador de interfaz periférico). El PIC original se diseñó para ser usado con la nueva CPU de 16 bits CP16000. Siendo en general una buena CPU, ésta tenía malas prestaciones de entrada y salida, y el PIC de 8 bits se desarrolló en 1975 para mejorar el rendimiento del sistema quitando peso de entrada/salida a la CPU. El PIC utilizaba microcódigo simple almacenado en ROM para realizar estas tareas; y aunque el término no se usaba por aquel entonces, se trata de un diseño RISC que ejecuta una instrucción cada 4 ciclos del oscilador. En 1985 la división de microelectrónica de General Instrument se separa como compañía independiente que es incorporada como filial (el 14 de diciembre de 1987 cambia el nombre a Microchip Technology y en 1989

es adquirida por un grupo de inversores) y el nuevo propietario canceló casi todos los desarrollos, que para esas fechas la mayoría estaban obsoletos.

El PIC, sin embargo, se mejoró con EPROM para conseguir un controlador de canal programable. Hoy en día multitud de PIC vienen con varios periféricos incluidos (módulos de comunicación serie, UART, núcleos de control de motores, etc.) y con memoria de programa desde 512 a 32 000 palabras (una palabra corresponde a una instrucción en lenguaje ensamblador, y puede ser de 12, 14, 16 o 32 bits, dependiendo de la familia específica de PICmicro).

La arquitectura del PIC es sumamente minimalista. Está caracterizada por las siguientes prestaciones:

- Área de código y de datos separadas (Arquitectura Harvard).
- Un reducido número de instrucciones de longitud fija.
- Un solo acumulador (W), cuyo uso (como operador de origen) es implícito (no está especificado en la instrucción).

- Todas las posiciones de la RAM funcionan como registros de origen y/o de destino de operaciones matemáticas y otras funciones.¹
- Una pila de hardware para almacenar instrucciones de regreso de funciones.
- Una relativamente pequeña cantidad de espacio de datos direccionable (típicamente, 256 bytes), extensible a través de manipulación de bancos de memoria.
- El espacio de datos está relacionado con el CPU, puertos, y los registros de los periféricos.
- El contador de programa está también relacionado dentro del espacio de datos, y es posible escribir en él (permitiendo saltos indirectos).

A diferencia de la mayoría de otros CPU, no hay distinción entre los espacios de memoria y los espacios de registros, ya que la RAM cumple ambas funciones, y esta es normalmente referida como "archivo de registros" o simplemente, registros.

Para transferir el código de un ordenador al PIC normalmente se usa un dispositivo llamado

programador. La mayoría de PIC que Microchip distribuye hoy en día incorporan ICSP (In Circuit Serial Programming, programación serie incorporada) o LVP (Low Voltage Programming, programación a bajo voltaje), lo que permite programar el PIC directamente en el circuito destino. Para la ICSP se usan los pines RB6 y RB7 (En algunos modelos pueden usarse otros pines como el GP0 y GP1 o el RA0 y RA1) como reloj y datos y el MCLR para activar el modo programación aplicando un voltaje de 13 voltios.

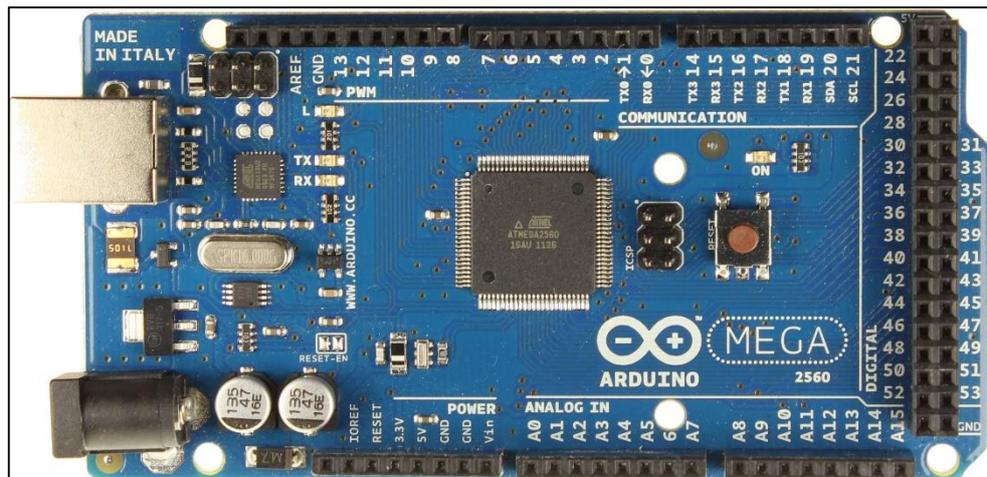
Existen muchos programadores de PIC, desde los más simples que dejan al software los detalles de comunicaciones, a los más complejos, que pueden verificar el dispositivo a diversas tensiones de alimentación e implementan en hardware casi todas las funcionalidades. Muchos de estos programadores complejos incluyen ellos mismos PIC preprogramados como interfaz para enviar las órdenes al PIC que se desea programar.

Arduino

Arduino es una plataforma de código abierto basado en prototipos de electrónica flexible y fácil de usar hardware y software (Cabrera et al, 2013, p.285). Arduino es una compañía de hardware libre, la cual desarrolla placas de desarrollo que integran un microcontrolador y un entorno de desarrollo (IDE), diseñado para facilitar el uso de la electrónica en proyectos multidisciplinarios. El hardware consiste en una placa de circuito impreso con un microcontrolador, usualmente Atmel AVR, y puertos digitales y analógicos de entrada/salida, los cuales pueden conectarse a placas de expansión (shields) que expanden las características de funcionamiento de la placa arduino, como se ilustra en la figura 21.

Por otro lado, el software consiste en un entorno de desarrollo (IDE) basado en el entorno de Processing y lenguaje de programación basado en Wiring, así como en el cargador de arranque (bootloader) que es ejecutado en la placa. El microcontrolador de la placa se programa a través de un computador, haciendo uso de comunicación serial mediante un convertidor de niveles RS-232 a TTL serial.

Figura 21. Placa Arduino.



Fuente: (Cabrera et al, 2013, p.285).

La primera placa Arduino fue introducida en el 2005, ofreciendo un bajo costo y facilidad de uso para novatos y profesionales buscando desarrollar proyectos interactivos con su entorno mediante actuadores y sensores. A partir de Octubre del año 2012, se incorporaron nuevos modelos de placas de desarrollo que hacen uso de microcontroladores CortexM3, ARM de 32 bits, que coexisten con los originales modelos que integran microcontroladores AVR de 8 bits. ARM y AVR no son plataformas compatibles a nivel binario, pero se pueden programar y compilar bajo el IDE clásico de Arduino sin ningún cambio.

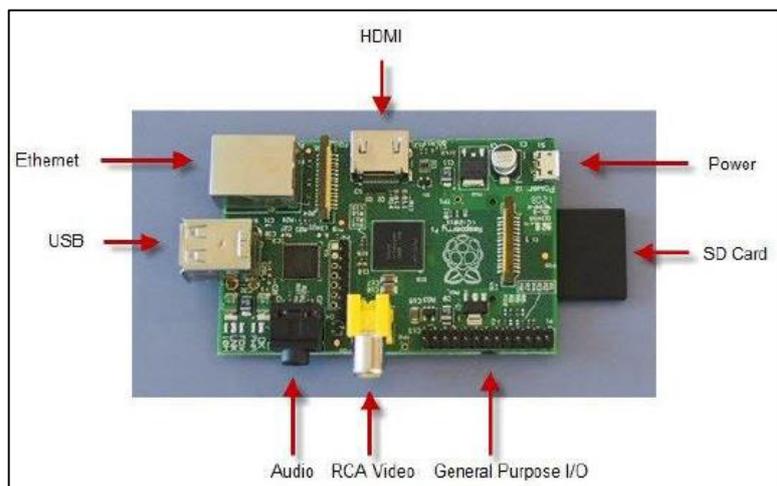
Las placas Arduino están disponibles de forma ensambladas o en forma de Kits "Hazlo tú mismo" (por sus siglas en inglés <<DIY>>). Los esquemáticos de diseño del Hardware están disponibles bajo licencia Libre, permitiendo a cualquier persona crear su propia placa arduino sin necesidad de comprar una prefabricada. Adafruit Industries estimó a mediados del año 2011 que alrededor de 300,000 placas arduinos habían sido producidas comercialmente, y en el año 2013 estimó que alrededor de 700,000 placas oficiales de la empresa Arduino estaban en manos de los usuarios.

Arduino se puede utilizar para desarrollar objetos interactivos autónomos o puede ser conectado a software tal como Adobe Flash, Processing, Max/MSP, Pure Data. Una tendencia tecnológica es utilizar Arduino como tarjeta de adquisición de datos desarrollando interfaces en software como JAVA, Visual Basic y LabVIEW. Las placas se pueden montar a mano o adquirirse. El entorno de desarrollo integrado libre se puede descargar gratuitamente.

Raspberry pi

Raspberry Pi es un ordenador de placa reducida o (placa única) (SBC) de bajo coste desarrollado en Reino Unido por la Fundación Raspberry Pi, con el objetivo de estimular la enseñanza de ciencias de la computación en las escuelas, como se ilustra en la figura 22.

Figura 22. Raspberry Pi.



Fuente: Canelo et al., (2016, p.43).

Aunque no se indica expresamente si es hardware libre o con derechos de marca, en su sección de preguntas y respuestas frecuentes (FAQs) explican que disponen de contratos de distribución y venta con dos empresas, pero al mismo tiempo cualquiera puede convertirse en revendedor o redistribuidor de las tarjetas RaspBerry Pi, por lo que se entiende que es un producto con propiedad registrada pero de uso

libre. De esa forma mantienen el control de la plataforma pero permitiendo su uso libre tanto a nivel educativo como particular.

Tampoco deja claro si es posible utilizarlo a nivel empresarial u obtener beneficios con su uso, asunto que se debe consultar con la fundación. En cambio el software sí es open source, siendo su sistema operativo oficial una versión adaptada de Debian, denominada RaspBian, aunque permite otros sistemas operativos, incluido una versión de Windows 10. El diseño incluye un System-on-a-chip Broadcom BCM2835, que contiene un procesador central (CPU) ARM1176JZF-S a 700 MHz (el firmware incluye unos modos "Turbo" para que el usuario pueda hacerle overclock de hasta 1 GHz sin perder la garantía), un procesador gráfico (GPU) VideoCore IV, y 512 MB de memoria RAM (aunque originalmente al ser lanzado eran 256 MB). El diseño no incluye un disco duro ni unidad de estado sólido, ya que usa una tarjeta SD para el almacenamiento permanente; tampoco incluye fuente de alimentación ni carcasa. El 29 de febrero de 2012 la fundación empezó a aceptar órdenes de compra del modelo B, y el 4 de febrero de 2013 del modelo A. La fundación da soporte para las

descargas de las distribuciones para arquitectura ARM, Raspbian (derivada de Debian), RISC OS 5, Arch Linux ARM (derivado de Arch Linux) y Pidora (derivado de Fedora); y promueve principalmente el aprendizaje del lenguaje de programación Python (Canelo, Vera, 2016, p.44).

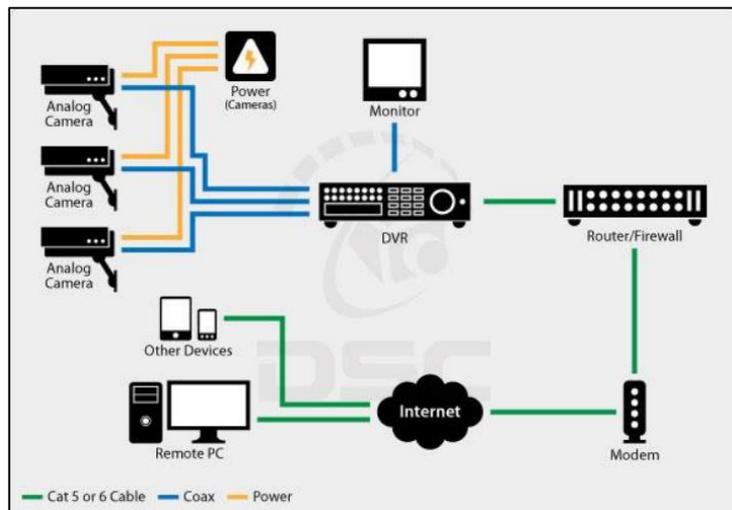
2.4.2.6.4 Video Vigilancia.

Se define video vigilancia como la supervisión presencial o remota de una determinada localidad utilizando una o más cámaras de video, puede ser para fines de seguridad, o de supervisión de personal, así como para muchas otras funciones. Los sistemas de video vigilancia actuales se clasifican en de acuerdo a la tecnología que utilizan en:

- **Sistemas analógicos:** los sistemas de vigilancia analógica o, también conocidos como circuito cerrado de televisión tienen como característica principal el uso de cámaras analógicas para la captura de señal analógica, la cual es transmitido por medio de cables coaxiales a un Grabador de Video Digital o DVR por sus siglas en inglés, tal como se muestra en la figura 23. En el DVR se realiza la conversión de la señal analógica en

digital, y se realiza la compresión del video, para posteriormente poder transmitirla al dispositivo en el cual se dará el almacenamiento. También permite transmitir en broadcast a todos los dispositivos de la red.

Figura 23. Sistema de video vigilancia analógico.



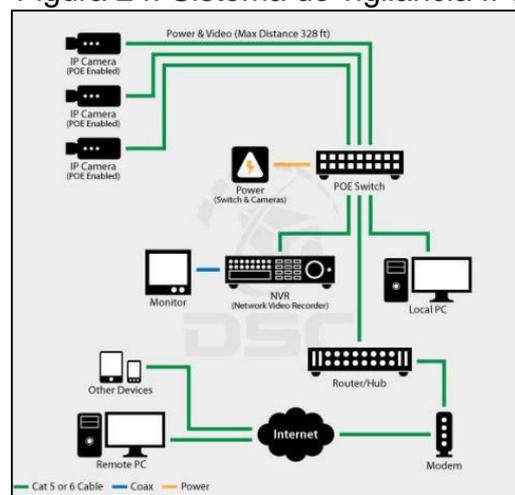
Fuente: Guerra, (2013, p.15).

- Sistema de vigilancia IP: es una tecnología de vigilancia visual que combina los beneficios analógicos de los tradicionales CCTV (Circuito Cerrado de Televisión) con las ventajas digitales de las redes de comunicación IP (Internet Protocol), tal como se muestra en la figura 24, permitiendo la supervisión local y/o remota de imágenes y audio así como el tratamiento digital de las imágenes, para aplicaciones como el reconocimiento de matrículas o reconocimiento facial, entre otras.

El despliegue resulta más sencillo y económico que un CCTV, puesto que aprovecha la red informática empresarial, es decir, el mismo cableado que se emplea para la comunicación de datos, acceso a Internet o correo electrónico, sin necesidad de desplegar una infraestructura de cableado coaxial específica para nuestra red de videovigilancia. La mayoría de las instalaciones más modernas están abandonando la tecnología analógica en favor de la videovigilancia IP, dada su versatilidad, funcionalidad, sencillez y optimización de las infraestructuras existentes en la compañía. Entre los avances más destacados de los últimos años, además de las capacidades inalámbricas que eliminan, incluso, el tendido de cables, se encuentran la alta resolución de imagen que ofrecen las cámaras megapixel (1,3 megapíxeles), la inclusión de sistemas de inteligencia para el tratamiento de video y gestión de eventos o contadores digitales. Es posible capturar vídeo y almacenarlo a pocos frames por segundo o activar la grabación sólo en determinadas circunstancias ya sea por la detección de movimientos en una zona determinada o por franjas horarias. La mejora de la resolución va acompañada de elevadas tasas

de compresión para evitar altos consumos de ancho de banda y espacio de almacenamiento, con estándares como H.264, que simplifican significativamente el almacenamiento en los NVR (Network Video Recorders) o servidores de vídeo respecto a otros formatos como vídeo Motion JPEG, MPEG-4. Estos avances tecnológicos han propiciado que consultoras como IP Video Market, en su informe 'Video Surveillance Market Size and Forecast Guide 2010', estimen que el mercado de la videovigilancia IP superará al de CCTV (analógico) entre 2010 y 2012, con un crecimiento de un 200%.

Figura 24. Sistema de vigilancia IP.



Fuente: (Guerra, 2013, p.15).

2.4.2.6.5 Medios de comunicación.

Comunicación alámbrica

Es aquella forma de comunicación eléctrica en la que se necesita un soporte físico para la transmisión de la señal eléctrica. Este soporte físico será un cable y dependiendo de las características de la comunicación el tipo de cable será distinto. Tipos de cables usados en comunicación alámbrica:

- Cable de pares o de par de par trenzado: Está formado por dos hilos de cobre recubiertos cada uno de ellos por un aislante. Los cables se trenzan uno alrededor del otro para evitar que se separen físicamente. Es el cable más simple y barato que se emplea en las comunicaciones aunque su velocidad para la transmisión de datos es inferior a la que se obtiene con otros soportes y en ocasiones producen interferencias (ruidos).
- Cable coaxial: está formado por dos conductores: uno central de cobre y de sección tubular revestido por una capa de aislante (este conductor es el que realmente transmite la señal) y otro en forma de malla que rodea al aislante del primero. Este segundo conductor es una especie de toma a tierra que evita interferencias electromagnéticas. Todo el

conjunto se aísla exteriormente por medio de un segundo aislante.

- Cable de fibra óptica: está formado por un núcleo central de plástico o vidrio por el que circula la luz, normalmente ultravioleta, gracias a las propiedades de reflexión de la luz. Este núcleo está revestido por varias capas de aislante y permite la transmisión de grandes cantidades de información a grandes distancias y a gran velocidad sin interferencias.
- Cable mixto fibra-coaxial: se emplea aprovechando instalaciones de televisión por cable y proporciona un ancho de banda importante.

Comunicación inalámbrica

La comunicación inalámbrica o sin cables es aquella en la que la comunicación (emisor/receptor) no se encuentra unida por un medio de propagación físico, sino que se utiliza la modulación de ondas electromagnéticas a través del espacio. En este sentido, los dispositivos físicos sólo están presentes en los emisores y receptores de la señal, entre los cuales encontramos: antenas, computadoras

portátiles, PDA, teléfonos móviles, etc. Entre medios de comunicación inalámbrica tenemos:

- Wifi: es un mecanismo de conexión de dispositivos electrónicos de forma inalámbrica. Los dispositivos habilitados con wifi (como una computadora personal, un televisor inteligente, una videoconsola, un teléfono inteligente o un reproductor de música) pueden conectarse a internet a través de un punto de acceso de red inalámbrica. Dicho punto de acceso tiene un alcance de unos veinte metros en interiores, distancia que es mayor al aire libre. Wi-Fi es una marca de la Wi-Fi Alliance, la organización comercial que adopta, prueba y certifica que los equipos cumplen las normas 802.11 relacionados a redes inalámbricas de área local.
- Bluetooth es una especificación industrial para Redes Inalámbricas de Área Personal (WPAN) que posibilita la transmisión de voz y datos entre diferentes dispositivos mediante un enlace por radiofrecuencia en la banda ISM de los 2,4 GHz. Los principales objetivos que se pretenden conseguir con esta norma son:
 - Facilitar las comunicaciones entre equipos móviles.

- Eliminar los cables y conectores entre éstos.
- Ofrecer la posibilidad de crear pequeñas redes inalámbricas y facilitar la sincronización de datos entre equipos personales.

Los dispositivos que con mayor frecuencia utilizan esta tecnología pertenecen a sectores de las telecomunicaciones y la informática personal, como PDA, teléfonos móviles, computadoras portátiles, ordenadores personales, impresoras o cámaras digitales.

2.4.3 Interfaces de usuario.

2.4.3.1 Definición.

La interfaz de usuario es el medio con que el usuario puede comunicarse con una máquina, equipo, computadora o dispositivo, y comprende todos los puntos de contacto entre el usuario y el equipo.

Las interfaces básicas de usuario son aquellas que incluyen elementos como menús, ventanas, contenido gráfico, cursor, los beeps y algunos otros sonidos que la computadora hace, y en general, todos aquellos canales por los cuales se permite la comunicación entre el ser humano y la computadora. La mejor

interacción humano-máquina a través de una adecuada interfaz (de usuario), que le brinde tanto comodidad, como eficiencia (Rodríguez, 2012, p. 105).

2.4.3.2 Tipos de interfaces de usuario.

Según la forma de interactuar del usuario

Atendiendo a como el usuario puede interactuar con una interfaz, existen varios tipos de interfaces de usuario:

- Interfaz de línea de comandos (Command-Line Interface, CLI): Interfaces alfanuméricas (intérpretes de comandos) que solo presentan texto.
- Interfaces gráficas de usuario (Graphic User Interface, GUI): Permiten comunicarse con la computadora de forma rápida e intuitiva representando gráficamente los elementos de control y medida.
- Interfaz natural de usuario (Natural User Interface, NUI): Pueden ser táctiles, representando gráficamente un «panel de control» en una pantalla sensible al tacto que permite interactuar con el dedo de forma similar a si se accionara un control físico; pueden funcionar mediante reconocimiento del habla, como por ejemplo Siri; o mediante movimientos corporales, como es el caso de Kinect.

Según su construcción

Pueden ser de hardware o de software:

- Interfaces de hardware: Se trata de un conjunto de controles o dispositivos que permiten que el usuario intercambie datos con la máquina, ya sea introduciéndolos (pulsadores, botones, teclas, reguladores, palancas, manivelas, perillas) o leyéndolos (pantallas, diales, medidores, marcadores, instrumentos).
- Interfaces de software: Son programas o parte de ellos, que permiten expresar las órdenes a la computadora o visualizar su respuesta.

2.4.3.3 Reconocimiento de voz.

2.4.3.3.1 Definición de voz.

La voz es un sonido emitido por un ser humano que se produce en el aparato fonador, en la laringe, cuando el aire procedente de los pulmones es forzado a través de la glotis, haciendo vibrar las cuerdas vocales, las cuales son más largas y gruesas en el hombre que en la mujer y en un niño.

2.4.3.3.2 Cualidades de voz.

La voz posee cuatro cualidades importantes:

- Tono o altura: se refiere al rango tonal o de frecuencia que le corresponde al individuo, según sexo y edad. La misma frecuencia de vibración da siempre un mismo tono, independientemente de las otras cualidades del cuerpo vibrante. Cuanto mayor es la frecuencia, más agudo es el tono y al revés. Dentro de este concepto, aparece el de Frecuencia Fundamental (F_0), que es el resultado de la vibración de los pliegues vocales. Cada individuo presenta una frecuencia fundamental propia que va descendiendo con la edad en el caso de las mujeres y en el caso de los hombres va ascendiendo.
- Intensidad: la intensidad o volumen de la voz, es la acción espiratoria de la respiración, es decir, el aire que sale desde los pulmones. La amplitud de vibración es la que da la sensación de intensidad, viene dada por la presión aérea espiratoria y puede ser también disminuida o aumentada. La intensidad es medida en decibeles (dB). A continuación se presentan los niveles de intensidad de la voz humana.

- **Timbre:** es la cantidad de armónicos que se forman al son de las frecuencias de los sonidos que se van emitiendo. Puede tener características diversas desde vivaz, estridente, monótono, pobre en armónicos, etc., esto estaría de alguna forma relacionado con el aspecto temperamental de cada persona. Para que la frecuencia de la voz sea comprensible se deberá estar entre 500 y 3500 Hz y se requiere la presencia de armónicos, de hecho se han encontrado espectros conteniendo hasta 35; por otro lado, la energía de voz está contenida en su mayoría en las bajas frecuencias.
- **Formantes:** El tracto vocal tiene cuatro o cinco resonadores llamados formantes. La frecuencia del formante es determinada por la forma del tracto vocal. Si el tracto vocal es un perfecto cilindro cerrado a nivel de la glotis y abierto al nivel de los labios y tiene una longitud de 17,5 cm, media aproximada de una laringe de hombre adulto, los primeros cuatro formantes estarán cerca de los 500, 1.500, 2.500 y 3.500 Hz. Agregando o acortando el tracto vocal, estas frecuencias básicas serán más graves o agudas; sin embargo, hay tres instrumentos para cambiar

la forma del tracto vocal. La frecuencia de un formante en particular se puede cambiar de una dirección a otra. De acuerdo con Sundberg (1977), estos instrumentos son la mandíbula, el cuerpo de la lengua y la punta de la lengua.

2.4.3.3.3 Técnicas de reconocimiento de voz.

Entre algunas técnicas se tiene:

- Comparación de patrones: el método de Plantilla o Patrones utiliza técnicas de Programación Dinámica (DTW) y básicamente consiste en comparar el patrón a reconocer (de entrada) con una serie de plantillas o patrones que representan a las unidades a reconocer.
- Modelos ocultos de Markov (HMM): El modelado estocástico de la señal de habla soluciona el problema que presentaba la técnica de alineamiento de plantillas, proporcionando los mejores resultados para el reconocimiento de habla aislada como continua y para independencia del locutor.
- Redes neuronales: las redes neuronales son estructuras de procesamiento paralelo de información, formadas por numerosos nodos

simples conectados entre sí mediante pesos y agrupados en diferentes capas, entre las que se deben distinguir la capa de entrada y la capa de salida. Debido a su naturaleza intrínsecamente no lineal, a su capacidad de clasificación, y sobre todo a la capacidad que tienen para aprender una determinada tarea a partir de pares observación-objetivo sin hacer suposición alguna sobre el modelo subyacente, se han convertido en una de las herramientas más atractivas para la solución del problema del reconocimiento de habla.

2.4.3.4 Aplicación web.

2.4.3.4.1 Definición.

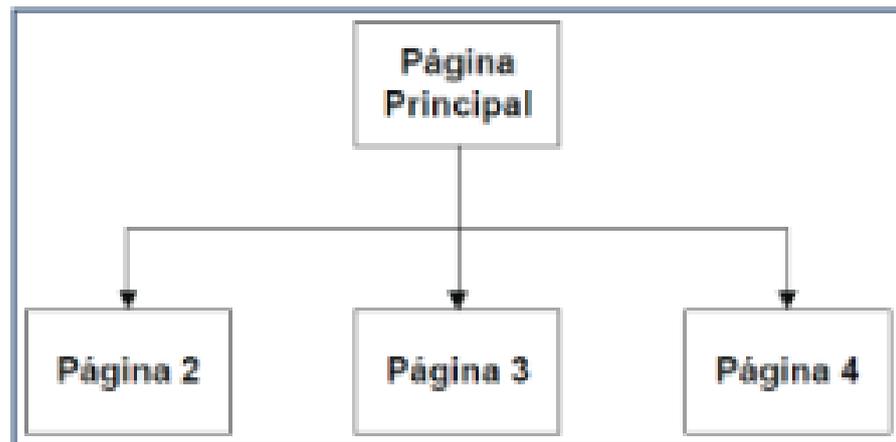
Una página web es el nombre que se da a un documento adaptado para la World Wide Web, más conocido como Internet, y que puede ser accedido por medio de un navegador integrado en un ordenador o un dispositivo móvil. Este documento se encuentra mayormente escrito en formato HTML o XHTML, y es posible proporcionar acceso a otras páginas web mediante enlaces de hipertexto (links).

Una página web puede presentar información de distintos formatos (texto, imagen, sonido, video, animación, etc.), estar asociada a datos de estilo o contar con aplicaciones interactivas. Se las puede clasificar en páginas web estáticas (contenido predeterminado), y dinámicas (generan contenido al momento de solicitar información a un servidor de web a través de lenguajes interpretados como JavaScript).

2.4.3.4.2 Página web estática.

Se la define como aquella que no requiere de actualizaciones constantes, está formada sólo por código XHTML y un estilo (CSS). Los sitios web estáticos son aquellos enfocados principalmente en mostrar una información permanente, donde el navegante se limita a obtener dicha información, sin que pueda interactuar con la página Web visitada. Las web estáticas están construidas principalmente con hipervínculos o enlaces (links) entre las páginas web que conforman el sitio, tal como se ilustra en la figura 25. Este tipo de páginas no soportan aplicaciones web como gestores de bases de datos, foros, consultas on-line y e-mails inteligentes.

Figura 25. Estructura página web estática.



Fuente: (Acosa et al., 2015, p.33)

Las características de la web estática son:

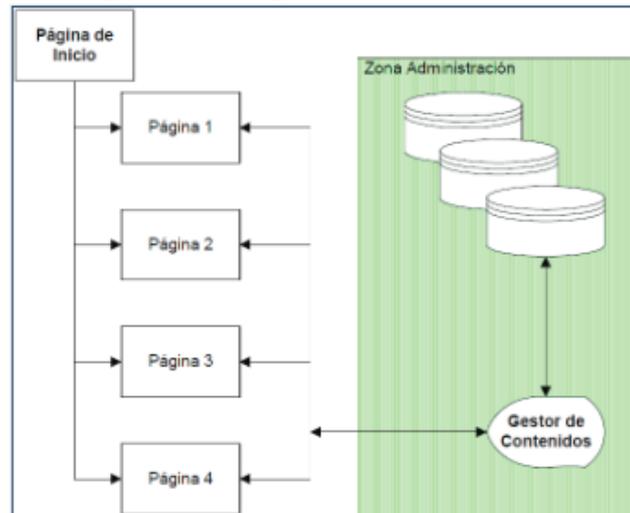
- Ausencia de movimiento y funcionalidades.
- Realizadas en XHTML o HTML.
- Para cambiar los contenidos de la página, es imprescindible acceder al servidor donde está alojada la página.
- El usuario no tiene ninguna posibilidad de seleccionar, ordenar o modificar los contenidos o el diseño de la página a su gusto.
- El proceso de actualización es lento, tedioso y esencialmente manual.
- No se pueden utilizar funcionalidades tales como bases dedatos, foros, etc.

La principal ventaja es lo económico que resulta crearlas, con un diseño vistoso e inclusión de imágenes y texto con los cuales se desee informar a los navegantes. Se la puede crear fácilmente sin necesidad de ningún tipo de programación especial (PHP, ASP, etc.). Destaca por su sencillez, rapidez, comodidad y, sobre todo, por ser la forma más económica de tener una página web profesional. Por otro lado, la gran desventaja reside en lo laborioso que resulta su actualización, así como la pérdida de potentes herramientas que ofrecen las bases de datos, como pueden ser la creación de registros históricos, pedidos on-line, etc.

2.4.3.4.3 Página web dinámica.

Los sitios web dinámicos son aquellos que permiten crear aplicaciones dentro de la propia Web, otorgando una mayor interactividad con el navegante, tal como se muestra en la figura 26. Permite desarrollar aplicaciones dinámicas como encuestas y votaciones, foros de soporte, libros de visita, envío de e-mails inteligentes, reserva de productos, pedidos on-line, atención al cliente personalizada, etc (Acosta et al, 2015).

Figura 26. Página web dinámica.



Fuente: Acosta et al., (2015, p.35).

Las características de la web dinámica son:

- El visitante puede alterar el diseño, contenidos o presentación de la página.
- Utiliza diversos lenguajes y técnicas de programación.
- Proceso de actualización sencillo, sin necesidad de entrar en el servidor.
- Gran número de funcionalidades (bases de datos, foros, contenido dinámico)
- Pueden realizarse íntegramente con software de libre distribución.
- Gran número de soluciones prediseñadas de libre disposición.

Una gran ventaja que proporciona crear sitios web dinámicos es la posibilidad de trabajar con bases de datos, las cuales pueden almacenar datos de interés para su posterior utilización en cualquier tipo de consultas. Por otro lado, el desarrollo de este tipo de sitios web es más complicado, pues requiere del conocimiento de lenguajes de programación específicos, así como de la creación y gestión de bases de datos.

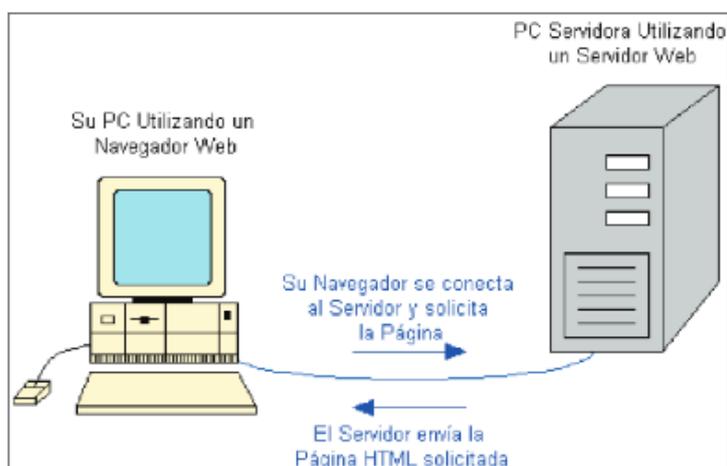
2.4.3.4.4 Servidor web.

Un servidor web es un ordenador que ejecuta un código (software de servidor web) cuya tarea es alojar documentos, sitios y/o aplicaciones, las cuales son accedidas por los clientes utilizando un navegador que se comunica con el servidor utilizando el protocolo HTTP. Básicamente un servidor web consta de un intérprete HTTP el cual se mantiene a la espera de peticiones de clientes y le responde con el contenido según sea solicitado. El cliente, una vez recibido el código, lo interpreta y lo exhibe en pantalla.

Además los servidores pueden disponer de un intérprete de otros lenguajes de programación que ejecutan código embebido dentro del código HTML de las páginas que contiene el sitio, antes de enviar el resultado al cliente. Esto se conoce como programación de lado del servidor y utiliza lenguajes como ASP, PHP, Perl y Ajax. Las ventajas de utilizar estos lenguajes radica en la potencia de los mismos ejecutando tareas más complejas como por ejemplo acceder a bases de datos, abstrayendo al cliente de toda la operación.

En la figura 27 se muestra el diagrama básico representativo de un servidor web en la configuración cliente servidor, en la cual el cliente solicita la página al servidor, y este a su vez devuelve la misma en formato HTML.

Figura 27. Diagrama representativo de un servidor web.



Fuente: (Acosta et al., 2015, p.36).

2.4.3.4.5 Lenguajes de programación.

El lenguaje en el lado del servidor se ejecuta en el servidor web antes de que se envíe la página a través de internet al cliente. Las páginas que se ejecutan en el servidor pueden acceder a bases de datos, conexiones de red, etc., para generar la página final que se mostrará al cliente. Los lenguajes de programación de lado servidor más utilizados son ASP, JSP, PERL y PHP.

ASP (Active Server Pages)

Es un lenguaje propietario de Microsoft, y es empleado por programadores para desarrollar sitios web, el cual incluye código que es interpretado en el servidor.

ASP.NET es el sucesor de la tecnología ASP y se ha desarrollado para resolver las limitantes que brindaba la misma, entre ellas generar contenido web dinámico, código de validación en el lado del cliente, acceder a bases de datos, etc.

Para el desarrollo de aplicaciones ASP.NET se puede utilizar lenguajes de programación como C#, VB.NET o J#.

JSP (Java Server Pages)

Es un lenguaje multiplataforma para la creación de sitios web dinámicos, acrónimo de Java Server Pages. JSP fue desarrollado por Sun Microsystems y comparte ventajas similares a las de ASP.NET, ha sido desarrollado para la creación de aplicaciones web potentes. Posee un motor de páginas basado en los servlets de Java, para su funcionamiento se necesita tener instalado un servidor Tomcat.

PHP (Hypertext Preprocessor)

Inicialmente 'PHP Tools' o 'Personal Home Page Tools', es un lenguaje de código abierto usado para la creación de aplicaciones para servidores, o creación de contenido dinámico para sitios web, y que puede ser incrustado en HTML. El PHP es un lenguaje de script incrustado dentro del HTML. La mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas de sí mismo. La meta del lenguaje es permitir rápidamente a los desarrolladores la generación dinámica de páginas.

CAPITULO III: DISEÑO E IMPLEMENTACION DEL SISTEMA

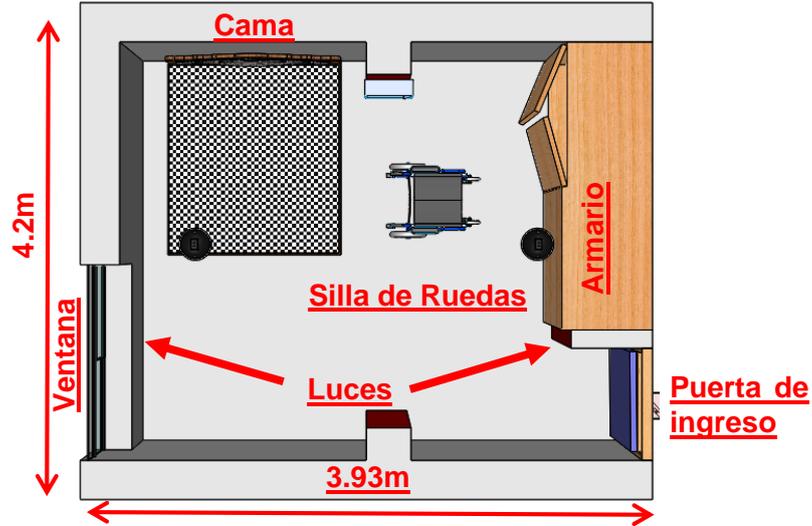
3.1 Características generales.

En este capítulo se muestra el diseño del sistema domótico para aplicaciones en usuarios parapléjicos. La finalidad de este trabajo es hacer que una persona con parapléjica pueda controlar un sistema domótico en una habitación de 16,5 m² con la finalidad de darle una mayor independencia en su vida diaria. En la figura 28 se muestra la distribución de la habitación. De acuerdo con el Informe Técnico sobre Domótica de Viviendas del Instituto de Investigación en Ingeniería de la Universidad de Zaragoza, “para contar con un sistema capaz de implementar todos los servicios necesarios habría que contar con los siguientes dispositivos y actuadores (...), entre ellos se encontrarían: motores para persianas, toldos, puertas, sirenas, rociadores, iluminación electrodomésticos” (Casas, Roy, 2005, p. 27). Por ello, el conjunto de actividades básicas que realizará la persona con paraplejia dentro de la habitación son:

- Actividad #1: Ingresar a la habitación por una puerta con mecanismo automático de apertura y cierre.
- Actividad #2: Encender las luces de la habitación.
- Actividad #3: Encender la calefacción para las noches de invierno.
- Actividad #4: Controlar una la cortina roller para iluminación en las mañanas.
- Actividad #5: Abrir la ventana de la habitación para una mejor ventilación.
- Actividad #6: Llamada de emergencia mediante alarma sonora.
- Actividad #7: Vigilancia de la habitación a través de una cámara IP.

Estas aplicaciones son fundamentales en la vida diaria de una persona con paraplejia y define el alcance a la que llegará nuestro trabajo de investigación.

Figura 28. Habitación de la persona con parapleja.

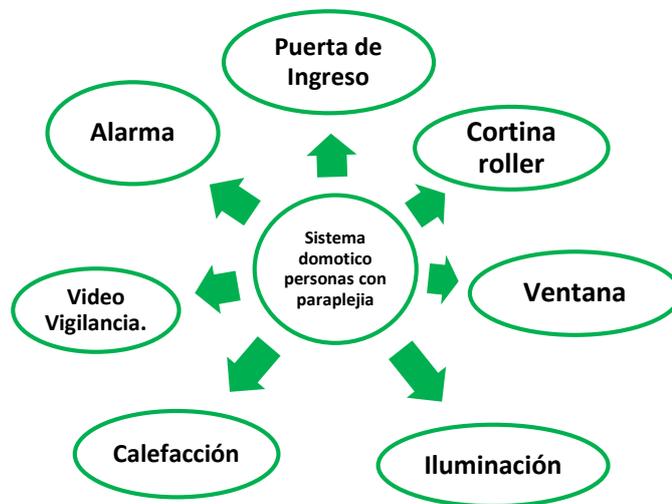


Fuente: Elaboración propia.

3.2 Diseño del Sistema Domótico.

Este tipo de sistemas es la unión de diferentes elementos que trabajan en sincronía para un alcanzar un objetivo común. El enfoque básico de la arquitectura de un sistema domótico para aplicaciones en usuarios con parapleja se visualizan en la figura 29.

Figura 29. Arquitectura del Sistema Domótico.



Fuente: (Carlos, Roy, 2005)

Con esto, en los siguientes apartados se describe el diseño y selección de cada uno de estos componentes.

3.2.1 Puerta de ingreso.

3.2.1.1 Consideraciones previas.

Los usuarios de sillas de ruedas experimentan dificultades para abrir y cerrar puertas, ya que les es difícil alcanzar y manipular la chapa, girar la llave, así como jalar y empujar las puertas. Esta dificultad para manipular las puertas limita el acceso a la casa habitación de forma autónoma y requiere la presencia y ayuda de un acompañante para entrar y salir de la propia casa. En países industrializados los usuarios de sillas de ruedas no requieren de la asistencia de otras personas para entrar y salir de su casa, ya que existen abre-puertas automáticos. Los abre-puertas automáticos aumentan la independencia, seguridad y calidad de vida de las personas con discapacidad.

Para esto, se necesita seleccionar un mecanismo de transmisión y un motor para mover la puerta que se muestra en la figura 30.

Figura 30. Puerta de ingreso.



Fuente: Elaboración propia.

Las características de la puerta de ingreso son las siguientes:

- Cuenta con dimensiones de 2,5m x 0,84m x 4cm.
- El material construido es de madera contrachapada.
- Su peso es de 55kg.
- Utiliza tres bisagras tipo libre para la abertura o cierre.

A continuación, se describe el diseño y selección de los componentes mecánicos y electrónicos para el abre-puerta automático.

3.2.1.2 Diseño del Mecanismo.

La norma internacional ANSI/BHMA 156.19 (American National Standard for Power Assist and Low Energy Power Operated

Doors) brinda requerimientos que deben ser revisados cuando se diseña abre-puertas automáticos. Esta norma indica las especificaciones de velocidad angular y fuerza para puertas giratorias de bisagra lateral de diferentes tamaños y pesos, indica el tiempo mínimo que debe permanecer la puerta completamente abierta antes de volver a cerrar, así como las variaciones permitidas de estas especificaciones, después de los ciclos de prueba.

Según esta norma, el tiempo mínimo de abertura desde 0° hasta 80° depende del ancho y peso de la puerta, y se calcula usando la ecuación 1.

$$T = \frac{D * \sqrt{W}}{2260} \dots\dots\dots(1)$$

donde:

T: tiempo de abertura o cierre (s).

D: Ancho de puerta (mm).

W: Peso de la puerta (kg)

Reemplazando:

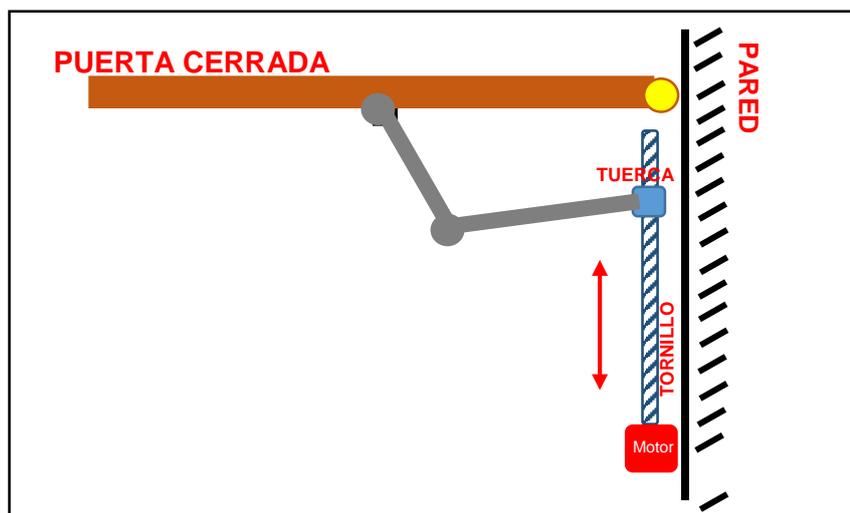
$$T = \frac{840 \text{ mm} * \sqrt{20 \text{ kg}}}{2260} = 2,76 \text{ s}$$

Se escoge un tiempo de 3,2 segundos (casi medio segundo más para tener un tiempo mayor al calculado) para la abertura o cierre de la puerta de 0° a 80°. Asimismo, la norma indica que los 10° sobrantes deben realizarse en un tiempo mínimo de 1,5 segundos.

Se selecciona un mecanismo tornillo-tuerca empotrada en la pared y conectada a la puerta mediante dos eslabones, tal como se muestra en la figura 31. Este mecanismo presenta las siguientes ventajas:

- Puede funcionar para puertas derechas o izquierdas.
- Para cualquier tipo de puerta.
- Reemplazo fácil del motor-reductor.
- Larga vida útil.

Figura 31. Mecanismo tornillo-tuerca.

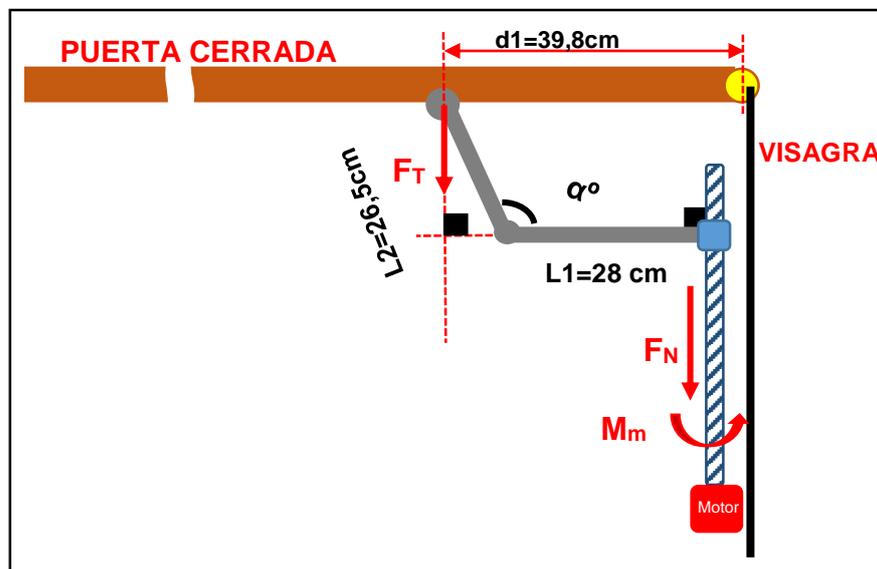


Fuente: Elaboración propia.

Calculo del Torque

El torque necesario para abrir y cerrar la puerta no se modifica dependiendo de la posición del eje del mecanismo y es igual al torque en el tornillo. La figura 32 muestra el diagrama de posición de los eslabones con respecto a la bisagra en vista superior.

Figura 32. Diagrama de posición de los eslabones de la puerta.



Fuente: Elaboración propia.

Tomando como referencia la bisagra de la puerta, el torque necesario para abrir la puerta se puede calcular con la ecuación de tornillos de potencia (Tyler, 1981, p. 3.43)

$$T = \frac{F_T d_m (1 + \pi \mu d_m \sec \beta)}{2(\pi d_m - \mu L \sec \beta)} \dots\dots\dots (2)$$

donde:

F_T : Fuerza para abrir la puerta (N).

d_m : diámetro medio del tornillo.

μ : coeficiente de fricción.

β : ángulo de la hélice del tornillo.

L: paso del tornillo.

Se realizaron mediciones de fuerza para conocer el torque necesario para abrir la puerta utilizando un dinamómetro con rango de 200 g a 20 kg y resolución de 200 gr posicionado en la esquina superior izquierda con respecto a la bisagra de la puerta. La fuerza mínima para vencer la inercia de la puerta fue de 50 N.

Por otro lado, se selecciona un tornillo de diámetro exterior de 3/8", diámetro medio de 0,32" y 1/16" de paso tal como se muestra en la figura 33. El coeficiente de fricción entre tuerca y tornillo es de 0,2 y ángulo de hélice de 30°.

Figura 33. Tornillo seleccionado.



Fuente: Elaboración propia.

De esta manera, el torque mínimo que deberá tener el motor es de:

$$T = \frac{(50 \text{ N})(8,128\text{mm})(1 + \pi(0,1)(8,128\text{mm}) \sec 30^\circ)}{2(\pi(8,128\text{mm}) - (0,1)(1.5875) \sec 30^\circ)}$$

$$= 24 \text{ N.mm} \cong 0,25 \text{ kgf - cm}$$

Aplicando un factor de seguridad del 75% (multiplicando 1,75 al resultado anterior), para sobredimensionar nuestra selección, se tiene que el torque necesario del motor es:

$$T = 0,25 \times 1.75 = 0,4375 \text{ kgf - cm}$$

Se escoge un motor-reductor de 2 kgf-cm porque se puede controlar su movimiento y velocidad a través de pulsos PWM. Además, es ideal para trabajar en lazo abierto y su mantenimiento es reducido porque no posee escobillas. El motor seleccionado es de la marca japonesa TSUKASA tal como se muestra en la figura 34 y su ficha técnica en la tabla 1.

Tabla 1. Características del motor-reductor TSUKASA.

Modelo	TG-35F-AG-18-A275
Torque	2 kgf-cm
Alimentación	24VDC
Corriente	0,5A
Velocidad	450 RPM
Peso	95gr

Fuente: www.tsukasa.com

Figura 34. Motor-reductor marca TSUKASA.

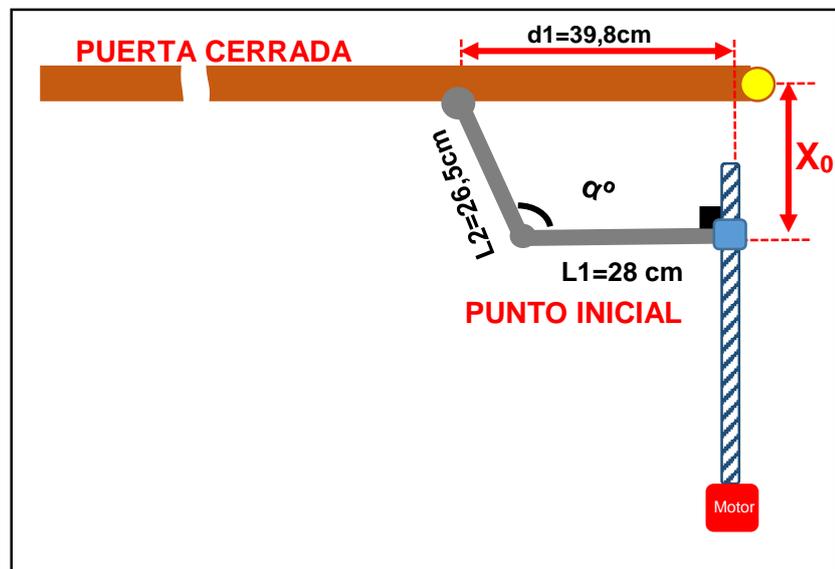


Fuente: Elaboración propia.

Calculo de la Longitud Total del Tornillo

Se debe calcular la longitud total del tornillo y determinar el punto inicial y final de carrera. En la figura 35 se muestra el mecanismo en su posición inicial (puerta cerrada).

Figura 35. Mecanismo de dos barras en posición inicial.



Fuente: Elaboración propia.

Tomando como referencia la bisagra de la puerta y considerando que el eslabón 2 forma un ángulo de 90° con el

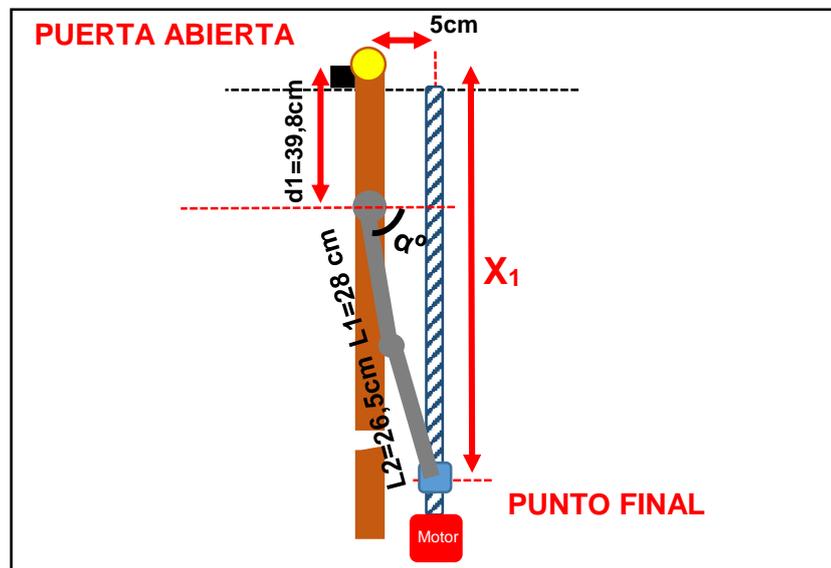
tornillo, entonces el punto inicial X_0 puede ser calculado utilizando el teorema de Pitágoras:

$$X_0 = \sqrt{L_2^2 - (d_1 - L_1)^2} = \sqrt{26,5^2 - (39,8 - 28)^2}$$

$$X_0 = 23,7 \text{ cm}$$

En la figura 36 se muestra la posición del mecanismo cuando la puerta está abierta.

Figura 36. Mecanismo de dos barras en posición final.



Fuente: Elaboración propia.

Tomando como referencia la bisagra de la puerta y considerando que el eslabón 1 y 2 forma un ángulo de 180° , es decir, se encuentran colineales entonces el punto final X_1 puede ser calculado utilizando el teorema de Pitágoras:

$$X_1 = d_1 + \sqrt{(L_1 + L_2)^2 - 5^2}$$

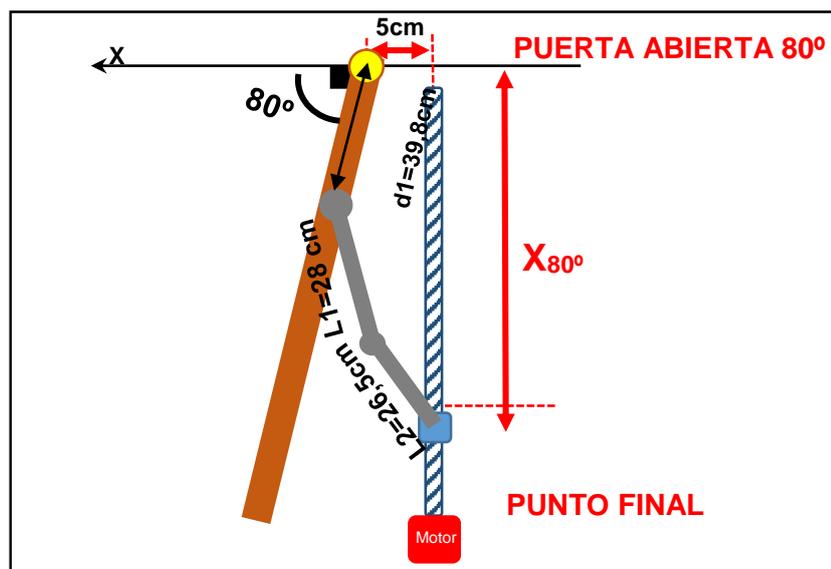
$$X_1 = 39,8 + \sqrt{(28 + 26,5)^2 - 5^2} \approx 94 \text{ cm}$$

Por lo tanto, se debe seleccionar una longitud de tornillo mayor de 94 cm. Para esta tesis, se selecciona un tornillo de longitud de 100 cm.

Cálculo de la velocidad de giro

Como se mencionó anteriormente, la puerta debe hacer un recorrido de 80° en un tiempo de 3,2 segundos y 10° en un tiempo de 1,5 segundos según norma. La figura 37 muestra el mecanismo en 80° de apertura.

Figura 37. Mecanismo de barras cuando está a 80° del eje X.



Fuente: Elaboración propia.

Al realizar un análisis de posición analíticamente, la distancia X_{80° se puede determinar como sigue:

$$X_{80^\circ} + d_1 \sin 80^\circ = \sqrt{(L_1 + L_2)^2 - (\sin(90^\circ - 80^\circ) + 5)^2}$$

$$X_{80^\circ} + (39,8) \sin 80^\circ = \sqrt{(28,5 + 26)^2 - (\sin(90^\circ - 80^\circ) + 5)^2}$$

$$X_{80^\circ} = 92,38 \text{ cm}$$

Entonces, el motor paso a paso debe hacer un recorrido desde 23,70 cm hasta 92,38 cm con respecto al eje X en un tiempo de 3,2 segundos y desde 92,39 cm hasta 94 cm con respecto al eje X en un tiempo de 1,5 segundos. Las velocidades lineales se determinan mediante la ecuación 3 (Tyler, 1981, p. 3.43):

$$V = \frac{x_f - x_i}{\Delta t} \dots\dots\dots (3)$$

donde:

V: velocidad lineal.

X_f: posición final.

X_i: posición inicial.

Δt: diferencial de tiempo.

Para el primer recorrido, el moto-reductor debe tener una velocidad lineal de:

$$V_1 = \frac{92,38\text{cm} - 23,70\text{cm}}{3,2\text{s}} = 21,5 \frac{\text{cm}}{\text{s}} \cong 0,215 \text{ m/s}$$

Para el segundo recorrido, el moto-reductor debe tener una velocidad lineal de:

$$V_2 = \frac{94\text{cm} - 92,38\text{cm}}{1,5\text{s}} = 1,08 \frac{\text{cm}}{\text{s}} \cong 0,108 \text{ m/s}$$

Ahora bien, se debe convertir la velocidad lineal en angular utilizando la ecuación 4 (Tyler, 1981, p. 3.43):

$$V = \omega * r \dots\dots\dots (4)$$

donde:

V: velocidad lineal.

ω : velocidad angular.

r: radio del tornillo.

Para el primer recorrido, el motor-reductor debe tener una velocidad angular de:

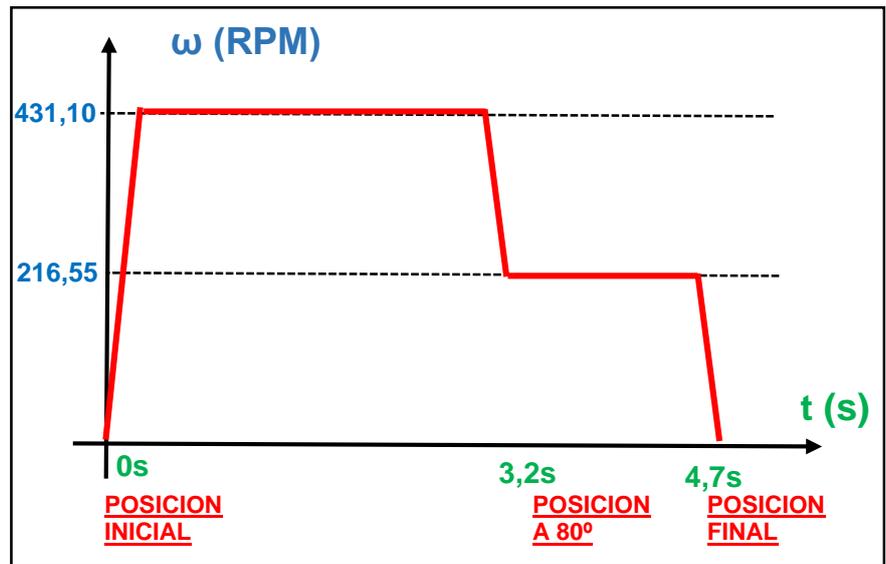
$$\omega_1 = \frac{0,215m/s}{4,7625mm} = \frac{45,14rad}{s} \cong 431,10 \text{ RPM}$$

Para el segundo recorrido, el motor paso a paso debe realizar el siguiente número de pasos:

$$\omega_2 = \frac{0,108m/s}{4,7625mm} = \frac{22,68rad}{s} \cong 216,55 \text{ RPM}$$

Estos son los datos necesarios para la programación de pulsos PWM en la tarjeta electrónica. En la figura 38 se muestra el diagrama de velocidad que deberá tener el moto-reductor para realizar la secuencia descrita anteriormente.

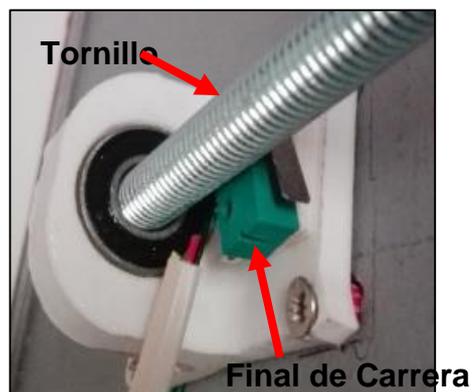
Figura 38. Velocidad del motor-reductor de la puerta según norma.



Fuente: Elaboración propia.

Para la detección de la posición inicial, a 80° y final se utiliza finales de carrera normalmente abiertos como se muestra en la figura 39.

Figura 39. Final de Carrera en Posición Inicial.



Fuente: Elaboración propia.

3.2.1.3 Diseño Electrónica de Potencia.

La variable a tener en consideración es la corriente máxima del motor-reductor que es de 0.5A según las especificaciones técnicas mostradas en la tabla 1. Para ello se utiliza transistores BJT con polarización NPN que funcionen como interruptores (región de saturación) para formar un puente H. Seleccionando un encapsulado TO-220, porque no requiere disipadores de calor, y el transistor de mediana potencia TIP122 con una potencia de disipación de 2W. Posteriormente, se calcula el voltaje colector-emisor mediante la ecuación 5:

$$V_{c-e} = P/I_c \dots\dots\dots (5)$$

donde:

V_{c-e} : Voltaje Colector-Emisor.

P: Potencia de disipación.

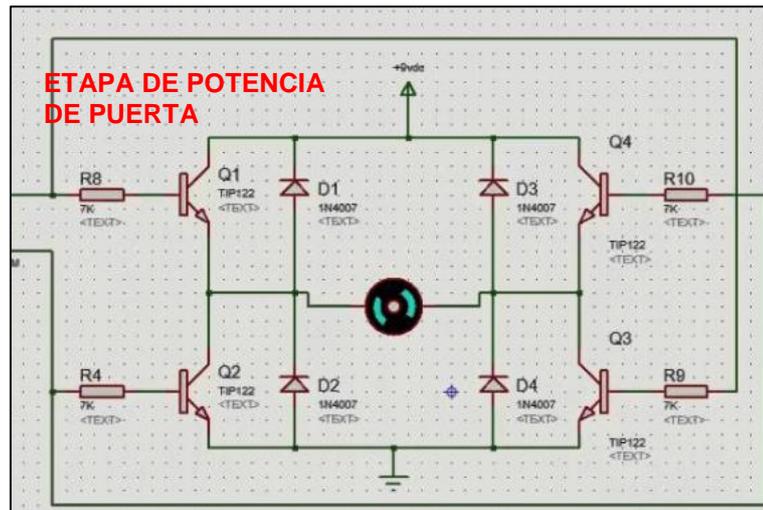
I_c : Máxima corriente en colector.

Despejando:

$$V_{c-e} = \frac{2}{0.5} = 4v$$

Como se muestra, se debe asegurar que el voltaje en la base del transistor no supere los 4 voltios para evitar que no se quemé. En la figura 40, se muestra el diseño final del puente H.

Figura 40. Puente H para motor-reductor de la puerta.



Fuente: Elaboración propia.

3.2.2 Cortina automática.

3.2.2.1 Consideraciones previas.

Los usuarios de sillas de ruedas experimentan dificultades para abrir y cerrar las cortinas. Esta dificultad para manipular las cortinas impide quitar total o parcialmente la luz en el momento del sueño y requiere la presencia y ayuda de un acompañante. Vale mencionar que, las cortinas sirven para no permitir el acceso del frío, porque al momento de frías corrientes o lluvias, el vidrio ingresa a la habitación bajando la temperatura.

Para esto, se necesita seleccionar un motor-reductor que pueda enrollar y desenrollar la cortina roller que se muestra en la figura 41.

Figura 41. Cortina roller.



Fuente: Elaboración propia.

Las características de la cortina roller son las siguientes:

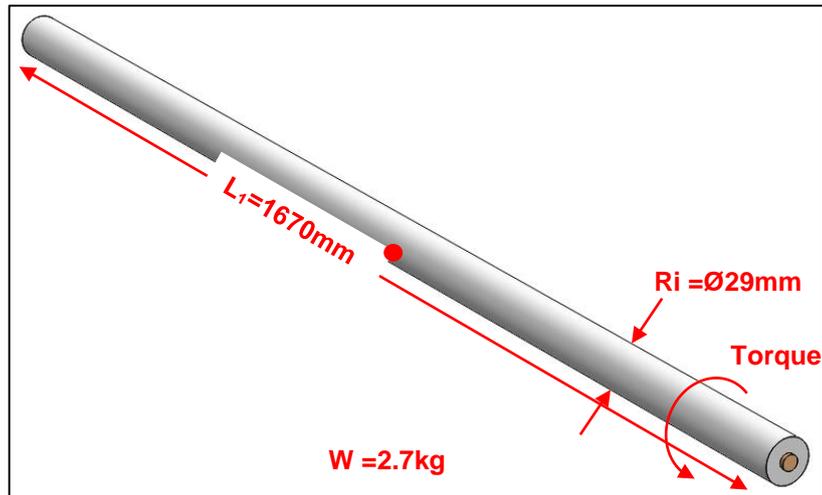
- El material es de tela con 0.5 mm de espesor.
- Su peso es de 2.7 kg.
- Sus dimensiones al ser enrollada son de $\text{Ø}58 \times 1670$ mm.
- Se empotra a la pared mediante tornillos con tarugos.
- Su longitud máxima es de 1720 mm.
- Sus dimensiones al ser desenrollada son de $\text{Ø}38 \times 1670$ mm.

A continuación, se describe el diseño y selección de los componentes mecánicos y electrónicos para la cortina automática.

3.2.2.2 Selección del Motor.

La cortina roller enrollada es decir, en sus condiciones iniciales, se muestra en la figura 42.

Figura 42. Cortina roller enrollada.



Fuente: Elaboración propia.

El torque mínimo para hacer girar la cortina y vencer el momento de inercia se puede calcular con la ecuación 6 (Tyler, 1981, p. 3.43):

$$\text{Torque} = F * r \quad \dots\dots\dots(6)$$

$$\text{Torque} = \left(2,7\text{kg} * 9,81 \frac{\text{m}}{\text{s}^2}\right) * \frac{0,058\text{m}}{2}$$

$$\text{Torque} = 0,77 \text{ N.m}$$

Para un factor de seguridad de 2, para sobredimensionar la selección, se necesita un motor con torque mayor a 1.54 N.m. Buscando en el mercado local, se encontró el motor-reductor

de la marca SWF VALEO de 2 N.m a 24 VDC tal como se muestra en la figura 43 y su ficha técnicas en la tabla 2.

Figura 43. Motor-reductor SWF VALEO.



Fuente: Elaboración propia.

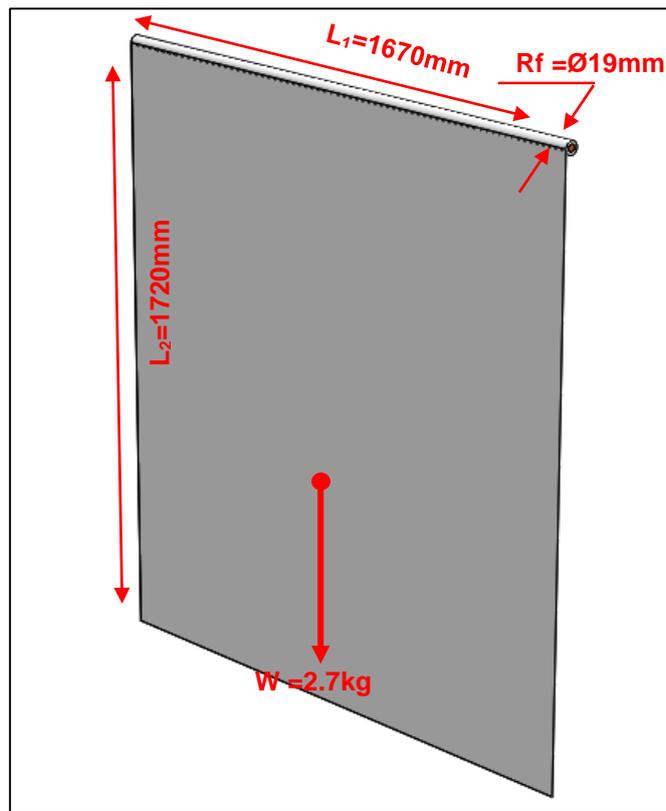
Tabla 2. Características técnicas del motor-reductor SWF VALEO.

Modelo	SWF404304
Torque	2 N.m
Alimentación	24VDC
Velocidad	115 RPM
Corriente	3 ^a

Fuente: www.smolka-berlin.de

Ahora bien, la cortina roller desenrollada, es decir, en sus condiciones finales de movimiento, se muestra en la figura 44.

Figura 44. Cortina roller desenrollada.



Fuente: Elaboración propia.

Considerando que la cortina se va a desenrollar hasta su longitud máxima (1720 mm), el tiempo de desenrollado se calcula utilizando la ecuación 7 (Tyler, 1981, p. 3.43):

$$t = \frac{2 * d}{V_f + V_i} \dots\dots\dots (7)$$

donde:

t: tiempo de desenrollado.

V_f: velocidad tangencial final.

V_i: velocidad tangencial inicial.

D: distancia de enrollado (1720 mm)

La velocidad tangencial inicial se calcula en el instante donde la cortina se encuentra en sus condiciones iniciales, es decir, cuando el diámetro es mayor. Podemos calcularlo utilizando la ecuación 8 (Tyler, 1981).

$$V_i = \omega * R_i \dots\dots\dots(8)$$

donde:

ω : velocidad angular inicial (115 RPM).

R_i : radio inicial de la cortina (29mm).

V_i : velocidad tangencial inicial.

Despejando en la ecuación 8, se tiene que:

$$V_i = 115RPM * 29mm = 349,24 \frac{mm}{s}$$

La velocidad tangencial final se calcula en el instante donde la cortina se encuentra en sus condiciones finales, es decir, cuando el diámetro es menor. Utilizando la ecuación 9 (Tyler, 1981, p. 3.43):

$$V_f = \omega * R_f \dots\dots\dots(9)$$

donde:

ω : velocidad angular inicial (115 RPM).

R_f : radio final de la cortina (19mm).

V_f : velocidad tangencial final.

Despejando en la ecuación 9, se tiene que:

$$V_f = 115RPM * 19mm = 228,81 \frac{mm}{s}$$

Por último, según la ecuación 7, el tiempo de desenrollado de la cortina es de:

$$t = \frac{2 * 1720mm}{228,81mm/s + 349,24mm/s} = 5,95 s$$

El tiempo de desenrollado calculado es de aproximadamente seis segundos. Vale recalcar que, para detectar la posición inicial de enrollado y final de desenrollado para detener el motor-reductor, se selecciona un final de carrera tal como se muestra en la figura 45.

Figura 45. Finales de carrera de cortinas.



Fuente: Elaboración propia.

3.2.2.3 Diseño Electrónica de Potencia.

La variable a tener en consideración es la corriente máxima del motor-reductor que es de 3 A según las especificaciones

técnicas mostradas en la tabla 2. Para ello se utiliza transistores BJT con polarización NPN que funcionen como interruptores (región de saturación) para formar un puente H. Seleccionando un encapsulado TO-220, porque no requiere disipadores de calor, y el transistor de mediana potencia TIP122 con una potencia de disipación de 2 W. Posteriormente, se calcula el voltaje colector-emisor mediante la ecuación 10:

$$V_{c-e} = P/I_c \quad \dots\dots\dots (10)$$

donde:

V_{c-e} : Voltaje Colector-Emisor.

P: Potencia de disipación.

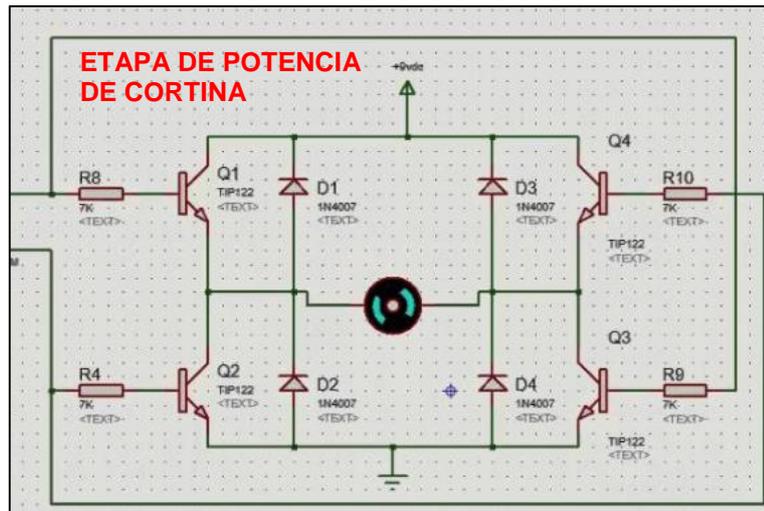
I_c : Máxima corriente en colector.

Despejando:

$$V_{c-e} = \frac{2}{3} = 0,6 \text{ v}$$

Como se muestra, se debe asegurar que el voltaje en la base del transistor no supere los 0.6 voltios para evitar que no se queme. En la figura 46, se muestra el diseño final del puente H.

Figura 46. Puente H para motor-reductor de la cortina.



Fuente: Elaboración propia.

3.2.3 Ventanas.

3.2.3.1 Consideraciones previas.

Los usuarios de sillas de ruedas experimentan dificultades para la apertura y el cierre de sus ventanas sobre todo cuando son altas.

Para esto, se necesita diseñar un mecanismo que pueda abrir y cerrar la ventana que se muestra en la figura 47.

Figura 47. Ventana de la habitación.



Fuente: Elaboración propia.

La ventana cuenta con una parte fija y otra móvil. Esta última realiza un movimiento de traslación en dirección horizontal sobre las guías inferior y superior. Sus características técnicas son las siguientes:

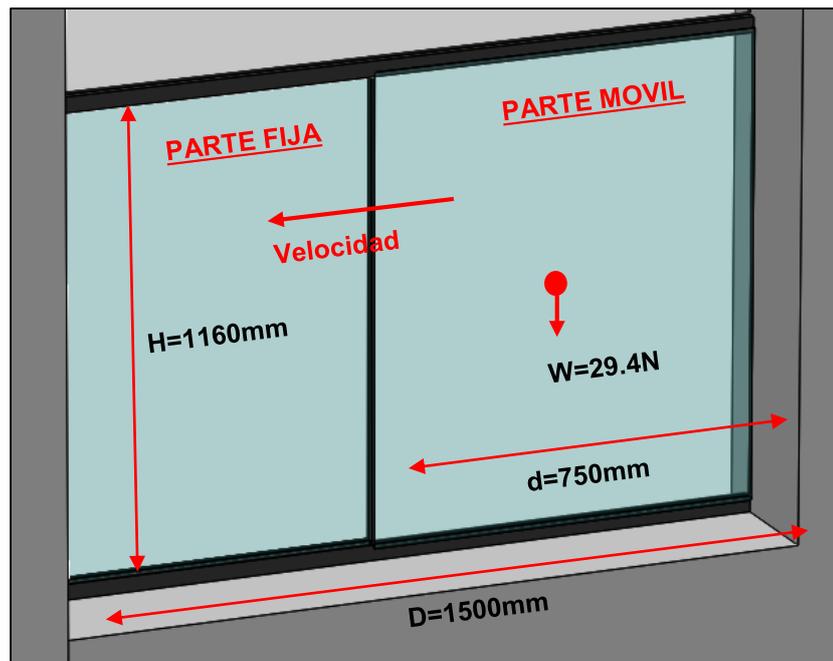
- El material es de cristal con marco de aluminio.
- Sus dimensiones totales son de 1.16x1.5x0.008 m.
- Dimensiones de parte fija y móvil: 1.16x0.75x0.008 m.
- Peso de la parte móvil: 3 kg.

A continuación, se describe el diseño y selección de los componentes mecánicos y electrónicos para la ventana automática.

3.2.3.2 Selección del motor-reductor.

La ventana cerrada es decir, en sus condiciones iniciales, se muestra en la figura 48.

Figura 48. Ventana cerrada.



Fuente: Elaboración propia.

Considerando un coeficiente de fricción de 0.35 por la superficie de contacto aluminio-aluminio de la corredera, la fuerza mínima para mover la parte móvil de la ventana se calcula mediante la ecuación 11 (Tyler, 1981, p. 3.43).

$$F = \mu * W \quad \dots\dots\dots(11)$$

$$F = 0.35 * 29.4\text{N} = 10.29\text{N}$$

Con una ventosa de succión que sirve para mover materiales con superficies lisas como vidrios una correa dentada de marca, que se extiende a todo el ancho de la ventana y dos poleas de 30 mm de diámetro se realiza el movimiento de la parte móvil de la ventana.

El torque mínimo que deberá tener el motor para mover la ventana se calcula mediante la ecuación 12 (Tyler, 1981, p. 3.43).

$$T = F * r \dots\dots\dots(12)$$

$$T = 10.29N * 15mm = 0.1544 N$$

Con esto, se escoge un motor paso a paso porque se puede controlar su movimiento y velocidad a través de pulsos. Además, es ideal para trabajar en lazo abierto y su mantenimiento es reducido porque no posee escobillas.

El motor seleccionado es de la marca MINEBEA-MATSUSHITA con un torque igual a 0.38 N.m (mayor al calculado). Su estructura se muestra en la figura 49 y su ficha técnica en la tabla 3.

Tabla 3. Características del motor paso a paso MINEBEA-MATSUSHITA.

Modelo	17PM-K142B (Bipolar)
Torque	0.38 N.m
Alimentación	12VDC
Corriente	1A
Paso	1,8°
Velocidad	8 RPM
Peso	0,3kg

Fuente: www.m-motor.com

Figura 49. Motor paso a paso MINEBEA-MATSUSHITA.



Fuente: Elaboración propia.

Conociendo que la velocidad angular del motor paso a paso que es de 8 RPM, entonces la velocidad lineal a la que se

mueve la ventana puede ser calculada utilizando la ecuación 8:

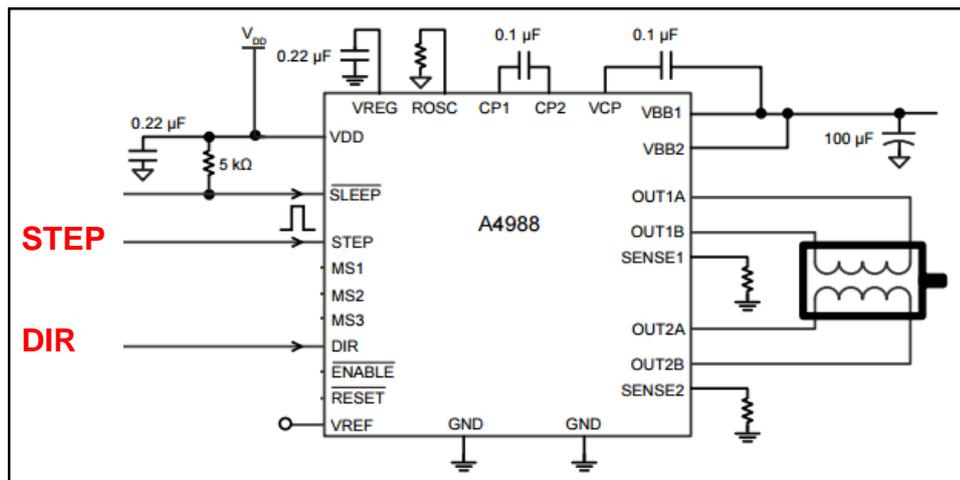
$$V_i = \omega * R_i = 8 \text{ RPM} \times 15 \text{ mm}$$

$$V_i = 12,56 \text{ mm/s}$$

3.2.3.3 Diseño Electrónica de Potencia.

Una vez seleccionado el motor, es necesario diseñar una etapa que pueda brindar la potencia necesaria para el movimiento del mismo. Además, esta etapa debe recibir las señales de control del sistema de reconocimiento de voz y el sistema de control por internet. La variable a tener en consideración es la corriente máxima del motor-reductor que es de 1 A según las especificaciones técnicas mostradas en la tabla 3. Para ello se selecciona un margen de seguridad del 100% y luego normalizando con lo encontrado en el mercado se selecciona el driver A499 que es un circuito que te permite controlar un motor paso a paso bipolar de hasta 2 A y 35 V. La figura 50 muestra la tarjeta de potencia.

Figura 50. Driver A4989.



Fuente: Elaboración propia.

3.2.4 Iluminación.

3.2.4.1 Consideraciones previas.

La iluminación es la acción o efecto de iluminar. En la técnica se refiere al conjunto de dispositivos que se instalan para producir ciertos efectos luminosos, tanto como prácticos como decorativos. Con la iluminación se pretende, en primer lugar conseguir un nivel de iluminación, o iluminancia, adecuado al uso que se quiere dar al espacio iluminado. Las consideraciones para seleccionar el sistema de iluminación para la habitación son:

- Satisfacer las necesidades visuales del usuario con paralejía.
- Crear un ambiente confortable en la habitación.

- Hacer uso racional de la energía para contribuir a minimizar el impacto ecológico y ambiental.

3.2.4.2 Selección.

Cumpliendo con las consideraciones previas se selecciona un foco ahorrador de 18 W para un área de 16 m², marca Phelix GSX-18 amigable al medio ambiente, de luz blanca y tiempo de duración de 4000 horas. Este foco espiral ahorra hasta un 50% porque usan solo una cuarta parte de la energía y duran diez veces más que los focos incandescentes. El foco se muestra en la figura 51 y su ficha técnica en la tabla 4.

Tabla 4. Características del foco ahorrador marca Phelix.

Modelo	GSX-18
Potencia	18W
Voltaje	220VAC
Corriente	82mA
Duración	4000 horas
Frecuencia	50/60 Hz

Fuente: www.feenilux.com

Figura 51. Foco ahorrador marca Phelix.



Fuente: Elaboración propia.

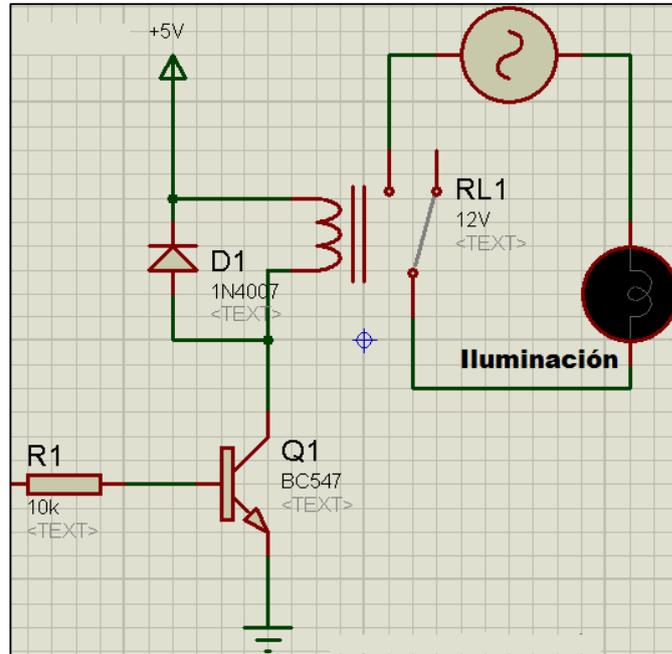
3.2.4.3 Diseño Electrónica de Potencia.

Una vez seleccionado el foco ahorrado, es necesario diseñar una etapa que pueda brindar la potencia necesaria para el encendido o apagado del mismo. Además, esta etapa debe recibir las señales de control del sistema de reconocimiento de voz y el sistema de control por internet.

La variable a tener en consideración es la corriente máxima que es de 81 mA según las especificaciones técnicas mostradas en la tabla 4. Para ello se selecciona un módulo de relés que se encuentra en el mercado local cuyas bobinas se polarizan con 5 VDC y soportan 10 A a través de sus contactos. El módulo de

relés se muestra en la figura 52. Para este caso, sólo se utiliza un relé dejando a los demás para otras aplicaciones.

Figura 52. Circuito de potencia para iluminación.



Fuente: Elaboración propia.

3.2.5 Calefacción.

3.2.5.1 Consideraciones previas.

Con el fin de crear un ambiente más confortable en la habitación del usuario con paroplejía durante los meses de invierno, se necesita seleccionar un calefactor eléctrico con las siguientes consideraciones:

- No debe necesitar instalación extra.
- No debe necesitar mantenimiento.
- No debe necesitar acopio de combustible.

- Debe ser cómodo y seguro.

3.2.5.2 Selección.

Las estufas son la solución más sencilla para mejorar la temperatura en ambientes pequeños, como es el caso de la habitación de 17 m² de la presente tesis, porque no requieren de ningún tipo de instalación, basta con que se enchufe a la corriente eléctrica para que empiece a expulsar aire caliente. Además, las resistencias van sumergidas en un aceite que se calienta de manera uniforme en cada celda. Una de sus desventajas es que demora un tiempo para calentar el aceite sin embargo, una vez que alcance la temperatura configurada los periodos de encendido y apagado con más largos. Ahora bien, la potencia adecuada de la estufa puede calcularse utilizando la ecuación 13 (Tyler, 1981, p. 3.471):

$$P_{estufa} > A * 80 \quad \dots\dots\dots (13)$$

donde:

P_{estufa} : potencia de la estufa (W).

A: Área de la habitación (m²).

$$P_{estufa} > 17 * 80$$

$$P_{estufa} > 1360 W$$

Por ello, se selecciona una estufa de 2000 W marca Imaco, con tres niveles de temperatura y 11 celdas cuyas características se muestran en la tabla 5 y su estructura en la figura 53.

Tabla 5. Características de la estufa marca Imaco.

Modelo	OFR11AO
Potencia	2000W
Voltaje	220VAC
Corriente	9A
Celdas	11
Frecuencia	50/60 Hz

Fuente: www.imacosa.com

Figura 53. Estufa marca Imaco.



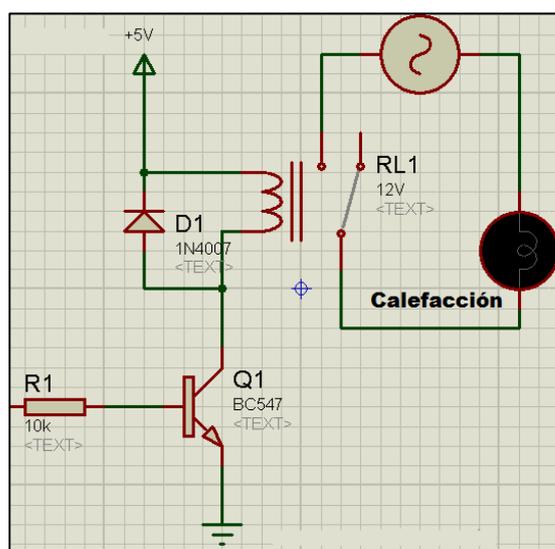
Fuente: Elaboración propia.

3.2.5.3 Diseño Electrónica de Potencia.

Una vez seleccionado la estufa, es necesario diseñar una etapa que pueda brindar la potencia necesaria para el encendido o apagado del mismo. Además, esta etapa debe recibir las señales de control del sistema de reconocimiento de voz y el sistema de control por internet.

La variable a tener en consideración es la corriente máxima que es de 9 A según las especificaciones técnicas mostradas en la tabla 5. Considerando un margen de seguridad del 10% y buscando en el mercado local se selecciona un módulo de relés cuyas bobinas se polarizan con 5 VDC y soportan 10 A a través de sus contactos. El módulo de relés se muestra en la figura 54. Para este caso, sólo se utiliza un relé dejando a los demás para otras aplicaciones.

Figura 54. Circuito de potencia para calefacción.



Fuente: Elaboración propia.

3.2.6 Alarma.

3.2.6.1 Consideraciones previas.

La integración de los diferentes elementos en la domótica tiene como objetivo no solo aumentar el confort de los usuarios, sino también trabajar para que la habitación sea más segura. Por ello, consideramos añadir un sistema de alarma tipo sonora para que pueda ser activada por el usuario parapléjico en alguna emergencia.

3.2.6.2 Selección.

Se requiere un mecanismo de aviso sonoro que se active cuando exista una emergencia. Se selecciona un transductor electroacústico que produce un sonido o zumbido continuo de un mismo tono generalmente agudo, sus características técnicas se muestran en la tabla 6 y su estructura en la figura 55. Su construcción consta de dos elementos: un electroimán y una lámina metálica de acero. Cuando una corriente pasa por la bobina del electroimán se produce un campo magnético variable que hace vibrar la lámina de acero sobre la armadura.

Tabla 6. Características de la alarma sonora.

Modelo	BZZ04
Voltaje	9VDC
Tipo	Generador Piezoeléctrico
Intensidad acústica	85dB a 3m
Corriente	0,2A

Fuente: www.lecomperu.com

Figura 55. Transductor electroacustico para alarma.



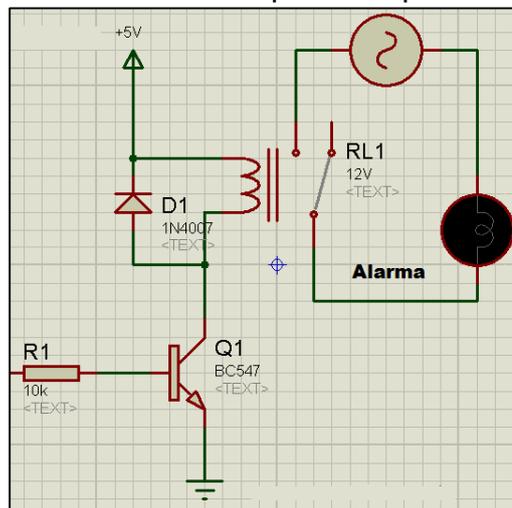
Fuente: Elaboración propia.

3.2.6.3 Diseño Electrónica de Potencia.

Una vez seleccionado el transductor electroacustico, es necesario diseñar una etapa que pueda brindar la potencia necesaria para el encendido o apagado del mismo. Además, esta etapa debe recibir las señales de control del sistema de reconocimiento de voz y el sistema de control por internet.

La variable a tener en consideración es la corriente máxima que es de 200 mA según las especificaciones técnicas mostradas en la tabla 6. Para ello se selecciona un módulo de relés encontrado en el mercado local, cuyas bobinas se polarizan con 5 VDC y soportan 10 A a través de sus contactos. El módulo de relés se muestra en la figura 56. Para este caso, sólo se utiliza un relé dejando a los demás para otras aplicaciones.

Figura 56. Circuito de potencia para alarma.



Fuente: Elaboración propia.

3.2.7 Video Vigilancia.

3.2.7.1 Consideraciones previas.

Las cámaras de video vigilancia han tenido una gran aceptación como elemento que ofrece seguridad a entidades tales como bancos u oficinas del gobierno, así como en

hogares. Además de ello, es importante mencionar que a un sistema de seguridad basado en cámaras de vigilancia se le puede agregar un elemento importante como lo es la domótica, incluyendo en este sistema la posibilidad de controlar de manera remota los aparatos eléctricos y las luces en el hogar para brindar seguridad, así como comodidad a la persona con paraplejía.

Es por ello, que a la habitación se le adicionará un sistema de seguridad basado en video vigilancia para que se pueda detectar cualquier evento. Además, un familiar puede observar al usuario con paraplejía para monitorearlo en caso de una emergencia. Por ello, se podrá acceder a las imágenes capturadas por las cámaras a través de una interfaz web de fácil acceso desde cualquier dispositivo que cuente con internet. El diseño de esta interfaz se explica en los siguientes apartados.

3.2.7.2 Selección.

Existen diferentes criterios para seleccionar un video vigilancia idóneo:

- Aplicación prevista: se refiere al nivel de detalle en resolución. En nuestra aplicación no es necesaria un nivel alto de resolución
- Canal de comunicación: se refiere a la utilización de redes IP o cámaras analógicas. Se escogen las cámaras IP debido a que su instalación es fácil y económica.
- Acceso remoto: las cámaras IP y analógicas permiten la vigilancia desde casi cualquier lugar, solo es necesaria una conexión a internet a través de un router. Sin embargo, los sistemas analógicos necesitan que los datos de video se digitalicen. Una razón más, para seleccionar cámaras IP.

Con todo esto, se selecciona la cámara DCS-930L de la marca D-link que constituye un sistema independiente que incorpora su propia CPU, evitando así requisitos especiales de hardware y software. Admite la conexión en red TCP/IP y el protocolo HTTP, y mediante un explorador web estándar se puede configurar y gestionar la cámara directamente desde una página web a través de Internet. Eso significa, que es posible diseñar un sistema de control remoto por web con la cámara seleccionada. Además, recoge las características analizadas en los criterios anteriores, tal como lo muestra la tabla 7 y la figura 61.

Tabla 7. Características de la cámara de vigilancia.

Modelo	DCS-930L
Voltaje	5VDC
Red	Ethernet – conector RJ45
Resolución	640x480
Ángulo de Visión	45,3°
Peso	41gr

Fuente: www.dlink.com

3.3 Diseño de las Interfaces de Usuario.

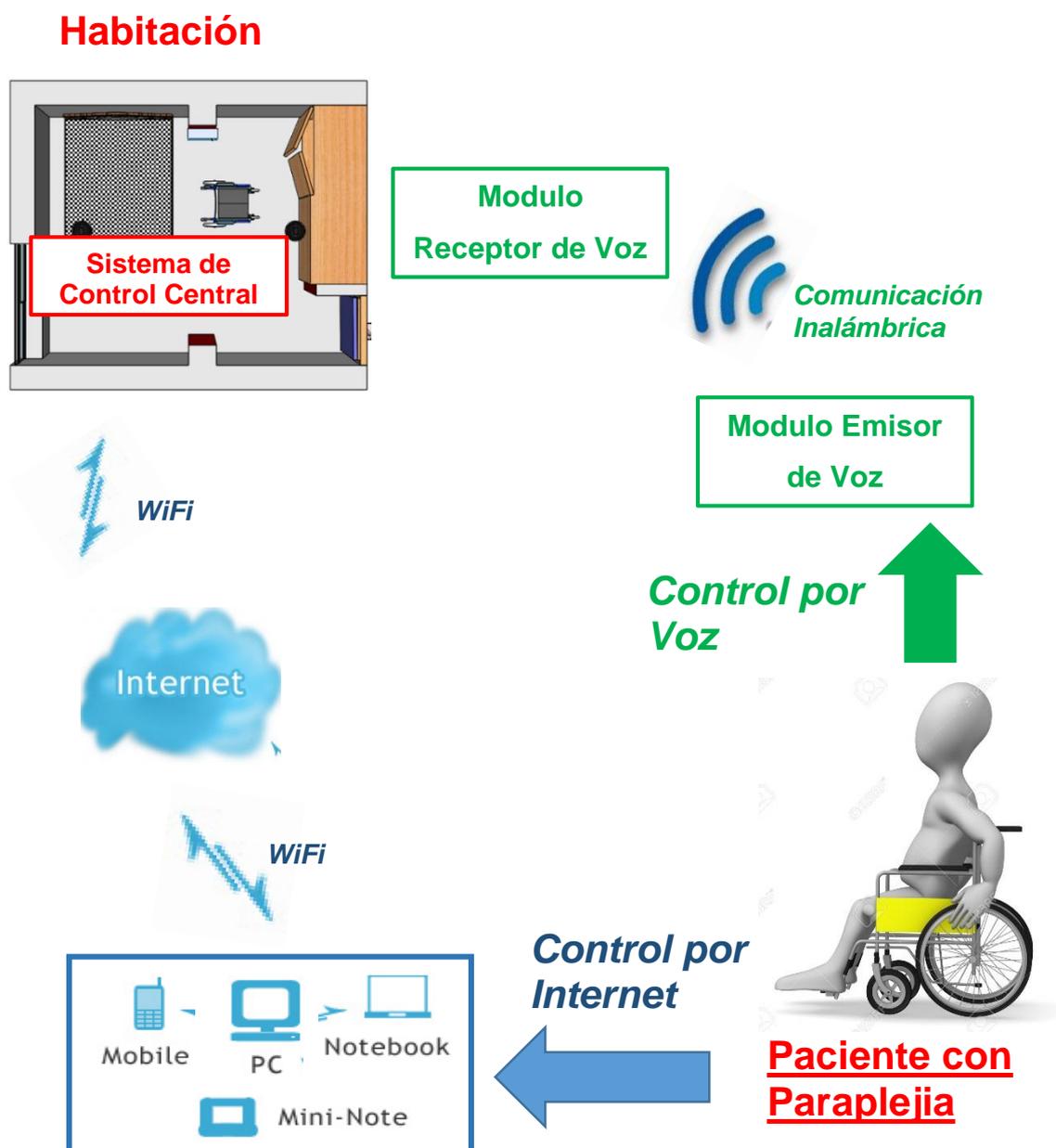
Con la finalidad de ayudar a la persona con discapacidad parapléjica a tener un mayor grado de autonomía y confort dentro de su hogar se pretende diseñar dos sistemas de control distintos:

1. Diseñar un sistema de control por internet a través de una página web que puede ser accedido desde cualquier dispositivo Móvil, Tablet, Smart TV o Laptop.
2. Diseñar un sistema de control de reconocimiento de voz a través de un micrófono inalámbrico con una cobertura de 3m.

Ambos sistemas de control permitirán gestionar actividades básicas como son: iluminación, puertas, persianas, ventanas, calefacción y video-vigilancia. Así, la presente tesis permite de alguna manera ayudar a las personas con paraplejía a tomar decisiones para su correcto desplazamiento en el interior de su vivienda o habitación. En la figura 57 se describe en un diagrama de bloques los modos de control de la habitación.

Vale mencionar que es necesario diseñar un sistema de control central que reciba los datos provenientes de los dos sistemas. Por ello en las siguientes secciones se describen el proceso de diseño de los sistemas y selección de sus respectivos componentes comenzando con el Sistema de Control Central (SCC), sistema de control por internet (SCI) y por último, el sistema de control por reconocimiento de voz (SCV).

Figura 57. Descripción de los sistemas de control.



Fuente: Elaboración propia.

3.3.1 Sistema de Control Central.

3.3.1.1 Consideraciones previas.

Es necesario tener un sistema que recepciones las ordenes de control de los sistemas por internet y voz. Por un lado, se requiere seleccionar un microcontrolador en donde se pueda conectarse a las 14 entradas/salidas del sistema domótico que se muestra en la tabla 8.

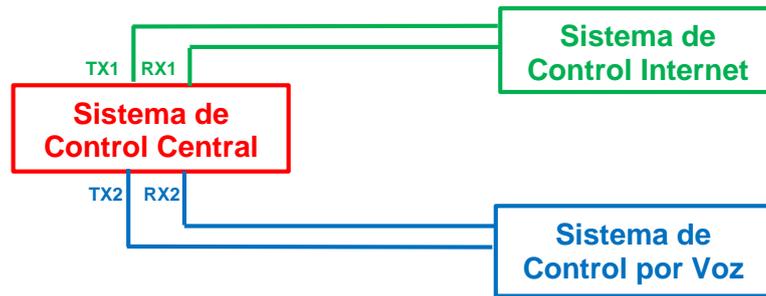
Tabla 8. Entradas y Salidas Digitales del Sistema Domótico.

SISTEMA DOMOTICO	ENTRADAS	SALIDAS
Puerta	2	2
Persiana	2	2
Ventana	0	3
Iluminación	0	1
Calefacción	0	1
Alarma	0	1
TOTAL	4	10

Fuente: Elaboración propia.

Por otro lado, se requiere de un microcontrolador capaz de comunicarse serialmente con dos sistemas de control al mismo tiempo tal como se muestra en la figura 58.

Figura 58. Comunicación serial de los sistemas de control.



Fuente: Elaboración propia.

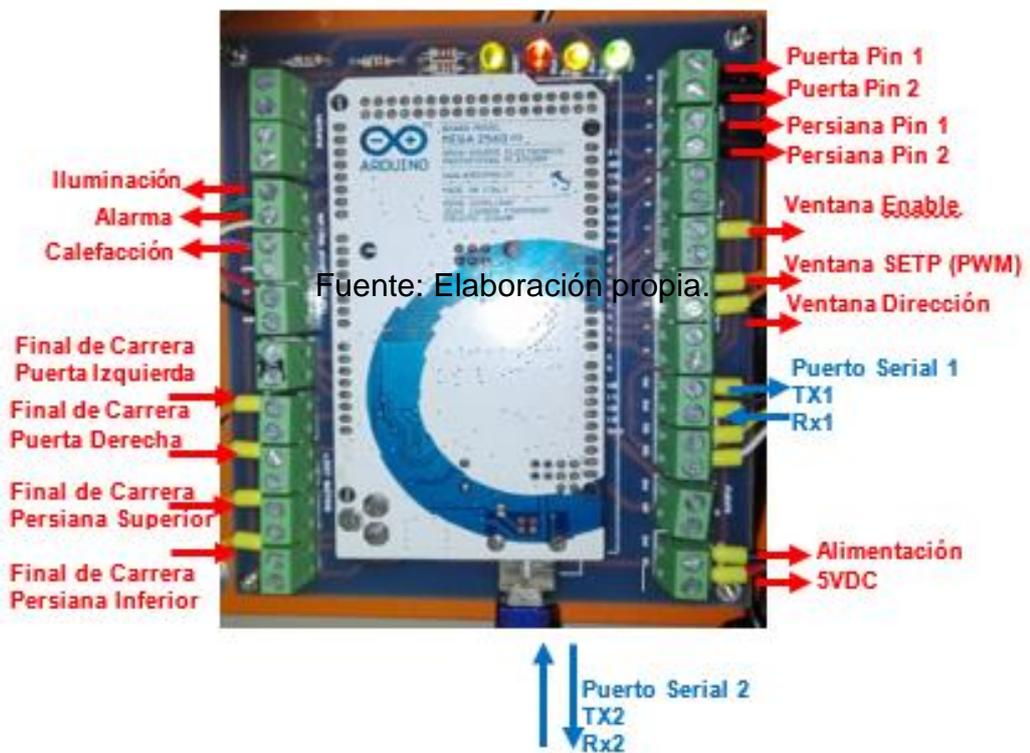
3.3.1.2 Controlador.

Para esta investigación se selecciona un Arduino ATmega 2560 porque cumple los requisitos mencionados como se muestra en sus características técnicas:

- Voltaje Operativo: 5 V.
- Voltaje de Entrada: 7-12 V.
- Voltaje de Entrada (límites): 6-20 V.
- **Pines digitales de Entrada/Salida: 54 (de los cuales 15 proveen salida PWM).**
- **Puertos seriales: 3.**
- Pines análogos de entrada: 16.
- Corriente DC por cada Pin Entrada/Salida: 40 Ma.
- Corriente DC entregada en el Pin 3.3 V: 50 mA.
- Memoria Flash: 256 KB (8KB usados por el bootloader).
- SRAM: 8 KB.
- EEPROM: 4 KB.
- Clock Speed: 16 MHz.

En la figura 59 se muestra al Arduino Mega conectado a las diferentes entradas y salidas del sistema domótico en el tablero eléctrico. En la tabla 9 se muestra en que pines del Arduino Mega está conectado cada entrada/salida y que nombre de etiqueta lleva en el código de programación.

Figura 59. Controlador del Sistema de Control General.



Fuente: Elaboración propia.

Tabla 9. Entradas y Salidas del Arduino y nombre de etiquetas.

ENTRADA/SALIDA	PIN	ETIQUETA
Iluminación	52	R1
Calefacción	50	R2
Alarma	48	R3
Persiana Pin 1	31	P1
Persiana Pin 2	29	P2
Puerta Pin 1	30	D1
Puerta Pin 2	28	D2
Ventana Dirección	2	DIR
Ventana Step	4	STEP
Ventana Enable	5	EN
Final Carrera Puerta Izquierda	11	Dtop
Final Carrera Puerta Derecha	12	Dbot
Final Carrera Persiana Superior	9	Ptop
Final Carrera Persiana Inferior	10	Pbot

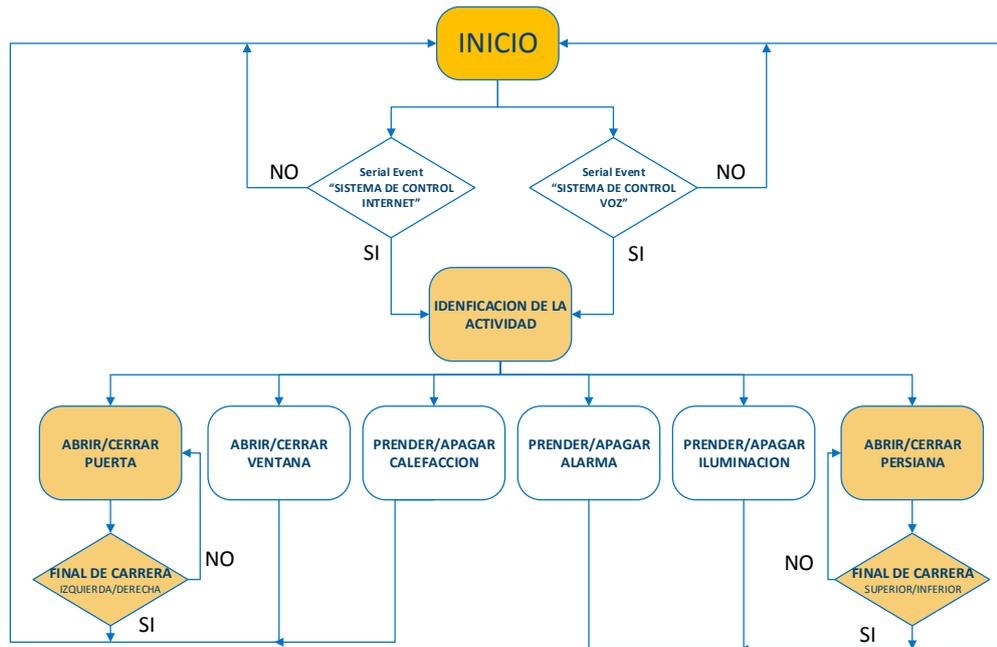
Fuente: Elaboración propia.

3.3.1.3 Algoritmo.

Un algoritmo es una serie de pasos o instrucciones con el fin de solucionar un problema. El algoritmo del Sistema de Control Central se muestra en la figura 60. El algoritmo inicia esperando comandos seriales provenientes del Sistema de Control por Internet o Sistema de Control por Voz para luego identificar qué actividad se requiere efectuar. Entre las actividades tenemos a: Abrir/Cerrar Puerta, Abrir/Cerrar Persiana, Abrir/Cerrar Ventana, Prender/Apagar Calefacción, Prender/Apagar Alarma y Prender/Apagar Iluminación. Solo las

primeras dos actividades confirmar su activación a través de finales de carrera Izquierda/Derecha y Superior/Inferior respectivamente. En las demás actividades no se realiza retroalimentación.

Figura 60. Algoritmo de Sistema de Control Central.



Fuente: Elaboración propia.

A continuación se comenzará a describir el algoritmo de programación para cada etapa del diagrama de flujo.

Algoritmo de Comunicación Serial:

Se estableció que los Sistemas de Control por Internet (SCI) y Sistema de Control por Voz (SCV) se comuniquen serialmente con el Sistema de Control Central (SCC). El primero a través del puerto

USB (SerialEvent) y el segundo a través de los puertos seriales TXD y RXD (SerialEvent1) del Arduino Mega. Ambos se configuran a una velocidad de 9600 baudios y se ejecutan en paralelo esperando comandos de actividad. El código de programación se muestra en la figura 61.

Figura 61. Código de programación Comunicación SCC.

```
void setup() {
  Serial.begin(9600);
  Serial.setTimeout(5);
  Serial1.begin(9600);
  Serial1.setTimeout(5);
  ...
void serialEvent() {
  digitalWrite(L4, !digitalRead(L4));
  delay(100);
  digitalWrite(L4, !digitalRead(L4));
  while(Serial.available()) {
    char Data=Serial.read();
    switch (Data) {
      ...
void serialEvent1() {
  digitalWrite(L4, !digitalRead(L4));
  delay(100);
  digitalWrite(L4, !digitalRead(L4));
  while(Serial.available()) {
    char Data=Serial.read();
    switch (Data) {
      ...
  ...
}
```

Fuente: Elaboración propia.

Los comandos de actividad son códigos codificados que envían los SCI y SCV al SCC a través de los puertos de comunicación. Estos comandos corresponden a cada actividad y se muestran en la tabla 10.

Tabla 10. Comandos de actividad.

COMANDO	ACTIVIDAD
R1	PRENDER/APAGAR ILUMINACION
R2	PRENDER/APAGAR CALEFACCION
R3	PRENDER/APAGAR ALARMA
D0	DETENER PUERTA
D1	ABRIR PUERTA
D2	CERRAR PUERTA
P0	DETERNER PERSIANA
P1	ABRIR PERSIANA
P2	CERRAR PERSIANA
V0	DETERNER VENTANA
V1	ABRIR VENTANA
V2	CERRAR VENTANA

Fuente: Elaboración propia.

Algoritmo de Abrir/Cerrar Puerta:

La puerta se abre o se cierra cuando se recibe el comando de actividad d1 y d2 respectivamente. Es posible detener el movimiento de la puerta en cualquier instante sí el SCC recibe el comando d0. El código de programación se muestra en la figura 62 y consta de sentencias de control “if” y “else if”.

Figura 62. Código de programación Puerta SCC.

```
void Puerta(int id){
  if (id==0){
    digitalWrite(D1,LOW);
    digitalWrite(D2,LOW);
  }
  else if (id==1){
    digitalWrite(D1,HIGH);
    digitalWrite(D2,LOW);
  }
  else if (id==2){
    digitalWrite(D1,LOW);
    digitalWrite(D2,HIGH);
  }
}
```

Fuente: Elaboración propia.

Algoritmo de Abrir/Cerrar Persiana:

La persiana se abre o se cierra cuando se recibe el comando de actividad p1 y p2 respectivamente. Es posible detener el movimiento de la persiana en cualquier instante sí el SCC recibe el comando p0. El código de programación se muestra en la figura 63 y consta de sentencias de control “if” y “else if”.

Figura 63. Código de programación Persiana SCC.

```
void Persiana(int ip){
  if (ip==0){
    digitalWrite(P1,LOW);
    digitalWrite(P2,LOW);
  }
  else if (ip==1){
    digitalWrite(P1,LOW);
    digitalWrite(P2,HIGH);
  }
  else if (ip==2){
    digitalWrite(P1,HIGH);
    digitalWrite(P2,LOW);
  }
}
```

Fuente: Elaboración propia.

Algoritmo de Abrir/Cerrar Ventana:

La ventana se abre o se cierra cuando se recibe el comando de actividad v1 y v2 respectivamente. Es posible detener el movimiento de la puerta en cualquier instante sí el SCC recibe el comando v0. El código de programación se muestra en la figura 64 y consta de sentencias de control “if” y “else if”.

Figura 64. Código de programación Ventana SCC.

```
void Ventana (int iv){
  if (iv==0) {
    stepv=3;          //DETENGO LA VENTANA
    digitalWrite(EN,HIGH);
    nvant=nv;
  }
  else if (iv==1) {
    digitalWrite(DIR,HIGH);
    digitalWrite(EN,LOW);
    stepv=1;
    stepcount+=1;
  }
  else if (iv==2) {
    digitalWrite(DIR,LOW);
    digitalWrite(EN,LOW);
    stepv=2;
    stepcount-=1;
  }
  if (stepcount==500 || stepcount==0){
    nvant=nv;
    digitalWrite(EN,HIGH);
    stepv=3;
  }
}
```

Fuente: Elaboración propia.

Algoritmo de Prender/Apagar Iluminación, Calefacción y Alarma:

La iluminación se prende o se apaga cuando se recibe el comando de actividad r1. La calefacción se prende o se apaga cuando se recibe el comando de actividad r2. La alarma se prende o se apaga cuando se recibe el comando de actividad r3. El código de programación se muestra en la figura 65 y consta de sentencias de control “if”, “case/break” y “else if”.

Figura 65. Código de programación Iluminación/Calefacción/Alarma

```
case 'r' :
nr=Serial.parseInt();
//Serial.print("Relay ");Serial.println(nr);
if (nr==1) digitalWrite(R1,!digitalRead(R1));
else if (nr==2) digitalWrite(R2,!digitalRead(R2));
else if (nr==3) digitalWrite(R3,!digitalRead(R3));
break;
```

Fuente: Elaboración propia.

3.3.2 Sistema de Control por Internet.

3.3.2.1 Consideraciones previas.

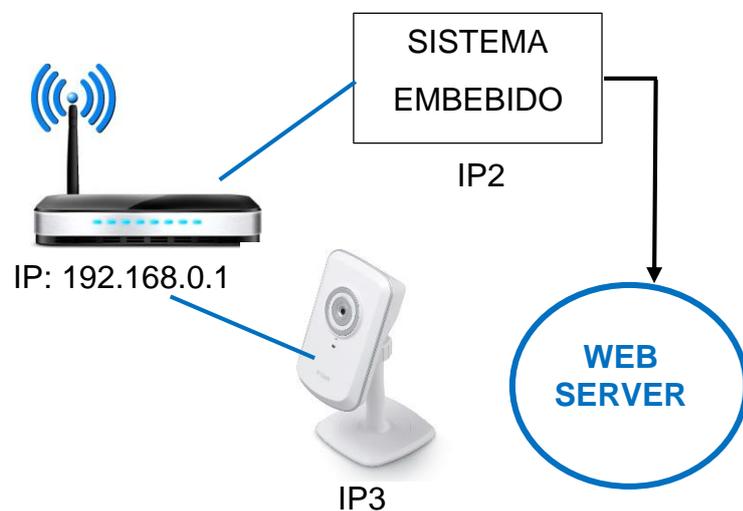
Para que el usuario parapléjico pueda interactuar con el sistema domótico, se ha desarrollado un software de control que sirva de interfaz de usuario permitiendo activar o desactivar los actuadores y revisar la cámara de video-vigilancia.

Una de las opciones es desarrollar una aplicación de escritorio e instalarlo en una computadora con sistema operativo Windows, sin embargo, se excluiría otros sistemas operativos, que actualmente también son conocidos, como Linux, Android, Mac OS, etc. Por ello, se elige desarrollar una aplicación web, la cual no necesita ser instalada porque se ejecuta a través de un navegador. Esta opción, permite que el usuario parapléjico pueda interactuar cómodamente con el sistema desde cualquier dispositivo (computadora, laptop, Smartphone, SmartTV, etc) o sistema operativo.

El sistema se basa en un sistema embebido, el cual ejecuta una interfaz HMI capaz de conectarse a un servidor web. Los Web Service permiten intercomunicar dos sistemas remotos a través de la Web manteniendo el estado de los objetos en la transferencia. Un Web Service nos permite ejecutar un método

remotamente y recibir su resultado como si se tratara de un método local a nuestro sistema (Canelo, Vera, 2015, p.156). En el servidor web se deberá visualizar las imágenes de la cámara IP D-Link y para ello se necesita conectarlo al router para formar una red de área local, tal como se muestra en la figura 66.

Figura 66. Red Local para el SCI.



Fuente: Elaboración propia.

3.3.2.2 Selección del Sistema Embebido

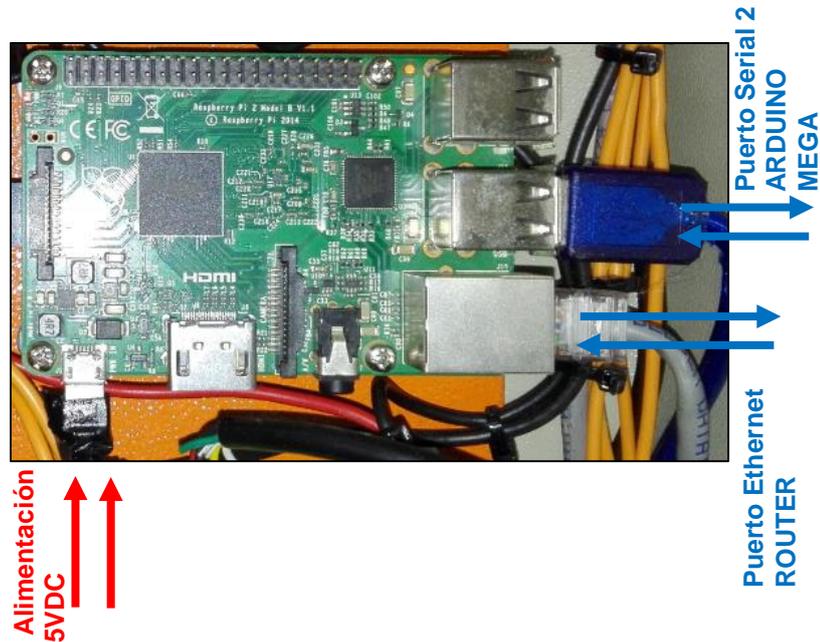
Entre los parámetros a considerar en la selección del sistema embebido se tiene:

- Costo: el costo monetario de comprar el sistema embebido manufacturado.

- Tamaño físico y peso: el sistema debería ser lo más ligero y pequeño posible para poder ser transportado por un usuario parapléjico.
- Fiabilidad: debe ofrecer garantías para su correcta operación, evitando fallos.
- Flexibilidad: la habilidad de cambiar la funcionalidad del sistema sin incurrir en un gran costo.
- Interfaces de comunicación: debe comunicarse a través de RS232 con el Sistema de Control Central y por Red Ethernet con el router de la habitación.
- Acceso a un Red: puede ser LAN o WAN para la conexión al Servidor Web.

Para esta investigación, se selecciona el sistema embebido Raspberry Pi 3 que tiene un procesador ARM, memoria RAM de 1GB, 1 Puerto Ethernet, 4 Puertos USB's, ROM, interfaces de entrada-salida y cumple con las características mencionadas anteriormente (Canelo, Vera, 2016, p.42). La comunicación con el Arduino Mega se realiza mediante el puerto USB y con el router se realiza con el Puerto Ethernet, tal como se muestra en la figura 67.

Figura 67. Puertos de comunicación del Raspberry Pi 3.



Fuente: Elaboración propia.

Se utilizará el sistema operativo embebido NOOBS porque es básica, completa y recomendada por la página web de Raspberry Pi: <https://www.raspberrypi.org/downloads/noobs/>

3.3.2.3 Configuración de la red con IP estática.

En una red de IP dinámicas se tiene la posibilidad de que cuando se reinicie el Router o se reconfigure un puerto del Raspberry Pi las IP cambian a otras. Para evitar estos inconvenientes, se configurará el Raspberry Pi con IP estáticas. La IP asignada es 192.168.0.122. Esta configuración se inicia ejecutando el comando que se muestra en la figura 68.

Figura 68. Configuración de IP estática en Raspberry Pi.

```
RASPBERRY PI 3
sudo nano /etc/dhcpd.conf

Interface eth0

Static ip_addres=192.168.0.122/24

Static routers=192.168.0.1

Static domain_name_servers= 192.168.0.1
```

Fuente Elaboración propia.

3.3.2.4 Programación del Servidor Web

Para convertir el Raspberry Pi se necesita instalar un Servidor Web como Lighttpd, Tomcat o Apache. Se selecciona el servidor web Apache porque es uno de los más populares y existe mucha información técnica. Apache es un servidor web HTTP de código abierto ejecutable bajo diferentes sistemas operativo. Una de sus grandes ventajas es que es un servidor modular, es decir, permite ir ampliando sus capacidades con el paso del tiempo. Además, es altamente configurable permitiendo personalizar la respuesta ante posibles errores y trabajar con bases de datos.

Para instalar Apache en el Raspberry Pi se tiene que verificar que se encuentre actualizado. Para luego escribir el código que se muestra en la figura 69.

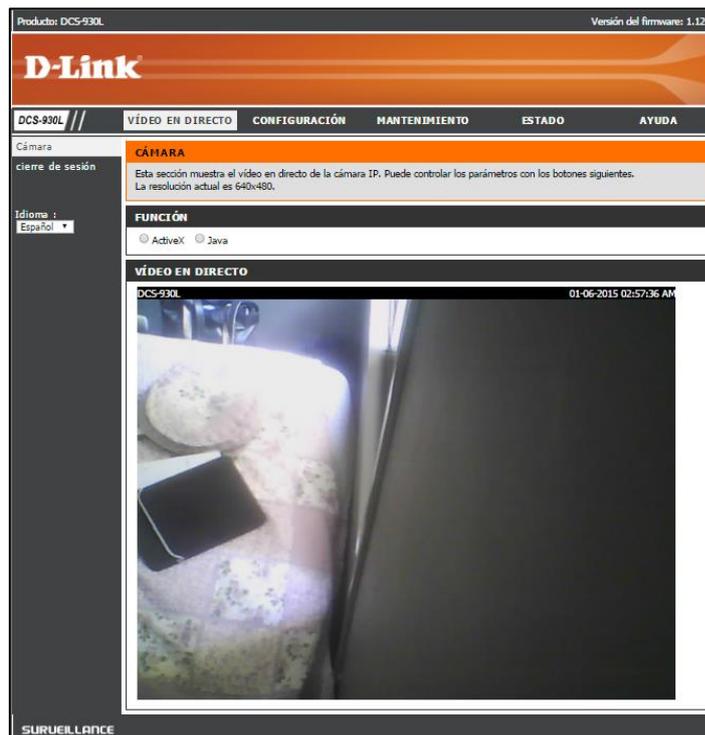
Figura 69. Instalación del Apache en Raspberry Pi 3

RASPBERRY PI 3
<i>Sudo apt update</i>
<i>Sudo apt-get install apache2 php5 libapache2-mod-php5</i>
<i>En el proceso solicita permiso para ocupar espacio en la SD, escribir "y"</i>
<i>Sudo /etc/init.d/apache2 restart</i>

Fuente: Elaboración propia.

Luego, se tiene que conectar la Cámara IP D-Link a una red local. Para ello, se debe configurar una IP fija utilizando su propia configuración web, tal como se muestra en la figura 70. La IP que se asigna a la cámara es de: 192.168.0.150.

Figura 70. Configuración Web Cámara D-Link.



Fuente: Elaboración propia.

Con ambas configuraciones de IP y el servidor Web Apache instalado, se procede a diseñar la página web principal con las siguientes características:

- Debe mostrar las imágenes de la cámara D-Link.
- Se debe poder activar todos los actuadores del sistema domótico.
- Debe tener botones grandes para una mejor visualización del usuario.
- Debe tener gráficos que se asemejen a los actuadores.

Las páginas web no se limitan a presentar información al usuario, sino que requiere interactuar con él, por lo que se

necesita de lenguajes de programación. Éstos pueden ser básicamente de dos tipos: los lenguajes de programación del lado servidor, que son ejecutados e interpretados por el propio servidor y los lenguajes del lado cliente que son aquellos que se envían al cliente junto al lenguaje HTML. Se optó por emplear PHP al ser un lenguaje del lado del servidor y por ser uno de los soportados por Apache. En definitiva, la aplicación web consiste en un programa escrito en PHP, alojado en un servidor y que incorpora código HTML. Vale mencionar que PHP aporta la funcionalidad a la aplicación web mientras HTML se encarga de los aspectos visuales de la interfaz.

Dicho esto, cada actuador del sistema domótico tendrá su respectiva página web y todas serán llamadas desde la página web principal denominada “index”. En la tabla 11, se muestra las diferentes páginas web y sus descripciones.

Tabla 11. Páginas Webs del SCI.

Nombre de pagina	Descripción
Index.php	<p>Al inicializar se carga las siguientes páginas:</p> <ul style="list-style-type: none"> • <camera.php> • <luces_on.php> • <luces_off.php> • <alarma.php> • <calec_on.php> • <calec_off.php> • <persiana_on.php> • <persiana_off.php> • <puerta_on.php>

	<ul style="list-style-type: none"> • <puerta_off.php> • <ventana_on.php> • <ventana_off.php> • <detener.php>
Camera.php	Responsable de mostrar las imágenes de la Camara D-LINK.
Luces_on.php	Responsable de enviar el comando para encender las luces y ejecuta la página <luces_off.php>
Luces_off.php	Responsable de enviar el comando para apagar las luces y ejecuta la página <luces_on.php>
Alarma.php	Responsable de enviar el comando para encender la alarma.
Calec_on.php	Responsable de enviar el comando para encender la calefacción y ejecuta la página <calec_off.php>
Calec_off.php	Responsable de enviar el comando para apagar las luces y ejecuta la página <calec_on.php>
Persiana_off.php	Responsable de enviar el comando para abrir la persiana y ejecuta la página <persiana_on.php>
Persiana_on.php	Responsable de enviar el comando para cerrar la persiana y ejecuta la página <persiana_off.php>
Puerta_on.php	Responsable de enviar el comando para abrir la puerta y ejecuta la página <puerta_on.php>
Puerta_off.php	Responsable de enviar el comando para cerrar la puerta y ejecuta la página <puerta_off.php>

Ventana_on.php	Responsable de enviar el comando para abrir la ventana y ejecuta la página <ventana_on.php>
Ventana_off.php	Responsable de enviar el comando para cerrar la ventana y ejecuta la página <ventana_off.php>
Detener.php	Responsable de enviar el comando para detener todas las actividades que incluyan motores. Esta no afecta a las salidas tipo relé.

Fuente: Elaboración propia.

En la tabla 12, se muestra la lista de comandos enviados por las páginas web y su equivalencia con respecto a los comandos de actividad interpretados por el Arduino Mega (tabla 10).

Tabla 12. Comandos Webs del SCI.

Comandos Web	Comandos Arduino	Actividad
a	v1	Ventana abre
b	v2	Ventana Cierra
c	d1	Puerta Abre
d	d2	Puerta Cierra
e	p1	Persiana Abre
f	p2	Persiana Cierra
g	r2	Calefacción Enciende
h	r2	Calefacción Apaga
i	r1	Luces Enciende
j	r1	Luces Apaga
k	r3	Encender Alarma
m	m	Persiana detener
n	n	Puerta detener
o	o	Ventana detener

q	-	Salir del Webserver.py
t	t	Apagar Todo

Fuente: Elaboración propia.

El diseño de la página principal del Sistema de Control por Internet se muestra en la figura 71. Vale mencionar que los códigos PHP y HTML de las páginas se encuentran documentadas en el Anexo.

Figura 71. Página Principal del SCI.



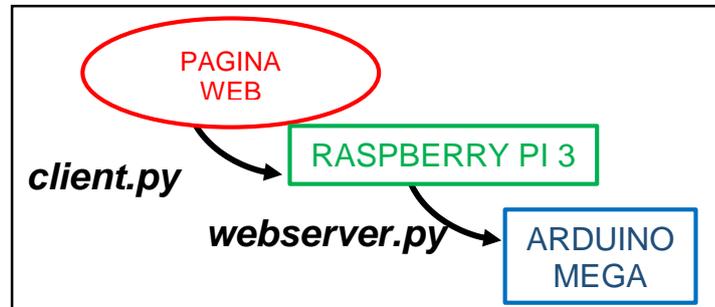
Fuente: Elaboración propia.

3.3.2.5 Librerías/scripts en Python para manejo del SCC

Cada página web envía un comando de acuerdo al actuador que se desea activar o desactivar. Estos comandos deben ser recepcionados por el Raspberry Pi 3 y enviados al Arduino Mega. Para realizar esto se desarrolla dos librerías, hechas en

el lenguaje de programación Python, llamadas “client.py” y “webserver.py” tal como se muestra en la figura 72.

Figura 72. Librerías Python para SCI.



Fuente: Elaboración propia.

Como se muestra, la librería “client.py” recolecta información desde las páginas web y las almacena en una variable llamada “data”, tal como se muestra en código de la figura 73.

Figura 73. Código Python de librería client.py

```
RASPBERRY PI 3
import socket
import sys
TCP_IP = '192.168.0.122'
TCP_PORT = 5005
BUFFER_SIZE = 1024
MESSAGE = str(sys.argv[1])
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect((TCP_IP, TCP_PORT))
s.send(MESSAGE)
data = s.recv(BUFFER_SIZE)
s.close()
#print "received data:", data
```

Fuente: Elaboración propia.

Por otro lado, la librería “webserver.py” recibe la variable data y envía los comandos de actividad al Arduino Mega de

acuerdo con lo establecido en la tabla 12. En la figura 74 se muestra el código de programación de esta librería.

Figura 74. Inicio de Código Python de librería webserver.py

```
RASPBERRY PI 3
#WEBSERVER.PY DATOS PRINCIPALES PARA EL CONTROL
# a VENTANA ON
# b VETANA OFF
# c PUERTA ON
# d PUERTA OFF
# e PERSIANA ON
# f PERSIANA OFF
# g CALEFACCION ON
# h CALEFACCION OFF
# i LUCES ON
# j CUCES OFF
# k ALARMA ON
import socket
import serial
import time
import subprocess
dev=subprocess.check_output('ls /dev/ttyACM*',shell=True)
print dev
try:
    ser=serial.Serial(dev.strip(),9600)
    print "Arduino Conectado"
    ser.write('11')
    time.sleep(0.5)
except:
    print "Arduino Desconectado"
...
```

Fuente: Elaboración propia.

Luego de que el Arduino se encuentra conectado, el algoritmo espera a recibir el comando web, lo guarda en la variable “data” y la compara utilizando la estructura de control “if-else”. En la figura 75, se muestra que sí se cumple la condición de “data==a” entonces se envía el comando “v1”.

Figura 75. Estructura de Control If-else I de código Python.

```
RASPBERRY PI 3
```

```

...def server():
    global ser
    while True:
        conn, addr = s.accept()
        print 'Direccion de Coneccion: ', addr
        data = conn.recv(BUFFER_SIZE)
        if not data: continue
        print "Informacion recibida: ", data
        if data=='a':
            conn.close()
            ser.write('v1')
            print "se envia v1"
            time.sleep(0.5)
    ...

```

Fuente: elaboración propia.

En la figura 76, se muestra que sí se cumple la condición de “data==b” entonces se envía el comando “v2”.

Figura 76. Estructura de Control If-else II de código Python.

RASPBERRY PI 3
<pre> ... elif data=='b': conn.close() ser.write('v2') print "se envia v2" time.sleep(0.5) ... </pre>

Fuente: Elaboración propia.

En la figura 77, se muestra que sí se cumple la condición de “data==c” entonces se envía el comando “d1”.

Figura 77. Estructura de Control If-else III de código Python.

RASPBERRY PI 3
<pre> ... elif data=='c': conn.close() ser.write('d1') print "se envia d1" time.sleep(0.5) ... </pre>

Fuente: Elaboración propia.

En la figura 78, se muestra que sí se cumple la condición de “data==d” entonces se envía el comando “d2”.

Figura 78. Estructura de Control If-else IV de código Python.

```
RASPBERRY PI 3
...
    elif data=='d':
        conn.close()
        ser.write('d2')
        print "se envia d2"
        time.sleep(0.5)
...
```

Fuente: Elaboración propia.

En la figura 79, se muestra que sí se cumple la condición de “data==e” entonces se envía el comando “p1”.

Figura 79.Estructura de Control If-else V de código Python.

```
RASPBERRY PI 3
...
    elif data=='e':
        conn.close()
        ser.write('p1')
        print "se envia p1"
        time.sleep(0.5)
...
```

Fuente: Elaboración propia.

En la figura 80, se muestra que sí se cumple la condición de “data==f” entonces se envía el comando “p2”.

Figura 80. Estructura de Control If-else VI de código Python.

```
RASPBERRY PI 3
...
    elif data=='f':
        conn.close()
        ser.write('p2')
        print "se envia p2"
        time.sleep(0.5)
...
```

Fuente: Elaboración propia.

En la figura 81, se muestra que sí se cumple la condición de “data==g” entonces se envía el comando “r2”.

Figura 81. Estructura de Control If-else VII de código Python.

```
RASPBERRY PI 3
...
    elif data=='g':
        conn.close()
        ser.write('r2')
        print "se envia r2"
        time.sleep(0.5)
...
```

Fuente: Elaboración propia.

En la figura 82, se muestra que sí se cumple la condición de “data==h” entonces se envía el comando “r2”.

Figura 82. Estructura de Control If-else VIII de código Python.

```
RASPBERRY PI 3
...
    elif data=='h':
        conn.close()
        ser.write('r2')
        print "se envia r2"
        time.sleep(0.5)
...
```

Fuente: Elaboración propia.

En la figura 83, se muestra que sí se cumple la condición de “data==i” entonces se envía el comando “r1”.

Figura 83. Estructura de Control If-else IX de código Python.

```
RASPBERRY PI 3
...
    elif data=='i':
        conn.close()
        ser.write('r1')
        print "se envia r1"
```

```
...  
        time.sleep(0.5)
```

Fuente: Elaboración propia.

En la figura 84, se muestra que sí se cumple la condición de “data==j” entonces se envía el comando “r1”.

Figura 84. Estructura de Control If-else X de código Python.

```
RASPBERRY PI 3  
...  
        elif data=='j':  
            conn.close()  
            ser.write('r1')  
            print "se envia r1"  
            time.sleep(0.5)  
...
```

Fuente: Elaboración propia.

En la figura 85, se muestra que sí se cumple la condición de “data==k” entonces se envía el comando “r3”.

Figura 85. Estructura de Control If-else XI de código Python.

```
RASPBERRY PI 3  
...  
        elif data=='k':  
            conn.close()  
            ser.write('r3')  
            print "se envia r3"  
            time.sleep(0.5)  
...
```

Fuente: Elaboración propia.

En la figura 86, se muestra que sí se cumple la condición de “data==t” entonces se envía el comando “t”.

Figura 86. Estructura de Control If-else XII de código Python.

```
RASPBERRY PI 3  
...  
        elif data=='t':
```

```
conn.close()
ser.write('t')
print "se envia t"
time.sleep(0.5)
...
```

Fuente: Elaboración propia.

En la figura 87, se muestra que sí se cumple la condición de “data==m” entonces se envía el comando “r3”.

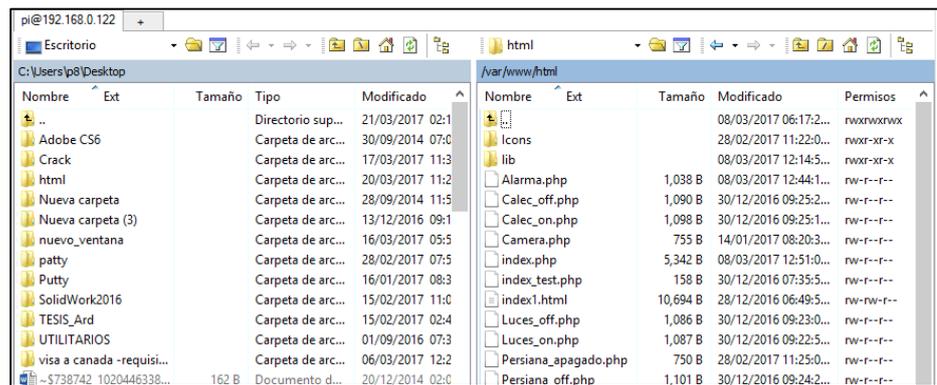
Figura 87. Estructura de Control If-else XIII de código Python.

```
RASPBERRY PI 3
...
elif data=='m':
    conn.close()
    ser.write('r3')
    print "se envia r3"
    time.sleep(0.5)
...
```

Fuente: Elaboración propia.

Luego, se usa el programa WinSCP para traspasar los códigos de programación del computador al Raspberry Pi 3. Las librerías (python) y las páginas web (PHP) se guardan en la dirección /var/www/html. En la figura 88 se muestra el EDT del programa WinSCP.

Figura 88. Programa WinSCP.



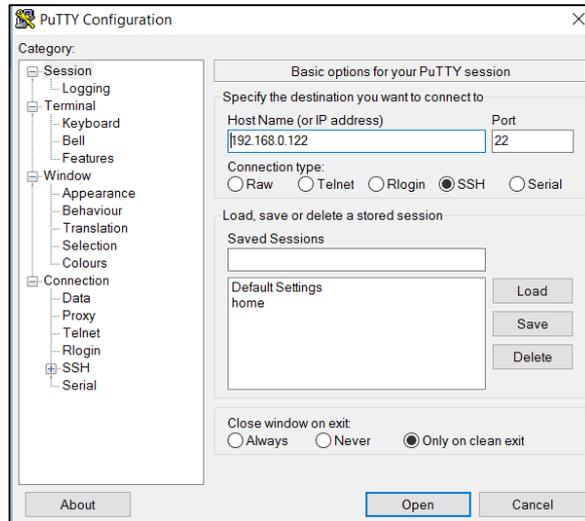
Fuente: Elaboración propia.

Por último, se inicializa el servidor web a través de PuTTY. PuTTY es un cliente SSH y Telnet para conectarse a servidores remotos y poder ejecutar comandos. Presenta las siguientes características:

- Es gratuito y de código abierto.
- Disponible para varias plataformas entre ellas Windows y Linux.
- Es una aplicación portable.
- Interfaz sencilla y manejable.
- Completo y ofrece una gran flexibilidad con multitud de opciones.
- Se encuentra en constante desarrollo.

Luego de descargar el programa, se procede a configurarlo como se muestra en la figura 89.

Figura 89. Configuración de PuTTY.



Fuente: Elaboración propia.

Por consiguiente, se coloca la siguiente instrucción en la consola: “*python /var/www/html/lib/webserver.py*”. Y de esta manera, se establece la comunicación con el Arduino Mega y se pone a la espera de los datos enviados por las páginas web, tal como se muestra en la figura 90.

Figura 90. Servidor Web y Arduino conectado.

```
login as: pi
pi@192.168.0.122's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Mar 21 15:17:10 2017
pi@raspberrypi:~$ python /var/www/html/lib/webserver.py
/dev/ttyACM0

Arduino Conectado
Servidor Iniciado
█
```

Fuente: Elaboración propia.

3.3.3 Sistema de Control por Voz.

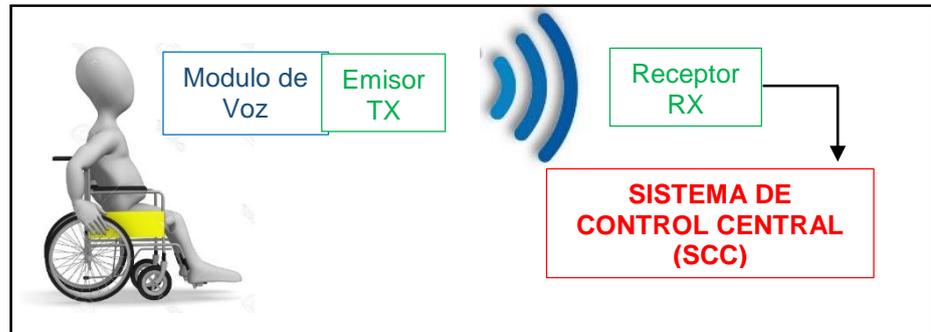
3.3.3.1 Consideraciones previas.

Existen diferentes niveles de paraplejia que impiden el movimiento de las extremidades tanto inferiores como superiores. Además, La falta de los servicios de apoyo necesarios puede hacer que las personas con discapacidad tengan una dependencia excesiva de los miembros de la familia (Banco Mundial, 2011, p.153). Por ello, se pretende diseñar una interfaz de comunicación que no comprometa movimientos motrices sino que solo se utilice la propia voz del usuario. Las características que debe tener este sistema son las siguientes:

- Transportable: se refiere a que no debe ser un problema para el usuario parapléjico tenerlo entre sus objetos personales.
- Reducido en tamaño.
- Inalámbrico: con el objetivo de ser lo más portable posible y no depender de cables en la habitación.
- Poco consumo: al ser inalámbrico, se necesita de un dispositivo que consuma poca corriente para no estar cambiando de baterías frecuentemente y sea fastidioso para el usuario.

La figura 91 muestra un diagrama de bloques del sistema de control por reconocimiento de voz.

Figura 91. Diagrama de bloques del SCV.



Fuente: Elaboración propia.

3.3.3.1 Selección del módulo de comunicación

Con respecto a la comunicación inalámbrica, existen varias tecnologías que se puede seleccionar y las más conocidas utilizadas en domótica son: WiFi, RF y Bluetooth:

- Bluetooth: es una tecnología inalámbrica utilizada para conectar dispositivos electrónicos entre sí con altos niveles de seguridad. A través del Bluetooth, podemos transmitir cómodamente datos entre aparatos tan empleados en la actualidad como los teléfonos móviles, los ordenadores, etc.
- WiFi: es un mecanismo de conexión de dispositivos electrónicos de forma inalámbrica. Los dispositivos

habilitados con Wi-Fi, tales como: un ordenador personal, una consola de videojuegos, un Smartphone o un reproductor de audio digital, pueden conectarse a Internet a través de un punto de acceso de red inalámbrica.

- RF: La Radio Frecuencia, es la característica que define a un grupo o subconjunto de ondas electromagnéticas que se propagan en el espectro en unos rangos utilizados principalmente en las comunicaciones de radio.

Entre las alternativas mostradas, se selecciona la tecnología bluetooth porque nos asegura una transferencia segura de datos, gran alcance y es utilizada en proyectos de domótica. Se selecciona el modulo Bluetooth HC-05 porque tiene la capacidad de comunicarse con un microcontrolador por el protocolo UART y tiene otras características que se muestran en la tabla 13. Vale mencionar que se necesita dos módulos, uno como emisor y otro como receptor.

Este último se ubica en el tablero del Sistema de Control Central, tal como se muestra en la figura 92.

Tabla 13. Características del módulo Bluetooth HC-05 para SCV.

Modelo	HC-05
Especificación	Bluetooth v2.0
Voltaje	3.3-5VDC
Chip	BC417143
Alcance	10m
Modulación	GFSK
Nivel	TTL
Frecuencia	2,4GHz, Banda ISM
Velocidad	1200bps – 1.3Mbps

Fuente: www.lecomperu.com

Figura 92. Modulo Receptor Bluetooth HC-05.



Fuente: Elaboración propia.

3.3.3.2 Selección del módulo de voz

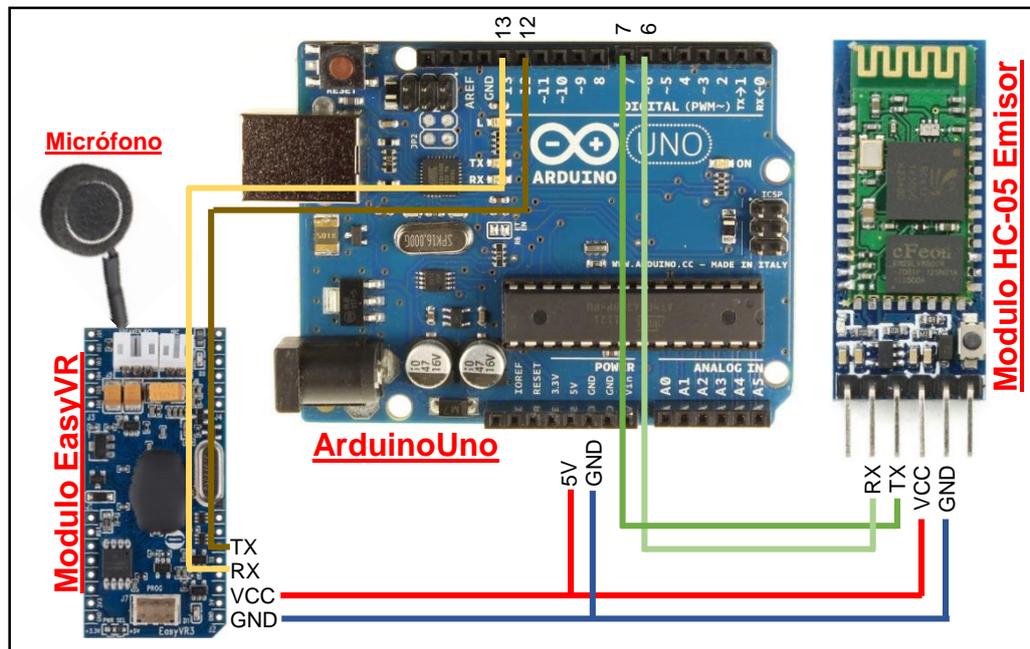
Se necesita seleccionar un módulo de voz que reconozca un número limitado de palabras aisladas para poder activar o desactivar los actuadores del sistema domótico. Lógicamente, el procedimiento se debe realizar gravando previamente unos patrones de voz. Para la elaboración de esta investigación, se opta por la utilización del módulo de voz EasyVR 3.0 que realiza un reconocimiento de voz multipropósito y presenta las siguientes características:

- 26 comandos y voces pre-programados (SI) en: en inglés, italiano, japonés, alemán, francés y español.
- Admite hasta 32 comandos y sus correspondientes voces definidas por el usuario (SD) en cualquier idioma.
- Los comandos y voces SD se pueden organizar en hasta 16 grupos.
- Se dispone de un interface gráfico de usuario (GUI) para Windows que permite una rápida familiarización con el uso del módulo.
- El módulo se puede conectar fácilmente con cualquier tipo de controlador mediante una sencilla comunicación serial.
- Dispone de un potente protocolo de comandos para el desarrollo de cualquier aplicación basada en el reconocimiento de voz.
- Alimentación de 3.3V a 5V.

- Líneas de E/S o GPIO (IO1, IO2 e IO3).

El módulo EasyVR es capaz de comunicarse con cualquier microcontrolador a través de comunicación serial. En este caso se selecciona un Arduino Uno que se encuentra en el mercado local. La conexión entre estos dos dispositivos y el modulo emisor Bluetooth HC-05 se muestra en la figura 93.

Figura 93. Conexión EasyVR – ArduinoUno – Emisor HC-05.



Fuente: Elaboración propia.

3.3.3.3 Configuración de comandos de voz

El módulo de voz se configura a través del programa EasyVR Commander que se puede descargar gratuitamente desde internet. En este programa se puede ingresar los comandos a reconocer y entrenar el reconocimiento de voz. Para realizar esto, es necesario que el módulo se conecte con la plataforma

ArduinoUno en modo bootloader. Al conectar el ArduinoUno a la PC se genera un puerto COM, el mismo que se debe sincronizar con el programa EasyVR Commander.

Luego, la inserción de los comandos de voz es ubicada en diferentes grupos (0-15). Primero se decidió ubicarlos en el grupo 2 pero se tuvo un retardo de 3 segundos en la respuesta del sistema. Como señala el Manual de usuario de EasyVR “la función de reconocimiento del Easy VR trabaja en un solo grupo al mismo tiempo (...) y lee de arriba hacia abajo todos los grupos” (RoboTech, 2010, p.58). Sobre la base de lo mencionado antes se cambió los comandos al grupo 1 y se disminuyó el retado a un segundo. Estos comandos se muestran en tabla 14.

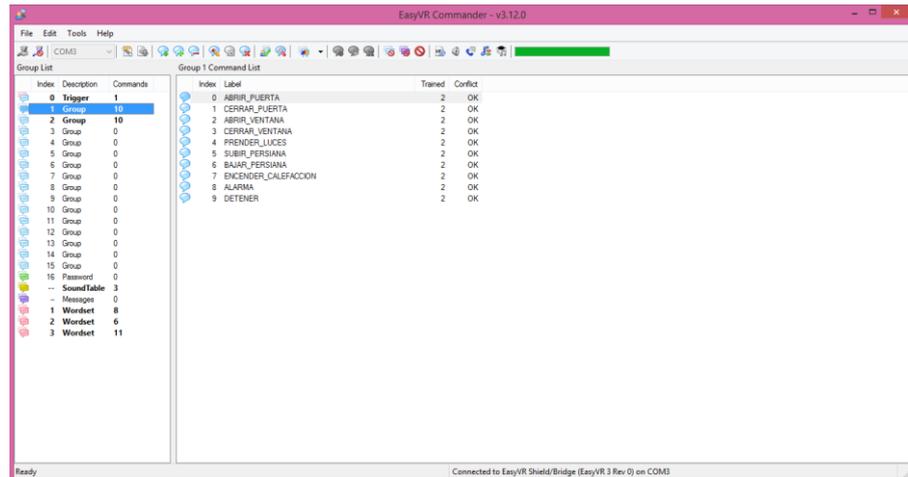
Tabla 14. Comandos de voz para el SCV.

Comandos de voz	Etiqueta	Actividad
“Abrir Puerta”	G1_ABRIR_PUERTA	Abre Puerta
“Cerrar Puerta”	G1_CERRAR_PUERTA	Cierra Puerta
“Abrir Ventana”	G1_ABRIR_VENTANA	Abre Ventana
“Cerrar Ventana”	G1_CERRAR_VENTANA	Cierra Ventana
“Prender Luces”	G1_PRENDER_LUCES	Prende Luces
“Subir Persiana”	G1_SUBIR_PERSIANA	Sube Persiana
“Bajar Persiana”	G1_BAJAR_PERSIANA	Baja Persiana
“Encender Calefacción”	G1_ENCENDER_CALEF ACCION	Enciende/Apaga Calefacción
“Alarma”	G1_ALARMA	Enciende/Apaga Alarma
“Detener”	G1_DETENER	Detiene movimientos los motores

Fuente: Elaboración propia.

La configuración de comandos y entrenamiento se realizan haciendo click en el botones “Insert Command” y “Train Command” en el menú “Edit”. En la figura 94 se muestra el programa EasyVR Commander con los comandos entrenados.

Figura 94. Comandos entrenados en EasyVR Commander.



Fuente: Elaboración propia.

Por último, generamos el código haciendo click en el botón “Generate Code” del menú File. En el código generado solo se debe agregar sentencias de control que conviertan los comandos de voz hacia los comandos de actividad utilizados en el ArduinoMega en el SCC, para luego enviarlos serialmente hacia el modulo emisor bluetooth HC-05, tal como se muestra en el código de la figura 95. Los pines 7 y 6 del ArduinoUno se configuran como puerto serie utilizando la siguiente instrucción: *SoftwareSerial BTMaestro (7,6)*. Vale mencionar, que el código completo se muestra en los Anexos de este documento.

Figura 95. Código de Programación del ArduinoUno.

```

ArduinoUno
...
case GROUP_1:
  switch (idx)
  {
    case G1_ABRIR_PUERTA:
      BTMaestro.print("d1");
      // write your action code here
      // group = GROUP_X; <-- or jump to another group X for composite commands
      break;
    case G1_CERRAR_PUERTA:
      BTMaestro.print("d2");
      // write your action code here
      // group = GROUP_X; <-- or jump to another group X for composite commands
      break;
    case G1_ABRIR_VENTANA:
      BTMaestro.print("v1");
      // write your action code here
      // group = GROUP_X; <-- or jump to another group X for composite commands
      break;
    case G1_CERRAR_VENTANA:
      BTMaestro.print("v2");
      // write your action code here
      // group = GROUP_X; <-- or jump to another group X for composite commands
      break;
    case G1_PRENDER_LUCES:
      BTMaestro.print("r1");
      // write your action code here
      // group = GROUP_X; <-- or jump to another group X for composite commands
    case G1_SUBIR_PERSIANA:
      BTMaestro.print("p1");
      // write your action code here
      // group = GROUP_X; <-- or jump to another group X for composite commands
      break;
    case G1_BAJAR_PERSIANA:
      BTMaestro.print("p2");
      // write your action code here
      // group = GROUP_X; <-- or jump to another group X for composite commands
      break;
    case G1_ENCENDER_CALEFACCION:
      BTMaestro.print("r2");
      // write your action code here
      // group = GROUP_X; <-- or jump to another group X for composite commands
      break;
    case G1_ALARMA:
      BTMaestro.print("r3");
      // write your action code here
      // group = GROUP_X; <-- or jump to another group X for composite commands
      break;
  }

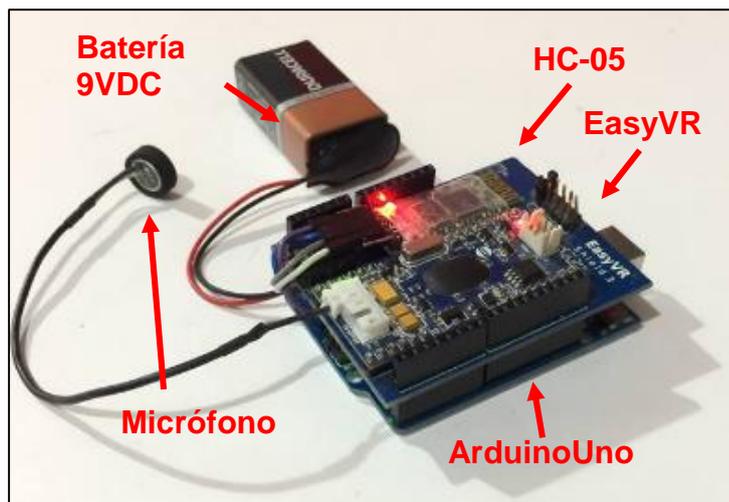
```

```
case G1_DETENER:  
  BTMaestro.print("t");  
  // write your action code here  
  // group = GROUP_X; <-- or jump to another group X for composite commands  
  break;  
}  
break;  
...
```

Fuente: Elaboración propia.

Finalmente, en la figura 96 se muestra la integración del módulo EasyVR, Arduino Uno, Módulo Emisor Bluetooth HC-05 y su respectiva fuente de alimentación conformando el Sistema de Control por Voz.

Figura 96. Integración del SCV.



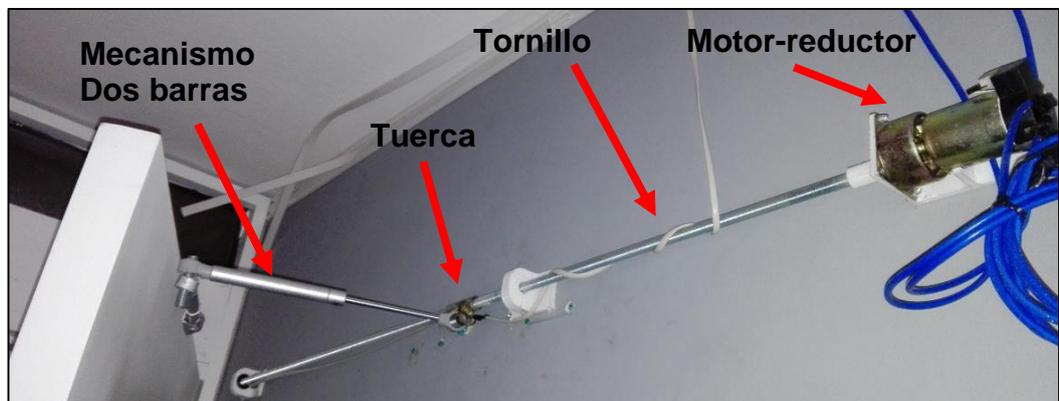
Fuente: Elaboración propia.

3.4 Implementación del Sistema Domótico.

3.4.1 Implementación de la puerta de ingreso y cortina.

La implementación del mecanismo de tornillo-tuerca, mecanismo de dos barras, motor-reductor y la puerta es realizada como se muestra en la figura 97. Primero, se empotró el mecanismo tornillo-tuerca en la pared adyacente con el motor-reductor para luego ensamblar el mecanismo de dos eslabones en el marco superior de la puerta. Los cables del motor se dirigieron hacia el tablero eléctrico para conectarse con su circuito de electrónica de potencia tal como se muestra en la figura 97.

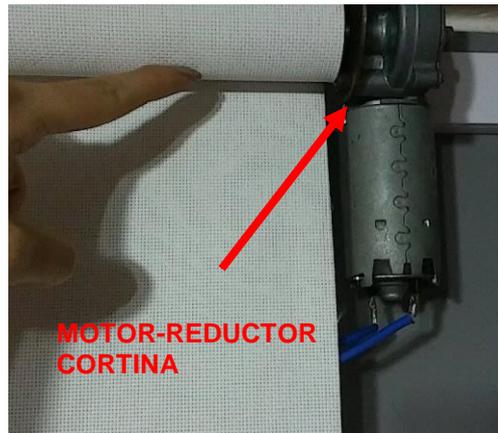
Figura 97. Instalación del motor paso a paso a la puerta.



Fuente: Elaboración propia.

Por otro lado, la unión del motor-reductor y la cortina es tal como se muestra en la figura 98. Primero, se empotró a cortina en el marco superior de la ventana, para luego acoplar el eje del motor-reductor por uno de sus extremos. Los cables del motor se dirigieron hacia el tablero eléctrico para conectarse con su circuito de electrónica de potencia.

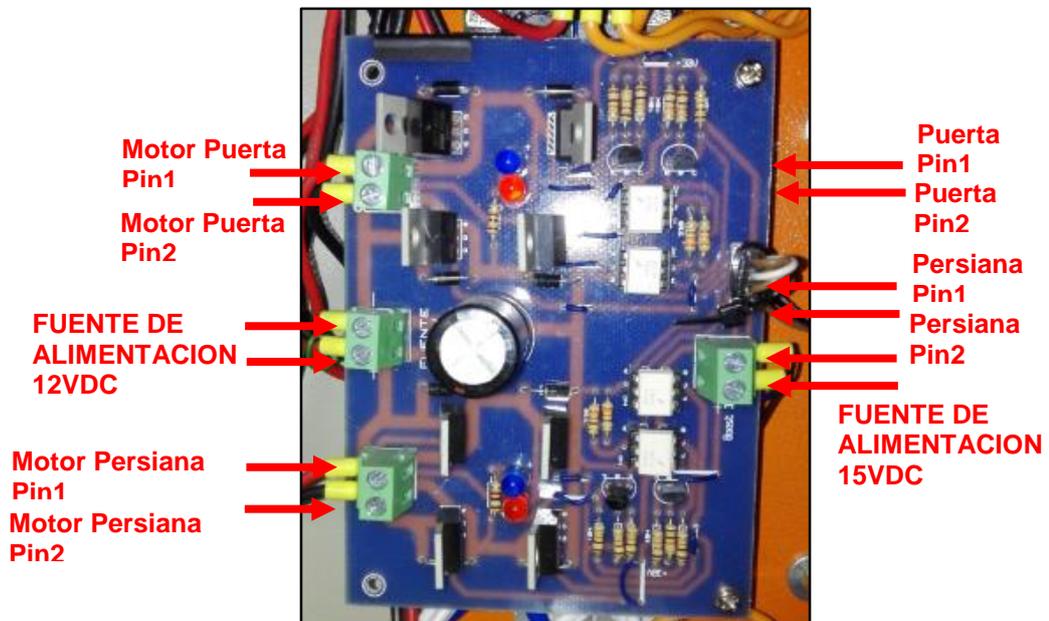
Figura 98. Instalación motor-reductor y cortina.



Fuente: Elaboración propia.

Es válido mencionar, que las etapas de potencia del motor-reductor de la puerta y motor-reductor de la cortina son las mismas. Por ello, se ha considerado implementarlas en un mismo circuito PCB tal como se muestra en la figura 99.

Figura 99. Circuito PCB de los motor-reductores de la puerta y cortina.

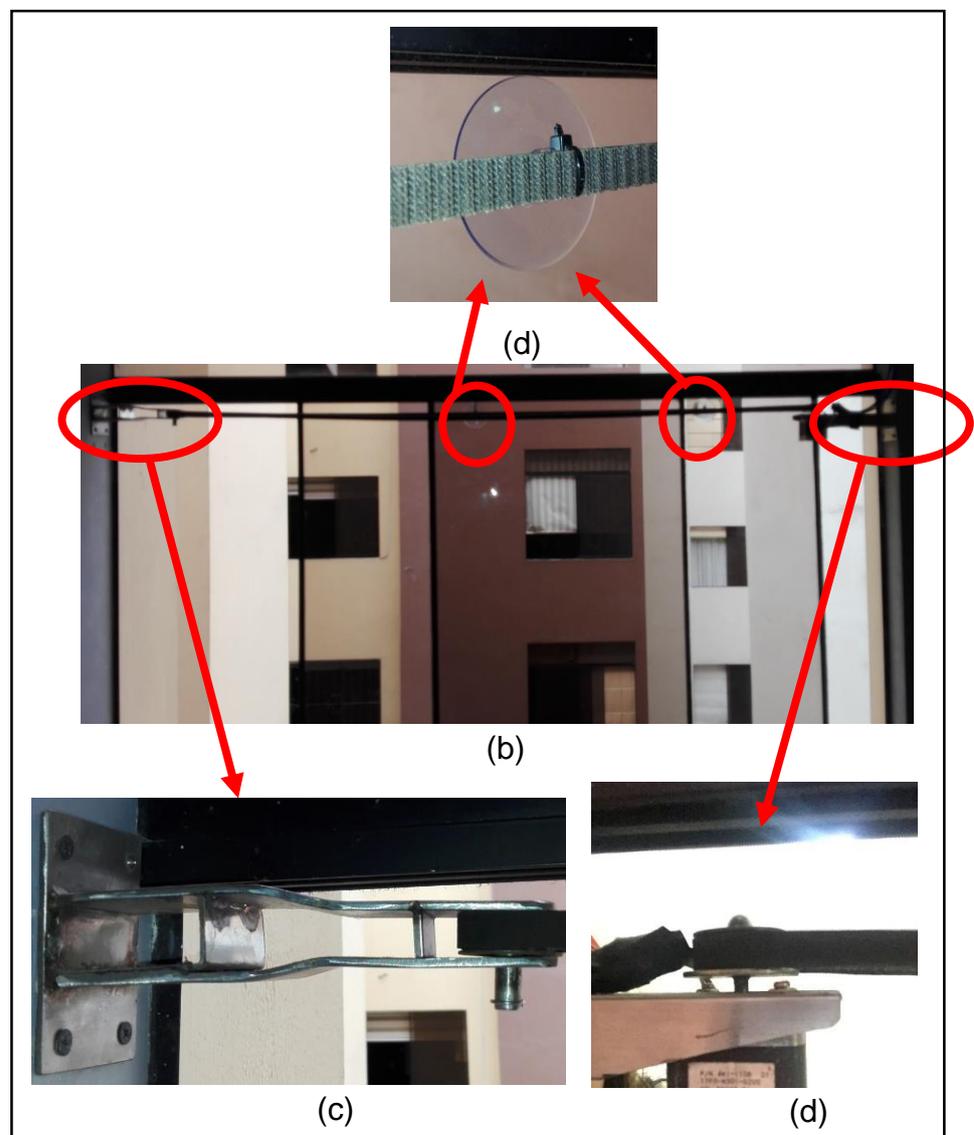


Fuente: Elaboración propia.

3.4.2 Implementación de la ventana.

Con una ventosa de succión de 40 mm de diámetro, que sirve para mover materiales con superficies lisas como vidrios (figura 100a); una correa dentada de marca Optibelt modelo OMEGA 2525 5M, que se extiende a todo el ancho de la ventana (figura 100b), y dos poleas de 30 mm de diámetro (figura 100c y d), se realiza la implementación del mecanismo de la ventana.

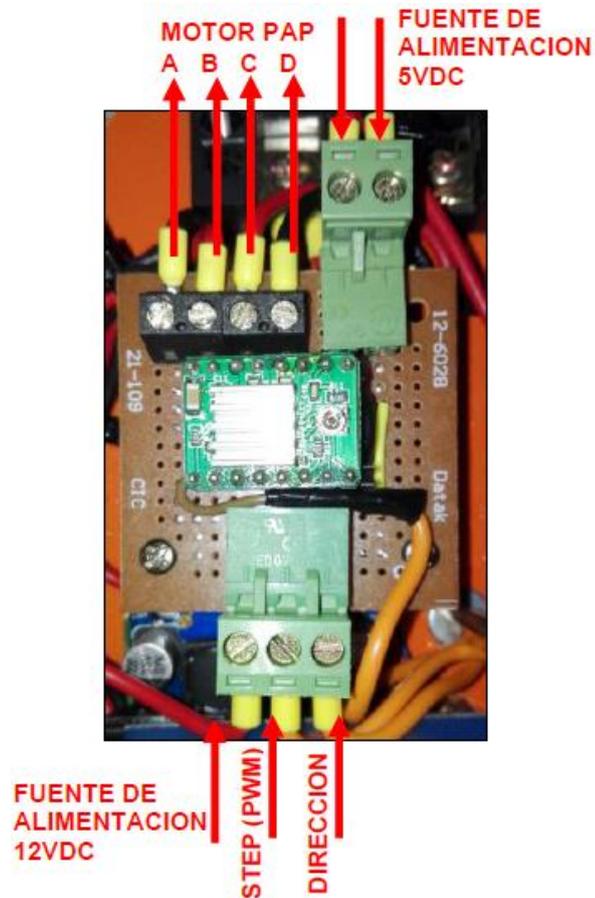
Figura 100. Mecanismo para el movimiento de la ventana.



Fuente: Elaboración propia.

Es válido mencionar, que la etapa de potencia de la ventana se ubica en el tablero de control tal como se muestra figura 101.

Figura 101. Implementación del Driver A999.



Fuente: Elaboración propia.

3.4.2 Implementación de la iluminación, alarma y calefacción.

La implementación de la iluminación se realizó en el cuarto del usuario reemplazando la luminario anterior por un foco ahorrador tal como se muestra en la figura 102. Con respecto a la alarma, se decidió colocarla dentro del tablero eléctrico tal como se muestra en la figura

103. Por otro lado, la calefacción se instaló debajo del tablero eléctrico cercano a la cama del usuario, tal como se muestra en la figura 104.

Figura 102. Instalación de iluminación.



Fuente: Elaboración propia.

Figura 103. Instalación de alarma.



Fuente: Elaboración propia.

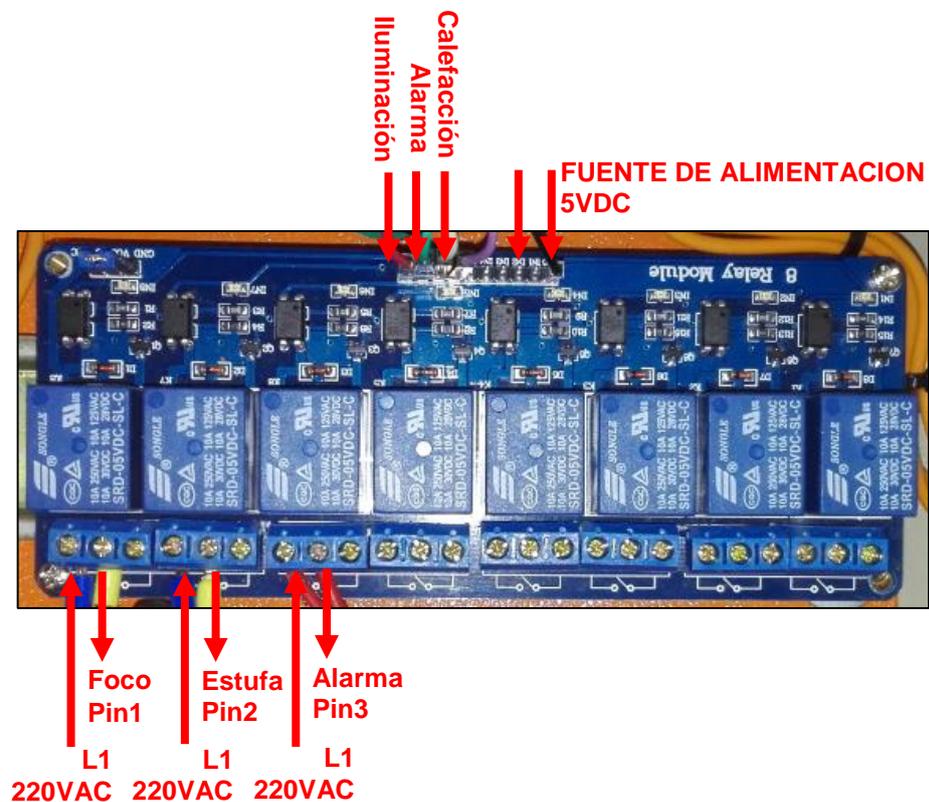
Figura 104. Instalación de calefacción.



Fuente: Elaboración propia.

Es válido mencionar, que las etapas de potencia de la iluminación, la alarma y la calefacción se encuentran en un módulo de ocho relés de 5 VDC y 10 A dejando libres cinco relés para futuras aplicaciones, como se muestra en la figura 105.

Figura 105. Módulo de ocho relés para calefacción.



Fuente: Elaboración propia.

3.4.3 Implementación de la cámara de videovigilancia.

La cámara DCS-930L de la marca D-link se instaló en una esquina del cuarto del usuario con la finalidad de que pueda captar un mayor rango de visión del cuarto, tal como se muestra en la figura 106.

Figura 106. Cámara IP DCS-930L.



Fuente: Elaboración propia.

3.5 Implementación de las Interfaces de usuario.

Se implementó un tablero eléctrico con el objetivo de proteger y juntar las tarjetas electrónicas de los sistemas de control central (Arduino ATmega; etapa de potencia de puerta, cortina, ventana, alarma, calefacción e iluminación; llave

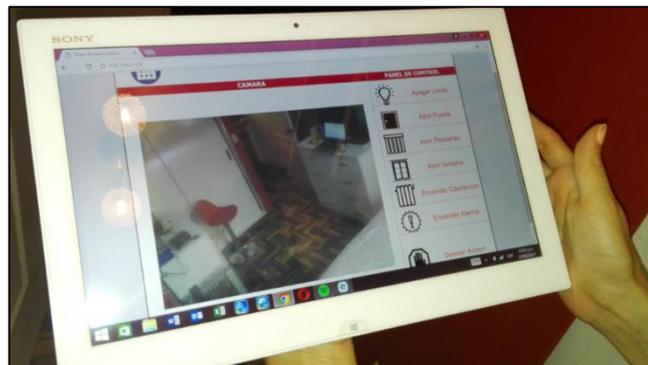
térmica bipolar y fuente de poder), sistema control por internet (Raspberry Pi) y sistema de control por voz (Modulo Bluetooth HC-06 receptor) tal se muestra en la figura 107. Por otro lado, se implementó la página web del sistema control por internet con IP estática 192.168.0.122 tal como se muestra en la figura 108. Por último, se implementó el sistema de control por voz mediante el módulo EasyVR, ArduinoUno, Modulo Bluetooth HC-06 emisor y el micrófono tal como se muestra en la figura 109.

Figura 107. Implementación del tablero electrico.



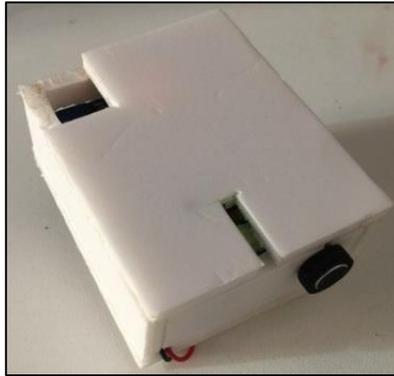
Fuente: Elaboración propia.

Figura 108. Implementación de la página web del SCI.



Fuente: Elaboración propia.

Figura 109. Implementación del módulo de voz SCV.



Fuente: Elaboración propia.

CAPITULO IV: PRUEBAS Y RESULTADOS

La presente investigación tiene como objetivo mejorar la calidad de vida de usuarios parapléjicos con el diseño e implementación de un sistema domótico controlado por Internet y Voz. Luego de haber diseñado el sistema, a continuación se describe las pruebas a las que se le sometió. En el primer apartado se describe las pruebas sin el voluntario con discapacidad con el fin de verificar aspectos técnicos de la misma. Por último, se describe las pruebas realizadas con el usuario parapléjico para comprobar su adaptación con el sistema.

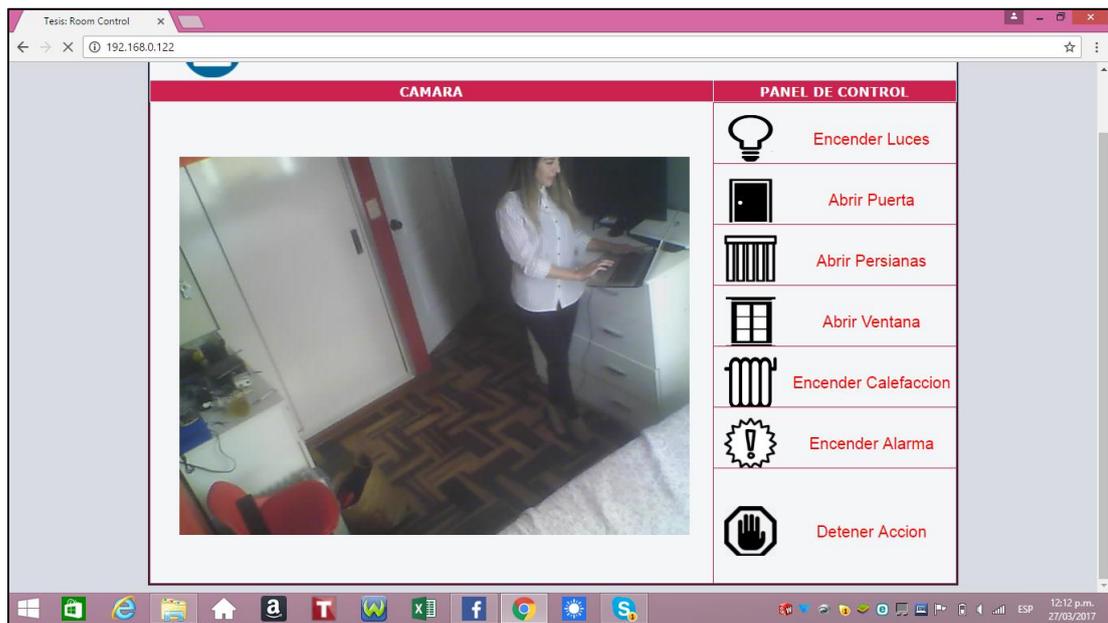
4.1 Pruebas sin usuario con paraplejía.

Estas pruebas fueron, básicamente, la integración del Sistemas de Control Central con los Sistemas de Control por Internet y Sistemas de Control por Voz.

Integración de Sistema de Control Central y Sistema de Control por Internet

Primero se estableció conexión con el servidor web a través del navegador Chrome y la IP 192.168.0.122, tal como se muestra en la figura 110. De esta manera, se comprobó el funcionamiento de la cámara D-Link y la página <index.php>

Figura 110. Pruebas del Sistema de Control por Internet.



Fuente: Elaboración propia.

Estas imágenes serán observadas por el usuario parapléjico teniendo la posibilidad de activar/desactivar los actuadores del sistema domótico.

Integración de Sistema de Control Central y Sistema de Control por Voz

Para la integración de esta prueba, se comunicó el módulo de voz EasyVR a través de los Modulos Bluetooth HC-05 emisor y receptor. No se necesitó ni un componente externo ya que estos mismos tienen la comunicación integrada en sus tarjetas. El módulo de voz se muestra en la figura 111(a). Las pruebas de la integración Sistema de Control Central y Sistemas de control por voz se muestran en la figura 111(b).

Figura 111. Pruebas del Sistema de Control por Voz.



(a)

(b)

Fuente: Elaboración propia.

4.2 Pruebas con el usuario parapléjico.

En este apartado se describe el grado de aceptación que el sistema recibió de un usuario con paraplejía. Antes, se hará una introducción sobre la historia del usuario voluntaria, su calidad de vida y limitaciones; para luego describir cómo se realizaron las pruebas.

Acerca del Usuario con Paraplejía.

Se trata de la Sra. Hermelinda Morales Silva de 86 años que sufre de distrofia muscular sin sensibilidad ni movimiento desde la cadera hacia abajo. Le aquejan algunas molestias como: adormecimiento por estar echada por largos periodos, dolores del hombro y cuello, molestias por la lenta digestión y artrosis. A pesar de todo esto, la Sra. Francisca sigue adelante con la ayuda de su familia e hijas.

Estas pruebas se realizaron el lunes 27 de Marzo del 2017, en su casa ubicada en el distrito de Surco. En la figura 112 se muestra a la Sra. Hermelinda sentada en un sillón de su casa.

Figura 112. La Sra. Hermelinda Morales antes de las pruebas.



Fuente: Elaboración propia.

Antes de iniciar, realizamos una explicación breve sobre cómo funcionaba nuestro sistema y sobre las pruebas a realizar, como se muestra en la figura 113.

Figura 113. Presentación del sistema al voluntario.



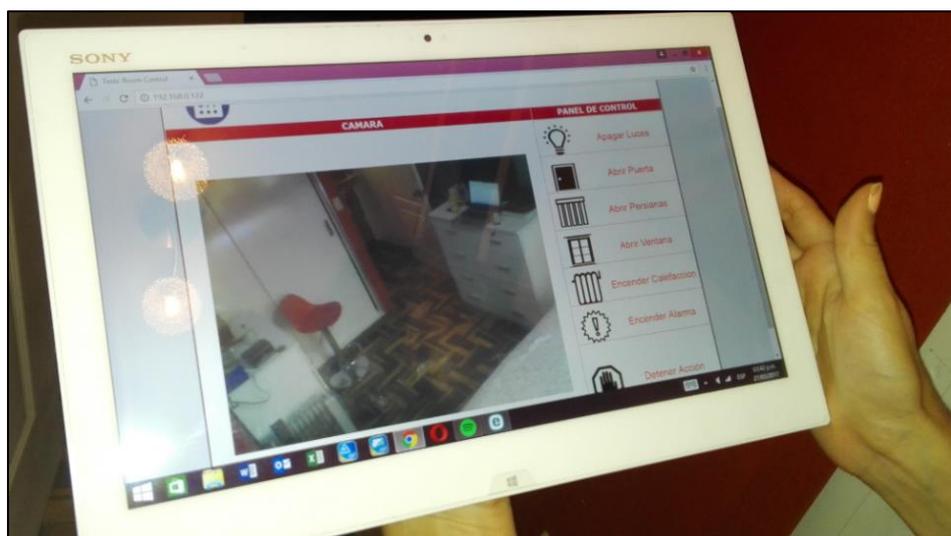
Fuente: Elaboración propia.

4.2.1 Pruebas 1: Control por Internet de la habitación domótica.

El objetivo de esta prueba fue demostrar que es posible controlar la habitación domótica a través de una página web. La Sra. Hermelinda Morales no estaba acostumbrada a usar una computadora de escritorio o laptop, sin embargo tiene una Tablet de marca SONY de 9,4" de pantalla y de 570 gr de peso; por donde se conecta a Skype regularmente. La ventaja de las tablet's es que poseen una pantalla táctil lo que hace fácil su manejo.

Primero, se conectó la Tablet a la red Wifi del hogar para luego abrir un navegador web y colocar la IP del web server: 192.168.0.122. Al hacer esto, automáticamente se puede observar la página web <index.php> con los botones de actividades y las imágenes de la cámara D-Link tal como se muestra en la figura 114.

Figura 114. Prueba 1: Conexión con el página web principal

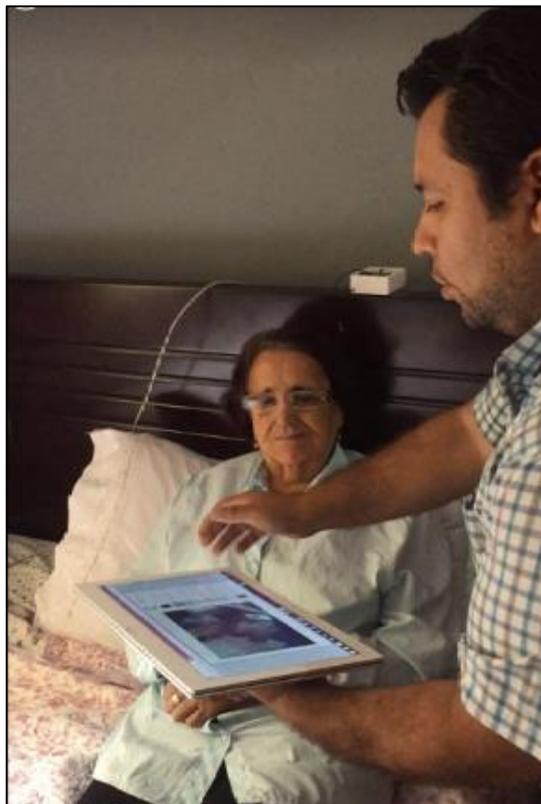


Fuente: Elaboración propia.

Así, nuestra voluntaria logro abrir la puerta, cerrar la persiana y la ventana, prender las luces, la calefacción y la alarma pulsando en los botones respectivos; y al mismo tiempo visualizaba lo que ocurría en la habitación desde las imágenes de la cámara D-Link. Por otro lado, no existió un retardo apreciable en la respuesta del sistema.

Esta prueba resulto como un entrenamiento para que la Sra. Hermelinda Morales se adapte a la página web y pueda utilizar sus funcionalidades. En la figura 115, podemos observarla con su Tablet durante la prueba.

Figura 115. Prueba 1: Control por Internet de la habitación domótica.



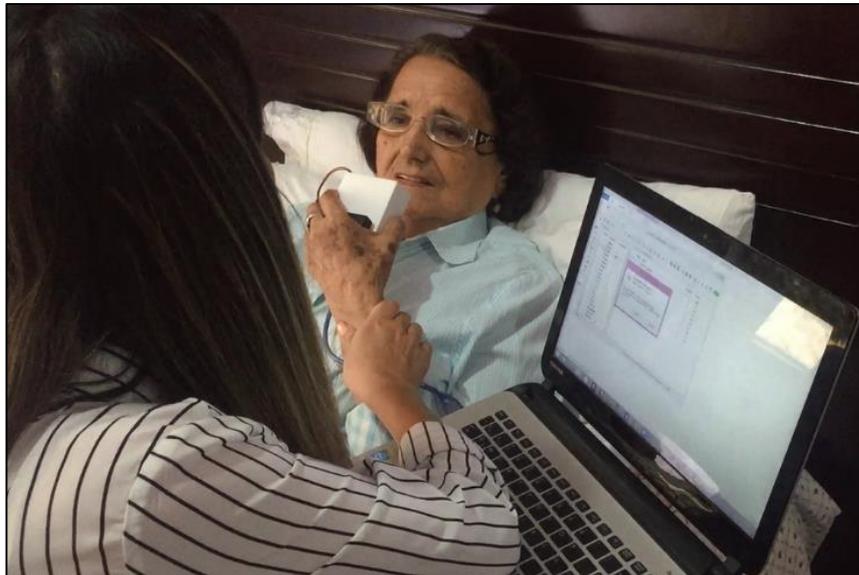
Fuente: Elaboración propia.

4.2.2 Pruebas 2: Control por Voz de la habitación domótica.

El objetivo de esta prueba fue demostrar que es posible controlar la habitación por comandos de voz. Primero, se tuvo que realizar el entrenamiento de los comandos a través del software EasyVR Commander lo cual tomo unos 8 minutos.

Del mismo modo, nuestra voluntaria logro activar o desactivar todos los actuadores de la habitación. En la figura 116 se observa el desarrollo de esta prueba.

Figura 116. Prueba 2: Control por Voz de la habitación.



Fuente: Elaboración propia.

4.2.3 Pruebas 3: Cambio de grupo de los comandos de voz.

Luego, de los resultados de la prueba anterior se decide mejorar el rendimiento del sistema porque primero se decidió ubicar los comandos de voz en el grupo 2 en el EasyVR Commander pero se obtuvo un retardo de 3 segundos en la respuesta del sistema. Sin embargo, existe una prioridad de lectura en el orden de los grupos tal como se señala en el manual de usuario: “la función de reconocimiento del Easy VR trabaja en un solo grupo al mismo tiempo (...) y lee de arriba hacia abajo todos los grupos” (RoboTech, 2010, p.58). Sobre la base de lo mencionado se cambió los comandos al grupo 1 y se disminuyó el retardo mejorando el tiempo de respuesta del sistema. En la figura 117 se muestra el desarrollo de esta prueba.

Figura 117. Prueba 3: Cambio de grupo en cambios de voz.



Fuente: Elaboración propia.

4.2.4 Entrevista a la voluntaria.

Luego de las pruebas se realizó una entrevista a la voluntaria para conocer su satisfacción con el sistema diseñado, sugerencias y opiniones. Esta entrevista tuvo una duración de 10 minutos. A continuación se muestra la entrevista transcrita:

Entrevistador: ¿Cuál de los dos módulos le pareció mejor para controlar la habitación: el control por internet o por voz?

Voluntaria: En mi caso personal, me resultó mejor el control por página web porque no tengo que recordar los comandos de voz. Me es más práctico usar mi Tablet que el micrófono del control por voz.

Entrevistador: ¿El diseño de la página web le parece amigable y entendible?

Voluntaria: Sí, me parece que el diseño y el tamaño de los botones son cómodos para mí.

Entrevistador: ¿Cree usted que este sistema le serviría en su vida diaria?

Voluntaria: Sí porque ya no tendría la necesidad de moverme de la cama para controlar mi cuarto ni incomodar a algún familiar para efectuar mis cosas.

Entrevistador: ¿Cree usted que son este sistema le ayudaría a desenvolverse sola?

Voluntaria: Sí podría desenvolverme sola, es funcional y con todos los adelantos que hay, sí por su puesto. Podría desenvolverme por mis propios medios.

Entrevistador: Finalmente, ¿Tienes algún comentario o algo que sugerir?

Voluntaria: Tal vez, sí se pudiese implementar este mecanismo en otras partes de mi casa.

4.2.5 Resultados de las pruebas con control por internet.

1. Notamos que nuestra voluntaria se sentía motivada por aprender el funcionamiento del sistema porque era una nueva tecnología para ella.
2. Las pruebas mostraron que realmente es posible controlar una habitación domótica por internet. Al mismo tiempo, mejorar la calidad de vida de una persona con paraplejia en la realización de sus tareas cotidianas.
3. La prueba de control por internet se realizó de manera exitosa sin retardo desde que se presionaba el botón hasta que se comenzaba a ejecutar la acción.
4. El tamaño de los botones y el diseño de la página web fue el amigable para la usuaria voluntaria.

4.2.6 Resultados de las pruebas con control por voz.

1. Notamos que nuestra voluntaria se sentía motivada por aprender el funcionamiento del sistema porque era una nueva tecnología para ella.
2. Las pruebas mostraron que realmente es posible controlar una habitación domótica por voz. Al mismo tiempo, mejorar la calidad de vida de una persona con paraplejia en la realización de sus tareas cotidianas.
3. Durante la prueba de control por voz, al principio la usuaria no pronunciaba correctamente los comandos de voz debido a la curva de aprendizaje pero luego se adaptó al sistema.
4. Este sistema pudo utilizarse en la cama, desde una silla de ruedas y en cualquier parte de la habitación porque que el módulo de voz es portátil e inalámbrico.

CONCLUSIONES

La presente tesis tuvo como objetivo diseñar e implementar un sistema domótico para aplicaciones en usuarios parapléjicos mediante control remoto a través de internet y reconocimiento de voz. A partir de ello, se genera las siguientes conclusiones y recomendaciones para investigaciones futuras:

1. Es posible diseñar e implementar un sistema domótico para aplicaciones en usuarios parapléjicos mediante control remoto a través de internet y reconocimiento de voz.
2. Sí es posible diseñar e implementar una arquitectura domótica que considere las principales necesidades que tiene una usuaria con paraplejia.
3. Sí es posible diseñar e implementar un sistema de control remoto a través de internet para un sistema domótico.
4. Se verifica que es posible diseñar e implementar un sistema de control por voz para un sistema domótico.
5. La usuaria con paraplejia se adaptó a los sistemas de control del sistema domótico.

RECOMENDACIONES

1. Se recomienda mejorar el sistema domótico dado que sus componentes son de programación libre.
2. Agregar nuevas tecnologías al Sistema de Control Central para poder aumentar el alcance del sistema domótico.
3. Adaptar y ampliar esta aplicación a otras necesidades que puedan tener los usuarios con discapacidad parapléjica.
4. Adaptar esta aplicación a personas con otro tipo de discapacidad como visual o auditiva.
5. Esta tesis fomenta el desarrollo de investigaciones aplicadas en el campo de tecnologías para personas con discapacidad. Se recomienda utilizar estos conocimientos para futuros desarrollos.

REFERENCIAS BIBLIOGRAFICAS

Acosta, C.; Tipán, C. (2015). *Implementación de una consola domótica residencial con control local y remoto a través del internet*. Tesis pregrado, Escuela Politécnica Nacional, Quito.

Alcobendas, M. (2011). *Efectividad del entrenamiento de la marcha mediante el sistema LOKOMAT en pacientes agudos con lesión modular incompleta*. Tesis Doctoral. Universidad Complutense de Madrid, España.

Alta, M.; Guijarro, J. (2015). *Diseño e implementación de un control por comandos de voz a una silla de ruedas*. Tesis pregrado. Facultad de Ingeniería en Ciencias Aplicadas, Universidad Técnica del Norte, Ecuador.

Alvarez, C.; Holguin, D.; Serrano, E. (2007). *Diseño de una Instalación Domótica en un Condominio para el Control de Seguridad e Iluminación mediante la Tecnología LonWorks*. Tesis pregrado. Escuela Superior Politecnica del Litoral, Guayaquil, Ecuador.

Atahualpa, J. (2014). *Estudio de factibilidad técnica para el diseño de un laboratorio de domótica en la facultad de educación técnica para el desarrollo*. Tesis pregrado. UNIVERSIDAD CATÓLICA DE SANTIAGO DE GUAYAQUIL, Guayaquil, Ecuador.

Banco Mundial (2011): *Informe Mundial sobre La Discapacidad*. Organización Mundial de la Salud. ISBN 978-92-4-356418.

Barbecho, J.; Fernandez, W. (2013). *Aplicaciones multimedia para control de un sistema domótico orientado a personas con discapacidad*. Tesis pregrado. UNIVERSIDAD DE CUENCA, Cuenca, Ecuador.

Builders Hardware Manufacturers Association – BHMA (2002): *BHMA A156.10 Power Assist and Low Energy Power Operated Doors*. New York, USA.

Cabrera, R.; Chavez, P. (2013). *Diseño e Implementación de un robot humanoide asistencial controlado por computador para aplicaciones en pacientes parapléjicos*. Tesis pregrado. Universidad Ricardo Palma, Lima, Perú.

Canelo, R.; Vera, M. (2016). *Diseño e implementación de un robot híbrido teleoperado con simulink y un web server embebido en raspberry pi para el laboratorio de ingeniería mecatrónica de la Universidad Ricardo Palma*. Tesis pregrado. Universidad Ricardo Palma, Lima, Perú.

Casas, R.; Roy, A. (2005). *Informe Técnico sobre Domótica de Viviendas y Edificios de la Milla Digital de Zaragoza*. Instituto de Investigación en Ingeniería de Aragón, Universidad de Zaragoza, España.

Carranza, S.; Pelcastre, U.; Jardón, L. (2014). *Persiana automatizada autosustentable*. Universidad Nacional Autónoma de México, México.

Cetinkunt, S.; Cárdenas, L. (2007): *Mecatrónica*. México: Patria, 2007.

Dujmovic, M. (2006). Diseño de un entrenador de microcontroladores PIC para el laboratorio de microcontroladores del Instituto Universitario Jesús Obrero. Universidad Simón Bolívar. Proyecto de Grado. Venezuela.

Gallegos, C; Jarrín, J. (2012). *Diseño e implementación de un módulo didáctico para clasificación, empaqueo e inspección de huevos aplicando plc*. Tesis de pregrado. Escuela Superior Politécnica de Chimborazo. Ecuador.

González, A; García, J. (2013). *Alternativa de energía eléctrica para la iluminación en el edificio principal de ingeniería*. Tesis pregrado. Universidad Nacional Autónoma de México, México.

Guerra, F. (2013). *Diseño de un sistema de control domótico y video vigilancia supervisado por un teléfono móvil*. Tesis pregrado. PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ, Lima, Perú.

Harke, W. (2010). *Domótica para viviendas y edificios*. 1era Ed. Marcombo.

Heriquez, M. (2005). *Investigación y desarrollo sistema prototipo de asistencia domótica para personas con movilidad limitada*. Tesis pregrado. UNIVERSIDAD AUSTRAL DE CHILE, Puerto Montt, Chile.

Hernandez, J.; López, E.; Miranda, D.; Rosas, F.; Sanchez, R. (2014). *Caracterización del consumo de energía en viviendas para proponer y evaluar tecnologías ahorradoras*. Tesis Pregrado. Universidad Nacional Autónoma de México, México.

Huidobro, J. (2008). *Domótica: Edificios inteligentes*. 1era Ed. Prentice Hall.

Jardón, A. (2008): *Metodología de diseño de robots asistenciales. Aplicación al robot portátil ASIBOT*. Tesis Doctoral. Universidad Carlos III de Madrid, España.

Mendez, A. (2014). *Diseño e implementación de plataforma basada en microcontroladores pic para facilitar el estudio practico y la elaboración de proyectos en la carrera de ingeniería en telecomunicaciones*. Universidad Católica de Santiago de Guayaquil. Tesis pregrado. Ecuador.

- Miranda, G.; Villacrés, R.; Villamar, F. (2009). *Diseño de un sistema domótico aplicado a una clínica de hemodiálisis*. Tesis pregrado. Escuela Superior Politecnica del Litoral, Guayaquil, Ecuador.
- Montalvo, C. (2010). *Control y Monitoreo de Edificios para Personas con Discapacidad, Mediante El Uso De Tecnología Mobile, Web, Sms y Reconocimiento de Voz*. Tesis pregrado. Universidad Técnica del Norte, Ecuador.
- Pais, C. (2011): *Treino de marcha suspensa robotizada (LOKOMAT) em pacientes com lesao vertebro-medular incompleta*. Licenciatura em fisioterapia, Universidade Fernando Pessoa, Brasil.
- Pérez, V. (2010). *Contribución al diseño de sistemas domóticos y de entretenimiento utilizando hardware libre y software de código abierto*. Tesis de Maestría. INSTITUTO POLITÉCNICO NACIONAL, Tijuana, México.
- Pérez, Y. (2015). *Efectos de la paraplejia a largo plazo en la función cognitiva y motora de la mano*. Tesis Doctoral. Universidad De Castilla-La Mancha, España.

Pueyo, S (2012). Proyecto de instalación eléctrica y control domótico de una vivienda unifamiliar mediante protocolo KNX. Universidad de la Rioja. Proyecto fin de carrera. España.

Reyes, F.; Cid J.; Vargas E. (2013). *Mecatrónica: Control y Automatización*. México: Alfaomega, 2013.

Rios. B. (1985). *El licenciado en trabajo social y el incapacitado parapléjico, centro de rehabilitación y educación especial*. Tesis Licenciatura. Universidad de Veracruz, México.

RoboTech (2015). *User Manual EasyVR 3 Release 1.0.11*. Recuperado de: www.veear.eu

Rodriguez, M. (2014). *Sistema de control remoto para aplicaciones domóticas a través de internet*. Tesis pregrado. UNIVERSIDAD AUTÓNOMA DE MADRID, Madrid.

Rodriguez, W. (2012). *Sistema de control domótico utilizando una central ip pbx basado en software libre*. Tesis pregrado. PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ, Lima, Perú.

Roman, R. (2011). *Diseño de un sistema domótico para control de iluminación y monitoreo de consumo eléctrico*. Tesis pregrado. UNIVERSIDAD INDUSTRIAL DE SANTANDER, Bucaramanga, Colombia.

Tyler, H. (1981) Ed. *Manual práctico de cálculos de ingeniería*. Reverté. Barcelona, España.

Vera, G. (2005). *Diseño mecánico de un manipulador robótico asistente para controlar los movimientos de un laparoscopio en intervenciones quirúrgicas*, Tesis de Maestría dirigido por J. Ramos, Instituto Politécnico Nacional, México.

ANEXOS

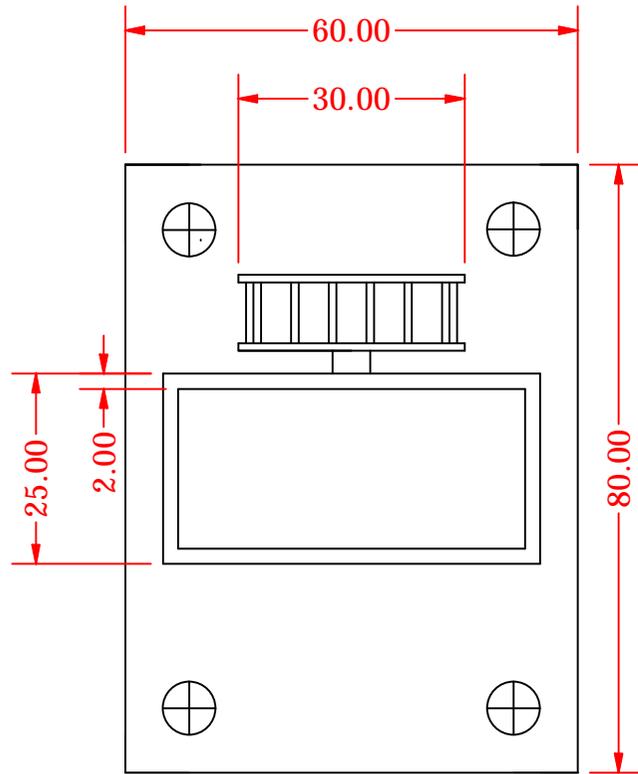
**Anexos Planos Mecánicos
Anexos Planos Eléctricos
Anexos Planos Electrónicos
Anexos de Programación
Anexos de Costos**

ANEXOS

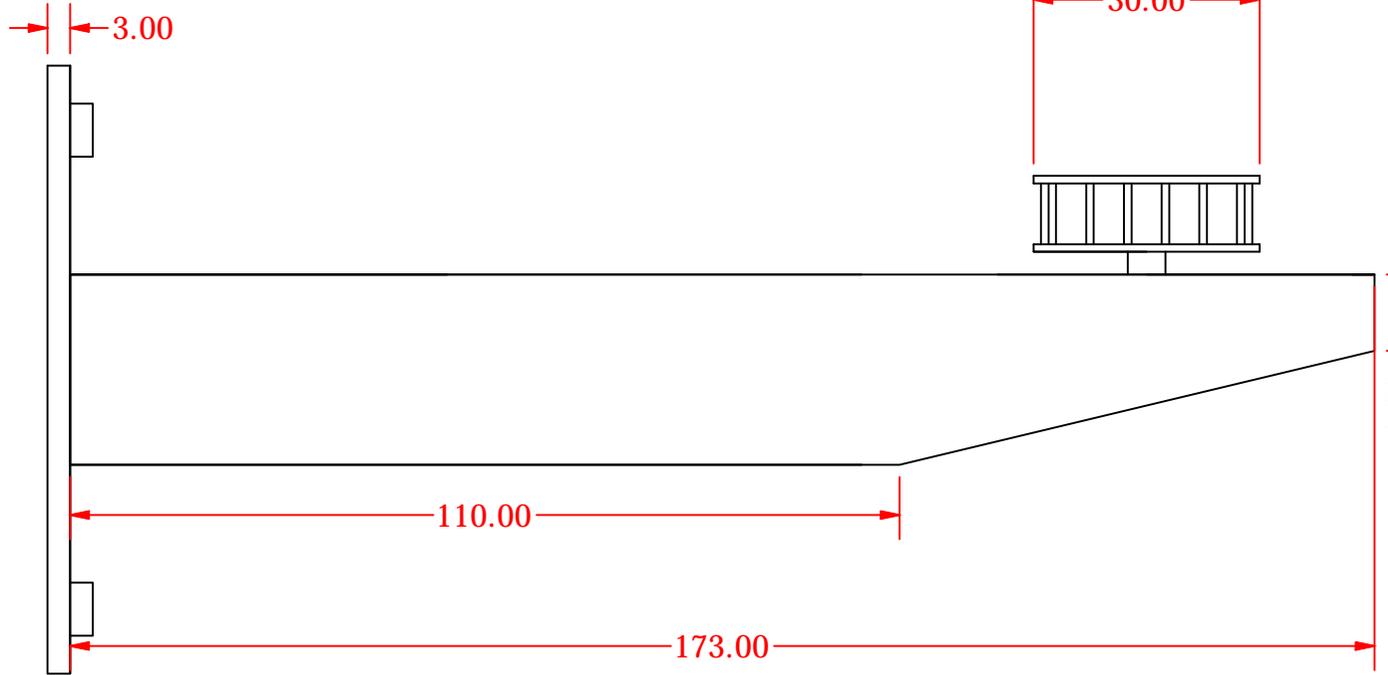
PLANOS

MECANICOS

SOPORTE I DE MOTOR PASO A PASO VENTANA



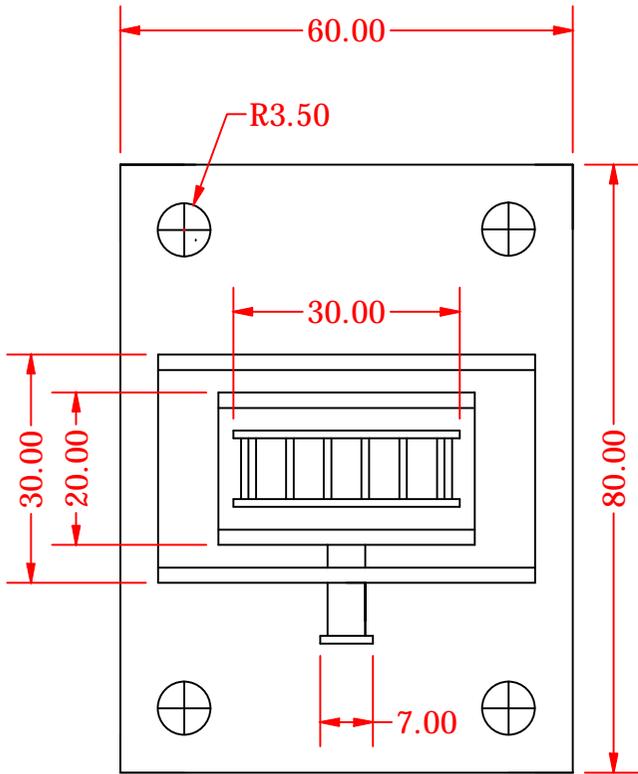
VISTA FRONTAL



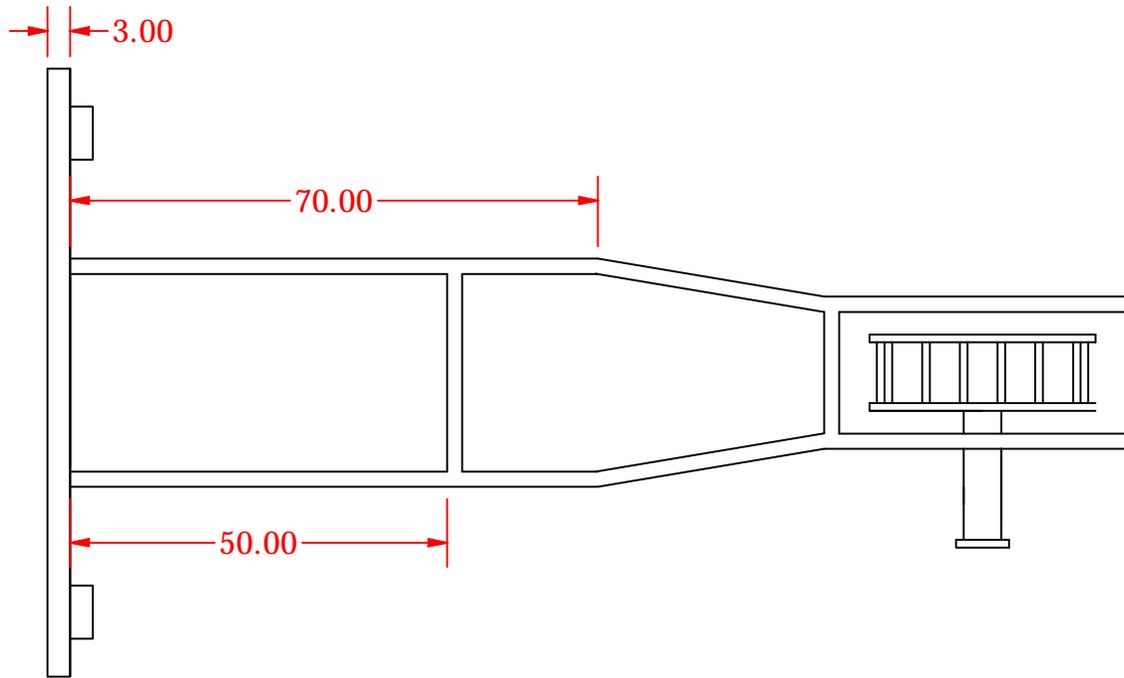
VISTA PERFIL

REFERENCIA:	SOPORTE I DE MOTOR PASO A PASO VENTANA	PLANO:	SD001
TESIS:	SISTEMA DOMOTICO PARA PACIENTES PARAPLEJICOS		
CANTIDAD:	1	MATERIAL:	
ASESOR:	FREDDY SOTELO VALER		FECHA: 17/03/2017
TESISTAS:	PATRICIA DIAZ JOSE YABAR		 Universidad Ricardo Palma <i>Formamos seres humanos para una cultura de paz</i>

SOPORTE II DE MOTOR PASO A PASO VENTANA



VISTA FRONTAL



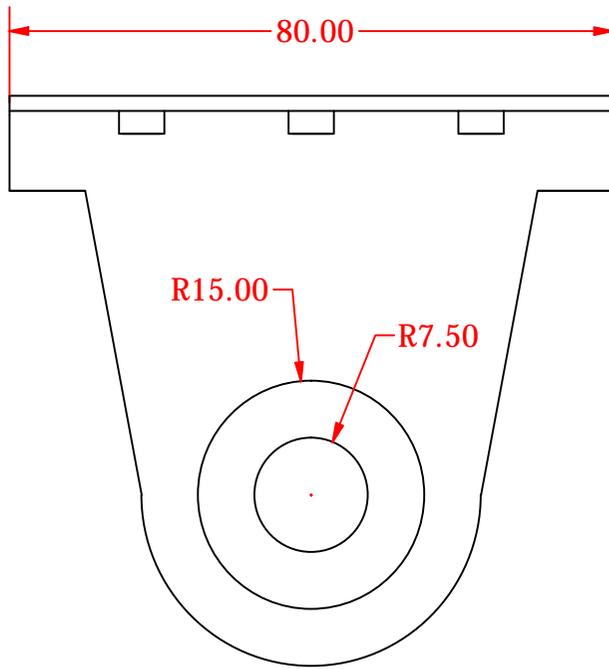
VISTA PERFIL

REFERENCIA:	SOPORTE II DE MOTOR PASO A PASO VENTANA	PLANO:	SD002
TESIS:	SISTEMA DOMOTICO PARA PACIENTES PARAPLEJICOS		
CANTIDAD:	1	MATERIAL:	
FECHA:	17/03/2017		
ASESOR:	FREDDY SOTELO VALER		
TESISTAS:	PATRICIA DIAZ JOSE YABAR		

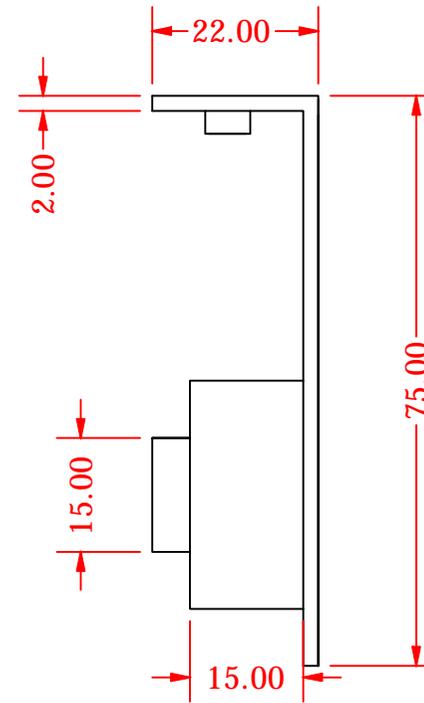


Universidad Ricardo Palma
Formamos seres humanos para una cultura de paz

SOPORTE DE MOTOR PASO A PASO PERSIANA



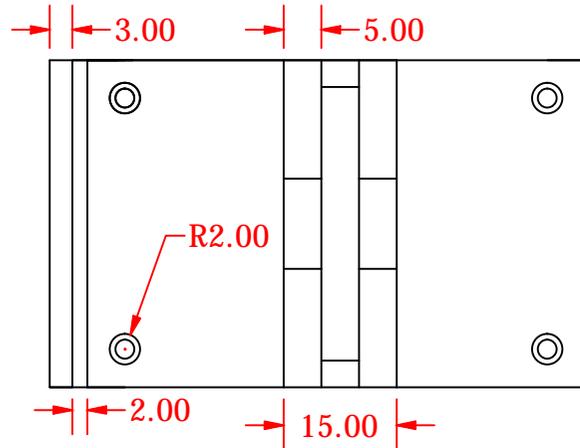
VISTA FRONTAL



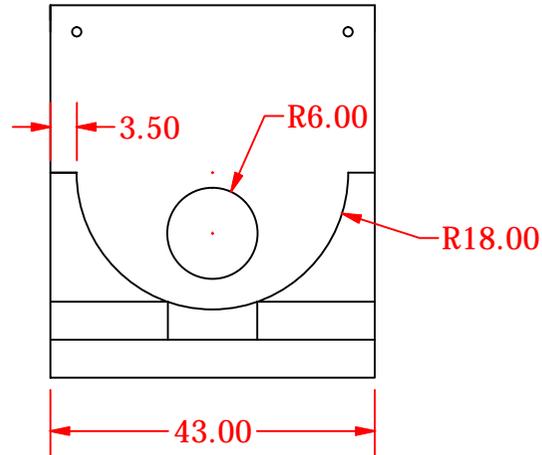
VISTA PERFIL

REFERENCIA:	SOPORTE DE MOTOR REDUCTOR PERSIANA	PLANO:	SD003
TESIS:	SISTEMA DOMOTICO PARA PACIENTES PARAPLEJICOS		
CANTIDAD:	1	MATERIAL:	
ASESOR:	FREDDY SOTELO VALER		
TESISTAS:	PATRICIA DIAZ JOSE YABAR		 Universidad Ricardo Palma <i>Formamos seres humanos para una cultura de paz</i>

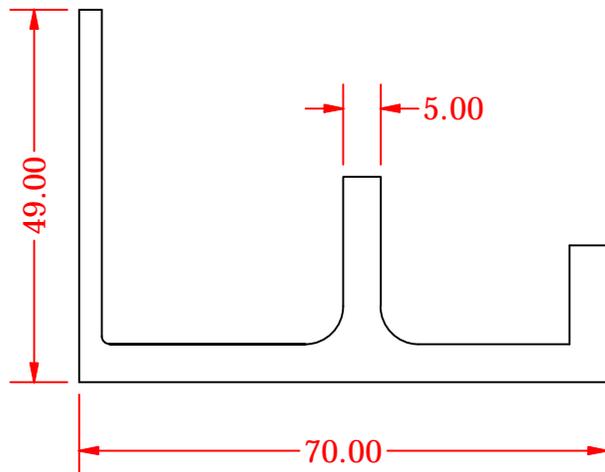
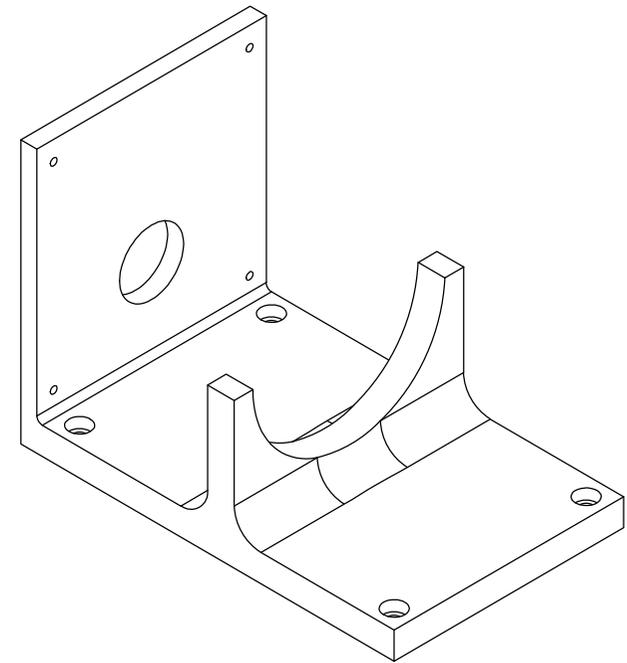
SOPORTE DE MOTOR PASO A PASO PUERTA



VISTA PLANTA



VISTA FRONTAL



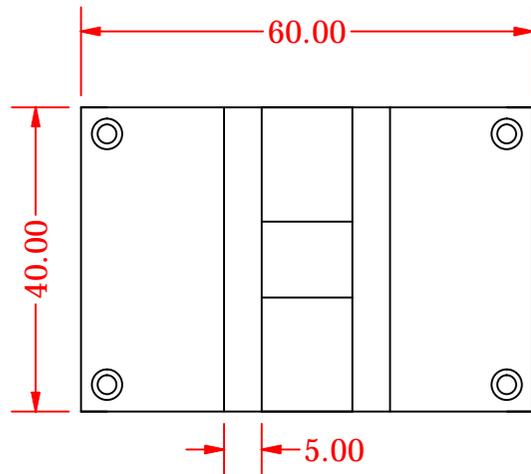
VISTA PERFIL

REFERENCIA:	SOPORTE DE MOTOR REDUCTOR PUERTA	PLANO:	SD004
TESIS:	SISTEMA DOMOTICO PARA PACIENTES PARAPLEJICOS		
CANTIDAD:	1	MATERIAL:	
FECHA:	17/03/2017		
ASESOR:	FREDDY SOTELO VALER		
TESISTAS:	PATRICIA DIAZ JOSE YABAR		

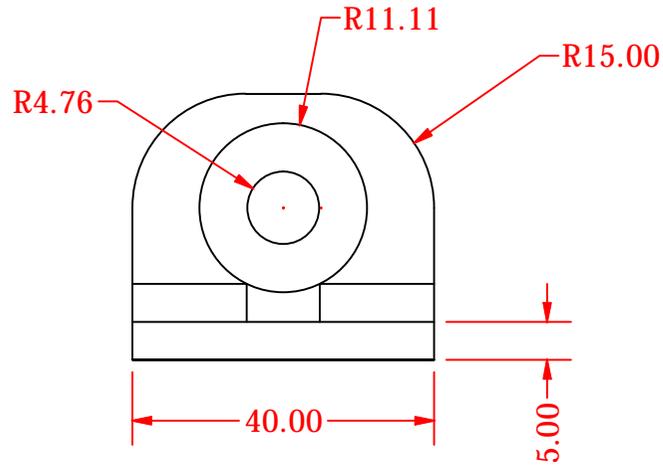


Universidad **Ricardo Palma**
Formamos seres humanos para una cultura de paz

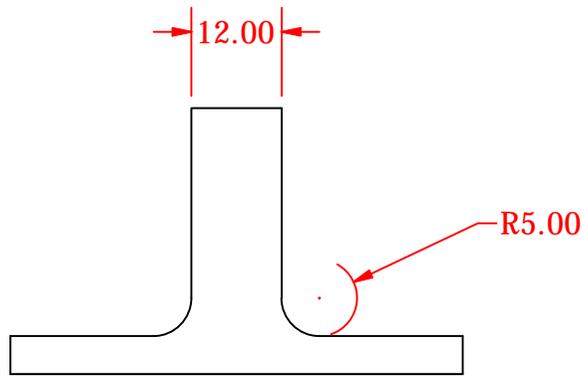
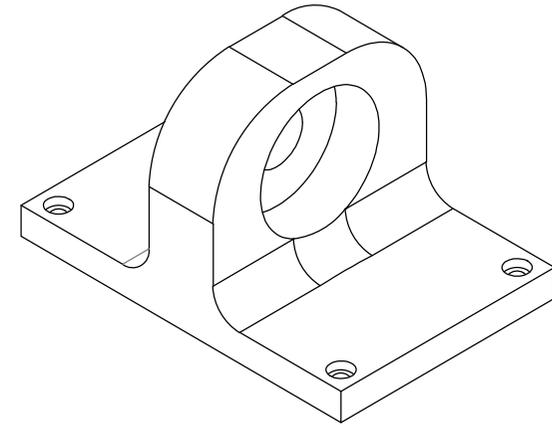
SOPORTE DE EJE PUERTA



VISTA PLANTA



VISTA FRONTAL



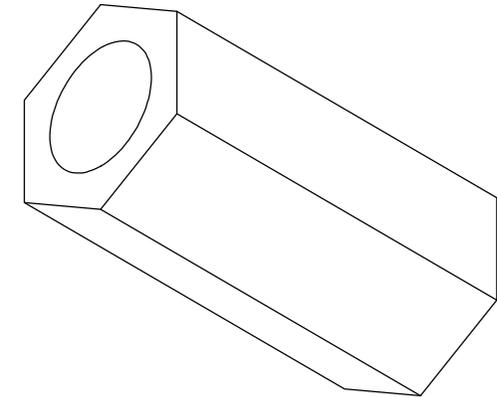
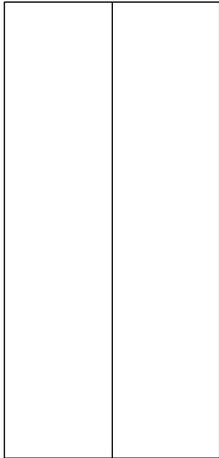
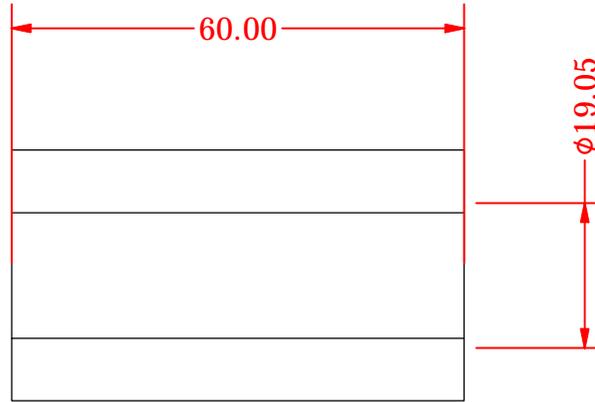
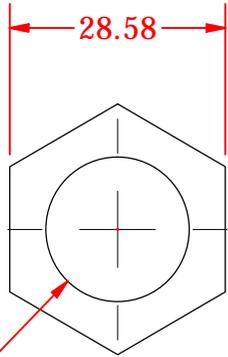
VISTA PERFIL

REFERENCIA:	SOPORTE DE EJE PUERTA	PLANO:	SD005
TESIS:	SISTEMA DOMOTICO PARA PACIENTES PARAPLEJICOS		
CANTIDAD:	1	MATERIAL:	
FECHA:	17/03/2017		
ASESOR:	FREDDY SOTELO VALER		
TESISTAS:	PATRICIA DIAZ JOSE YABAR		



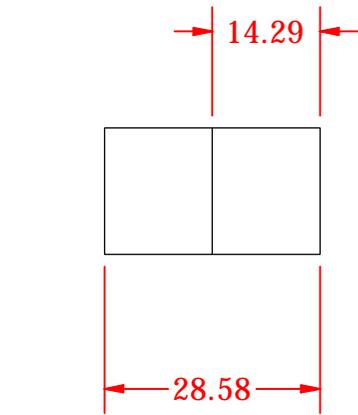
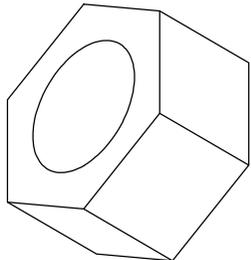
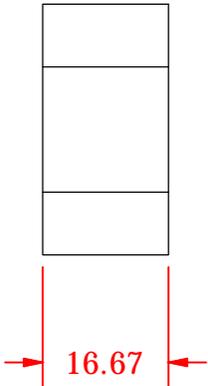
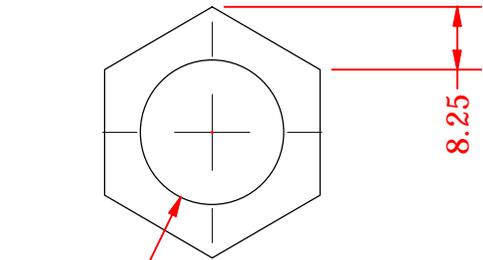
Universidad **Ricardo Palma**
Formamos seres humanos para una cultura de paz

SOPORTE TUERCA LARGA



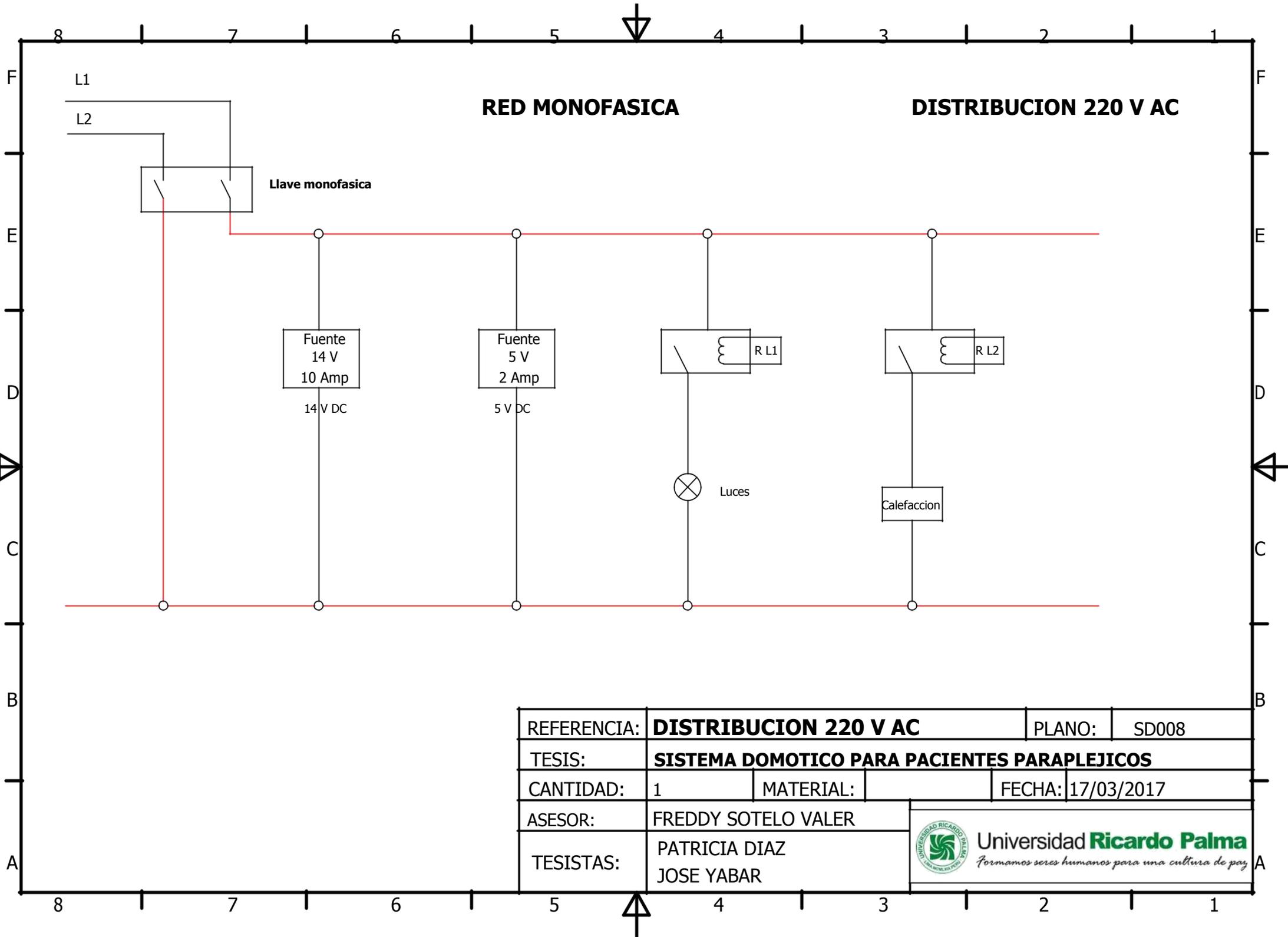
REFERENCIA:	SOPORTE TUERCA LARGA		PLANO:	SD006
TESIS:	SISTEMA DOMOTICO PARA PACIENTES PARAPLEJICOS			
CANTIDAD:	1	MATERIAL:		FECHA: 17/03/2017
ASESOR:	FREDDY SOTELO VALER		 Universidad Ricardo Palma <i>Formamos seres humanos para una cultura de paz</i>	
TESISTAS:	PATRICIA DIAZ JOSE YABAR			

SOPORTE TUERCA

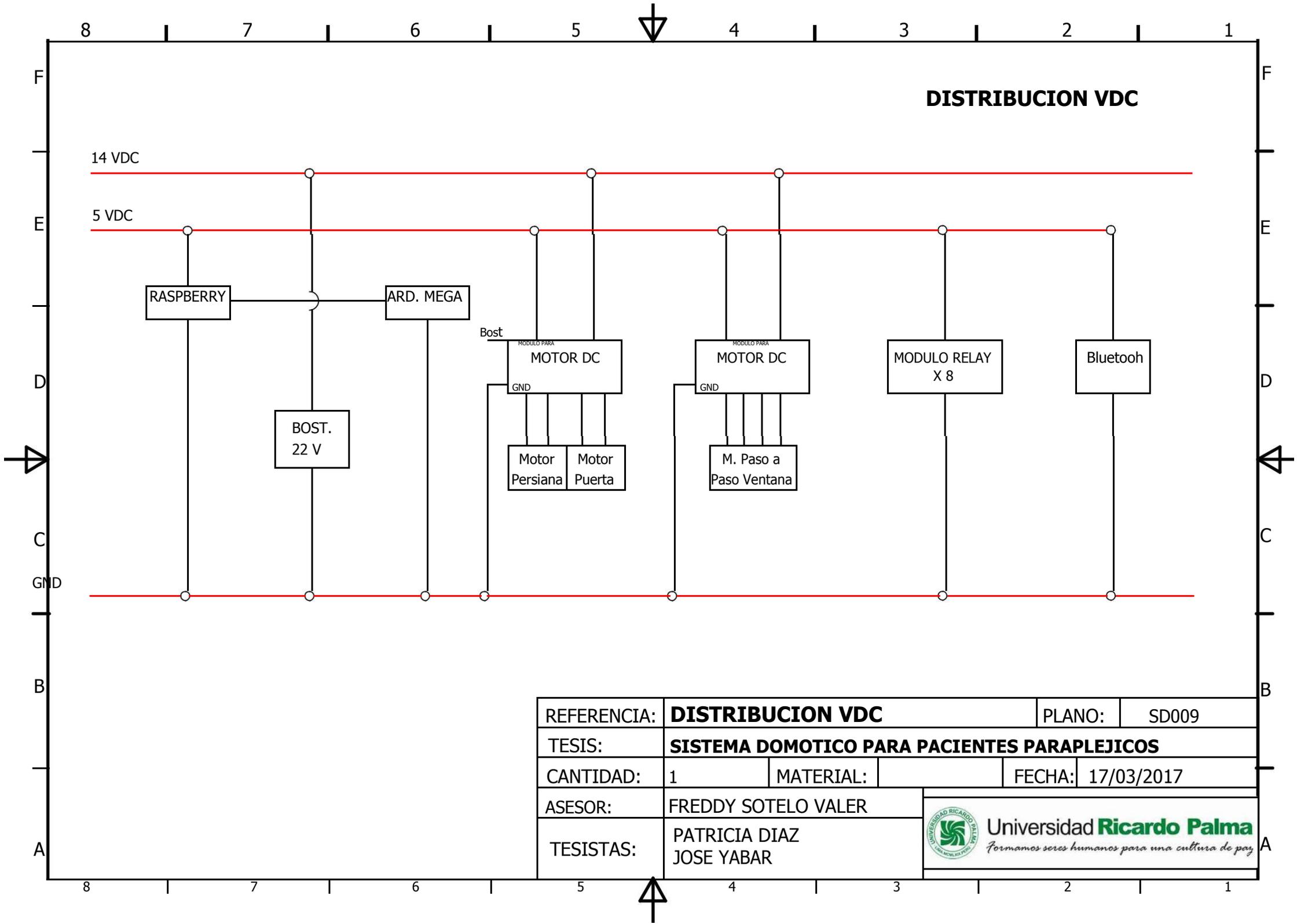


REFERENCIA:	SOPORTE TUERCA		PLANO:	SD007	
TESIS:	SISTEMA DOMOTICO PARA PACIENTES PARAPLEJICOS				
CANTIDAD:	1	MATERIAL:		FECHA:	17/03/2017
ASESOR:	FREDDY SOTELO VALER		 Universidad Ricardo Palma <i>Formamos seres humanos para una cultura de paz</i>		
TESISTAS:	PATRICIA DIAZ JOSE YABAR				

ANEXOS
PLANOS
ELECTRICOS



REFERENCIA:	DISTRIBUCION 220 V AC		PLANO:	SD008
TESIS:	SISTEMA DOMOTICO PARA PACIENTES PARAPLEJICOS			
CANTIDAD:	1	MATERIAL:	FECHA: 17/03/2017	
ASESOR:	FREDDY SOTELO VALER		 Universidad Ricardo Palma <i>Formamos seres humanos para una cultura de paz</i>	
TESISTAS:	PATRICIA DIAZ JOSE YABAR			

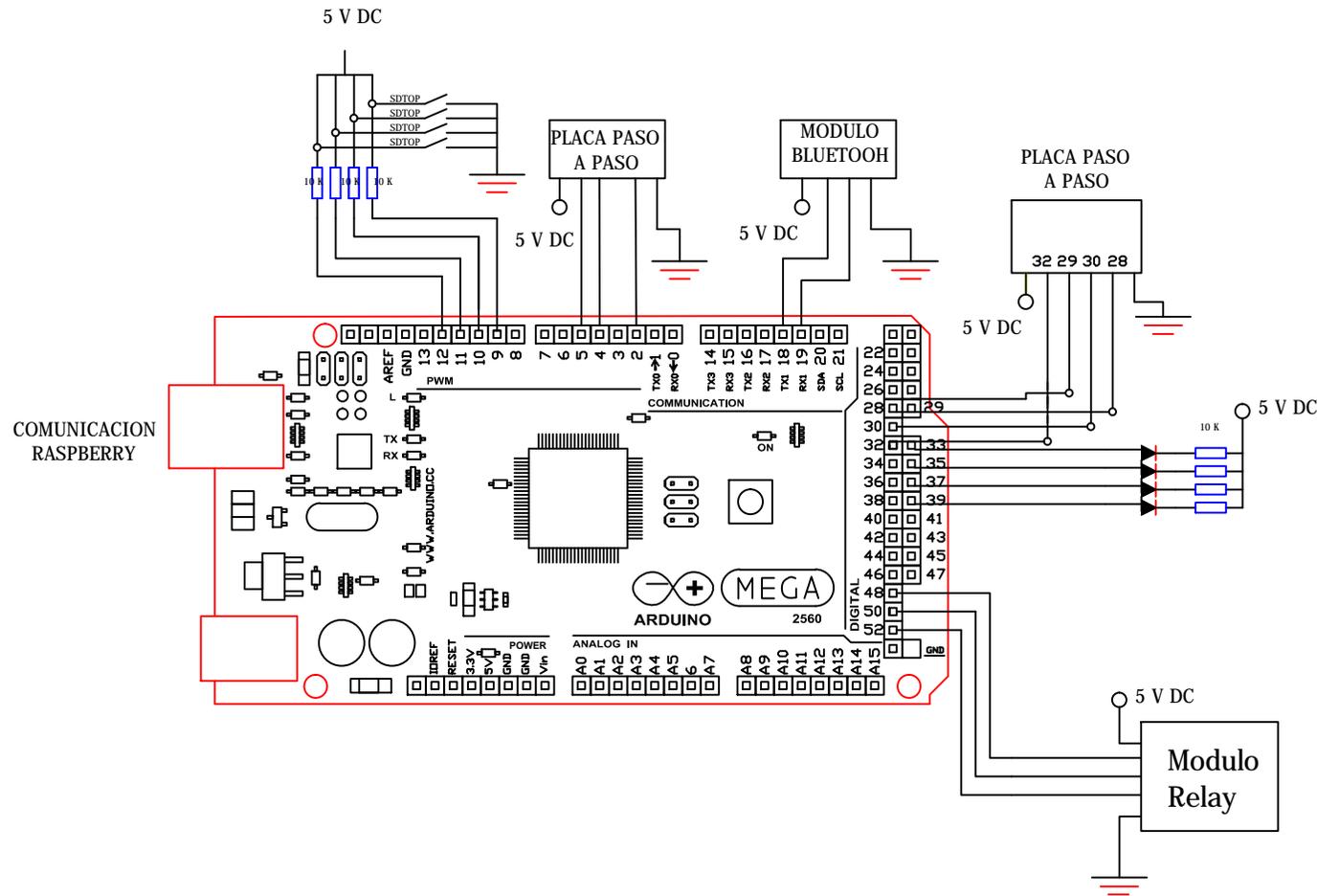


DISTRIBUCION VDC

REFERENCIA:	DISTRIBUCION VDC		PLANO:	SD009
TESIS:	SISTEMA DOMOTICO PARA PACIENTES PARAPLEJICOS			
CANTIDAD:	1	MATERIAL:	FECHA:	17/03/2017
ASESOR:	FREDDY SOTELO VALER			 Universidad Ricardo Palma <i>Formamos seres humanos para una cultura de paz</i>
TESISTAS:	PATRICIA DIAZ JOSE YABAR			

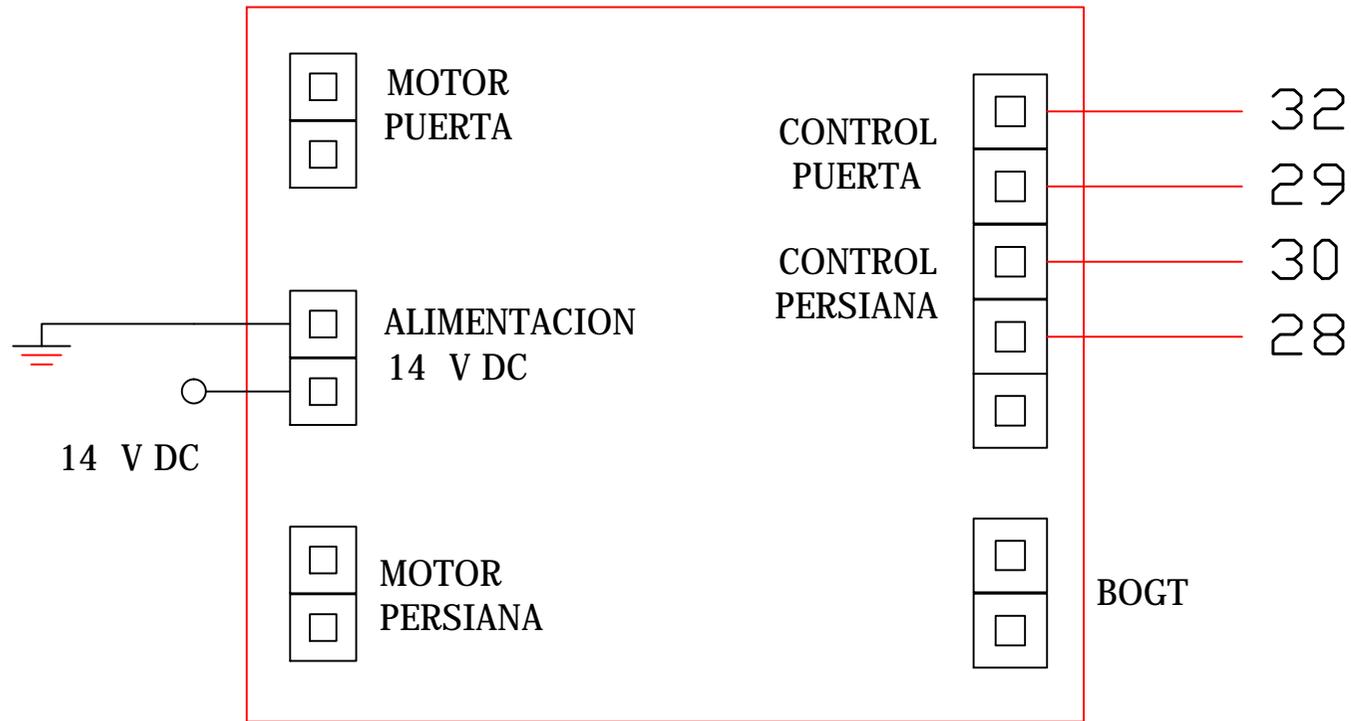
**ANEXOS
PLANOS
ELECTRONICOS**

CONEXION ARDUINO



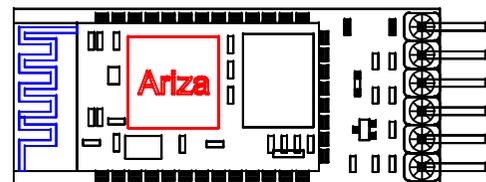
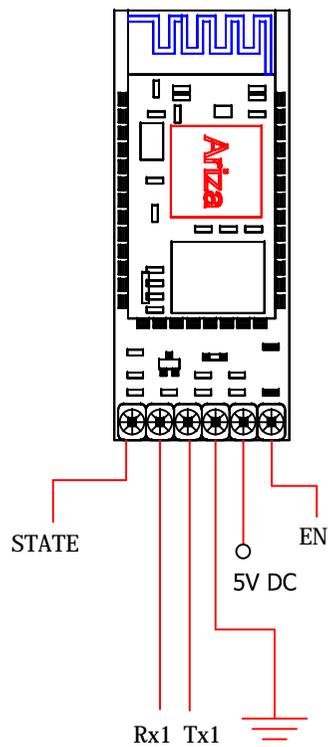
REFERENCIA:	CONEXION ARDUINO		PLANO:	SD010
TESIS:	SISTEMA DOMOTICO PARA PACIENTES PARAPLEJICOS			
CANTIDAD:	1	MATERIAL:	FECHA: 17/03/2017	
ASESOR:	FREDDY SOTELO VALER		 Universidad Ricardo Palma <i>Formamos seres humanos para una cultura de paz</i>	
TESISTAS:	PATRICIA DIAZ JOSE YABAR			

MODULO MOTOR DC



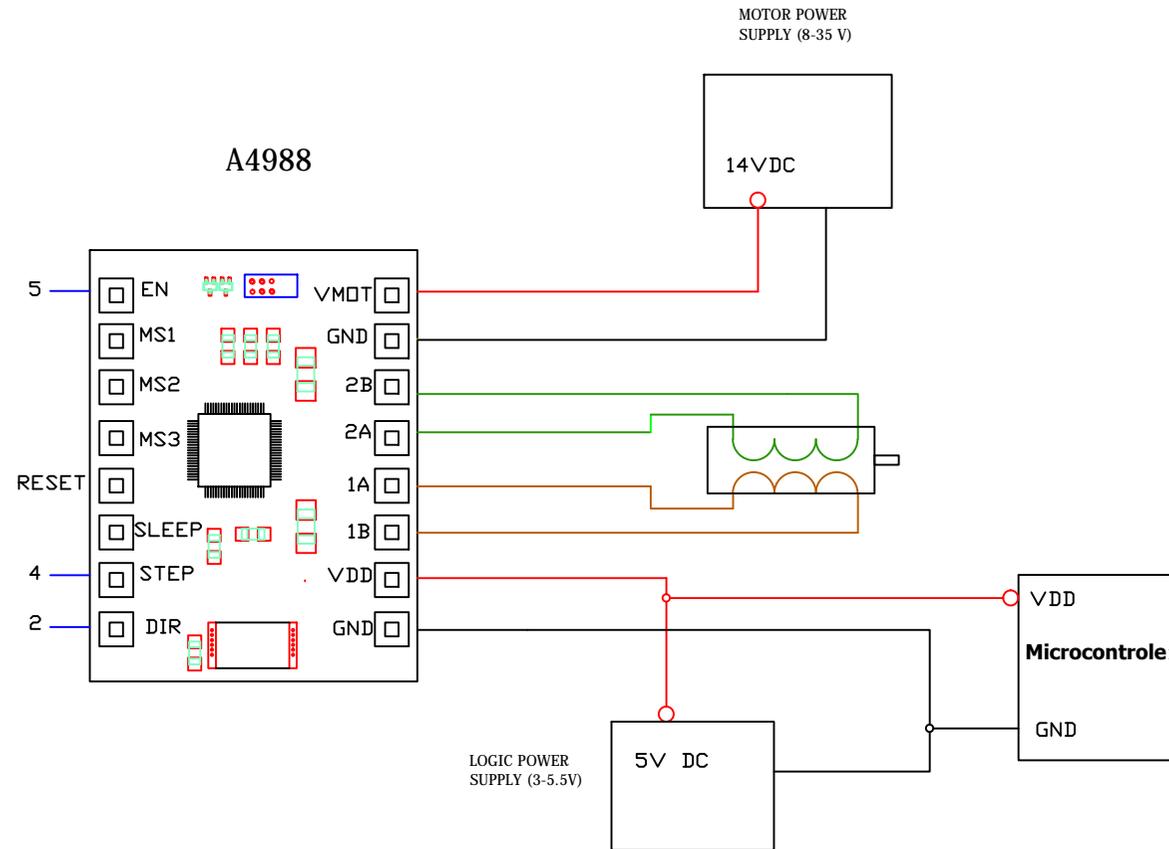
REFERENCIA:	MODULO MOTOR DC		PLANO:	SD011
TESIS:	SISTEMA DOMOTICO PARA PACIENTES PARAPLEJICOS			
CANTIDAD:	1	MATERIAL:	FECHA: 17/03/2017	
ASESOR:	FREDDY SOTELO VALER		 Universidad Ricardo Palma <i>Formamos seres humanos para una cultura de paz</i>	
TESISTAS:	PATRICIA DIAZ JOSE YABAR			

MODULO BLUETOOTH HC 05



REFERENCIA:	MODULO BLUETOOTH HC 05	PLANO:	SD012
TESIS:	SISTEMA DOMOTICO PARA PACIENTES PARAPLEJICOS		
CANTIDAD:	1	MATERIAL:	FECHA: 17/03/2017
ASESOR:	FREDDY SOTELO VALER		 Universidad Ricardo Palma <i>Formamos seres humanos para una cultura de paz</i>
TESISTAS:	PATRICIA DIAZ JOSE YABAR		

MODULO MOTOR PASO A PASO A4988

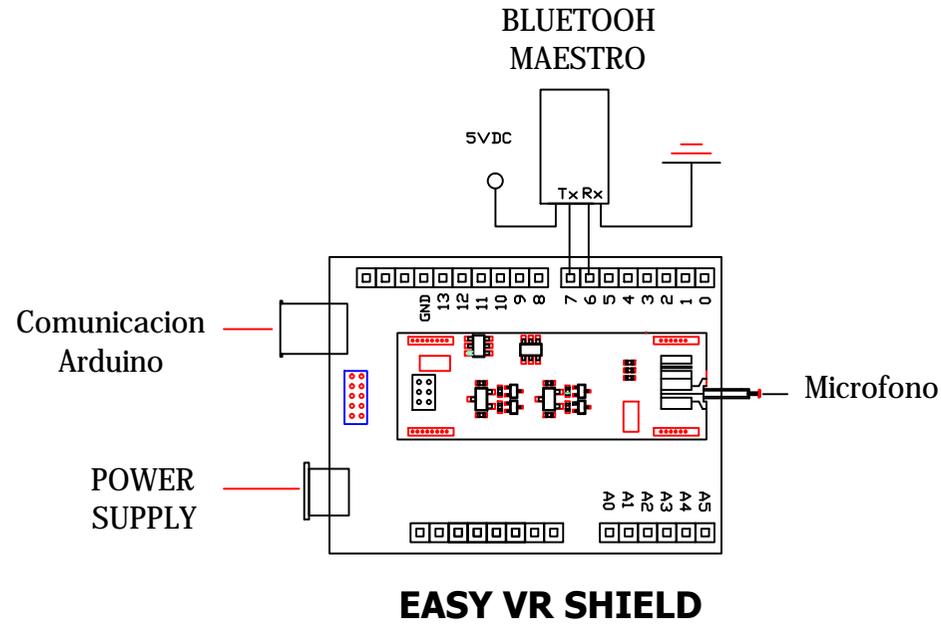


REFERENCIA:	MODULO MOTOR PASO A PASO A4988	PLANO:	SD013
TESIS:	SISTEMA DOMOTICO PARA PACIENTES PARAPLEJICOS		
CANTIDAD:	1	MATERIAL:	
FECHA:	17/03/2017		
ASESOR:	FREDDY SOTELO VALER		
TESISTAS:	PATRICIA DIAZ JOSE YABAR		



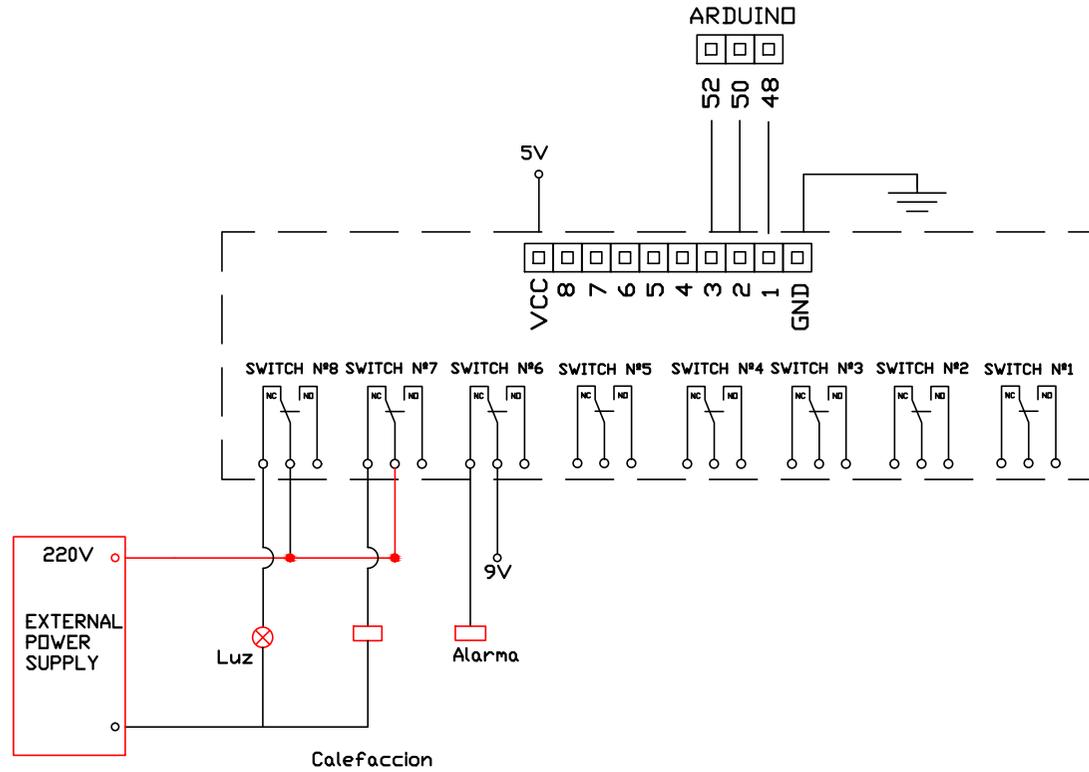
Universidad Ricardo Palma
Formamos seres humanos para una cultura de paz

CONEXION EASY VR SHIELD



REFERENCIA:	CONEXION EASY VR		PLANO:	SD014
TESIS:	SISTEMA DOMOTICO PARA PACIENTES PARAPLEJICOS			
CANTIDAD:	1	MATERIAL:	FECHA: 17/03/2017	
ASESOR:	FREDDY SOTELO VALER		 Universidad Ricardo Palma <i>Formamos seres humanos para una cultura de paz</i>	
TESISTAS:	PATRICIA DIAZ JOSE YABAR			

CONEXION MODULO RELAY X 8

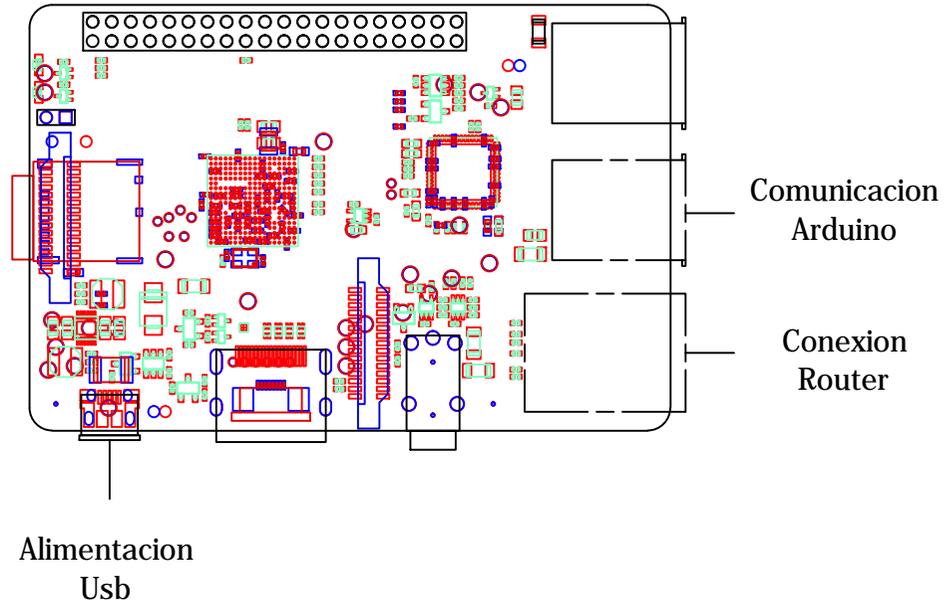


REFERENCIA:	CONEXION MODULO RELAY X 8	PLANO:	SD015
TESIS:	SISTEMA DOMOTICO PARA PACIENTES PARAPLEJICOS		
CANTIDAD:	1	MATERIAL:	FECHA: 17/03/2017
ASESOR:	FREDDY SOTELO VALER		
TESISTAS:	PATRICIA DIAZ JOSE YABAR		



Universidad Ricardo Palma
Formamos seres humanos para una cultura de paz

CONEXION RASPBERRY PI MODELO B



REFERENCIA:	CONEXION RASPBERRY PI		PLANO:	SD016
TESIS:	SISTEMA DOMOTICO PARA PACIENTES PARAPLEJICOS			
CANTIDAD:	1	MATERIAL:	FECHA: 17/03/2017	
ASESOR:	FREDDY SOTELO VALER		 Universidad Ricardo Palma <i>Formamos seres humanos para una cultura de paz</i>	
TESISTAS:	PATRICIA DIAZ JOSE YABAR			

ANEXOS DE PROGRAMACION

PAGINA WEB ALARMA

```
<!-- VENTANA PARA ACTIVAR ALARMA -->
<html>
<style>
    p{
        display: block;
        margin-top: 1em;
        margin-bottom: 1em;
        margin-left: 0;
        margin-right: 0;
    }
    table{
        font-family:arial, sans-serif;
        border-collapse: collapse;
        width: 300;
    }
    th{
        border: 0px solid #FFFFFF;
        text-align= left;
        padding:0px;
    }
    td{
        border:0px solid #FFFFFF;
        text-align= left; padding:
        0px;
    }
    .buttonSize{
        background-color:
        #F5F6F7; border: none;
        text-align: center;
        height:75px;
        width:225px;
    }
</style>
<body>
<?php
$command='k';
$a = "python /var/www/html/lib/client.py ";
$b= $a.$command;
$last_line = system($b, $retval);
?>
    <table bgcolor=#F5F6F7>
        <tr>
            <th width="75"> <form >
                <input type="image" src="Icons/Alarm.png" alt="Alarma"
                height="65" width="65"/>
            </th>
            <td width="225"> <form
                >
                <button class="ButtonSize" type="submit" >
                    <p style="color:red;font-weight:bold;font-
                    size:150%" > Encender Alarma </p>
                </button>
            </td>
        </tr>
    </table>
</body>
</html>
```

PAGINA WEB CALEFACCION ON

```
<!-- VENTANA PARA ENCENDER CALEFACTOR -->
<html>
<style>
    p{
        display: block;
        margin-top: 1em;
        margin-bottom: 1em;
        margin-left: 0;
        margin-right: 0;

    }
    table{
        font-family:arial, sans-serif;
        border-collapse: collapse;
        width: 300;
    }
    th{
        border: 0px solid #FFFFFF;
        text-align= left;
        padding:0px;
    }
    td{
        borders:0px solid #FFFFFF;
        text-align= left; padding:
        0px;
    }
    .buttonSize{
        background-color:
        #F5F6F7; border: none;
        text-align: center;
        height:75px;
        width:225px;
    }
</style>
<body>
<?php
$command='h';
$a = "python /var/www/html/lib/client.py ";
$b= $a.$command;
$last_line = system($b, $retval);
?>
    <table bgcolor=#F5F6F7>
        <tr>
            <th width="75">
                <form action="Calec_off.php">
                    <input type="image"
src="Icons/Calec_Off.png" alt="Calec Off" height="65" width="65"/>
                </form>
            </th>
            <th width="225">
                <form action="Calec_off.php">
                    <button class="ButtonSize" type="submit" >
                        <p style="color:red;font-weight:bold;font-
size:150%" > Encender Calefaccion </p> </button>
                </form>
            </th>
        </tr>
    </table>
</body>
</html>
```

PAGINA WEB CAMARA

```
<!-- VENTANA PARA LA CAMARA -->
<html>
<style>
<!-- CARACTERISTICAS DE LOS OBJETOS --
  > p{
        display: block;
        margin-top: 1em;
        margin-bottom:
        1em; margin-Left: 0;
        margin-right: 0;

    }
    table{
        font -family:arial, sans-serfi;
        border-collapse: collapse;
        width: 300;
    }
    th{
        border: 0px solid #FFFFFF;
        text-align= left;
        padding:0px;
    }
    td{
        borders:0px solid
        #FFFFFF; text-align= left;
        padding: 0px;
    }
.frameimg{
    align=center;
    border: none;
    frameborder: none;
    height:480px;
    width:640px;
    border-style:none;
}
<!-- CUERPO DEL PROGRAMA --
> </style>
<body>
<meta http-equiv="refresh" content="1"> <!-- REFRESH DE LA PAGINA CADA SEGUNDO -->
  <table bgcolor=#F5F6F7>
    <tr>
      <th width="75">
        <iframe class="frameimg" scrolling="no"
src="http://192.168.0.150/image/jpeg.cgi"></iframe> <!-- PIDO IMAGEN A LA CAMARA INSTALADA -->
      </th>
    </tr>
  </table>
</body>
</html>
```

PAGINA WEB BOTON DETENER

```
<!-- VENTANA PARA EL ABRIR PERSIANA -->
<html>
<style>
    p{
        display: block;
        margin-top: 1em;
        margin-bottom:
        1em; margin-left: 0;
        margin-right: 0;

    }
    table{
        font -family:arial, sans-serfi;
        border-collapse: collapse;
        width: 300;

    }
    th{
        border: 0px solid #FFFFFF;
        text-align= left;
        padding:0px;

    }
    td{
        borders:0px solid
        #FFFFFF; text-align= left;
        padding: 0px;

    }
.buttonSize{
    background-color:
    #F5F6F7; border: none;
    text-align: center;
    height:75px;
    width:225px;
}
</style>
<body>
<?php
$command='t';
$a = "python /var/www/html/lib/client.py
"; $b= $a.$command;
$last_line = system($b, $retval);
?>
    <table bgcolor=#F5F6F7> <tr>

        <th width="75">
            <form action="Detener.php">
                <input type="image" src="Icons/Stop.png" alt="Detener Accion" height="65"
width="65"/>
            </form>
        </th>
        <th width="225">
            <form action="Detener.php">
                <button class="ButtonSize" type="submit" >
                    <p style="color:red;font-weight:bold;font-size:150%" >Detener Accion
                </p>
            </button>
        </form>
        </th>
    </tr>
</table>
</body>
</html>
```

PAGINA WEB BOTON DETENER

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
  <title>Tesis: Room Control</title>
  <style type="text/css" media="screen"> <!--
    CONTROL DE CARACTERISTICAS -->
* {
  margin: 0px 0px 0px 0px;
  padding: 0px 0px 0px 0px;
}

body, html {
  padding: 3px 3px 3px 3px;

  background-color: #D8DBE2;

  font-family: Verdana, sans-
  serif; font-size: 11pt;
  text-align: center;
}

div.main_page { <!--CARACTERISTICAS DEL ENCABEZADO-->
  position: relative;
  display: table;

  width: 1000px;

  margin-bottom: 3px;
  margin-left: auto; margin-
  right: auto; padding: 0px
  0px 0px 0px;

  border-width: 2px;
  border-color: #212738;
  border-style: solid;

  background-color: #FFFFFF;

  text-align: center;
}
div.page_header
{ height: 99px;
width: 100%;

  background-color: #F5F6F7;
}

div.page_header span {
  margin: 15px 0px 0px 50px;

  font-size: 180%;
  font-weight: bold;
}

div.page_header img {
  margin: 3px 0px 0px 40px;

  border: 0px 0px 0px;
}

div.table_of_contents
{ clear: left;
```

```
min-width: 200px;

margin: 3px 3px 3px 3px;

background-color: #FFFFFF;

text-align: left;
}

div.table_of_contents_item {
clear: left;

width: 100%;

margin: 4px 0px 0px 0px;

background-color: #FFFFFF;

color: #000000;
text-align: left;
}

div.table_of_contents_item a {
margin: 6px 0px 0px 6px;
}

div.content_section {
margin: 3px 3px 3px 3px;

background-color: #FFFFFF;

text-align: left;
}

div.content_section_text {
padding: 4px 8px 4px 8px;

color: #000000;
font-size: 100%;
}

div.content_section_text pre {
margin: 8px 0px 8px 0px;
padding: 8px 8px 8px 8px;

border-width: 1px;
border-style: dotted;
border-color: #000000;

background-color: #F5F6F7;

font-style: italic;
}

div.content_section_text p
{ margin-bottom: 6px;
}

div.content_section_text ul, div.content_section_text li {
padding: 4px 8px 4px 16px;
}

div.section_header {
padding: 3px 6px 3px 6px;

background-color: #8E9CB2;
```

```

color: #FFFFFF;
font-weight: bold;
font-size: 112%;
text-align: center;
}

div.section_header_red {
background-color: #CD214F;
}

div.section_header_grey {
background-color: #9F9386;
}

.floating_element {
position: relative;
float: left;
}

div.table_of_contents_item
a, div.content_section_text a
{ text-decoration: none; font-
weight: bold;
}

div.table_of_contents_item a:link,
div.table_of_contents_item a:visited,
div.table_of_contents_item a:active {
color: #000000;
}

div.table_of_contents_item a:hover {
background-color: #000000;

color: #FFFFFF;
}

div.content_section_text a:link,
div.content_section_text a:visited,
div.content_section_text a:active {
background-color: #DCDFE6;

color: #000000;
}

div.content_section_text a:hover {
background-color: #000000;

color: #DCDFE6;
}

div.validator {
}

table{
font-family:arial, sans-
serfi; border-collapse:
collapse; width:100%;
}
th{
border:1px solid
#FFFFFF; text-align= left;
padding:0px;
}
td{

```

```

        border:1px solid
        #CD214F; text-align= left;
        padding:0px;
    }
    .framecarac{
        align=left; border:
        none;
        frameborder:
        none; height:75px;
        width:225; border-
        style:none;
    }
    .frameimg{
        align=center;
        border: none;
        frameborder: none;
        height:480px;
        width:640px;
        border-style:none;
    }
}

</style>
</head>
<!--FIN DEL ENCABEZADO Y CONTROL DE CARACTERISTICAS -->
<!--INICIO DEL CUERPO -->
<body>

<!--SE INICIA LA LIBERIA DEL CLIENTE PARA EL CONTROL WEB -->
<?php
"python /var/www/html/lib/client.py";
?>
<!--TITULO DE LA PAGINA -->
<div class="main_page">
<div class="page_header floating_element">
<BR>

<span class="floating_element">
CONTROL WEB DE LA HABITACION
</span>

</div>
<br>
<br>
<!-- CUERPO DE LA PAGINA -->
<table bgcolor=#F5F6F7>
<tr>
<th width="70%">
<div class="section_header section_header_red">
<div id="Camara"></div> <!--
SBUTITULO DE CAMARA -->
CAMARA
</div>
</th>
<th width="30%">
<div class="section_header section_header_red">
<div id="Cpanel"></div>
PANEL DE CONTROL <!--SUBTITULO
PANEL DE CONTROL -->
</div>
</th>
</tr>
<!--ESPACIO PARA LOS DIFERENTES INTERACCIONES -->
<tr>
<td rowspan=8 height="600">
<iframe class="frameimg" scrolling="no" src="Camera.php"></iframe> <!-- SE CARGA
LA PAGINA CAMERA.PHP -->

```

```

        </td>

        <td>
        </td>
    </tr>
    <tr>
        <td height="75">
        <iframe class="framecarac" scrolling="no" src="Luces_on.php"></iframe>          <!-- SE CARGA LA PAGINA
LUCES_ON.PHP -->
        </td>
    </tr>
    <tr>
        <td height="75">
        <iframe class="framecarac" scrolling="no" src="Puerta_on.php"></iframe>          <!-- SE CARGA LA PAGINA
PUERTA_ON.PHP -->
        </td>
    </tr>
    <tr>
        <td height="75">
        <iframe class="framecarac" scrolling="no" src="Persiana_on.php"></iframe>          <!-- SE CARGA
LA PAGINA PERSIANA_ON.PHP -->
        </td>
    </tr>
    <tr>
        <td height="75">
        <iframe class="framecarac" scrolling="no" src="Ventana_on.php"></iframe>          <!-- SE CARGA
LA PAGINA VENTANA_ON.PHP -->
        </td>
    </tr>
    <tr>
        <td height="75">
        <iframe class="framecarac" scrolling="no" src="Calec_on.php"></iframe>          <!-- SE CARGA LA PAGINA
CALEC_ON.PHP -->
        </td>
    </tr>
    <tr>
        <td height="75">
        <iframe class="framecarac" scrolling="no" src="Alarma.php"></iframe>          <!-- SE CARGA LA PAGINA
ALARMA.PHP -->
        </td>
    </tr>
    <tr>
        <td>
        <iframe class="framecarac" scrolling="no" src="Detener.php"></iframe>          <!-- SE CARGA LA PAGINA
PUERTA_APAGADO1.PHP -->
        </td>
    </tr>
</table>

<!-- FIN DEL CUERPO -->

</body>
</html>

<!-- CIERRE DE PROGRAMA -->

```

PAGINA WEB LUCES ON

```
<!-- VENTANA PARA EL ENCENDIDO DE LUCES -->
<html>
<style>
    p{
        display: block;
        margin-top: 1em;
        margin-bottom:
        1em; margin-left: 0;
        margin-right: 0;
    }
    table{
        font -family:arial, sans-serfi;
        border-collapse: collapse;
        width: 300;
    }
    th{
        border: 0px solid #FFFFFF;
        text-align= left;
        padding:0px;
    }
    td{
        borders:0px solid
        #FFFFFF; text-align= left;
        padding: 0px;
    }
.buttonSize{
    background-color:
    #F5F6F7; border: none;
    text-align: center;
    height:75px;
    width:225px;
}
</style>
<body>
<?php
$command='j';
$a = "python /var/www/html/lib/client.py
"; $b= $a.$command;
$last_line = system($b, $retval);
?>
    <table bgcolor=#F5F6F7> <tr>
        <th width="75">
            <form action="Luces_off.php">
                <input type="image" src="Icons/foco_off.png" alt="Foco Off" height="60"
width="60"/>
            </form>
        </th>
        <th width="225">
            <form action="Luces_off.php">
                <button class="ButtonSize" type="submit" >
                    <p style="color:red;font-weight:bold;font-size:150%" > Encender Luces
                </p>
            </button>
        </form>
        </th>
    </tr>
</table>
</body>
</html>
```

PAGINA WEB PERSIANA ON

```
<!-- VENTANA PARA EL ABRIR PERSIANA -->
<html>
<style>
    p{
        display: block;
        margin-top: 1em;
        margin-bottom:
        1em; margin-left: 0;
        margin-right: 0;

    }
    table{
        font-family:arial, sans-serif;
        border-collapse: collapse;
        width: 300;

    }
    th{
        border: 0px solid #FFFFFF;
        text-align= left;
        padding:0px;

    }
    td{
        borders:0px solid
        #FFFFFF; text-align= left;
        padding: 0px;

    }
.buttonSize{
    background-color:
    #F5F6F7; border: none;
    text-align: center;
    height:75px;
    width:225px;
}
</style>
<body>
<?php
$command='f';
$a = "python /var/www/html/lib/client.py
"; $b= $a.$command;
$last_line = system($b, $retval);
?>
    <table bgcolor=#F5F6F7> <tr>
        <th width="75">
            <form action="Persiana_off.php">
                <input type="image" src="Icons/Persiana_Off.png" alt="Persiana off"
height="65" width="65"/>
            </form>
        </th>
        <th width="225">
            <form action="Persiana_off.php">
                <button class="ButtonSize" type="submit" >
                    <p style="color:red;font-weight:bold;font-size:150%" > Abrir Persianas
                </p>
            </button>
        </th>
    </tr>
</table>
</body>
</html>
```

PAGINA WEB PUERTA ON

```
<!-- VENTANA PARA EL ABRIR PUERTA -->
<html>
<style>
    p{
        display: block;
        margin-top: 1em;
        margin-bottom:
        1em; margin-left: 0;
        margin-right: 0;

    }
    table{
        font -family:arial, sans-serfi;
        border-collapse: collapse;
        width: 300;

    }
    th{
        border: 0px solid #FFFFFF;
        text-align= left;
        padding:0px;

    }
    td{
        borders:0px solid
        #FFFFFF; text-align= left;
        padding: 0px;

    }
.buttonSize{
    background-color:
    #F5F6F7; border: none;
    text-align: center;
    height:75px;
    width:225px;
}
</style>
<body>
<?php
$command='d';
$a = "python /var/www/html/lib/client.py
"; $b= $a.$command;
$last_line = system($b, $retval);
?>
    <table bgcolor=#F5F6F7> <tr>
        <th width="75">
            <form action="Puerta_off.php">
                <input type="image" src="Icons/Door_Off.png" alt="Door Off" height="55"
width="55"/>
            </form>
        </th>
        <th width="225">
            <form action="Puerta_off.php">
                <button class="ButtonSize" type="submit" >
                    <p style="color:red;font-weight:bold;font-size:150%" > Abrir Puerta
                </p>
            </button>
        </form>
        </th>
    </tr>
</table>
</body>
</html>
```

PAGINA WEB VENTANA ON

```
<!-- VENTANA PARA EL ABRIR VENTANA -->
<html>
<style>
    p{
        display: block;
        margin-top: 1em;
        margin-bottom:
        1em; margin-left: 0;
        margin-right: 0;

    }
    table{
        font -family:arial, sans-serfi;
        border-collapse: collapse;
        width: 300;

    }
    th{
        border: 0px solid #FFFFFF;
        text-align= left;
        padding:0px;

    }
    td{
        borders:0px solid
        #FFFFFF; text-align= left;
        padding: 0px;

    }
.buttonSize{
    background-color:
    #F5F6F7; border: none;
    text-align: center;
    height:75px;
    width:225px;
}
</style>
<body>
<?php
$command='b';
$a = "python /var/www/html/lib/client.py
"; $b= $a.$command;
$last_line = system($b, $retval);
?>
    <table bgcolor=#F5F6F7> <tr>

        <th width="75">
            <form action="Ventana_off.php">
                <input type="image" src="Icons/Ventana_off.png" alt="Ventana Off" height="65"
width="65"/>
            </form>
        </th>
        <th width="225">
            <form action="Ventana_off.php">
                <button class="ButtonSize" type="submit" >
                    <p style="color:red;font-weight:bold;font-size:150%" > Abrir Ventana
                </p>
            </button>
        </th>
    </tr>
</table>
</body>
</html>
```

LIBRERIAS CLIENT

```
import socket
import sys

TCP_IP = '192.168.0.122'
TCP_PORT = 5005
BUFFER_SIZE = 1024
MESSAGE = str(sys.argv[1])

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect((TCP_IP, TCP_PORT))
s.send(MESSAGE)
data = s.recv(BUFFER_SIZE)
s.close()

#print "received data:", data
```

LIBRERIAS WEBSERVER

```
#WEBSERVER.PY DATOS PRINCIPALES PARA EL CONTROL
# a VENTANA ON
# b VETANA OFF
# c PUERTA ON
# d PUERTA OFF
# e PERSIANA ON
# f PERSIANA OFF
# g CALEFACCION ON
# h CALEFACCION OFF
# i LUCES ON
# j CUCES OFF
# k ALARMA ON

import socket
import serial
import time
import subprocess

dev=subprocess.check_output('ls
/dev/ttyACM*',shell=True) print dev
try:
    ser=serial.Serial(dev.strip(),9600)
    print "Arduino Conectado"
    ser.write('l1')
    time.sleep(0.5)

except:
    print "Arduino Desconectado"

def server():
    global ser

    while True:
        conn, addr = s.accept()
        print 'Direccion de Coneccion: ', addr
```

```
data = conn.recv(BUFFER_SIZE)
if not data: continue
print "Informacion recibida: ", data
if data=='a':
    conn.close()
    ser.write('v1')
    print "se envia v1"
    time.sleep(0.5)
elif data=='b':
    conn.close()
    ser.write('v2')
    print "se envia v2"
    time.sleep(0.5)
elif data=='c':
    conn.close()
    ser.write('d1')
    print "se envia d1"
    time.sleep(0.5)
elif data=='d':
    conn.close()
    ser.write('d2')
    print "se envia d2"
    time.sleep(0.5)
elif data=='e':
    conn.close()
    ser.write('p1')
    print "se envia p1"
    time.sleep(0.5)
elif data=='f':
    conn.close()
    ser.write('p2')
    print "se envia p2"
    time.sleep(0.5)
elif data=='g':
    conn.close()
    ser.write('r2')
    print "se envia r2"
    time.sleep(0.5)
```

```
elif data=='h':
    conn.close()
    ser.write('r2')
    print "se envia r2"
    time.sleep(0.5)
elif data=='i':
    conn.close()
    ser.write('r1')
    print "se envia r1"
    time.sleep(0.5)
elif data=='j':
    conn.close()
    ser.write('r1')
    print "se envia r1"
    time.sleep(0.5)
elif data=='k':
    conn.close()
    ser.write('r3')
    print "se envia r3"
    time.sleep(0.5)
elif data=='t':
    conn.close()
    ser.write('t')
    print "se envia t"
    time.sleep(0.5)
elif data=='m':
    conn.close()
    ser.write('m')
    print "se envia r3"
    time.sleep(0.5)
elif data=='n':
    conn.close()
    ser.write('n')
    print "se envia n"
    time.sleep(0.5)
elif data=='o':
    conn.close()
    ser.write('o')
```

```
        print "se envia o"  
        time.sleep(0.5)  
    elif data=='q':  
        conn.close()  
        ser.close()  
        exit(0)
```

```
TCP_IP = '192.168.0.122'  
TCP_PORT = 5005  
BUFFER_SIZE = 20 # Normally 1024, but we want fast response
```

```
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)  
s.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)  
s.bind((TCP_IP, TCP_PORT))  
s.listen(5)
```

```
print 'Servidor  
Iniciado' server()
```

EMISOR VR ARDUINO BLUE

```
//*****Proyecto domotico ---- Controlador Emisor
*****

#define state1 9

#include "Arduino.h"
#if !defined(SERIAL_PORT_MONITOR)
  #error "Arduino version not supported. Please update your IDE to the
latest version."
#endif

#if defined(SERIAL_PORT_USBVIRTUAL)
  // Shield Jumper on HW (for Leonardo and
  Due) #define port SERIAL_PORT_HARDWARE
  #define pcSerial SERIAL_PORT_USBVIRTUAL
#else
  // Shield Jumper on SW (using pins 12/13 or 8/9 as RX/TX)
  #include "SoftwareSerial.h"
  SoftwareSerial port(12, 13);
  SoftwareSerial BTMaestro(7,6);

  #define pcSerial
SERIAL_PORT_MONITOR #endif

#include "EasyVR.h"

EasyVR easyvr(port);

//Groups and Commands
enum Groups
{
  GROUP_0 = 0,
  GROUP_1 = 1,
  GROUP_2 = 2,
  GROUP_3 = 3,
```

```
GROUP_4 = 4,  
GROUP_5 = 5,  
GROUP_6 = 6,  
};
```

```
enum Group0  
{  
    G0_OPEN = 0,  
};
```

```
enum Group1  
{  
    G1_ABRIR_PUERTA = 0,  
    G1_CERRAR_PUERTA = 1,  
    G1_ABRIR_VENTANA = 2,  
    G1_CERRAR_VENTANA = 3,  
    G1_PRENDER_LUCES = 4,  
    G1_SUBIR_PERSIANA = 5,  
    G1_BAJAR_PERSIANA = 6,  
    G1_ENCENDER_CALEFACCION = 7,  
    G1_ALARMA = 8,  
    G1_DETENER= 9,  
};
```

```
enum Group2  
{  
    G1_HOLA = 0,  
    G1_CHAU = 1,  
    G1_PERSIANA = 2,  
    G1_CLOSE = 3,  
};
```

```
enum Group3  
{  
    G3_LLAMAR_EMERGENCIA = 0,  
    G3_ABRIR_VENTANA = 1,  
    G3_ABRIR_PUERTA = 2,
```

G3_PRENDER_LUZ = 3,

```
};
```

```
enum Group4
```

```
{
```

```
  G4_ABRIR_VENTANA = 0,
```

```
  G4_ABRIR_PUERTA = 1,
```

```
  G4_LLAMAR_EMERGENCIA = 2,
```

```
};
```

```
enum Group5
```

```
{
```

```
  G5_PERUANO = 0,
```

```
  G5_HOLA_PERU = 1,
```

```
  G5_CHAU_PERU = 2,
```

```
};
```

```
enum Group6
```

```
{
```

```
  G6_CERRAR_PUERTA = 0,
```

```
  G6_ABRIR_VENTANA = 1,
```

```
  G6_CERRAR_VENTANA = 2,
```

```
  G6_XZXZ = 3,
```

```
};
```

```
int8_t group, idx;
```

```
void setup()
```

```
{
```

```
  // setup PC serial port
```

```
  pcSerial.begin(9600);
```

```
  BTMaestro.begin(9600);
```

```
  // bridge mode?
```

```
  int mode = easyvr.bridgeRequested(pcSerial);
```

```
  switch (mode)
```

```
  {
```

```
    case EasyVR::BRIDGE_NONE:
```

```

// setup EasyVR serial port
port.begin(9600);
// run normally pcSerial.println(F("----"));
pcSerial.println(F("Bridge not
started!")); break;

case EasyVR::BRIDGE_NORMAL:
// setup EasyVR serial port (low
speed) port.begin(9600);
// soft-connect the two serial ports (PC and
EasyVR) easyvr.bridgeLoop(pcSerial);
// resume normally if aborted
pcSerial.println(F("----"));
pcSerial.println(F("Bridge connection
aborted!")); break;

case EasyVR::BRIDGE_BOOT:
// setup EasyVR serial port (high
speed) port.begin(115200);
// soft-connect the two serial ports (PC and
EasyVR) easyvr.bridgeLoop(pcSerial);
// resume normally if aborted
pcSerial.println(F("----"));
pcSerial.println(F("Bridge connection
aborted!")); break;
}

while (!easyvr.detect())
{
Serial.println("EasyVR not
detected!"); delay(1000);
}

easyvr.setPinOutput(EasyVR::IO1, LOW);
Serial.println("EasyVR detected!");
easyvr.setTimeout(5);
easyvr.setLanguage(4);

```

```

group = GROUP_1; //<-- start group (customize) INICIO DEL GRUPO

//
delay(3000);
}

void action();

void loop()
{
  easyvr.setPinOutput(EasyVR::IO1, HIGH); // LED on (listening)

  Serial.print("Say a command in Group ");
  Serial.println(group);
  easyvr.recognizeCommand(group);

  do
  {
    // can do some processing while waiting for a spoken command
  }
  while (!easyvr.hasFinished());

  easyvr.setPinOutput(EasyVR::IO1, LOW); // LED off

  idx = easyvr.getWord();
  if (idx >= 0)
  {
    // built-in trigger (ROBOT)
    // group = GROUP_X; <-- jump to another group
    X return;
  }
  idx = easyvr.getCommand();
  if (idx >= 0)
  {
    // print debug message
    uint8_t train = 0;
    char name[32];

```

```

Serial.print("Command:
"); Serial.print(idx);
if (easyvr.dumpCommand(group, idx, name, train))
{
    Serial.print(" = ");
    Serial.println(name);
}
else
    Serial.println();
easyvr.playSound(0, EasyVR::VOL_FULL);
// perform some action
action();
}
else // errors or timeout
{
    if (easyvr.isTimeout())
        Serial.println("Timed out, try again...");
    int16_t err = easyvr.getError();
    if (err >= 0)
    {
        Serial.print("Error ");
        Serial.println(err, HEX);
    }
}
}

void action()
{
    switch (group)
    {
    case GROUP_0:
        switch (idx)
        {
        case GO_OPEN:
            // write your action code here
            // group = GROUP_X; <-- or jump to another group X for composite
            commands
            break;

```

```

}
break;
case GROUP_2:
switch (idx)
{
case G2_HOLA:
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
break;
case G2_CHAU:
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
break;
case G2_PERSIANA:
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
break;
case G2_CLOSE:
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
break;
}
break;
case GROUP_1:
switch (idx)
{
case G1_ABRIR_PUERTA:
BTMaestro.print("d1");
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
break;
case G1_CERRAR_PUERTA:
BTMaestro.print("d2");

```

```
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
break;
case G1_ABRIR_VENTANA:
BTMaestro.print("v1");
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
break;
case G1_CERRAR_VENTANA:
BTMaestro.print("v2");
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
break;
case G1_PRENDER_LUCES:
BTMaestro.print("r1");
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
case G1_SUBIR_PERSIANA:
BTMaestro.print("p1");
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
break;
case G1_BAJAR_PERSIANA:
BTMaestro.print("p2");
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
break;
case G1_ENCENDER_CALEFACCION:
BTMaestro.print("r2");
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
```

```

    break;
case G1_ALARMA:
    BTMaestro.print("r3");
    // write your action code here
    // group = GROUP_X; <-- or jump to another group X for composite
commands
    break;
    case G1_DETENER:
    BTMaestro.print("t");
    // write your action code here
    // group = GROUP_X; <-- or jump to another group X for composite
commands
    break;
}
break;
case GROUP_3:
    switch (idx)
    {
    case G3_LLAMAR_EMERGENCIA:
        // write your action code here
        // group = GROUP_X; <-- or jump to another group X for composite
commands
        break;
    case G3_ABRIR_VENTANA:
        // write your action code here
        // group = GROUP_X; <-- or jump to another group X for composite
commands
        break;
    case G3_ABRIR_PUERTA:
        // write your action code here
        // group = GROUP_X; <-- or jump to another group X for composite
commands
        break;
    case G3_PRENDER_LUZ:
        // write your action code here
        // group = GROUP_X; <-- or jump to another group X for composite
commands
        break;

```

```

}
break;
case GROUP_4:
switch (idx)
{
case G4_ABRIR_VENTANA:
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
break;
case G4_ABRIR_PUERTA:
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
break;
case G4_LLAMAR_EMERGENCIA:
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
break;
}
break;
case GROUP_5:
switch (idx)
{
case G5_PERUANO:
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
break;
case G5_HOLA_PERU:
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
break;
case G5_CHAU_PERU:
// write your action code here

```

```

    // group = GROUP_X; <-- or jump to another group X for composite
commands
    break;
}
break;
case GROUP_6:
switch (idx)
{
case G6_CERRAR_PUERTA:
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
break;
case G6_ABRIR_VENTANA:
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
break;
case G6_CERRAR_VENTANA:
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
break;
case G6_XZXZ:
// write your action code here
// group = GROUP_X; <-- or jump to another group X for composite
commands
break;
}
break;
}
}

```

PROGRAMA ARDUINO MEGA

```
//DECLARACION DE PUERTO I/O
```

```
#define L1 33 //LED1
```

```
#define L2 35 //LED2
```

```
#define L3 37 //LED3
```

```
#define L4 39 //LED4
```

```
#define R1 52 // LUCES -- 220AC OJO
```

```
#define R2 50 // CALEFACCION -- 220AC OJO
```

```
#define R3 48 // ALARMA -- 220AC OJO
```

```
#define P1 31 // PERSIANA PIN1
```

```
#define P2 29 // PERSIANA PIN2
```

```
#define D1 30 // PUERTA PIN1
```

```
#define D2 28 // PUERTA PIN2
```

```
#define DIR 2 //MOTOR PAP PARA VENTANA DIRECCION
```

```
#define STEP 4 //PULSOS
```

```
#define EN 5 // HABILITA MOV. DEL MOTOR
```

```
#define Ptop 9 //FINAL DE CARRERA PERSIANA TOP
```

```
#define Pbot 10 //FINAL DE CARRERA PERSIANA BOT
```

```
#define Dtop 11 //FINAL DE CARRERA PUERTA ABIERTA
```

```
#define Dbot 12 //FINAL DE CARRERA PUERTA CERRADA
```

```
#define Velocidad 400 //VELOCIDAD DEL MOTOR PARA VENTANA
```

```
long timeac=0, time3=0; //TEMPORIZADOR
```

```
int nr, nl, nd=0, np=0, nv=0, stepv=0; //ESTADOS PARA LA PROGRAMACION
```

```
int npant=0, nvant=0, ndant=0; //ESTADO ANTERIOERS
```

```
long stepcount=0; //PULSOS DE LA VENTANA
```

```
void INICIALIZA(){
```

```
  //LEDS
```

```
  pinMode(L1,OUTPUT);
```

```
pinMode(L2,OUTPUT);
pinMode(L3,OUTPUT);
pinMode(L4,OUTPUT);
//RELAYS
pinMode(R1,OUTPUT);
pinMode(R2,OUTPUT);
pinMode(R3,OUTPUT);
//PERSIANA
pinMode(P1,OUTPUT);
pinMode(P2,OUTPUT);
//PUERTA
pinMode(D1,OUTPUT);
pinMode(D2,OUTPUT);
//VENTAN
pinMode(DIR,OUTPUT);
pinMode(STEP,OUTPUT);
pinMode(EN,OUTPUT);
digitalWrite(EN,HIGH);
// PARA LOS LEDS
digitalWrite(L1,LOW); //EL L1 SIEMPRE ENCENDIDO
digitalWrite(L2,HIGH);
digitalWrite(L3,HIGH);
digitalWrite(L4,HIGH); //INDICA EL ESTADO DE LA COMUNICACION
// PARA LOS RELAYS
digitalWrite(R1,HIGH);
digitalWrite(R2,HIGH);
digitalWrite(R3,HIGH);
// PARA LOS MOTORES DC
digitalWrite(P1,LOW);
digitalWrite(P2,LOW);
digitalWrite(D1,LOW);
digitalWrite(D2,LOW);
// PARA LOS FINALES DE CARRERA
pinMode(Ptop,INPUT);
pinMode(Pbot,INPUT);
pinMode(Dtop,INPUT);
pinMode(Dbot,INPUT);
}
```

```

void setup() {
  Serial.begin(9600); //PUERTO DE COMUNICACION PARA LA PC O EL
RASPEBERRY
  Serial.setTimeout(5); //TIEMPO DE ADQUISICION DE DATOS
  Serial1.begin(9600); //PUERTO DE COMUNICACION PARA EL MODULO BT
  Serial1.setTimeout(5); //TIEMPO DE ADQUISICION DE DATOS
  INICIALIZA();
}

void Persiana(int ip){ //FUNCION PERSIANA( 0:DETENIDO, 1:ABRE, 2:CIERRA)
  if (ip==0){
    digitalWrite(P1,LOW);
    digitalWrite(P2,LOW);
  }
  else if (ip==1){
    digitalWrite(P1,LOW);
    digitalWrite(P2,HIGH);
  }
  else if (ip==2){
    digitalWrite(P1,HIGH);
    digitalWrite(P2,LOW);
  }
}

void Puerta(int id){
  if (id==0){//FUNCION PUERTA( 0:DETENIDO, 1:ABRE,
  2:CIERRA) digitalWrite(D1,LOW);
  digitalWrite(D2,LOW);
  }
  else if (id==1){
    digitalWrite(D1,HIGH);
    digitalWrite(D2,LOW);
  }
  else if (id==2){
    digitalWrite(D1,LOW);
    digitalWrite(D2,HIGH);
  }
}

```

```

void Ventana (int iv){ //FUNCION VENTANA( 0:DETENIDO, 1:ABRE, 2:CIERRA)
  if (iv==0) {
    stepv=3; //DETENGO LA VENTANA
    digitalWrite(EN,HIGH);
    nvant=nv;
  }
  else if (iv==1) {
    digitalWrite(DIR,HIGH);
    digitalWrite(EN,LOW);
    stepv=1;
    stepcount+=1; //CUENTA DE PULSOS INCREMENTO
  }
  else if (iv==2) {
    digitalWrite(DIR,LOW);
    digitalWrite(EN,LOW);
    stepv=2;
    stepcount-=1; //CUENTA DE PULSOS DECREMENTO
  }
  if (stepcount==1000 || stepcount==0){ //SI CUENTA DE PULSOS LLEGO AL
  TOPE DETENGO EL
    MOTOR nvant=nv;
    digitalWrite(EN,HIGH);
    stepv=3;
  }
  //Serial.print("Paso: "); Serial.print(stepv);
  //Serial.print("\t Nro Paso: "); Serial.println(stepcount);
  switch(stepv){
    case 1: // PULSO DE MOTOR
      digitalWrite(STEP,HIGH);
      delayMicroseconds(400);
      digitalWrite(STEP,LOW);
      delayMicroseconds(400);
      break;
    case 2:
      digitalWrite(STEP,HIGH);
      delayMicroseconds(400);
      digitalWrite(STEP,LOW);
      delayMicroseconds(400);

```

```

    break;
case 3: //DETENER MOTOR
    digitalWrite(STEP, LOW);
    digitalWrite(EN,HIGH);
    break;
}
}

void loop() {
    timeac=millis();

    if (npant!=np) {          //COMPARA EL ESTADO ANTERIOR DE LA
    PERSIANA SI ESTADO ANTERIOR DIFERENTE AL ACTUAL ACTUALIZA ESTADOS.
        Persiana(np);        //LLAMA A LA FUNCION PERSIANA CON EL DATO
    ACTUALIZADO
        npant=np;
    }
    if (ndant!=nd){          //COMPARA EL ESTADO ANTERIOR DE LA PUERTA SI
    ESTADO ANTERIOR DIFERENTE AL ACTUAL ACTUALIZA ESTADOS.
        Puerta(nd);          //LLAMA A LA FUNCION PERSIANA CON EL DATO
    ACTUALIZADO
        ndant=nd;
    }
    if (nvant!=nv){          //COMPARA EL ESTADO ANTERIOR DE LA VENTANA
    SI ESTADO ANTERIOR DIFERENTE AL ACTUAL ACTUALIZA ESTADOS.
        Ventana(nv);        //LLAMA A LA FUNCION PERSIANA CON EL DATO
    ACTUALIZADO
    }
    if (timeac-time3==2000){ //TIEMPO EN EL QUE ESTA ACTIVADA LA
    ALARMA
        timeac=millis();
        digitalWrite(R3,HIGH);
    }
    // INSTRUCCIONES PARA DETENER LA PERSIANA O PUERTA EN CASO
    LLEGUEN AL FINAL DE SU RECORRIDO
    if (digitalRead(Ptop)==0 & np==1){ //SI LA PERSIANA SE ESTA ABRIENDO
    NP=1 Y EL SENSOR SE ACTIVA PTOP=1 SE
        DETIENE. Persiana(0);

```

```

    np=0;
}
if (digitalRead(Pbot)==0 & np==2){ //SI LA PERSIANA SE ESTA CERRANDO
NP=2 Y EL SENSOR SE ACTIVA PBOT=1 SE DETIENE.
    Persiana(0);
    np=0;
}
if (digitalRead(Dtop)==0 & nd==1){ //SI LA PUERTA SE ESTA ABRIENDO
ND=1 Y EL SENSOR SE ACTIVA DTOP=1 SE
DETIENE. Puerta(0);
    nd=0;
}
if (digitalRead(Dbot)==0 & nd==2){ //SI LA PUERTA SE ESTA CERRANDO
ND=2 Y EL SENSOR SE ACTIVA PBOT=1 SE
DETIENE. Puerta(0);
    nd=0;
}
}
//INTERRUPCION PARA LA LECTURA DEL SERIAL PC O
RP void serialEvent(){
    digitalWrite(L4,!digitalRead(L4)); //FLAHS PARA INDICAR QUE LLEGARON
DATOS
    delay(100);
    digitalWrite(L4,!digitalRead(L4));

while(Serial.available()){
    char Data=Serial.read();
    switch (Data){          //VER TABLA DE ACCIONES
        case 'l' :
            nl=Serial.parseInt();
            //Serial.print("Led "); Serial.println(nl);
            if (nl==1) digitalWrite(L1,!digitalRead(L1));    //ON_OFF LED1
            else if (nl==2) digitalWrite(L2,!digitalRead(L2)); //ON_OFF LED2
            else if (nl==3) digitalWrite(L3,!digitalRead(L3)); //ON_OFF LED3
            else if (nl==4) digitalWrite(L4,!digitalRead(L4)); //ON_OFF LED4
            break;

        case 'r' :

```

```

nr=Serial.parseInt();
//Serial.print("Relay ");Serial.println(nr);
if (nr==1) digitalWrite(R1,!digitalRead(R1)); //ON_OFF RELAY1
else if (nr==2) digitalWrite(R2,!digitalRead(R2)); //ON_OFF RELAY2
else if (nr==3) { //ON_OFF RELAY3
    time3=millis(); //ACTUALIZO EL TIEMPO EN EL QUE
PERMANECERA ENCENDIO LA ALARMA
    //Serial.println(time3);
    digitalWrite(R3,!digitalRead(R3));
}
break;

case 'p' : //PERSIANA( 0:DETENIDO, 1:ABRE,
2:CIERRA)
    np=Serial.parseInt();
    /*Serial.print("Persiana ");
    if (np==0) Serial.println("Apagado");
    else if (np==1) Serial.println("Sube");
    else if (np==2) Serial.println("Baja");*/
    break;

case 'd' : //PUERTA( 0:DETENIDO, 1:ABRE,
2:CIERRA)
    nd=Serial.parseInt();
    /*Serial.print("Puerta ");
    if (nd==0) Serial.println("Apagado");
    else if (nd==1) Serial.println("Abre");
    else if (nd==2) Serial.println("Cierra");*/
    break;

case 'v' : //VENTANA( 0:DETENIDO, 1:ABRE,
2:CIERRA)
    nv=Serial.parseInt();
    /*Serial.print("Ventana ");
    if (nv==0) Serial.println("Apagado");
    else if (nv==1) Serial.println("Abre");
    else if (nv==2)
Serial.println("Cierra");*/ break;

```

```

        case 't':                //DETENGO TODA ACCION DE LOS
MOTORES
        np=0;
        nd=0;
        nv=0;
        //nr=3;
        break;
        case 'm':                //DENTENGO PERSIANA
        np=0;
        break;
        case 'n':                //DETENGO PUERTA
        nd=0;
        break;
        case 'o':                //DETENGO VENTANA
        nv=0;
        break;
    }
}
}
//INTERRUPCION A LA LECTURA DEL BLUETOOTH ESTA ES IGUAL A LA
INTERRUPCION DEL RP O PC.
void serialEvent1(){          //FLAHS PARA INDICAR QUE
LLEGARON DATOS
    digitalWrite(L4,!digitalRead(L4));
    delay(100);
    digitalWrite(L4,!digitalRead(L4));

    while(Serial1.available()){
        char Data=Serial1.read();
        switch (Data){
            case 'l' :
                nl=Serial1.parseInt();
                //Serial.print("Led "); Serial.println(nl);
                if (nl==1) digitalWrite(L1,!digitalRead(L1)); else
                if (nl==2) digitalWrite(L2,!digitalRead(L2)); else
                if (nl==3) digitalWrite(L3,!digitalRead(L3)); else
                if (nl==4) digitalWrite(L4,!digitalRead(L4));
                break;

```

```
case 'r' :
nr=Serial1.parseInt();
//Serial.print("Relay ");Serial.println(nr);
if (nr==1) digitalWrite(R1,!digitalRead(R1)); else
if (nr==2) digitalWrite(R2,!digitalRead(R2)); else
if (nr==3) {
    time3=millis();
    Serial.println(time3);
    digitalWrite(R3,!digitalRead(R3));
}
break;
```

```
case 'p' :
np=Serial1.parseInt();
/*Serial1.print("Persiana ");
if (np==0) Serial.println("Apagado");
else if (np==1) Serial.println("Sube");
else if (np==2)
Serial.println("Baja");*/ break;
```

```
case 'd' :
nd=Serial1.parseInt();
/*Serial.print("Puerta ");
if (nd==0) Serial.println("Apagado");
else if (nd==1) Serial.println("Abre");
else if (nd==2)
Serial.println("Cierra");*/ break;
```

```
case 'v' :
nv=Serial1.parseInt();
/*Serial.print("Ventana ");
if (nv==0) Serial.println("Apagado");
else if (nv==1) Serial.println("Abre");
else if (nv==2)
Serial.println("Cierra");*/ break;
```

```
case 't':
    np=0;
```

```
    nd=0;
    nv=0;
    break;
    case 'm':
    np=0;
    break;
    case 'n':
    nd=0;
    break;
    case 'o':
    nv=0;
    break;
    }
    }
}
```

ANEXOS DE COSTOS

Lista de Materiales

Mecanicos

Cantidad	Nombre de la pieza	Descripción	Precio
01	Cinta negra	Aislante	S/. 3.00
02	Chupones de ventana	Transparentes	S/. 2.00
01	Correa dentada	2 m	S/. 15.00
01	Varilla roscada	1/8" x 1	S/. 13.00
01	Paquete tuercas	hexagonales	S/. 3.00
01	Motor-reductor TSUKASA	G-35F-AG-18-A275 0,5 A	S/. 30.00
01	Motor-reductor SWF VALEO	SWF404304 3A	S/. 50.00
01	Motor paso a paso MINEBEA-M	17PM-K142B (Bipolar) 1A	S/. 35.00
01	Tuerca larga esparrago	hexagonal	S/. 2.00
			S/. 153.00

Electricos

Cantidad	Nombre de la pieza	Descripción	Precio
01	Rollo cable azul	100 m - # 14	S/. 67.00
01	Tablero adosable	40 x 30 x 15 cm	S/. 40.00
01	Fuente de poder	12 v - 10 Amp	S/. 120.00
01	Bateria duracel	9 v	S/. 12.00
01	Caja modular + tomacorriente	_____	S/. 10.00
01	Interruptor	Doble	S/. 5.00
01	Llave monofasica	Marca Bticino	S/. 20.00
04	Finales de carrera	Color verde	S/. 4.00
01	Conector 2 pines	_____	S/. 1.70
01	Conector 3 pines	_____	S/. 2.40
01	Espadin hembra	10 mm x 90 mm	S/. 3.00
01	Modulo relays	8 canales	S/. 40.00
01	Cargador celular	_____	S/. 10.00
15 m	Cable de internet	UTP Cat 6	S/. 25.00
4	Conectores de red	_____	S/. 4.00
10	Cable telefono	de 4 hilos	S/. 8.00
04	Canaletas	40 x 25 x 2000	S/. 32.00
02	Canaletas	20 x 20 x 2000	S/. 14.00
01	Camara TpLink	blanca movable	S/. 40.00
			S/. 458.10

Electrónicos

Cantidad	Nombre de la pieza	Descripción	Precio
01	Modulo relays	8 canales	S/. 40.00
01	Arduino Uno	R3	S/. 80.00
01	Arduino Mega	2560	S/. 200.00
01	Modulo reconocimiento de voz	Easy VR Shield Arduino	S/. 150.00
01	Arduino Modulo Bluetooth	Bluetooth Hc-05	S/. 20.00
01	Arduino Modulo Bluetooth	Bluetooth Hc-06	S/. 20.00
			S/. 510.00

TOTAL

S/. 1,121.10