



UNIVERSIDAD DE LOS ANDES

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA ELÉCTRICA

www.bdigital.ula.ve

SERVIDOR DE CONTROL DOMÓTICO

Br. Gleysa A. Muñoz M.

Mérida, diciembre 2020

Reconocimiento-No comercial- Compartir igual

UNIVERSIDAD DE LOS ANDES
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA ELÉCTRICA

SERVIDOR DE CONTROL DOMÓTICO

Trabajo de Grado presentado como requisito parcial para optar al título de
Ingeniero Electricista

Br. Gleysa A. Muñoz M.

Tutor: Prof. Francisco J. Vilorio

Mérida, diciembre 2020

UNIVERSIDAD DE LOS ANDES
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA ELÉCTRICA

SERVIDOR DE CONTROL DOMÓTICO

Br. Gleysa Muñoz

www.bdigital.ula.ve

Trabajo de Grado, presentado en cumplimiento parcial de los requisitos exigidos para optar al título de Ingeniero Electricista, aprobado en nombre de la Universidad de Los Andes por el siguiente Jurado.

Ana Angelissis Arraiz Budovalchew

Luís Ramón Araujo Rangel

RESUMEN

Gleysa A. Muñoz M. Servidor de control domótico. Universidad de Los Andes.

Tutor: Francisco J. Vilorio. M. abril 2020

Este trabajo está enfocado en el diseño e implementación de un sistema de control domótico basado en un software de código abierto que proporciona la integración de diferentes sistemas y tecnologías de dispositivos de automatización del entorno mediante el uso simultáneo de la electricidad, la electrónica, la informática y las telecomunicaciones, garantizando así un nivel de automatismo que le permite al usuario el control o monitorización de su entorno de forma fácil y segura. El sistema de control domótico a implementar usa una plataforma de software/hardware basada en PHP, MySQL, JavaScript, HTML5, CSS3 y lenguaje basado en C++; así como el uso de sensores, actuadores y microprocesadores, en los que el módulo de adquisición de datos opera de manera inalámbrica, por lo que el sistema monitorea de manera remota diversas variables de interés a través de la tecnología WIFI. Para este tipo de comunicación, se utilizó el protocolo TCP/IP, una interfaz gráfica que actúa como cliente y los microprocesadores, que actúan como cliente/servidor. Finalmente, el procesamiento, control, almacenamiento y presentación de la información de las variables analógicas y digitales se presenta de una manera sencilla, intuitiva y natural con elementos visuales como indicadores y gráficas a través de la interfaz de usuario creada.

Descriptor: Servidor, domótica, WIFI, software, interfaz.

ÍNDICE GENERAL

APROBACIÓN.....	i
RESUMEN	ii
INTRODUCCIÓN	1
CAPITULO I: EL PROBLEMA DE INVESTIGACIÓN.....	3
1.1 Planteamiento del problema.....	3
1.2 Objetivos	5
1.3 Justificación	5
CAPITULO II: MARCO TEÓRICO.....	8
2.1 ANTECEDENTES DE LA INVESTIGACIÓN	8
2.2 DOMÓTICA.....	11
2.2.1 Elementos de un sistema de control domótico.....	12
2.2.2 Tipos de sensores	14
2.2.3 Tipos de actuadores:.....	15
2.2.4 Normativas existentes para los entornos domóticos	16
2.3 REDES DE TRANSMISIÓN DE DATOS.....	16
2.3.1 Redes de área local (LAN).....	17
2.3.2 El protocolo TCP/IP.....	17
2.3.4 El TCP (<i>Transmission Control Protocol</i>)	17
2.3.5 El IP (<i>Internet Protocol</i>)	17
2.3.6 Protocolo de transferencia de hipertexto.....	17
2.4 METODOLOGÍA DE DESARROLLO DE SOFTWARE	18
2.4.1 Modelo secuencial o cascada	18
2.4.2 Etapa 1: analisis	18
2.4.3 Etapa 2: diseño	19
2.4.4 Etapa 3: codificación.....	20
2.4.5 Etapa 4: pruebas	20
2.4.6 Etapa 5: implementación.....	20
2.5 SOFTWARE LIBRE Y LENGUAJES DE ROGRAMACIÓN	21
2.5.1 Base de datos.....	21
2.5.2 Lenguaje de etiquetas HTML5	21
2.5.3 Lenguaje de clases CSS3	21
2.5.4 Lenguaje de programación JavaScript	21

2.5.5 Lenguaje de consulta SQL	22
2.5.6 Lenguaje de programación PHP	22
2.5.7 Lenguaje de programación basado en C++	22
2.5.8 Gestor de base de datos MySQL.....	22
CAPITULO III: MARCO METODOLÓGICO	23
3.1 Tipo de investigación.....	23
3.2 Diseño de investigación	23
3.3 Metodología para el diseño del sistema de control domótico	23
3.4 ETAPA 1: ANALISIS	24
3.4.1 Requerimientos generales del sistema	25
3.4.2 Técnicas y herramientas para la recopilación de información	26
3.4.3 Definición de la infraestructura tecnológica	27
3.4.4 Análisis de la plataforma hardware/software.....	27
3.4.5 Requisitos en hardware y software	27
3.4.6 Requisitos de implementación.....	28
3.5 ETAPA 2: DISEÑO	29
3.5.1 Especificaciones técnicas en hardware	30
3.5.2 Especificaciones técnicas en software	31
3.5.3 Microcontrolador ESP8266.....	31
3.5.4 Restricciones y limitaciones de diseño	33
3.5.5 Interconexión de los dispositivos	34
3.5.6 Interconexión de los sensores	34
3.5.7 Interconexión de los actuadores	35
3.5.8 Circuitos necesarios para la conexión.....	36
3.5.9 Esquema de conexión del sistema domótico.....	37
3.5.10 Base de datos.....	38
3.5.11 Descripción de la interfaz	47
3.5.12 Prototipos de pantalla.....	47
3.5.13 Identidad visual	50
3.6 ETAPA 3: CODIFICACIÓN	52
3.6.1 Elección de los lenguajes de condificación.....	52
3.6.2 Actividades de la codificación	53
3.6.3 Asignación de un identificador <i>id</i> a cada dispositivo	56
3.6.4 HTTP <i>request</i> de los ESP8266.....	57

3.6.5 Recepción de datos de los ESP8266	58
3.7 ETAPA 4: PRUEBAS	60
CAPITULO IV: IMPLEMENTACIÓN	63
4.1 <i>Login</i>	633
4.2 Crear proyecto de automatización.....	63
4.2.1 Primer paso: configurar proyecto.....	64
4.2.2 Segundo paso: configurar ambientes.....	64
4.2.3 Tercer Paso: Cargar plano.....	65
4.2.4 Cuarto paso: finalizar configuración.....	65
4.3 Pantalla principal	66
4.4 Menú de la plataforma	66
4.5 Items o dispositivos de automatización	67
4.6 Proyectos.....	71
4.7 Ambientes	73
4.8 Agregar ítems.....	76
4.9 Configuración.....	81
4.10 Configuración de ítems	82
4.11 Eliminar ítem asociado.....	85
4.12 Sensores	86
4.13 Base de datos.....	87
4.14 Control	89
4.15 Programar actuadores.....	91
CONCLUSIONES.....	93
RECOMENDACIONES.....	93
REFERENCIAS	95

ÍNDICE DE FIGURAS

Figura	pp.
Figura 2.1 Fases del modelo secuencial de desarrollo de software.....	18
Figura 3.1 Etapas del proceso metodológico	24
Figura 3.2 Descripción del proceso de análisis del sistema de control domótico	25
Figura 3.3. Requerimientos generales del sistema domótico.	26
Figura 3.4 Requisitos en hardware para el sistema de control domótico.....	28
Figura 3.5 Requisitos en software para el sistema de control domótico	28
Figura 3.6 Descripción del proceso de diseño del sistema de control domótico.	30
Figura 3.7 Especificaciones técnicas en software del servidor de control domótico	31
Figura 3.8 Especificaciones técnicas del microcontrolador ESP8266	32
Figura 3.9 Parámetros de WIFI correspondientes al ESP8266	32
Figura 3.10 Parámetros de Hardware correspondientes al ESP8266	33
Figura 3.11 Parámetros de Software correspondientes al ESP8266	33
Figura 3.12 Interconexión de los dispositivos del sistema.....	34
Figura 3.13 Interconexión de los sensores del sistema	35
Figura 3.14 Interconexión de los actuadores del sistema.....	35
Figura 3.15 Conexión cableada de los sensores y actuadores.....	36
Figura 3.16 Circuito de alimentación de potencia	36
Figura 3.17 Circuito de alimentación auxiliar	37
Figura 3.18 Circuito de interconexión del nodo a los sensores y actuadores.....	37
Figura 3.19 Esquema de conexión del sistema domótico	38
Figura 3.20 Estructura de la base de datos del sistema	39
Figura 3.21 Estructura de la tabla ambientes	39
Figura 3.22 Estructura de la tabla configuración	40
Figura 3.23 Estructura de la tabla proyecto_config	41
Figura 3.24 Estructura de la tabla fc37 como referencia	42
Figura 3.25 Estructura de la tabla ESP8266	42
Figura 3.26 Estructura de la tabla ítems.....	43
Figura 3.27 Estructura de la tabla programar_actuadores.....	44
Figura 3.28 Estructura de la tabla proyectos	44
Figura 3.29 Diagrama entidad-relación de la base de datos 1.....	45
Figura 3.30 Diagrama entidad-relación de la base de datos 2.....	45

Figura 3.31 Prototipo de pantalla para menú de la interfaz de usuario.....	48
Figura 3.32 Prototipo de pantalla de <i>login</i>	49
Figura 3.33 Prototipo de pantalla de inicio.....	49
Figura 3.34 Boceto final del logotipo de la plataforma.....	51
Figura 3.35 Logotipo de la plataforma.....	51
Figura 3.36 Aplicación logotipo de la plataforma.....	51
Figura 3.37 Esquema de los lenguajes de codificación del software.....	53
Figura 3.38 Diagrama de flujo para la programación de los ESP8266.....	54
Figura 3.39 Fragmento de código de programación de los ESP8266.....	57
Figura 3.40 Ejemplo de datos enviados al servidor.....	58
Figura 3.41 Envío de datos desde el servidor a los nodos.....	59
Figura 3.42 Fragmento de código de programación para la recepción de datos 1.....	60
Figura 3.43 Fragmento de código de programación para la recepción de datos 2.....	60
Figura 3.44 Esquema de circuito electrónico de prueba 1.....	61
Figura 3.45 Esquema de circuito electrónico de prueba 2.....	62
Figura 4.1 Interfaz de <i>login</i>	63
Figura 4.2 Creación de proyecto.....	64
Figura 4.3 Segundo paso para crear proyecto.....	64
Figura 4.4 Selección de proyecto a utilizar.....	65
Figura 4.5 Pantalla de inicio.....	66
Figura 4.6 Menú principal de la plataforma.....	67
Figura 4.7 Ítems o dispositivos de automatización predeterminados.....	68
Figura 4.8 Configuración o personalización de ítems.....	68
Figura 4.9 Personalizando ítem de entrada 1.....	69
Figura 4.10 Personalizando ítem de entrada 3.....	70
Figura 4.11 Lista de dispositivos agregados.....	70
Figura 4.12 Lista de dispositivos agregados 3.....	71
Figura 4.13 Proyectos creados.....	71
Figura 4.14 Proyectos creados 2.....	72
Figura 4.15 Interfaz para borrar proyecto.....	72
Figura 4.16 Interfaz para configuración de proyectos.....	73
Figura 4.17 Sub menú para la configuración de proyectos.....	73
Figura 4.18 Configuración de ambientes.....	74
Figura 4.19 Agregar imagen de ambiente.....	74

Figura 4.21 Sección de pantalla de configuración de proyectos	75
Figura 4.22 Pantalla de configuración de proyectos para ambiente habitación	76
Figura 4.23 Pantalla de configuración de proyectos para ambiente baño	76
Figura 4.24 Lista de dispositivos para ser asociados a un proyecto.....	777
Figura 4.25 Configurar ubicación de dispositivo en el plano principal	78
Figura 4.26 Configurar ubicación de dispositivo en el ambiente.....	78
Figura 4.27 Pantalla de configuración de proyecto con un ítem asociado	79
Figura 4.28 Pantalla de configuración de proyecto con un ítem asociado en entorno	79
Figura 4.29 Pantalla de configuración de proyecto con varios dispositivos asociados.....	80
Figura 4.30 Pantalla de configuración de proyecto con varios dispositivos asociados en habitación.....	80
Figura 4.31 Pantalla para añadir microcontrolador al sistema.....	81
Figura 4.32 Hardware asociado al sistema.....	82
Figura 4.33 Asociación de ESP8266 a un proyecto	82
Figura 4.34 Métodos para acceder a la configuración del ítem	83
Figura 4.35 Pantalla de configuración de proyecto con varios dispositivos asociados en sala de estar	83
Figura 4.36 Sub menú para ítem no configurado.....	84
Figura 4.37 Selección de hardware para dispositivos asociados.....	85
Figura 4.38 Dispositivos configurados y no configurados.....	85
Figura 4.39 Eliminar ítems asociados a un proyecto	86
Figura 4.40 Gráfica de sensores.....	87
Figura 4.41 Consultar base de datos del sistema	87
Figura 4.42 Consulta en la base de datos de un proyecto seleccionado.....	88
Figura 4.43 Pantalla de consulta de base de datos para sensor dht11	88
Figura 4.44 Interfaz para accionar actuadores	89
Figura 4.45 Ejecutar dispositivos actuadores.....	89
Figura 4.46 Control de actuador desde pantalla principal.....	90
Figura 4.47 Control de actuador desde pantalla principal 2.....	90
Figura 4.48 Accionar sg-90 o 28by-48 desde pantalla principal	91
Figura 4.49 Accionar dispositivos de salida	91
Figura 4.50 Transición de ventanas al seleccionar condición.....	92
Figura 4.51 Programación para accionar dispositivos de salida guardada.....	92

ÍNDICE DE TABLAS

Tabla	pp.
Tabla 1. Hardware en sensores y actuadores	30
Tabla 2. Asignación de identificadores en el nodo 1	56
Tabla 3. Asignación de identificadores en el nodo 2	56

www.bdigital.ula.ve

INTRODUCCIÓN

Un entorno domótico es aquel que incorpora mecanismos de automatización que son capaces de comunicarse entre ellos y con los usuarios a través de un sistema de control que coordina y controla todos los dispositivos electrónicos con el fin de mejorar la calidad de vida de los usuarios, aumentar su seguridad, reducir el trabajo doméstico, racionalizar los diferentes consumos y optimizar los recursos [1].

Así pues, este proyecto surge a partir de la problemática actual de generar ambientes domóticos que permitan la integración de diferentes dispositivos tecnológicos con diferentes protocolos de comunicación a través de un sistema que agrupe dichos dispositivos y permita relacionarlos entre sí.

La presente investigación aborda como principal propósito el diseño e implementación de un entorno de control para la integración de los diferentes elementos que conforman un sistema domótico, el cual consiste en un software de código abierto que permite el control de actuadores y monitorización de sensores mediante una plataforma de software/hardware basada en PHP, MySQL, JavaScript, HTML5, CSS3, SQL, lenguajes basados en C, microcontroladores, entre otros dispositivos electrónicos. El presente trabajo de grado se encuentra estructurado de la siguiente manera:

En el Capítulo I se proporciona una amplia descripción de la situación objeto de estudio, planteando así el problema de investigación, sus objetivos y respectiva justificación.

Adicionalmente, el Capítulo II está basado en el aspecto preliminar de efectuar una revisión bibliográfica del estado del arte de los sistemas domóticos, además de establecer las bases conceptuales de un sistema que integre diferentes tipos de sensores y actuadores, así como los mecanismos y protocolos de comunicación a usarse entre la central domótica y los sensores y actuadores que forman parte del entorno.

El Capítulo III proporciona una descripción detallada del proceso de desarrollo del sistema domótico empleando un modelo de metodología que define los criterios necesarios para el diseño del sistema, el cual consiste en un proceso secuencial constituido por un conjunto de etapas que establecen el orden de las fases del proceso de desarrollo del sistema software/hardware del entorno domótico, así como su implementación y pruebas.

Seguidamente, el Capítulo IV detalla la implementación del sistema domótico, facilitando una descripción sobre el manejo y uso de la interfaz de usuario para la monitorización de sensores y control de actuadores asociados al entorno domótico que se desea automatizar.

www.bdigital.ula.ve

CAPITULO I:

EL PROBLEMA DE INVESTIGACIÓN

A continuación, se presentará una amplia descripción de la situación objeto de estudio de la presente investigación, la cual consiste en el planteamiento del problema de investigación, sus objetivos y su respectiva justificación.

1.1 PLANTEAMIENTO DEL PROBLEMA

El avance de las nuevas tecnologías de la información y las comunicaciones, así como el continuo desarrollo tecnológico de los circuitos integrados y la electrónica, han permitido contar en la actualidad con una diversidad de dispositivos novedosos y de bajo costo, cuyas características permiten su uso como herramientas eficientes en una amplia gama de aplicaciones, lo cual ha posibilitado a las personas mejorar efectivamente su calidad de vida y a partir de estas nuevas tecnologías ha surgido el concepto del internet de las cosas [2]. El “Internet de las Cosas” puede definirse como una tecnología conformada por un conjunto de “cosas” o dispositivos conectados a internet e interrelacionados entre sí, que envían o reciben datos para analizarlos o actuar en base a ellos [3]. El internet de las cosas es una tecnología que pretende consolidarse como una de las más revolucionarias, con mayor proyección y que cambiará al planeta y el estilo de vida de sus habitantes mediante la creación de ambientes domóticos o casas inteligentes [4].

Una casa inteligente o un entorno domótico es aquel que incorpora la integración de distintas tecnologías mediante el uso simultáneo de la electrónica, la informática y las telecomunicaciones, lo cual permite un control inteligente del entorno, proporcionando de esta manera un nivel de automatismo que le permite al usuario el control o monitorización de su entorno de forma fácil y segura [5].

Ahora bien, se estima que en 2020 haya 12,8 mil millones de dispositivos conectados a la red y que en edificios de uso no residencial haya cerca de 8 mil millones de dispositivos conectados [6], esto sugiere una necesidad real de la

integración de los mismos a través de un sistema de control domótico que agrupe todos estos dispositivos y que le facilite al usuario la interacción con su entorno.

Hoy en día es posible generar ambientes inteligentes contruidos alrededor de dispositivos basados en el Internet de las Cosas, en tecnologías alámbricas e inalámbricas, así como dispositivos electrónicos disponibles comercialmente, no obstante, el modo en que estos dispositivos se relacionan, comunican e interactúan entre sí, su facilidad de uso, su controlabilidad y flexibilidad son problemáticas que no se han resuelto completamente y sobre las cuales muchas empresas actualmente se encuentran trabajando, razón que ha dificultado la distribución de ambientes inteligentes en el mercado. Cada vez es mayor la cantidad de dispositivos electrónicos que tenemos a nuestro alrededor y a pesar de que todos ellos están diseñados para mejorar el estilo de vida y proporcionar un mayor nivel de confort, comúnmente no disponen de un lenguaje común o una forma sencilla de comunicarse entre sí. Lo cual provoca una necesidad real de comunicar unos con otros para facilitar su uso y aplicación [7].

Así mismo, otra problemática que presentan los ambientes inteligentes es la falta de una interfaz de usuario que sea amigable, común e intuitiva. Esto se debe a que las soluciones actualmente disponibles en el mercado están diseñadas como aplicaciones cerradas, las cuales sólo conectan un pequeño número de dispositivos y son de propiedad de una empresa, por lo que el usuario debe pagar altos costos por el uso de estas interfaces que finalmente no interconectan diferentes tecnologías para controlar su entorno y la mayoría de veces, resultan ser poco intuitivas y difíciles de utilizar.

Por lo que, la problemática presentada por los ambientes domóticos consiste en la ausencia de soluciones integrales, existencia de equipos de difícil manejo, dificultades para el mantenimiento y demás elementos propios de un mercado dirigido desde la oferta y que se muestra poco sensible a los deseos de los usuarios [8]. En virtud de ello, cada día se hace más importante la creación de plataformas o sistemas de control domótico que permitan una solución a estas dificultades, logrando así la integración de estos diferentes dispositivos y sus correspondientes tecnologías.

Para abordar la problemática planteada, en el presente trabajo de grado se propone el diseño e implementación de un sistema de Control domótico basado en un software que hace uso de los lenguajes de código abierto PHP, MySQL, JavaScript, HTML5, CSS3 y lenguaje C, el cual permitirá la integración de los diferentes sistemas y tecnologías de los dispositivos de automatización que componen el sistema domótico, permitiendo ser gestionado a través de una interfaz de usuario amigable e intuitiva.

1.2 OBJETIVOS

A continuación, se expondrán los objetivos formulados en la presente investigación para abordar la problemática anteriormente planteada:

OBJETIVO GENERAL

Diseñar e implementar un entorno de control para integrar los diferentes elementos que conforman un sistema domótico.

OBJETIVOS ESPECÍFICOS

- Revisión bibliográfica del estado del arte de los sistemas domóticos.
- Establecer las bases conceptuales de un sistema que integre diferentes tipos de sensores y actuadores.
- Establecer los mecanismos de comunicación a usarse entre la central domótica, los sensores y actuadores.
- Diseñar una interfaz de usuario basada en PHP, MySQL, JavaScript, HTML5 y CSS3.
- Implementar una plataforma software/hardware que permita el monitoreo de sensores y el control de actuadores.

1.3 JUSTIFICACIÓN

La domótica cada día tiene más auge y aunque las personas todavía no están arraigadas a los beneficios que ofrecen los sistemas de control domóticos, es un hecho que en un futuro cercano la domótica estará instalada en cualquier vivienda

o entorno debido a que pretende mejorar la calidad de vida de sus habitantes y ampliar sus posibilidades de comunicación, automatizando procesos domésticos e intercomunicando tanto estos procesos como los residentes del hogar o entorno entre sí y con el exterior [8].

Un servidor de control domótico representa grandes ventajas de seguridad, confort, automatización, ahorro y gestión energética del entorno. Estas ventajas se deben a que los servidores de control domóticos gestionan dispositivos electrónicos, cuyo control contribuye al ahorro energético, ahorro de agua y gas [5].

Los sistemas domóticos que representan ventajas en gestión de la energía ya que administran inteligentemente la iluminación, la climatización, el riego, el encendido y apagados de los electrodomésticos, entre otros dispositivos eléctricos, permitiendo el mejor aprovechamiento de los recursos naturales [9]. Por lo tanto, con un servidor domótico, el encendido y apagado de los dispositivos o elementos que conforman el entorno puede ser programado, lo que permite ahorrar en el consumo de los mismo. Es posible garantizar que todas las luces del entorno estén apagadas y que la temperatura de referencia de la climatización esté en modo ahorro cuando no haya nadie presente en el ambiente inteligente. Lo cual, sin duda, tiene efectos muy positivos en la economía y contribuye ahorro energético y al cuidado del ambiente [10]. Por otro lado, un sistema de control domótico puede aportar importantes ventajas de seguridad al entorno, como lo es la detección de incendios, la detección de fugas de gas, monitorización a distancia de cámaras de videovigilancia, gestión de alarmas de seguridad, simulación de presencia a través del apagado y encendido de luces a distancia, entre otros beneficios de seguridad.

Ahora bien, los sistemas de control domótico también pueden aportar grandes beneficios en términos de comunicación entre el entorno y el usuario, pudiendo este último interactuar en todo momento con el entorno en cualquier ocasión y en cualquier lugar. El empleo de un sistema domótico facilita sin lugar a dudas, las tareas y actividades a realizarse en el entorno automatizado, lo que permite una gran comodidad a la hora de ejecutar acciones desde la interfaz de usuario del servidor. Esto quiere decir que la manera más agradable y confortable de gestionar el hogar es automatizando la mayor cantidad posible de elementos o

dispositivos. Por lo tanto, encontramos una gran variedad de beneficios como lo establece Redolfi:

Entre ellos, tenemos el ahorro energético, el confort, la seguridad, la continua intercomunicación y la asistencia a personas enfermas. En el ahorro de la energía, la domótica nos permite consumir la potencia eléctrica de forma eficiente e inteligente. En el confort podemos controlar la iluminación, la temperatura, incluso apagar o encender los electrodomésticos de manera automatizada, como también interactuar con ellos. Por el lado de la seguridad, nos permite una rápida detección de problemas e intrusos, reduciendo el tiempo de respuesta de los servicios de emergencia. También tenemos la posibilidad de simular nuestra presencia. En lo que respecta a la comunicación y el entretenimiento, podemos acceder desde nuestra casa al universo de la información, y desde cualquier lugar del mundo a la información de nuestra casa, gracias a Internet [11].

Por las razones ya expuestas, los sistemas de control domóticos son cada vez más requeridos en la actualidad debido a la búsqueda de seguridad y control de una diversidad de dispositivos o variables en un entorno. La Facultad de Ingeniería ha desarrollado gran cantidad de proyectos relacionados con los sistemas de control domótico, no obstante, todos son proyectos aislados de automatización que intentan resolver un problema a la vez. Algunos de estos proyectos están relacionados con el consumo eléctrico, con el control de iluminación, el manejo remoto de cámaras, el manejo remoto de dispositivos usando RF, Bluetooth, XBee e Ethernet. Debido a que han sido proyectos independientes es muy difícil su integración. Previamente Jefferson Núñez hizo un intento para alcanzar este objetivo al desarrollar un sistema que permite integrar diversos tipos de sensores, sin embargo, la problemática aún no ha sido resuelta.

CAPITULO II: MARCO TEÓRICO

En este apartado se efectuará el estudio del estado del arte en el cual se expondrán las soluciones más destacadas que pueden encontrarse en previas investigaciones, así como los conceptos y las tecnologías más relevantes que son requeridas para el diseño e implementación del sistema domótico.

2.1 ANTECEDENTES DE LA INVESTIGACIÓN

Horro, M. (2008). En su trabajo de grado para obtener el título de Ingeniero Telemático, titulado “*Programa en Java de un entorno de control para un sistema domótico*” en la Universidad Carlos III de Madrid, persigue desarrollar e implementar una aplicación que permita controlar y visualizar el estado de todos los componentes que forman parte de un sistema domótico para la infraestructura de la empresa Ergodomus S.L. En su investigación Horro empleó módulos de control desarrollados por la empresa y sus respectivos protocolos de comunicación, entre los cuales, principalmente utilizó comunicaciones vía puerto serial. Por último, Horro consiguió satisfacer las expectativas de la empresa pudiendo facilitar enormemente la gestión y monitorización de la misma [12].

Durán, A. (2009). En su tesis de Grado para obtener el título de Ingeniero Industrial, titulada “*Instalación Domótica de una vivienda unifamiliar*”. Desarrollada en la Universidad Pontificia Comillas, Madrid, se plantea como objetivo principal la instalación de un sistema de Control domótico en una vivienda, con lo cual empleó una metodología en la que efectuaba un estudio comparativo de los diferentes sistemas domóticos actuales, distribuidos y centralizados. Posteriormente realizó un análisis de los componentes necesarios de toda aplicación domótica y sus costes, así como la estructura física y lógica y el modelo de gestión o arquitectura a aplicar en el sistema de control domótico. Finalmente, Durán obtuvo los datos necesarios para la implementación del sistema de control y

estableció la ventaja existente en instalar el sistema en una vivienda que se encuentra en fase de construcción [13].

Nuñez, J. (2014). En su trabajo de grado titulado *“Diseño e implementación de un entorno de control para un sistema domótico”* para optar por el título de Ingeniero electricista en la Universidad de Los Andes, implementó una solución viable para la automatización de cargas en el hogar creando una red que permitió controlar diversos dispositivos mediante una interfaz de control domótico. Los entornos de monitoreo y control que se desarrollaron en esta tesis estaban basados en software libre, al igual que los de la presente investigación. Nuñez utilizó lenguajes de programación de páginas web y Visual Basic. Sin embargo, el protocolo de comunicación empleado para el desarrollo del entorno domótico fueron los protocolos industriales RS232 y RS485, cuyo medio de comunicación es el cable de par trenzado tipo UTP con conexión RJ45, de manera que el sistema de control domótico implementado utilizaba una adquisición de datos alámbrica. Por último, Nuñez obtuvo muy buenos resultados en la implementación de la comunicación serial y en la creación de la red 485, proporcionando una solución de implementación para un sistema domótico con redes cableadas [14].

J. Quinde, J. Israel, B. Zhingre y P. Buele (2015) en su tesis titulada *“Diseño de software para el control y monitoreo domótico en una minicomputadora basado en protocolo TCP/IP”* para la obtención del título de Ingenieros electrónicos en la Universidad Politécnica Salesiana de Cuenca, persiguieron el propósito de diseñar un software de control domótico, empleando una metodología que consistía en la definición de los parámetros necesarios para el diseño del sistema que efectúa el control y monitoreo de actuadores y sensores mediante una comunicación basada en el protocolo TCP/IP, la cual basaba en un recuento sobre temas existentes desarrollados para conocer sus características, ventajas y facilidades de uso, posteriormente recurrieron a la aplicación de herramientas que les permitieron el desarrollo del software basado en Android y el hardware necesarios para las pruebas y su posterior implementación. Finalmente, se determinó que fue factible la implementación de una aplicación Android que permitió a los usuarios mantener el control de su hogar desde cualquier parte del mundo a través de su teléfono inteligente [15].

Sol, V. (2016). En su trabajo de grado titulado “*Sistema para la monitorización y asistencia en el hogar usando OpenHAB y robótica móvil*” para la obtención del título de Ingeniero en Informática de la Universidad de Málaga, se enfoca en el desarrollo y creación de un sistema domótico, informático, robótico y automático que permita la monitorización de personas mayores. Obtuvo como resultado la incorporación de un sistema informático, robótico y automático que ofreció la posibilidad de proporcionar una visión clara del estado de la persona mayor que se deseaba monitorizar a través de la interfaz gráfica proporcionada por OpenHAB [16].

García, A. (2017). En su trabajo de grado titulado “*Plataforma domótica basada en Raspberry Pi y el protocolo MQTT*” para la obtención del título de Ingeniero de tecnologías de comunicación en la Universidad de Granada, propone la creación de un servidor domótico de bajo coste en el que emplea herramientas de licencias *open-source* o de coste reducido como el servidor domótico OpenHAB, Raspberry Pi y la placa de desarrollo ESP8266 NodeMCU, entre otros elementos. La metodología empleada en su tesis fue descrita en diversas fases de desarrollo, empezando por la definición del proyecto, la especificación de los requisitos, el estudio de las alternativas disponibles en el mercado, la familiarización con las herramientas que utilizó y finalmente el diseño, configuración, pruebas e implementación del sistema de control. García consiguió obtener una solución de bajo coste compatible con la mayoría de los dispositivos disponibles en el mercado, escalabilidad al permitir la integración de una gran variedad de elementos en el sistema domótico, así como seguridad en las comunicaciones a través del uso de protocolos como HTTPS o MQTTS [17].

Serna, P. (2018). En su tesis titulada “*Vivienda inteligente: proyecto domótico de bajo coste para una vivienda unifamiliar*” para la obtención del título de Ingeniero en tecnologías Industriales de la Escuela Técnica Superior de Ingenieros Industriales y de Telecomunicación de Universidad de Cantabria expuso como objetivo el desarrollo de un sistema de automatización que permitiera la monitorización del consumo eléctrico y la eficiencia en la iluminación de una vivienda unifamiliar mediante el empleo de dispositivos de coste reducido, cuya metodología consistió en el diseño y selección de la parte hardware de bajo coste,

en la que se incluyen los sensores y circuitos de control empleados y, por otro lado, la parte software que aglutina el código de programación de los circuitos de control, posteriormente efectuó las pruebas necesarias para lograr la implementación del sistema. Finalmente, Serna concluyó que fue posible mejorar la eficiencia energética en la iluminación de la vivienda, ya que el sistema de control permitió monitorizar e informar al usuario del consumo de energía eléctrica, con el objetivo de poder reducirlo y adicionalmente fue capaz de regular la intensidad lumínica de manera automática en función de las condiciones de luz de cada momento, garantizando de esta manera la eficiencia energética en términos de la iluminación [18].

Velasco, L. (2019). En su trabajo de grado titulado “*Interfaz de Usuario en dispositivos Domóticos.*” para optar por el título de Ingeniero electricista en la Universidad de Los Andes, implementó la configuración de una red domótica utilizando una aplicación Android que le permitió realizar un control domótico a través de microcontroladores ESP8266-01 con tecnología WIFI. Adicionalmente, Velasco creó servidores web para cada microcontrolador, permitiendo al usuario realizar el control domótico y modificar la configuración del mismo, siendo estas modificaciones guardadas en la memoria EEPROM de los dispositivos, facilitando así los cambios permanezcan inalterados cuando se efectúe un reinicio de los mismos. Como conclusión, le fue posible crear una interfaz de usuario para configurar los dispositivos domóticos que formaron parte de la red independientemente de la conexión a internet, además logró elaborar un servidor web que permitió al usuario modificar la configuración de cada uno de los dispositivos, así como modificar la ubicación de los dispositivos, el usuario y la contraseña WIFI del módulo principal [19].

2.2 DOMÓTICA

El término domótica encuentra su origen etimológico en la combinación de la palabra “casa” y la sugerencia de “automática”. Ahora bien, para ampliar este concepto:

La vivienda domótica es, por lo tanto, aquella que integra una serie de automatismos en materia de electricidad, electrónica, robótica, informática y telecomunicaciones, con el objetivo de asegurar al usuario un aumento del confort, de la seguridad, del ahorro energético, de las facilidades de comunicación, y de las posibilidades de entretenimiento [5].

Así pues, una vivienda domótica es aquella que incorpora un sistema orientado al control inteligente de la misma, lo cual permite una gestión fácil, automática, que proporciona seguridad y confort con la mínima intervención de usuario. Este concepto propone la integración de las distintas tecnologías en el hogar mediante el uso simultáneo de la electricidad, la electrónica, la informática y las telecomunicaciones. Además, el término domótica también representa un conjunto de servicios integrados al hogar:

Este conjunto de nuevos servicios integrados en la vivienda asegura al usuario mayores ventajas en aspectos relacionados con el confort, la seguridad, el ahorro y la gestión energética, las comunicaciones y las nuevas tecnologías de la información [13].

A partir de este concepto se puede entender que la domótica busca la integración de los aparatos o dispositivos del hogar con la finalidad de automatizar la vivienda. De esta manera la aplicación de la domótica proporciona algún nivel de automatismo que le permite al usuario el control o monitorización de su hogar de forma fácil y segura. De acuerdo con los conceptos expuestos anteriormente, la domótica puede entenderse como la incorporación de sistemas capaces de recoger información proveniente de unas entradas, ya sean sensores o mandos, procesarla y emitir órdenes a unos actuadores o salidas. Ahora bien, este sistema debe funcionar bajo las órdenes de una red de control o red domótica que conecte el hogar o entorno a automatizar con los diferentes sensores y actuadores. Por lo tanto, la domótica constituye la integración e instalación de varios dispositivos electrónicos en el hogar o espacio a automatizar, lo cual proporciona un control y monitorización tanto de forma local como remota, posibilitando de esta manera una cantidad ilimitadas de nuevas funcionalidades, aplicaciones y servicios al entorno domótico.

2.2.1 ELEMENTOS DE UN SISTEMA DE CONTROL DOMÓTICO

Una instalación domótica requiere percibir señales del exterior representadas por magnitudes físicas como la temperatura, presión, presencia de personas, entre otros. De forma genérica se puede decir que una instalación domótica está formada por los siguientes elementos:

Sensores: son dispositivos que transforman las magnitudes físicas en señales eléctricas, la cual comúnmente se trata de una señal analógica que puede ser convertida fácilmente a una señal digital por medio de un conversor analógico-digital integrado al sensor. Dependiendo del tipo de sensor, el valor digital recibido puede ser traducido a un protocolo de comunicación específico [11].

Nodos: los nodos son aquellos que reciben, procesan y envían las señales procedentes de dispositivos domóticos emisores (sensores) y la transmiten a los diferentes actuadores. En un sistema domótico pueden existir uno o más nodos interconectados entre sí. Los sistemas distribuidos o descentralizados poseen varios nodos interconectados entre sí a través de un bus de datos común [7].

Preactuadores: es un dispositivo que se encarga de conectar un actuador principal con el nodo del sistema domótico, permitiendo así que el actuador se ejecute de acuerdo a la señal recibida por el preactuador. Esto ocurre cuando el actuador a controlar no puede ser soportado directamente por la alimentación del nodo [7].

Actuadores: son dispositivos que permiten la recepción de una señal procedente de un sistema de control y ejecutan una acción en consecuencia, produciendo así una modificación del estado de un equipo o electrodoméstico. Comúnmente, los actuadores se ejecutan en función de una orden por el usuario o ya sea por la lectura recibida desde algún sensor [8].

Tipos de señales de los sensores y actuadores:

“En función del tipo de señal que envían o reciben, los sensores y actuadores respectivamente, pueden ser de dos tipos: digitales o analógicos” [7]. Adicionalmente, define cada tipo de señal de la siguiente manera:

- **Digitales:** son señales que adoptan dos valores: (1) y (0), es decir, un valor máximo y un valor mínimo.
- **Analógicas:** son señales que adoptan un rango de valores [7]

Acondicionadores de señal: los acondicionadores de señal se pueden entender como “circuitos eléctricos o electrónicos que modifican la señal que obtenemos del sensor para que podamos interpretarla de forma más precisa en la unidad de control”. [11]

Interfaz de Usuario: la interfaz de usuario es aquella que media entre el sistema y el usuario, es decir, garantiza una facilidad de comunicación entre los mismos, permitiendo así el funcionamiento y control del sistema domótico desde la interacción con el usuario. Las interfaces de usuario deben ser fáciles, amigables e intuitivas [11].

Software de gestión: se refiere a un software o conjunto de rutinas que permite la integración de múltiples herramientas para simplificar los procesos operativos del sistema de control domótico. El software de gestión ofrece la posibilidad de programar el sistema para que, de acuerdo a los valores de los sensores, un actuador pueda ser modificado o activado [11].

2.2.2 TIPOS DE SENSORES

Dentro del entorno de control domótico que se implementará en la presente investigación se emplearán algunos de los siguientes tipos de sensores:

Detectores de presencia o volumétricos (PIRs): son sensores piroeléctricos que envían una señal de acuerdo a los cambios en los niveles de radiación ante la presencia de movimiento. En los sistemas domóticos, este tipo de sensor es normalmente utilizado para el encendido y apagado de luces o para detectar intrusos [7].

Detectores de humo o fuego: los detectores de humo se encargan de detectar las fuertes concentraciones de humo, mientras que los de fuego son capaces de detectar altas temperaturas producidas por el fuego. Normalmente son utilizados en aplicaciones de seguridad en las viviendas domóticas [7].

Sensores de temperatura: son dispositivos que permiten medir los cambios de temperatura, y en consecuencia, facilitan la gestión de circuitos eléctricos en función de dichas mediciones [7].

Sensores de luminosidad: son sensores capaces de detectar los niveles de luz y son comúnmente utilizados para controlar circuitos eléctricos en función de los niveles detectados, como lo es la subida y bajada de persianas o el encendido y apagado automático de luces [7].

2.2.3 TIPOS DE ACTUADORES:

Los dispositivos de salida o actuadores son elementos controlados a través de un nodo domótico de acuerdo a los requerimientos específicos del usuario. Existen diferentes tipos de actuadores y se presentarán algunos de ellos a continuación:

Encendido On/Off: este tipo de actuadores se encarga de actuar sobre la fase de alimentación de los dispositivos a controlar mediante un elemento de corte, como por ejemplo un interruptor o un pulsador [7].

El regulador de luminosidad o Dimmer: “es un actuador de tipo electrónico que permite regular la tensión de alimentación de una lámpara para, así, conseguir diferentes niveles de luz (a menor tensión, menos luz, y viceversa” [7].

Electroválvulas: son actuadores que facilitan el control eléctrico de circuitos de fluidos como el agua o el gas. Normalmente son controlados eléctricamente a través de una bobina y son instalados en conducciones de fluidos [7].

Contactores: es un dispositivo electromagnético capaz de controlar cargas de potencia de forma remota. Habitualmente es utilizado para fabricar circuitos de automatismos cableados y en las aplicaciones domóticas [7].

Relés: Son dispositivos electromagnéticos que funcionan como un interruptor que es activado a través de la señal de un circuito eléctrico. Estos dispositivos son similares a los contactores [7].

Motores eléctricos: Son dispositivos capaces de convertir la energía eléctrica en energía mecánica de rotación debido a la acción de campos magnéticos. Estos dispositivos de salida pueden ser utilizados en innumerables aplicaciones domóticas, como, por ejemplo, el control de la subida y bajada de persianas [7].

Sirenas y avisadores: “son dispositivos acústicos y/o luminosos que se activan cuando se dispara un sistema de alarma. Se utilizan tanto para señalar el disparo de alarmas técnicas, como de seguridad”. [7]

2.2.4 NORMATIVAS EXISTENTES PARA LOS ENTORNOS DOMÓTICOS

Las normativas de la industria domótica posibilitan una arquitectura de carácter abierto, es decir, no limita a las empresas o instituciones a trabajar con tecnologías o proveedores específicos. Las normativas para los entornos domóticos establecen soluciones específicas y la adopción y cumplimiento de estas normativas permiten el crecimiento de las tecnologías y especializaciones domóticas. Los organismos encargados de la promoción, regularización, redacción y publicación de las normativas para entornos de control domótico son los siguiente:

- **ISO:** es una organización de normalización que promueve mundialmente las normas comerciales, industriales y tecnológicas.
- **IEEE:** es una organización que redacta las normativas en telecomunicaciones y electrónica a nivel mundial.
- **TIA:** publican las pautas en telecomunicaciones para celulares, satelitales, centros de datos, VoIP y redes para edificios inteligentes.
- **IEC:** redacta las normas eléctricas y electrónicas para telecomunicaciones, fibra óptica y otros.
- **INCITS:** publica las normas para campos como las comunicaciones, transporte y computación en la nube.
- **CENELEC:** redacta las normativas para redes e interconexiones inalámbricas y cableadas [20].

2.3 REDES DE TRANSMISIÓN DE DATOS

Para efectuar adecuadamente una transmisión de datos se debe establecer una red que permita la interconexión de los dispositivos a fin de alcanzar una transmisión satisfactoria. La red de datos define la transmisión y, de ser necesario, la conmutación, de la información digital [21].

2.3.1 REDES DE ÁREA LOCAL (LAN)

Una red de área local (LAN) es un medio de transmisión compartido y está constituido por la agrupación de hardware y software que funcionan de interfaz entre los dispositivos que se comunican y el medio. Así, una red de Área local se establece cuando los dispositivos a conectar están físicamente próximos [22].

2.3.2 EL PROTOCOLO TCP/IP:

Se refiere a un conjunto de protocolos de comunicación de datos y emplea dos importantes protocolos: el protocolo TCP (*Transmission Control Protocol*) y el protocolo IP (*Internet Protocol*) [23].

2.3.4 EL TCP (*Transmission Control Protocol*)

Es el que se encarga de la segmentación del mensaje en datogramas, el reensamblaje de destino, la retransmisión de los datogramas que se encuentran perdidos, así como la reordenación del mensaje para ser entregado al usuario [21].

2.3.5 El IP (*Internet protocol*)

El protocolo IP es aquel que define una unidad básica para la transmisión de datos y establece reglas que especifican el procesamiento de los paquetes de datos y el manejo de errores durante el envío, así como también define el formato de estos datos cuando viajan a través de una red TCP/IP, proporcionando un servicio de datagramas [22].

Dirección IP: Una dirección ip está formada por números de 32 bits y representan una dirección unívoca de un dispositivo conectado a una red que se comunique a través del protocolo TCP/IP [23].

2.3.6 PROTOCOLO DE TRANSFERENCIA DE HIPERTEXTO (HTTP: *Hypertext Transfer Protocol*)

El HTTP es un protocolo de comunicación cliente/servidor empleado en transacciones. Uno de los usos más comunes de este protocolo es a través de un navegador y un servidor web. Este protocolo utiliza TCP y cada una de sus

transacciones es tratada de forma independiente, esto quiere decir que, el HTTP creará una nueva conexión entre el servidor y el cliente para cada transacción y ésta misma será cerrada una vez que se haya completado [24].

2.4 METODOLOGÍA DE DESARROLLO DE SOFTWARE

El modelo de metodología del proceso de desarrollo de software en cascada es un proceso de desarrollo secuencial, en el que el desarrollo de software se concibe como un conjunto de etapas que se ejecutan una tras otra.

2.4.1 MODELO SECUENCIAL O CASCADA

El modelo lineal o secuencial es también conocido como el ciclo de vida del software y está conformado por las etapas de: análisis de requerimientos, diseño, codificación, pruebas e implementación. Este modelo sugiere un enfoque sistemático, secuencial para el desarrollo del software que empieza con el establecimiento de requisitos y termina en la implementación del software. El ciclo de vida de este modelo de metodología representa un conjunto de fases por las que se deberá pasar y cuyas funciones principales son establecer el orden de las fases del proceso de software, determinar los criterios de transición para pasar de una fase a otra y definir las entradas y salidas de cada fase. Este ciclo de vida está constituido por fases sucesivas compuestas por actividades planificadas, la cual puede ampliarse con bucles de retroalimentación [25].



Figura 2.1 Fases del modelo secuencial de desarrollo de software.

2.4.2 ETAPA 1: ANALISIS

En esta etapa se establece un análisis de las necesidades y características del software a desarrollar, en el cual se especifica todos los requerimientos del sistema sin entrar en detalles técnicos. El análisis consiste en llevar a cabo una descripción de los requisitos del software permitiendo estimar de forma rigurosa las necesidades del software antes de su diseño, incluidas la estimación del coste del software, los

riesgos y los plazos de desarrollo del mismo. Finalmente, con base en los requerimientos establecidos se analizan los componentes factibles de ser implementados para dicha necesidad. En esta parte del proceso metodológico el ingeniero intenta comprender la naturaleza de los programas que han de construirse, así como el dominio de la aplicación. Esta etapa comprende las siguientes características:

Definición de los requerimientos del usuario: Se hace necesario identificar los usuarios y entornos afectados por el problema, así como la correcta elección de las técnicas o herramientas que se utilizarán para facilitar la recopilación de la información y requerimientos del sistema.

Estudiar el dominio del problema: Consiste en identificar las relaciones con otros módulos o componentes del software. También se evalúan los recursos que se necesitan para la implementación, tales como infraestructura tecnológica (Servidores, equipos, sensores, módulos, dispositivos electrónicos, entre otros).

Especificar los requisitos del sistema: Consiste en determinar requerimientos de software, hardware e implementación, entre otros. Describir las interacciones entre el producto y su entorno, los aspectos visibles del software por el usuario, los lenguajes de programación a emplear, entre otros [25].

2.4.3 ETAPA 2: DISEÑO

En esta fase se describe la estructura interna del software, así como las relaciones entre las entidades que componen al producto. En esta etapa se establece la arquitectura general del software, representaciones de interfaz y algoritmos, a través de emplear la información obtenida en la etapa de análisis, elaborando de esta manera un diseño lógico del producto. Este nivel de diseño en la metodología secuencial o de cascada sugiere tres fases importantes:

Diseñar el modelo entidad-relación: Consiste en la utilización de una herramienta de modelado y se diseña cada una de las entidades y atributos principales que van a necesitarse para almacenar datos. En casos en que la base de datos no requiera ser modificada, esta parte puede obviarse.

Diseñar prototipos de pantalla: Consiste en representar prototipos de pantalla que

ayudan a normalizar las funcionalidades del sistema y permiten construir interfaces uniformes [25].

2.4.4 ETAPA 3: CODIFICACIÓN

Una vez que se cuenta con todos los requerimientos de la fase de análisis y del diseño de esta metodología; se inicia la fase de codificación. Evidentemente para codificar se necesita conocer la sintaxis del lenguaje de programación que se vaya a emplear. En esta parte, el desarrollador deberá seguir los lineamientos impuestos en el análisis y diseño diseños previos tomando en consideración siempre los requisitos funcionales y no funcionales [25].

2.4.5 ETAPA 4: PRUEBAS

Luego que el producto se ha terminado de codificar; debe ser instalado en el ambiente de pruebas. En esta parte se busca comprobar que el producto funciona correctamente y que cumple con los requerimientos del usuario. Las pruebas finales del producto son realizadas por el mismo usuario con la guía del desarrollador. En esta fase, el desarrollador deberá asegurarse que todos los componentes del producto se instalen correctamente para evitar que durante las pruebas se presenten errores o inconvenientes, esto incluye: objetos de base de datos, archivos de código fuente, datos iniciales en las tablas y otros recursos necesarios. Aquí también debe asegurarse que se instale la infraestructura necesaria. Es importante que la instancia de prueba tenga la misma configuración de los servicios (servidores) que el servidor de producción para detectar cualquier error en este punto y no cuando pase a producción [25].

2.4.6 ETAPA 5: IMPLEMENTACIÓN

Tomando como punto de partida el modelo de la fase anterior, se procede a implementar el producto entregable. El propósito de esta etapa es instalar el software y los requisitos necesarios para que el entregable pueda correr. En esta fase se considera desplegar los archivos fuente, implementar los objetos de la base de datos, verificar la funcionalidad del entregable y obtener conformidad del producto [25].

2.5 SOFTWARE LIBRE Y LENGUAJES DE PROGRAMACIÓN

A continuación, se presentarán los conceptos referentes al software y lenguajes que se emplearán para el diseño e implementación del sistema domótico.

2.5.1 BASE DE DATOS

Una base de datos es una colección organizada de datos en un formato específico y cuya estructura está definida por los metadatos [26], además, “una base de datos relacional es un programa capaz de almacenar grandes cantidades de información y de ofrecerla de manera rápida con capacidad para combinar y ordenar los datos, ajustándose a los requerimientos del usuario” [27]. Así pues, simplificando este concepto, se puede decir que “una base de datos, en su definición más sencilla, es una colección de archivos relacionados” [28].

2.5.2 LENGUAJE DE ETIQUETAS HTML5

Es un lenguaje de marcas de hipertexto desarrollado con la finalidad de mostrar archivos de texto a los que les pueda ser agregado colores, diseños y estilos. HTML se le conoce como “lenguaje de etiquetado” [29], además, provee estructura, estilo y funcionalidad. HTML5 es un producto que surge de la combinación del HTML, CSS y JavaScript, que son tecnologías independientes y que pueden actuar como una unidad organizada [30].

2.5.3 LENGUAJE DE CLASES CSS3

Corresponde a un lenguaje de estilos utilizado para la presentación y la apariencia de un documento HTML, XML o cualquier otro. Este lenguaje de estilos surge de la necesidad de separar el contenido de la presentación, permitiendo que el contenido este definido por un lenguaje como HTML, mientras que el diseño está enmarcado por CSS [31].

2.5.4 LENGUAJE DE PROGRAMACIÓN JAVASCRIPT

JavaScript es “un lenguaje interpretado basado en guiones que son integrados directamente en el código HTML. El código es transferido al cliente para

que este lo interprete al cargar la página” [32], además, se considera como un lenguaje de programación cuya principal aplicación es el desarrollo web [31].

2.5.5 LENGUAJE DE CONSULTA SQL

El lenguaje estructurado de consultas es un estándar que utilizan las bases de datos para actualizar, insertar, eliminar o recuperar información [27], por otro lado, SQL es un lenguaje de no procedimiento, lo que quiere decir que los resultados obtenidos de una operación dependerán del entorno fundamental del software [26].

2.5.6 LENGUAJE DE PROGRAMACIÓN PHP

Es un lenguaje de programación de código abierto cuya sintaxis está basada en C, Java y Perl. Este lenguaje es interpretado, lo que quiere decir que no requiere compilación [27]. PHP es un lenguaje interpretado del lado del servidor y se caracteriza por su robustez, modularidad y versatilidad, siendo uno de los lenguajes de programación más utilizados en el desarrollo web. Este lenguaje puede ser integrado directamente en el código HTML [32].

2.5.7 LENGUAJE DE PROGRAMACIÓN BASADO EN C++

El lenguaje de programación que se empleará para la programación de los microcontroladores ESP8266 del servidor de control domótico será un lenguaje basado en programación de alto nivel Processing que es similar a C++. Este lenguaje soporta todas las funciones del estándar C y algunas de C++ y es procesado en el entorno de desarrollo integrado Arduino. Principalmente, este lenguaje de puede dividir en tres grandes partes: estructura, valores (ya sea de variables y/o constantes), y funciones.

2.5.8 GESTOR DE BASE DE DATOS MySQL

Es un sistema de administración de base de datos de tipo relacional y es capaz de almacenar enormes cantidades de datos. MySQL utiliza el lenguaje SQL y permite crear base de datos, manipularlas, eliminarlas, agregar datos y recuperar datos de acuerdo a funciones específicas [28].

CAPITULO III: MARCO METODOLÓGICO

La presente investigación pretende desarrollar una plataforma de software/hardware innovadora, que ofrezca mejoras a nivel de procesos como la reducción del trabajo doméstico, el aumento de la seguridad, la racionalización de los diferentes consumos, la optimización de los recursos y el confort de los usuarios. Para ello se empleará el marco metodológico descrito a continuación.

3.1 TIPO DE INVESTIGACIÓN

Esta investigación reúne las condiciones metodológicas de una investigación de tipo aplicada, ya que, “la investigación aplicada o técnica tiende a la resolución de problemas o al desarrollo de ideas, dirigidas a conseguir innovaciones, mejoras de procesos o productos, etc.” [33].

3.2 DISEÑO DE INVESTIGACIÓN

El término diseño se refiere al plan o estrategia a seguir para el cumplimiento de los objetivos propuestos en la presente investigación, así pues, el diseño de la investigación objeto de estudio consistió en generar una serie de pasos a seguir para el logro del objetivo general, que estuvo enmarcado en una metodología de desarrollo de software que consiste en un modelo secuencial o de cascada que consta de etapas que establecen el orden de las fases del proceso de desarrollo de la investigación.

3.3 METODOLOGÍA PARA EL DISEÑO DEL SISTEMA DE CONTROL DOMÓTICO

A partir de la disponibilidad de los sistemas de hardware y software libre se desarrollará un sistema domótico para el control de cargas eléctricas compuesto por

los sistemas de control y los sistemas de monitoreo de cargas. En la presente investigación se utilizó un modelo secuencial o cascada como metodología para el diseño y desarrollo del software empleado en el monitoreo y control de las cargas, el cual fue abordado en el capítulo II. Este modelo de metodología consiste en un proceso secuencial constituido por un conjunto de etapas que establecen el orden de las fases del proceso de desarrollo del software del entorno domótico, así como también se emplean para definir los criterios necesarios para el diseño del sistema, administrar el progreso del desarrollo y proveer un espacio de trabajo para la definición de un proceso detallado. Tal y como se aprecia en la Figura 3.1, estas fases corresponden a: análisis de requerimientos, diseño, codificación, pruebas e implementación.

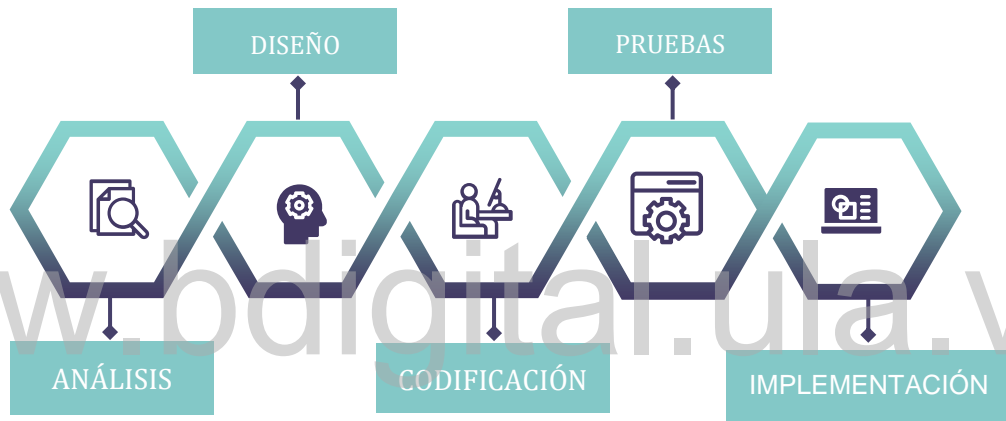


Figura 3.1 Etapas del proceso metodológico

3.4 ETAPA 1: ANALISIS

Se efectuó un análisis de las necesidades y características del hardware y software necesarios para el desarrollo del ambiente domótico de acuerdo a los requerimientos establecidos por los objetivos de la presente investigación. En el análisis presentado a continuación se especifica todos los requerimientos del sistema sin entrar en detalles técnicos, los cuales se presentarán en la etapa de diseño. Como se mencionó anteriormente, en esta parte del proceso metodológico se efectúa una descripción de los requisitos del software y hardware necesarios para el entorno domótico antes de su diseño, así como la estimación de los plazos de desarrollo del sistema. Esto puede apreciarse en la Figura 3.2 mostrada a continuación.

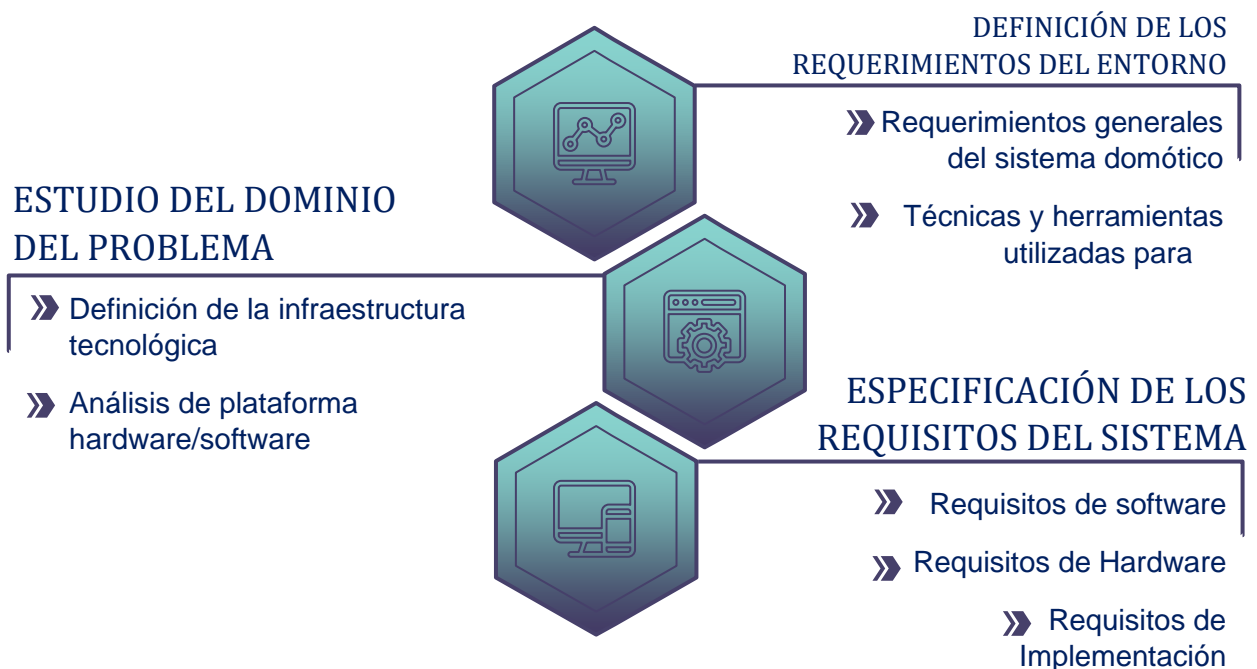


Figura 3.2 Descripción del proceso de análisis del sistema de control domótico

3.4.1 REQUERIMIENTOS GENERALES DEL SISTEMA

En esta fase del proceso metodológico correspondiente al análisis del entorno domótico en cuestión, se definirán los requerimientos de dicho entorno necesarios para garantizar una experiencia agradable al usuario. Un entorno domótico deberá contar con una interfaz de usuario que proporcione una las características que se describen a continuación:

Una buena interfaz de usuario debe ser fácil de aprender: ha de proporcionar ayuda suficiente para usuarios inexpertos (sin que esta facilidad en el aprendizaje se convierta en un obstáculo para aquellos usuarios más habituados a su manejo). Finalmente, una buena interfaz debe ser consistente. Una interfaz es consistente cuando mantiene un estilo, asistencia y pautas de interacción uniformes, con independencia de la tarea a realizar [7].

Finalmente, se estableció dentro del análisis inicial que los requerimientos del entorno domótico a desarrollar son los especificados en la Figura 3.3 que se observa a continuación:



Figura 3.3. Requerimientos generales del sistema doméstico.

3.4.2 TÉCNICAS Y HERRAMIENTAS PARA LA RECOPIACIÓN DE INFORMACIÓN

“Las técnicas de recolección de datos son las distintas formas o maneras de obtener la información” [34]. Adicionalmente, “Las técnicas de investigación o de investigación bibliográfica se relacionan con los procedimientos que se usan para obtener datos e información a través de los libros, y en general artículos que se refieren a determinadas materias y temas.” [35]. Así pues, las técnicas y herramientas utilizadas en la recopilación de la información del presente proyecto se establecieron a partir de la definición de la localización de las fuentes de información bibliográficas. Esto se llevó a cabo a partir de criterios de selección que garantizan que las fuentes consultadas pueden proporcionar información útil. El criterio empleado se centró en la recopilación de la información en cuatro tipos básicos:

Información primaria: proporciona documentos originales que tienen relación directa con el tema o el problema planteado.

Información secundaria: aporta información sobre cómo y dónde hallar fuentes primarias (bibliografías de libros, citas bibliográficas textuales o contextuales).

Información referencial y de consulta general: se refiere a obras que abarcan

temas diversos, dentro de los cuales pueden encontrarse referencias a cuestiones generales o específicas de interés para el investigador.

3.4.3 DEFINICIÓN DE LA INFRAESTRUCTURA TECNOLÓGICA

El sistema de control domótico a diseñar estará basado en un software de código abierto que proporcionará la integración de diferentes tecnologías y dispositivos de automatización del entorno mediante el uso simultáneo de la electrónica, la informática, la electricidad y las telecomunicaciones. Se empleará el protocolo de comunicación TCP/IP y la tecnología WIFI para la adquisición de datos de forma inalámbrica, lo cual permitirá que el sistema a diseñar se encargue de monitorear de manera remota diversas variables de interés, el cual estará constituido por la interfaz gráfica que actúa como cliente y por el microprocesador, que actuará como cliente/servidor.

3.4.4 ANÁLISIS DE LA PLATAFORMA HARDWARE/SOFTWARE

El sistema de control domótico a implementar en la presente investigación consiste en una plataforma de software/hardware basada en PHP, MySQL, JavaScript, HTML5, CSS3 y lenguaje C; así como el uso de sensores, actuadores, preactuadores y microprocesadores, además de un módulo de adquisición de datos que opera tanto de manera inalámbrica como alámbrica. Esta plataforma constituye el dominio del problema y representará una solución en términos de gestión domótica.

3.4.5 REQUISITOS EN HARDWARE Y SOFTWARE

Se realizó una selección de requerimientos tanto en hardware como en software para la implementación y el desarrollo del sistema domótico. Cabe destacar que los protocolos de comunicación a emplearse serán: TCP (*Transmisión Control Protocol*), IP (*Internet Protocol*), IPv4, TCP/IP, HTTP (*Hypertext Transfer Protocol*), MAC (*Media Access Control*), LLC (*Logical Link Control*), SPI (*Serial Peripheral Interface*) y UDP (*User Datagram Protocol*). A continuación, se observarán dos gráficos en los que se aprecia los requerimientos en hardware y software del sistema, mostrados en las figuras 3.4 y 3.5 respectivamente:

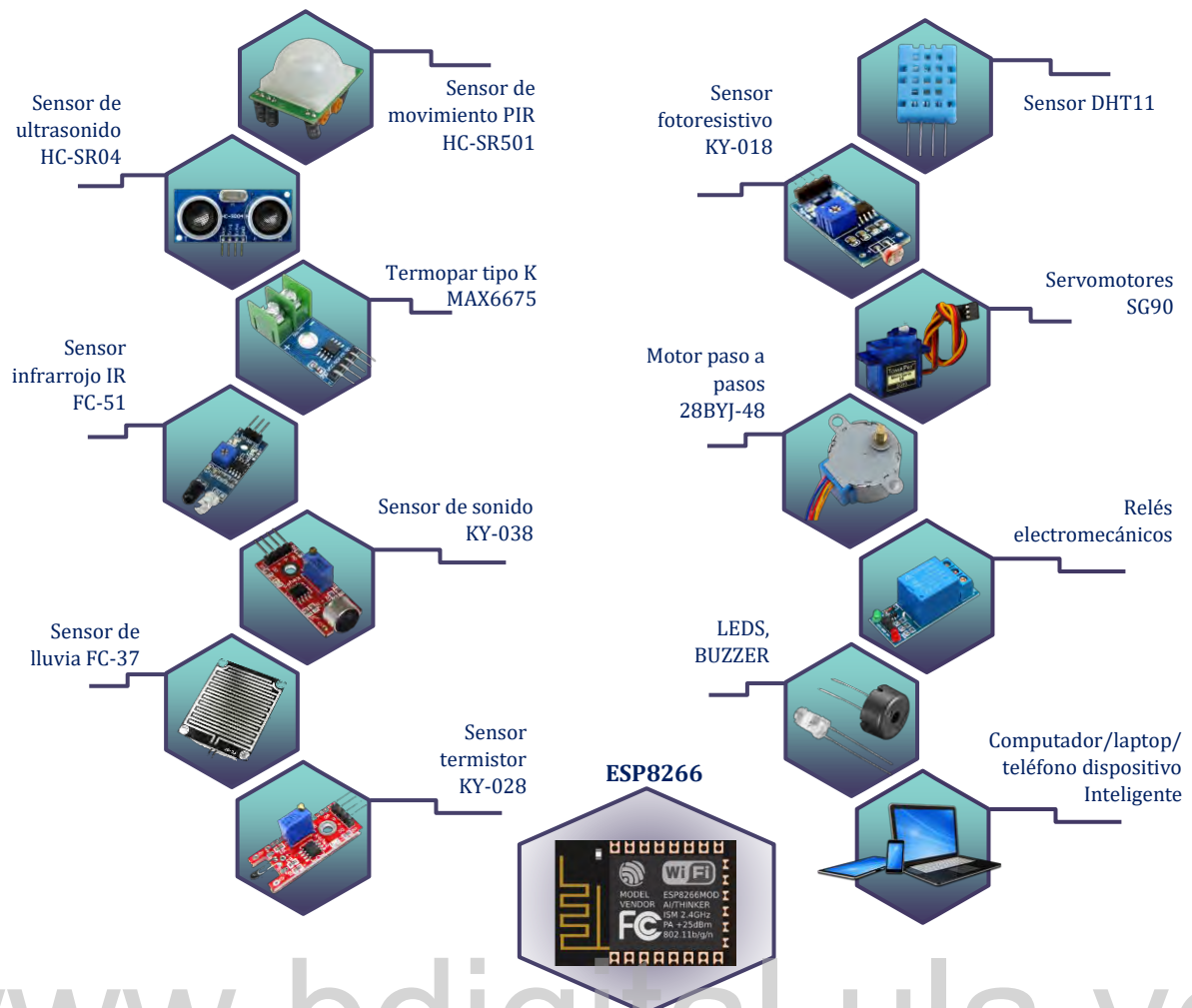


Figura 3.4 Requisitos en hardware para el sistema de control domótico

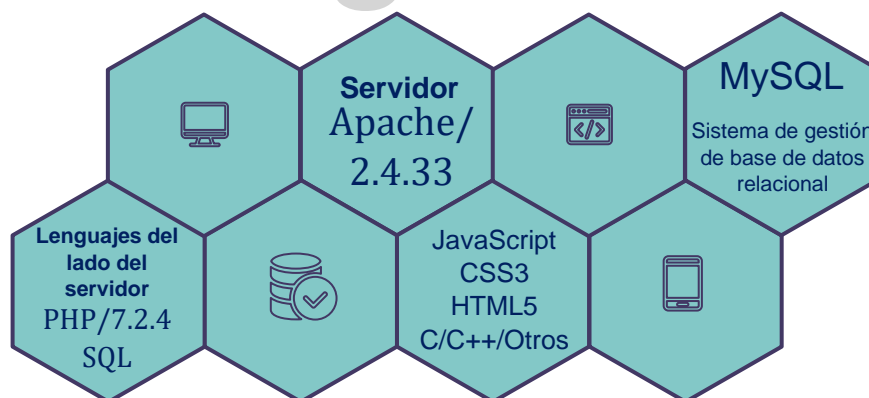


Figura 3.5 Requisitos en software para el sistema de control domótico

3.4.6 REQUISITOS DE IMPLEMENTACIÓN

De acuerdo a la definición de los requerimientos del entorno domótico mostrados anteriormente, se establecen los requisitos específicos necesarios para la implementación del software de control del entorno domótico. Este software debe cumplir con los requerimientos de confort, ahorro energético, seguridad y/o

alarmas, comunicación y accesibilidad descritos anteriormente. A continuación, se especificarán los requisitos de implementación:

- El usuario debe poder registrarse y acceder a la plataforma de forma fácil y amigable.
- El usuario debe poder crear diferentes proyectos de automatización para diferentes entornos domóticos de acuerdo a sus necesidades.
- El usuario debe poder crear o personalizar ítems o dispositivos de automatización que le permitirán monitorizar o controlar el ambiente. Estos dispositivos corresponden a sensores y actuadores. Además, tendrá la posibilidad de elegir dónde ubicar cada uno de estos diferentes dispositivos de acuerdo a sus necesidades y establecer una configuración para ellos.
- El usuario será capaz de configurar cada proyecto o entorno domótico. Podrá eliminar proyectos, agregar nuevos proyectos, así como eliminar o agregar ítems.
- El usuario podrá consultar los registros de la base de datos asociados a las lecturas de los diferentes sensores que configuró en su proyecto.
- El usuario deberá ser capaz de ejercer control de forma remota sobre los actuadores asociados al entorno domótico que está utilizando.
- El usuario podrá programar la ejecución de los actuadores de acuerdo a las lecturas recibidas por los sensores o de acuerdo a un horario.
- El usuario podrá asociar uno o más microcontroladores a su entorno domótico y podrá cambiar la configuración de WIFI de los diferentes microcontroladores que utilice en sus proyectos.

3.5 ETAPA 2: DISEÑO

En esta etapa, la cual constituye la primera fase de desarrollo del sistema, se traducen los requerimientos especiados en la etapa anterior, ya que se establece la arquitectura general del software y hardware, representaciones de interfaz y algoritmos, a través de la información obtenida en la etapa previa. En esta fase se describe la estructura interna del software diseñado, así como las relaciones entre las entidades que componen al producto. Aquí se aplican diversas técnicas y

principios con el propósito de definir el sistema con los suficientes detalles para permitir su implementación física. A continuación, en la Figura 3.6, se muestra un diagrama que especifica las actividades de diseño del sistema de control:

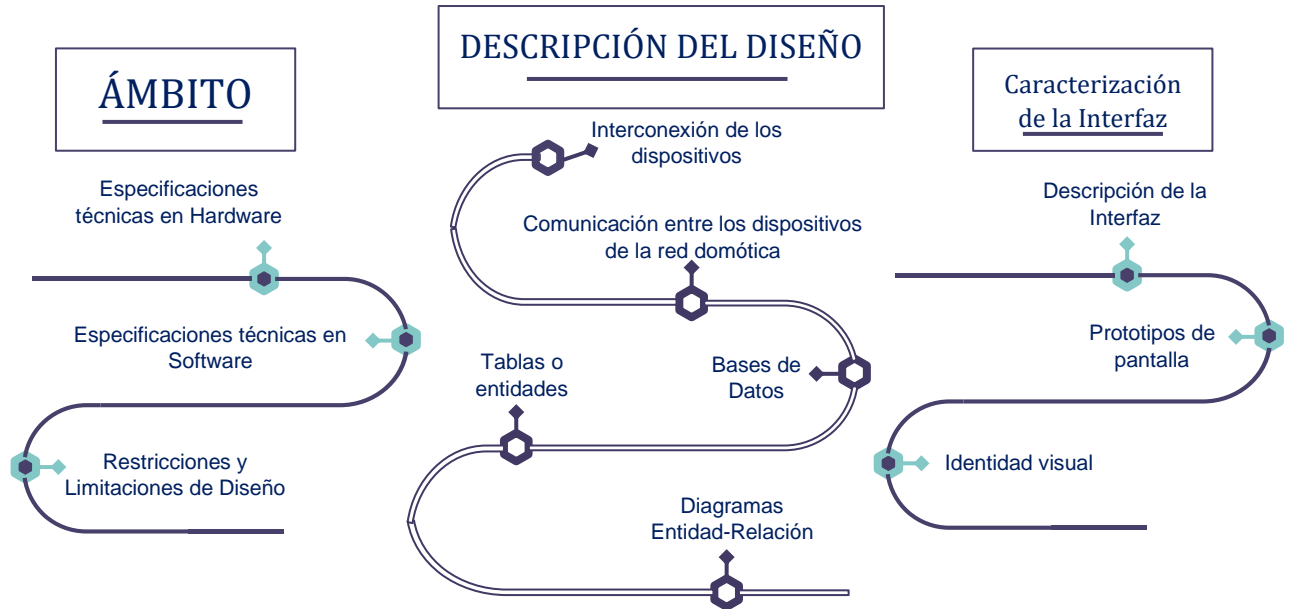


Figura 3.6 Descripción del proceso de diseño del sistema de control domótico.

3.5.1 ESPECIFICACIONES TÉCNICAS EN HARDWARE

De acuerdo a la disponibilidad de hardware libre, se seleccionaron los siguientes sensores, actuadores y dispositivos electrónicos necesarios para la conformación del entorno domótico. Dentro de este listado mostrado en la Tabla 1, se podrán encontrar las referencias de las especificaciones técnicas consultadas en la *datasheet* correspondiente a cada dispositivo.

Tabla 1. Hardware en sensores y actuadores

Dispositivo	Datasheet
Sensor de movimiento PIR HC-SR501	[36]
Sensor de ultrasonido HC-SR04	[37]
Termopar tipo K MAX6675	[38]
Sensor infrarrojo IR FC-51	[39]
Sensor de sonido KY-038	[40]
Sensor de lluvia FC-37	[41]
Sensor termistor KY-028	[42]

Continuación Tabla 1. Hardware en sensores y actuadores

Dispositivo	Datasheet
Sensor de humedad y temperatura DHT11	[43]
Sensor fotoresistivo KY-018	[44]
Servomotores SG90	[45]
Motor paso a paso 28BYJ-48	[46]
Relés electromecánicos	[47]
ULN2003 Motor driver	[48]

3.5.2 ESPECIFICACIONES TÉCNICAS EN SOFTWARE

A continuación, se presentan algunas características de software que forman parte en los requerimientos del entorno domótico analizado en la primera estaba, entre estas características se encuentran las especificaciones del servidor web y de la base de datos a emplear, tal y como se describe en la Figura 3.7.

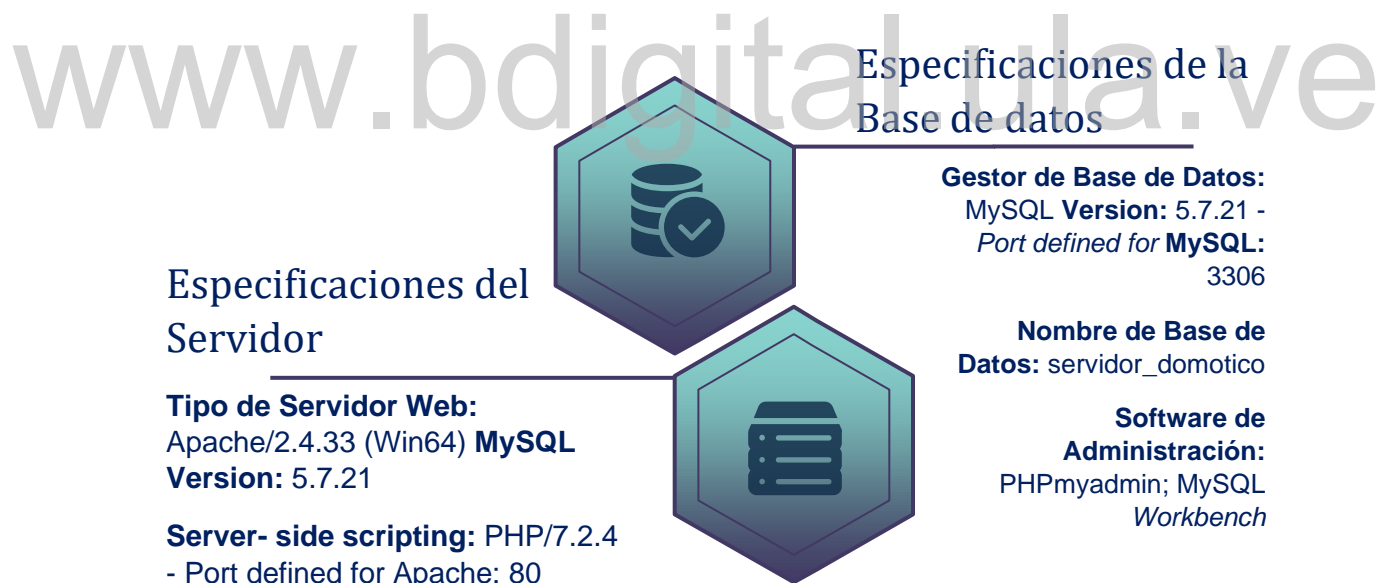


Figura 3.7 Especificaciones técnicas en software del servidor de control domótico

3.5.3 MICROCONTROLADOR ESP8266:

El microcontrolador ESP8266 es empleado para la transmisión y recepción de datos de forma inalámbrica desde el servidor domótico hacia los diferentes

dispositivos de entrada y salida. Algunas especificaciones técnicas obtenidas de la datasheet proporcionada por Espressif Systems [49] se mostrarán a continuación en las Figuras 3.8, 3.9, 3.10 y 3.11 respectivamente:

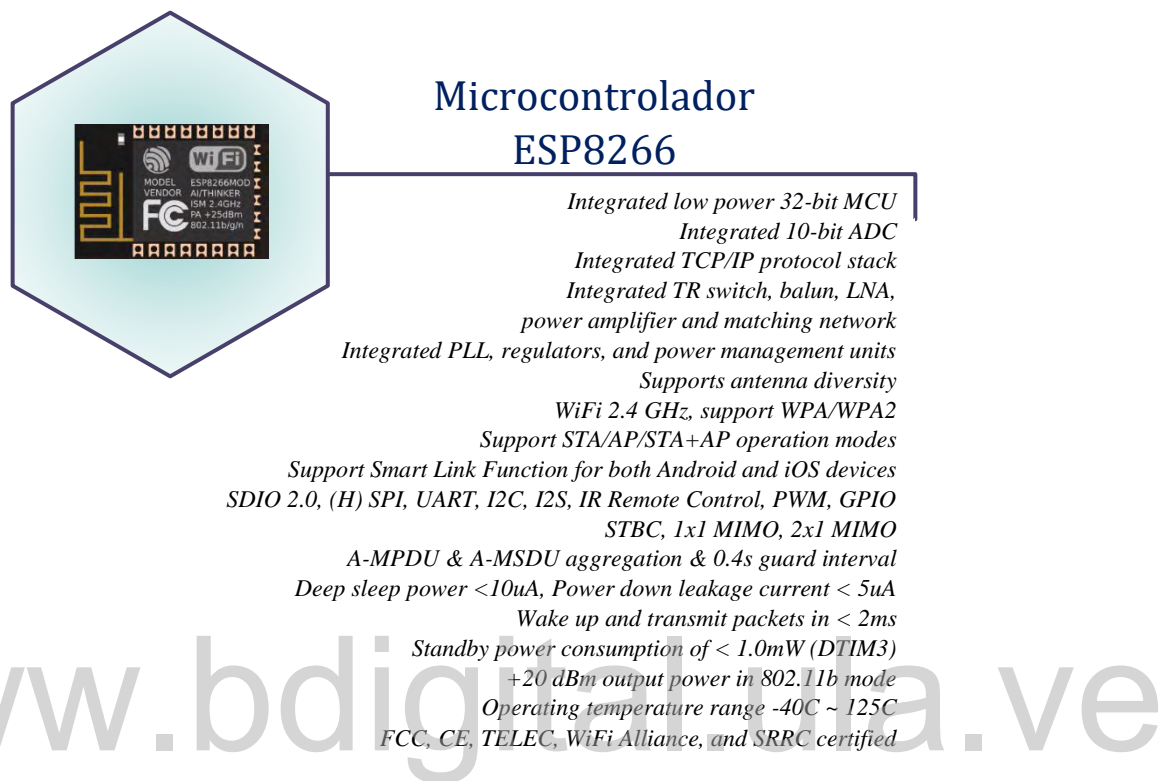


Figura 3.8 Especificaciones técnicas del microcontrolador ESP8266

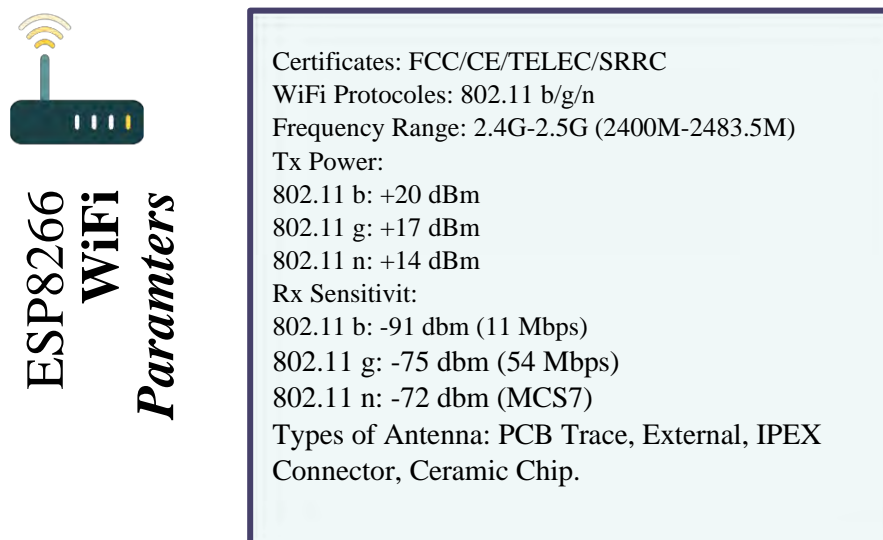


Figura 3.9 Parámetros de WIFI correspondientes al ESP8266



**ESP8266
Hardware
Parameters**

Peripheral Bus:
UART/SDIO/SPI/I2C/I2S/IR Remote Control
GPIO/PWM
Operating Voltage: 3.0~3.6V
Operating Current: Average value: 80mA
Operating Temperature: Range -40°~125°
Ambient Temperature Range: Normal temperature
Package Size: 5x5mm
External Interface: N/A

Figura 3.10 Parámetros de Hardware correspondientes al ESP8266



**ESP8266
Software
Parameters**

WiFi mode: station/softAP/SoftAP+station
Security: WPA/WPA2
Encryption: WEP/TKIP/AES
Firmware Upgrade: UART Download / OTA
(via network)
Software Development:
Supports Cloud Server Development / SDK for
custom firmware
development
Network Protocols: IPv4, TCP/UDP/HTTP/FTP

Figura 3.11 Parámetros de Software correspondientes al ESP8266

3.5.4 RESTRICCIONES Y LIMITACIONES DE DISEÑO

Las restricciones y limitaciones de diseño están supeditadas a la disponibilidad de software y hardware libre, así como los costos en hardware y la factibilidad de encontrar dispositivos electrónicos como actuadores y sensores que permitan al sistema de control funcionar para lo que está diseñado. La lista de sensores y actuadores previamente señalada en la primera etapa, especifica aquellos dispositivos que pudieron ser encontrados en el mercado actual y cuyos costos no resultaron ser excesivos.

3.5.5 INTERCONEXIÓN DE LOS DISPOSITIVOS

Los diferentes dispositivos que forman parte del conjunto de operación del sistema de control domótico se interconectan de la forma en que se especifica más adelante en la Figura 3.19. El servidor web basado en apache se comportará como cliente/servidor al igual que los microcontroladores ESP8266 (nodos) que se conecten al sistema. El servidor se comunicará y recibirá comunicación tanto con los microcontroladores como con la base de datos, mientras que los sensores enviarán información a los ESP8266 y los actuadores recibirán información de éstos. El sistema permitirá utilizar tantos nodos o microcontroladores ESP8266 como el usuario desee, esto se ilustra en la Figura 3.12.

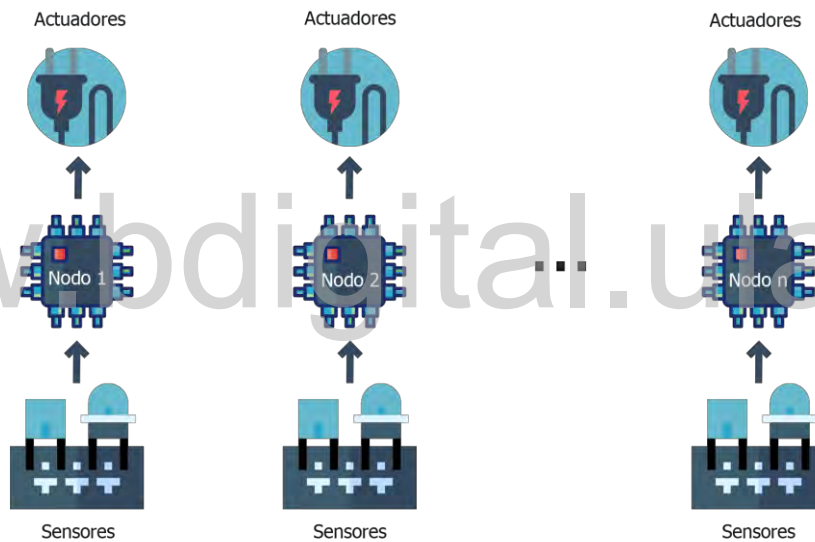


Figura 3.12 Interconexión de los dispositivos del sistema

3.5.6 INTERCONEXIÓN DE LOS SENSORES

Los elementos que conforman la instalación domótica están caracterizados por la descripción en hardware y software señaladas en la sección anterior. Tanto los sensores como los actuadores, entre otros dispositivos, forman parte de los elementos de la instalación domótica. En la imagen que se muestra a continuación, se puede observar cómo los sensores se interconectan con un nodo, el cual está conformado por un microcontrolador ESP8266 debido a que recibe, procesa y envía

las señales domóticas procedentes de los sensores hacia los actuadores. En la Figura 3.13 puede observarse que los sensores funcionan como dispositivos de entrada al nodo.

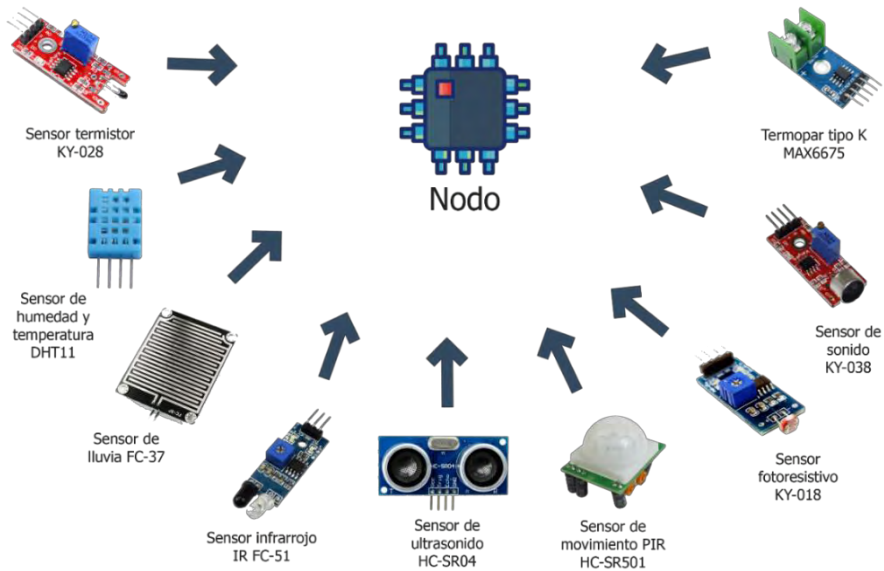


Figura 3.13 Interconexión de los sensores del sistema

3.5.7 INTERCONEXIÓN DE LOS ACTUADORES

En el caso de los actuadores del sistema domótico, éstos funcionan como salidas (véase Figura 3.14), las cuales, como se verá más adelante, pueden ser activadas a voluntad del usuario, ya sea de forma inmediata o programa. Usualmente la activación de un actuador está determinada por las lecturas proporcionadas por los sensores que forman parte del entorno domótico.

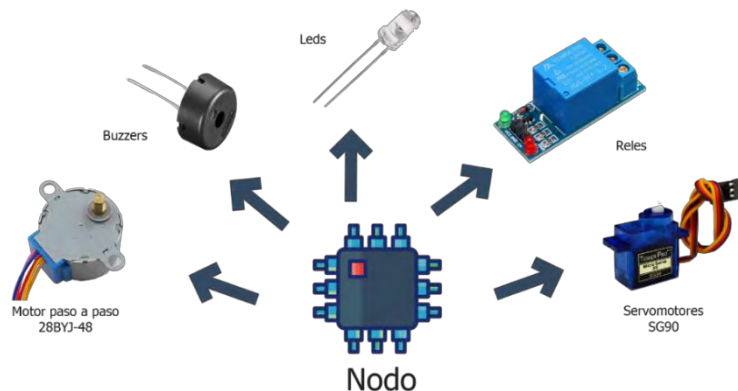


Figura 3.14 Interconexión de los actuadores del sistema

Nótese que tanto los actuadores como los sensores deberán estar conectados de forma cableada a su correspondiente nodo, mientras que el nodo se comunicará con el servidor de forma inalámbrica a través de la red WIFI. Esto quiere decir que el medio físico para el transporte de señales domóticas de los sensores y actuadores hacia el nodo es cableado. Esto se ilustra en la Figura 3.15 mostrada a continuación.

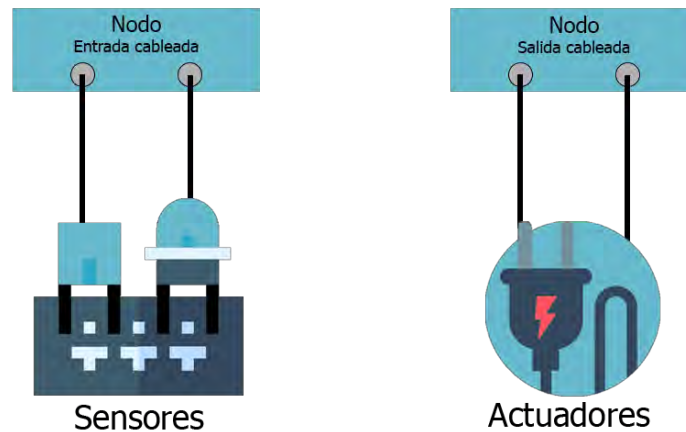


Figura 3.15 Conexión cableada de los sensores y actuadores

3.5.8 CIRCUITOS NECESARIOS PARA LA CONEXIÓN

El entorno domótico debe poder integrarse a la instalación eléctrica de la vivienda. No se debe olvidar que los circuitos domóticos, además de necesitar de la red eléctrica para su funcionamiento, también son los encargados de aplicar dicha alimentación a los diferentes receptores de potencia a través de los actuadores. Se puede decir de forma general que una instalación domótica está formada por uno o varios tipos circuitos eléctricos y una red domótica. Para el diseño del sistema domótico del presente trabajo de grado, se utilizará:

Circuito de alimentación de potencia: este circuito será utilizado para alimentar a 120 VCA los diferentes actuadores y receptores de la instalación, como lámparas, aparatos electrodomésticos, motores y persianas, entre otros. (Véase Figura 3.16).

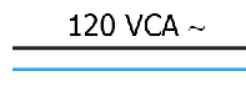


Figura 3.16 Circuito de alimentación de potencia

Circuito de alimentación auxiliar: este circuito será utilizado para alimentar determinados elementos del sistema domótico que requieren tensiones de corriente continua que no son las de la red eléctrica. En este caso, los sensores utilizados se alimentan a 5 Vcc como se observa en la Figura 3.17.

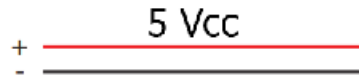


Figura 3.17 Circuito de alimentación auxiliar

Nodo domótico: este circuito será utilizado para el transporte de las señales (telegramas) de mando enviadas por los sensores y recibidas por los actuadores a través del nodo domótico (microcontroladores ESP8266). En la Figura 3.18 se observa una representación del bus domótico o circuito de interconexión del nodo a los diferentes sensores o actuadores.

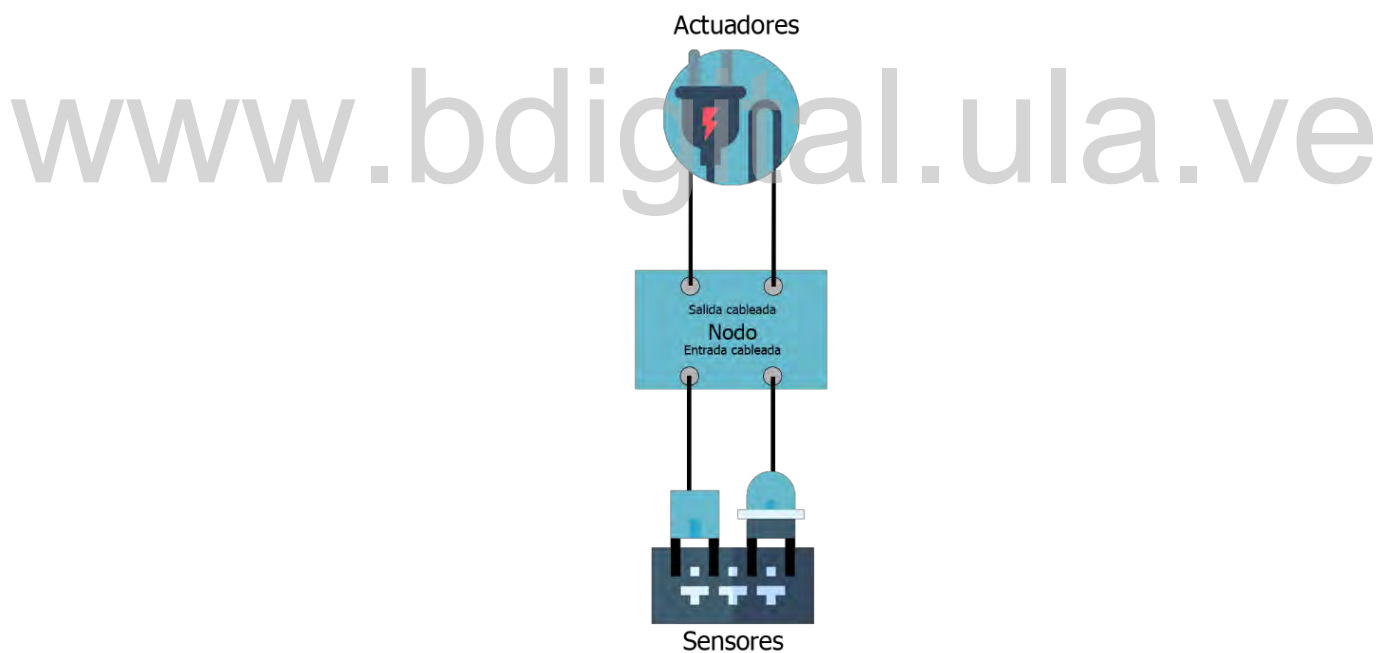


Figura 3.18 Circuito de interconexión del nodo a los sensores y actuadores

3.5.9 ESQUEMA DE CONEXIÓN DEL SISTEMA DOMÓTICO

En el siguiente esquema puede observarse que tanto el servidor, como los clientes y diferentes nodos (microcontroladores ESP8266) se comunican inalámbricamente por medio de una red WIFI a través del router de la red asignada

para el sistema domótico, el cual se encarga de interconectar estos dispositivos. Por otro lado, los actuadores y sensores están conectado de forma cableada al nodo correspondiente, tal y como se observa en la Figura 3.19.

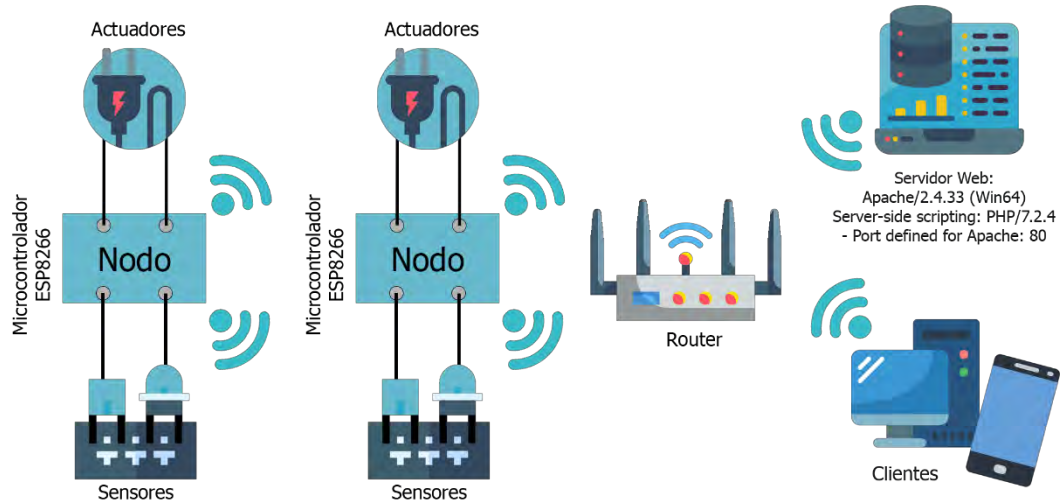


Figura 3.19 Esquema de conexión del sistema domótico

3.5.10 BASE DE DATOS

Como se estableció en la etapa de análisis, el diseño del sistema de control para un entorno domótico debe presentar ventajas en confort, ahorro energético, seguridad y/o alarmas, comunicación y accesibilidad. Esto constituye un requerimiento esencial en el diseño del sistema de control, así pues, la base de datos diseñada para este sistema debe contemplar todos estos aspectos. En total se determinó un número de 17 tablas o entidades que conforman la base de datos. Para la creación de la base de datos se hizo uso del gestor de base de datos MySQL Version: 5.7.21, el cual utiliza el puerto MySQL definido por: 3306. El nombre asignado para la base de datos es **servidor_domotico** y el software de administración de este gestor es PHPmyadmin. Las tablas pertenecientes a la base de datos pueden observarse en la Figura 3.20.

Tabla	Acción
<input type="checkbox"/> ambientes	★ Examinar Estructura Buscar Insertar Vaciar Eliminar
<input type="checkbox"/> configuracion	★ Examinar Estructura Buscar Insertar Vaciar Eliminar
<input type="checkbox"/> config_proyecto	★ Examinar Estructura Buscar Insertar Vaciar Eliminar
<input type="checkbox"/> dht11	★ Examinar Estructura Buscar Insertar Vaciar Eliminar
<input type="checkbox"/> esp8266	★ Examinar Estructura Buscar Insertar Vaciar Eliminar
<input type="checkbox"/> fc37	★ Examinar Estructura Buscar Insertar Vaciar Eliminar
<input type="checkbox"/> fc51	★ Examinar Estructura Buscar Insertar Vaciar Eliminar
<input type="checkbox"/> hcsr04	★ Examinar Estructura Buscar Insertar Vaciar Eliminar
<input type="checkbox"/> hcsr501	★ Examinar Estructura Buscar Insertar Vaciar Eliminar
<input type="checkbox"/> items	★ Examinar Estructura Buscar Insertar Vaciar Eliminar
<input type="checkbox"/> ky018	★ Examinar Estructura Buscar Insertar Vaciar Eliminar
<input type="checkbox"/> ky028	★ Examinar Estructura Buscar Insertar Vaciar Eliminar
<input type="checkbox"/> ky038	★ Examinar Estructura Buscar Insertar Vaciar Eliminar
<input type="checkbox"/> max6675	★ Examinar Estructura Buscar Insertar Vaciar Eliminar
<input type="checkbox"/> programar_actuadores	★ Examinar Estructura Buscar Insertar Vaciar Eliminar
<input type="checkbox"/> proyectos	★ Examinar Estructura Buscar Insertar Vaciar Eliminar
<input type="checkbox"/> usuarios	★ Examinar Estructura Buscar Insertar Vaciar Eliminar
17 tablas	Número de filas

Figura 3.20 Estructura de la base de datos del sistema

TABLAS O ENTIDADES

Las tablas o entidades corresponden a un modelo relacional y están formadas por atributos o campos referidos a un objeto real o abstracto.

TABLA AMBIENTES:

Describe los ambientes que han sido asociados a un proyecto. Tal y como se observa en la Figura 3.21, esta tabla contiene 5 atributos que corresponden a un **id** que identifica de forma única cada fila de la tabla, el nombre del **proyecto** al que pertenece el ambiente, el nombre del **ambiente**, así como el **usuario** que lo asoció, y por último el nombre de la **imagen** que el usuario cargó para representar dicho ambiente. Obsérvese que para esta tabla el atributo imagen puede ser nulo.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios
<input type="checkbox"/> 1	id 🔑	int(11)			No	Ninguna	
<input type="checkbox"/> 2	proyecto	varchar(30)	utf8_general_ci		No	Ninguna	
<input type="checkbox"/> 3	ambiente	varchar(40)	utf8_general_ci		No	Ninguna	
<input type="checkbox"/> 4	usuario	varchar(10)	utf8_general_ci		No	Ninguna	
<input type="checkbox"/> 5	imagen	varchar(37)	utf8_general_ci		Sí	NULL	

Figura 3.21 Estructura de la tabla ambientes

TABLA CONFIGURACIÓN:

La entidad de configuración consta de 8 atributos, los cuales corresponden a un **id** único para cada fila, los atributos **hijo_id** y **ip_dispositivo** que pueden ser nulos pero únicos en la tabla, el **usuario**, nombre del **proyecto**, el atributo **chipid** que identifica de forma única al microcontrolador ESP8266 que corresponde al **dispositivo**, el nombre de dicho dispositivo y el atributo **dispositivo_hijo** que se refiere al dispositivo al que se le asocia la configuración.

Las filas de esta tabla se llenan automáticamente cuando el microcontrolador se conecta a la red, enviando un *HTTP Request* donde el servidor recibe datos acerca de la configuración de hardware asociado al ESP8266 (véase con más detalle en la sección 3.6.3 y 3.6.4), esto quiere decir que, por cada sensor o actuador conectado al microcontrolador conectado a la red, se llenará una fila que contendrá la identificación del dispositivo (**id_dispositivo**), el tipo de dispositivo (**dispositivo**), la identificación de la placa ESP8266 (**chipid**). Una vez que el usuario agregue el microcontrolador a su cuenta y le asocie un proyecto, los campos **usuario** y **proyecto** se llenarán. Cuando el usuario establezca los ítems que desea asociar a esta configuración, entonces los atributos **hijo_id** y **dispositivo_hijo** se llenarán. Las entidades de esta tabla pueden apreciarse en la Figura 3.22.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
<input type="checkbox"/> 1	id 🔑	int(11)			No	Ninguna
<input type="checkbox"/> 2	usuario	varchar(10)	utf8mb4_general_ci		Sí	NULL
<input type="checkbox"/> 3	proyecto	varchar(30)	utf8mb4_general_ci		Sí	NULL
<input type="checkbox"/> 4	chipid	int(11)			No	Ninguna
<input type="checkbox"/> 5	dispositivo	varchar(20)	utf8mb4_general_ci		No	Ninguna
<input type="checkbox"/> 6	dispositivo_hijo	varchar(30)	utf8mb4_general_ci		Sí	NULL
<input type="checkbox"/> 7	hijo_id 🔑	int(11)			Sí	NULL
<input type="checkbox"/> 8	id_dispositivo 🔑	varchar(20)	utf8mb4_general_ci		No	Ninguna

Figura 3.22 Estructura de la tabla configuración

TABLA CONFIG_PROYECTO:

La tabla **config_proyecto** constituye una de las tablas más importantes del software de control, contando con un total de 16 atributos que establecen la configuración más esencial de cada proyecto o entorno domótico a automatizar.

En esta tabla se encuentra el atributo **id** que identifica de manera única cada fila de la tabla, cada fila contiene la configuración de un ítem que ha sido asociado a un proyecto, el resto de los atributos corresponden al nombre del **proyecto**, el id asociado al proyecto (**id_proyecto**), el **usuario**, el **ambiente** en el que se encuentra el ítem que ha sido asociado al proyecto, el nombre de dicho **ítem**, la especificación de si se trata de un ítem de entrada o de salida para el atributo **tipo**, además del nombre del **icono** con el que se representa el ítem asociado, las coordenadas “x” (**ejex**) y las coordenadas “y” (**ejej**) del ítem dentro de la imagen del plano del proyecto, así como las coordenadas “x” (**ejex_ambiente**) y las coordenadas “y” (**ejej_ambiente**) del ítem ubicándolo en el ambiente al que corresponde, así como el nombre del **dispositivo**, su identificador (**id_dispositivo**) y el identificador del ESP8266 (**chipid**), tal y como se observa en la Figura 3.23.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	id	int(11)			No	Ninguna
2	proyecto	varchar(30)	utf8mb4_general_ci		No	Ninguna
3	id_proyecto	int(11)			No	Ninguna
4	usuario	varchar(30)	utf8mb4_general_ci		No	Ninguna
5	ambiente	varchar(40)	utf8mb4_general_ci		No	Ninguna
6	item	varchar(20)	utf8mb4_general_ci		No	Ninguna
7	tipo	varchar(20)	utf8mb4_general_ci		No	Ninguna
8	nombre	varchar(30)	utf8mb4_general_ci		No	Ninguna
9	icono	varchar(20)	utf8mb4_general_ci		No	Ninguna
10	ejex	double			Sí	NULL
11	ejej	double			Sí	NULL
12	ejex_ambiente	double			Sí	NULL
13	ejej_ambiente	double			Sí	NULL
14	dispositivo	varchar(30)	utf8mb4_general_ci		Sí	NULL
15	id_dispositivo	varchar(13)	utf8mb4_general_ci		Sí	NULL
16	chipid	int(11)			Sí	NULL

Figura 3.23 Estructura de la tabla proyecto_config

TABLAS DHT11, FC37, FC51, HSCR04, HSCR501, KY018, KY028, KY038 Y MAX6675:

Estas tablas o entidades contienen la información que los microcontroladores ESP8266 reciben de los sensores disponibles y configurados por el usuario. Allí se almacenan los datos concernientes a las lecturas de los sensores mencionados. La especificación de estas tablas es básicamente la misma en términos de sus atributos, así pues, en general, constan del atributo **id** como todas las tablas, la fecha y hora en la que fue sensado el entorno (**fecha**), la **lectura** arrojada por el sensor, la identificación del sensor (**id_dispositivo**) que efectuó dicha lectura y la identificación de la placa ESP8266 (**chipid**) que contiene al sensor mencionado. Por esta razón, solo se mostrará la tabla en la Figura 3.24 a forma de referencia, ya que las tablas restantes contienen la misma información.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
<input type="checkbox"/>	1 id 🔑	int(11)			No	Ninguna
<input type="checkbox"/>	2 fecha	datetime			No	Ninguna
<input type="checkbox"/>	3 lectura	double			No	Ninguna
<input type="checkbox"/>	4 id_dispositivo	varchar(32)	utf8mb4_general_ci		No	Ninguna
<input type="checkbox"/>	5 chipid	int(11)			No	Ninguna

Figura 3.24 Estructura de la tabla fc37 como referencia

TABLA ESP8266:

Muestra información relacionada con cada ESP8266 que se comunique con el servidor web. Cada fila de esta tabla deberá contener información de una única placa o microcontrolador, teniendo como atributos un **id**, el identificador del ESP8266 (**chipid**), la ip que el router del sistema domótico le asigne al microcontrolador (**ip_chipid**), el **usuario** y el **proyecto**. (Véase la Figura 3.25).

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
<input type="checkbox"/>	1 id 🔑	int(11)			No	Ninguna
<input type="checkbox"/>	2 chipid 🔑	int(11)			No	Ninguna
<input type="checkbox"/>	3 ip_chipid	varchar(20)	utf8mb4_general_ci		No	Ninguna
<input type="checkbox"/>	4 usuario	varchar(20)	utf8mb4_general_ci		Sí	NULL
<input type="checkbox"/>	5 proyecto	varchar(30)	utf8mb4_general_ci		Sí	NULL

Figura 3.25 Estructura de la tabla ESP8266

TABLA ITEMS:

Esta tabla contiene información de los dispositivos de entrada y salida que pueden ser utilizados, ya sea que vengan predeterminados en el programa o que el usuario los haya personalizado. En la Figura 3.26 se puede apreciar que la tabla consta de los atributos **id**, el **usuario**, el nombre del ítem o dispositivo (**ítem**), si se trata de un dispositivo de entrada o de salida identificado con el atributo **tipo**, el **icono** al que está asociado el ítem, una breve descripción del ítem la cual puede ser opcional (**descripcion**) siendo este un campo nulo, y, por último, el **dispositivo**.


#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
<input type="checkbox"/>	1 id 	int(11)			No	Ninguna
<input type="checkbox"/>	2 usuario	varchar(20)	utf8_general_ci		No	Ninguna
<input type="checkbox"/>	3 ítem	varchar(40)	utf8_general_ci		No	Ninguna
<input type="checkbox"/>	4 tipo	varchar(15)	utf8_general_ci		No	Ninguna
<input type="checkbox"/>	5 icono	varchar(37)	utf8_general_ci		No	Ninguna
<input type="checkbox"/>	6 descripcion	varchar(80)	utf8_general_ci		Sí	NULL
<input type="checkbox"/>	7 dispositivo	varchar(30)	utf8_general_ci		No	Ninguna

Figura 3.26 Estructura de la tabla ítems

TABLA PROGRAMAR_ACTUADORES:

La presente tabla establece la ejecución programada de los actuadores configurados en el entorno domótico. La tabla incluye en campo **id**, el **usuario** y el id del proyecto (**id_proyecto**). Adicionalmente se observan los campos **condicion**, la cual especifica si se trata de una condición dada por la lectura de un sensor o debido a un horario programado por el usuario.

Los campos **id_sensor** e **id_actuador** identifican de manera inequívoca cuales son los dispositivos asociados a la ejecución del actuador, que son el actuador mismo y en caso de que aplique, el sensor cuyo valor determinará el accionar del actuador. Por otro lado, se presentan los atributos **descripcion**, el cual corresponde a una breve descripción de la actividad programa, y el campo **sensor** que muestra el nombre de dicho componente. El atributo **valor_condicion** determinará, en caso de que aplique, el valor que debe mostrar determinado sensor para que el actuador se accione. Estos campos pueden observarse en la Figura 3.27.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	id	int(11)			No	Ninguna
2	condicion	varchar(20)	utf8_general_ci		No	Ninguna
3	id_actuador	varchar(30)	utf8_general_ci		No	Ninguna
4	id_sensor	varchar(30)	utf8_general_ci		Sí	NULL
5	valor_condicion	varchar(30)	utf8_general_ci		No	Ninguna
6	usuario	varchar(10)	utf8_general_ci		No	Ninguna
7	id_proyecto	int(11)			No	Ninguna
8	descripcion	varchar(200)	utf8_general_ci		No	Ninguna
9	actuador	varchar(20)	utf8_general_ci		No	Ninguna
10	sensor	varchar(20)	utf8_general_ci		No	Ninguna

Figura 3.27 Estructura de la tabla programar_actuadores

TABLA PROYECTOS:

Guarda la descripción de todos los proyectos creados en el servidor, el atributo **id** identifica cada fila como única, el atributo **proyecto** encierra el nombre que el usuario le asignó a su proyecto, seguidamente se observa el atributo **usuario**, la **descripcion** del proyecto, el nombre del archivo que contiene la imagen del plano del proyecto (**plano**), así como un atributo llamado **listo** que determinará si el proyecto se ha terminado de configurar. Esto se muestra en la Figura 3.28.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	id	int(11)			No	Ninguna
2	proyecto	varchar(30)	utf8mb4_general_ci		No	Ninguna
3	usuario	varchar(10)	utf8mb4_general_ci		No	Ninguna
4	descripcion	varchar(200)	utf8mb4_general_ci		Sí	NULL
5	plano	varchar(37)	utf8mb4_general_ci		Sí	NULL
6	listo	varchar(2)	utf8mb4_general_ci		No	Ninguna

Figura 3.28 Estructura de la tabla proyectos

TABLA USUARIOS:

La tabla usuarios contienen la información de registro e ingreso de los usuarios que hacen uso del sistema de control domótico. Cabe destacar que el atributo **clave** es varchar(32) porque la clave que proporciona el usuario a la hora de registro es encriptada por el servidor bajo el hash md5, por lo que la clave no es visible para ninguna persona o máquina que tenga acceso a la base de datos como forma de seguridad. Sus atributos son **id**, **usuario**, **email** y **clave**.

DIAGRAMA ENTIDAD RELACIÓN (E-R)

Un diagrama entidad-relación o modelo entidad relación o ERD, es un tipo de diagrama de flujo que ilustra cómo las entidades o tablas que conforman una base de datos se relacionan entre sí dentro de un sistema. En las Figuras 3.29 y 3.30 se muestran los diagramas empleados para diseñar las bases de datos relacionales.

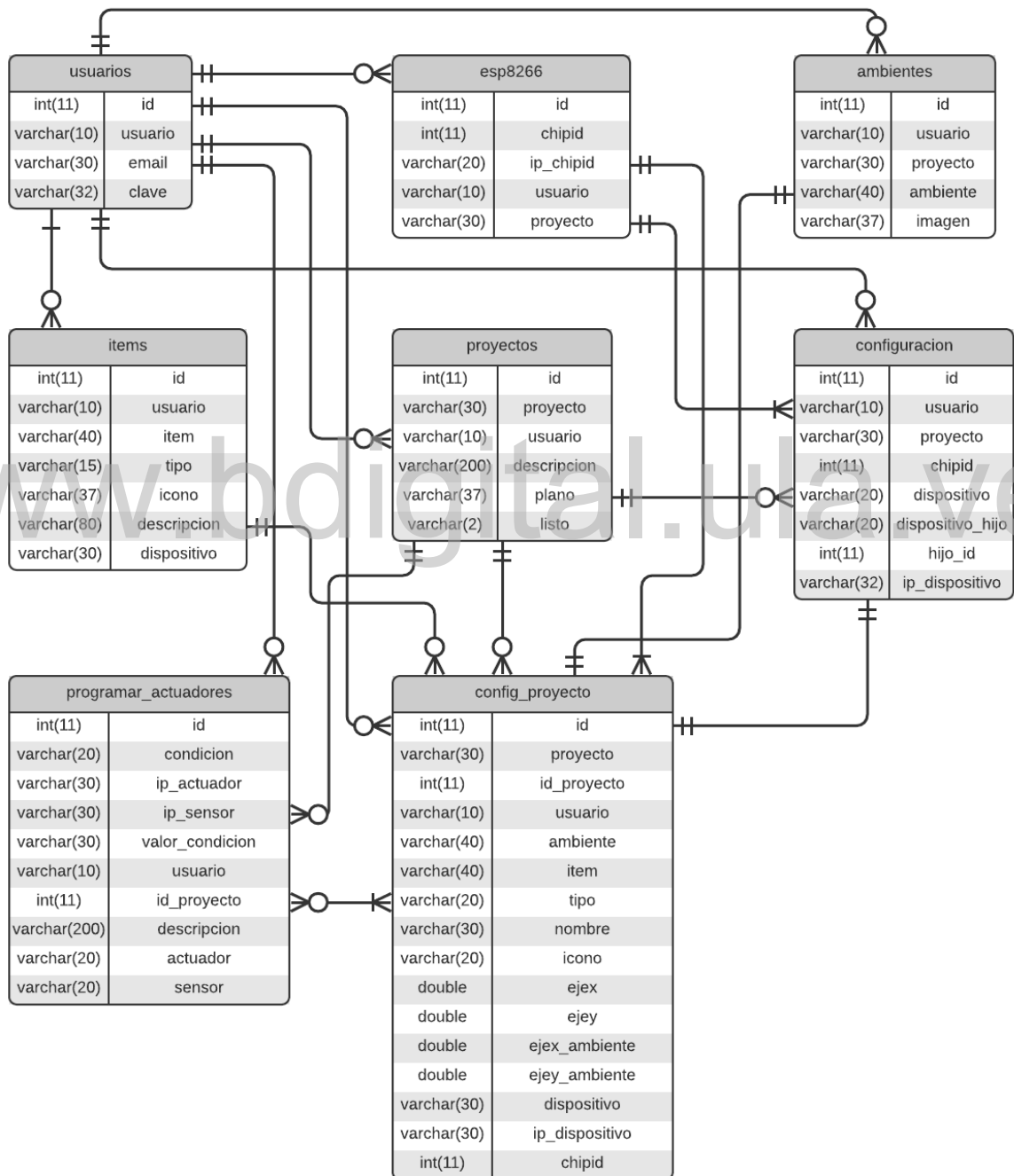


Figura 3.29 Diagrama entidad-relación de la base de datos 1

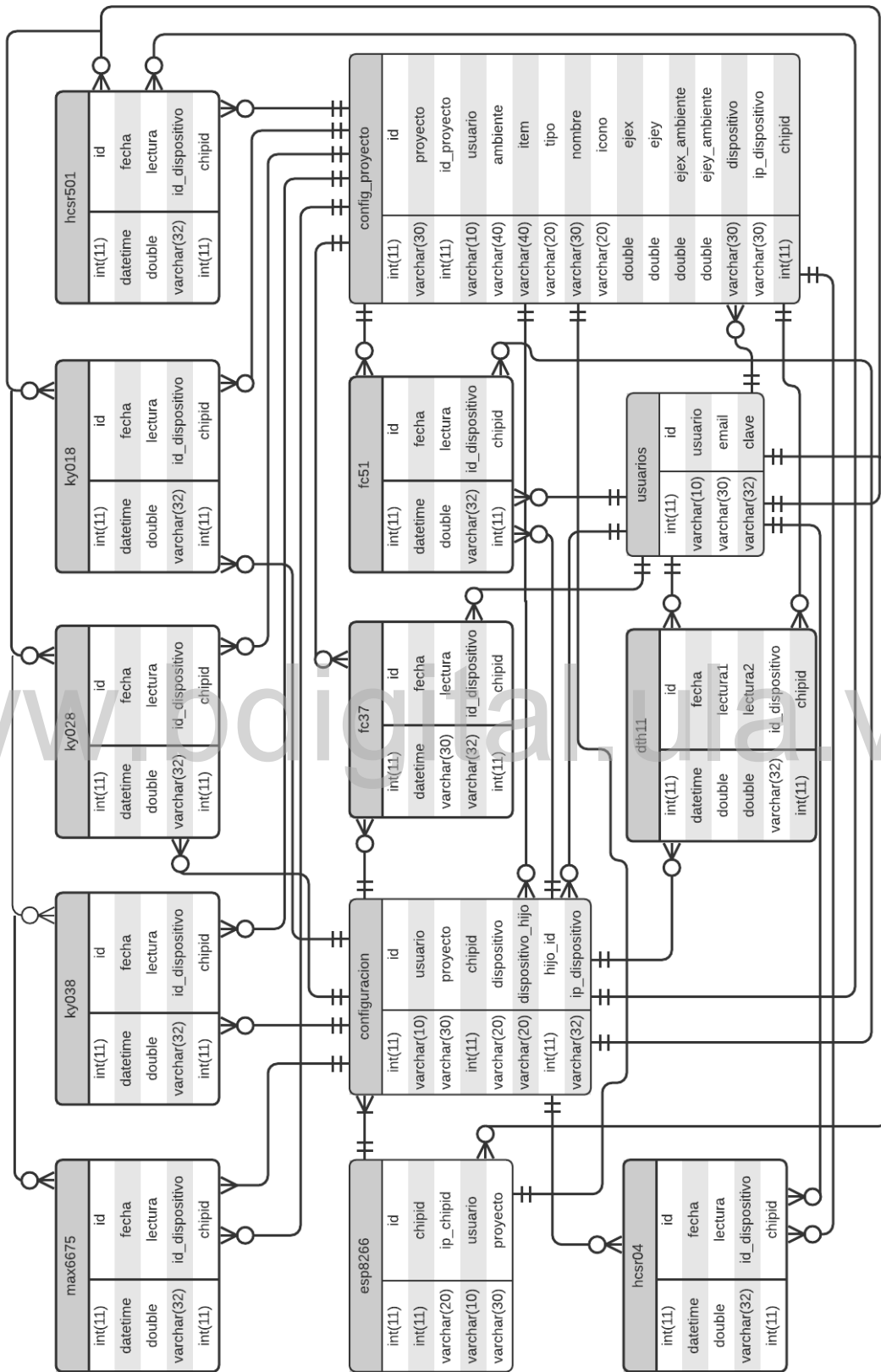


Figura 3.30 Diagrama entidad-relación de la base de datos parte 2

3.5.11 DESCRIPCIÓN DE LA INTERFAZ

Como resultado del análisis de la etapa anterior, la interfaz a diseñar debe cumplir con ciertos requerimientos, además de ser amigable, común e intuitiva. Adicionalmente se requiere que la interfaz de usuario sea *responsive* o adaptativa, de manera que pueda ser utilizada en cualquier tipo de pantalla. En general, los principios de diseño que se definirán para esta interfaz son los siguientes:

- Permitir que el usuario tenga una interacción flexible.
- Generar una interfaz que evite tecnicismos y el usuario pueda utilizar sin tener conocimientos de ingeniería.
- Facilitar la interacción del usuario con la interfaz, permitiendo que algunas funcionalidades puedan ser personalizadas.
- Definición de los modos de interacción con la finalidad de que el usuario no realice acciones innecesarias.
- La interacción del usuario debe ser interrumpible y reversible.
- Diseñar una interacción directa con objetos mostrados en la pantalla.
- Interfaz adaptativa o *responsive*.

3.5.12 PROTOTIPOS DE PANTALLA

Las características inherentes a la interfaz de usuario a diseñar deben ser: claridad, concisión, coherencia, legibilidad, interactividad, flexibilidad, familiaridad, eficiencia y estética. Por ello, se hace necesario establecer una idea de la visualización de los elementos que componen la interfaz, así pues, a continuación, se mostrarán los diferentes prototipos de pantalla diseñados para la caracterización de la interfaz que se codificará.

PROTOTIPO DE BARRA DE MENÚ

Para el diseño de la barra de menú, se considerará un menú lateral en disposición vertical que contenga los elementos descritos en la Figura 3.31 mostrada a continuación, los cuales corresponden a: inicio, proyectos, items o dispositivos, base de datos, sensores, indicadores, control y configuración.

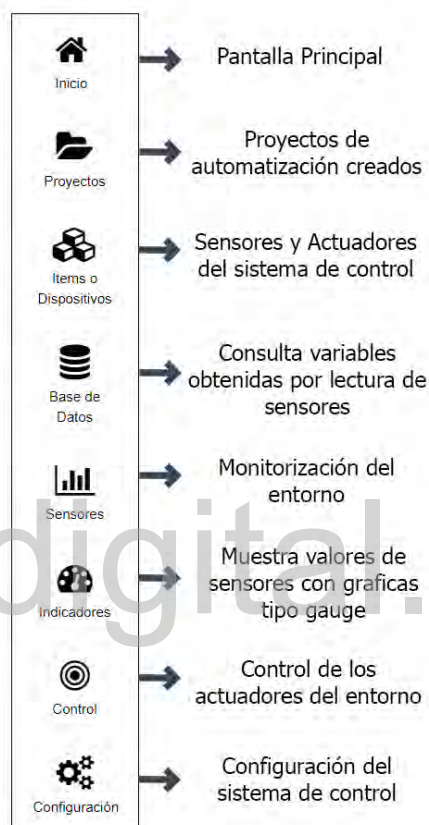


Figura 3.31 Prototipo de pantalla para menú de la interfaz de usuario

PROTOTIPO DE PANTALLA DE LOGIN Y REGISTRO

La pantalla de *login* o acceso al sistema deberá contener un botón que habilitará a los usuarios el registro en la aplicación, así como un formulario que permita el ingreso de los usuarios a la plataforma donde se requiera el ingreso del nombre del usuario, así como la clave asociada al mismo. También debe incluirse un botón para ejecutar el proceso de autenticación en la aplicación. Esto se observa en la Figura 3.32.

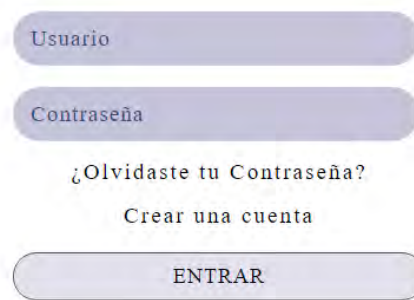


Figura 3.32 Prototipo de pantalla de login

La pantalla de registro del sistema de control domótico está constituida principalmente por un formulario web que permite al usuario introducir ciertos datos requeridos por la aplicación para la creación de una cuenta en el sistema. El prototipo de pantalla incluye la entrada de datos de: nombre de usuario, email, contraseña y una verificación de que la contraseña proporcionada está bien escrita.

PROTOTIPO DE PANTALLA DE INICIO

La pantalla de inicio del sistema será el punto de entrada a la plataforma de automatización domótica, aportando información general sobre la aplicación, además de información descriptiva del sistema.

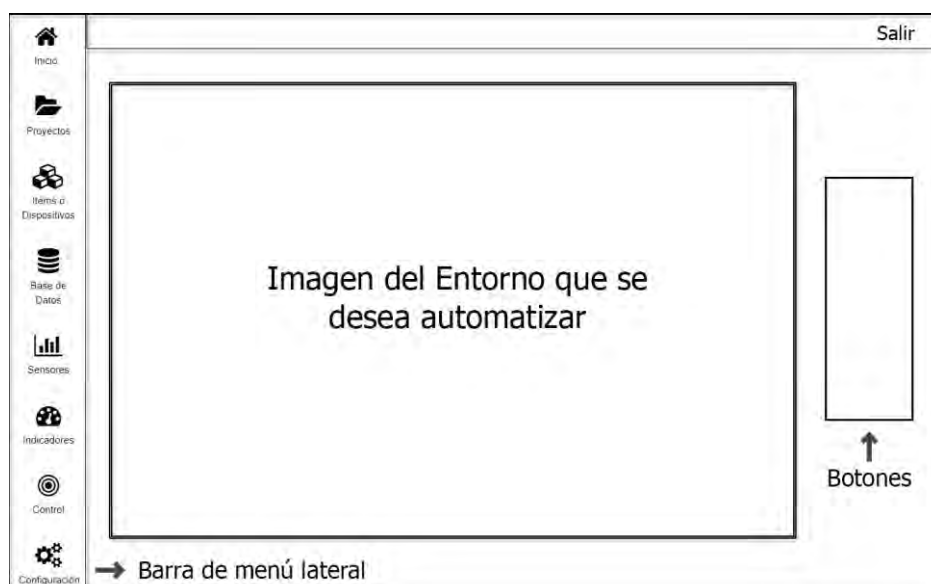


Figura 3.33 Prototipo de pantalla de inicio

Este prototipo de pantalla es un requisito funcional del sistema. Esta pantalla deberá contener la barra de menú principal, así como también un plano o imagen correspondiente al entorno domótico a automatizar, tal y como se mostró en la Figura 3.33.

3.5.13 IDENTIDAD VISUAL

El diseño de la identidad visual del sistema de control domótico corresponde a la expresión externa del producto o plataforma a implementar, lo cual constituye el medio fundamental para que el usuario asocie, identifique o reconozca la plataforma entre otras. El diseño de identidad visual incluye el nombre de la plataforma, logotipo y apariencia visual en general.

NOMBRE DE LA PLATAFORMA:

El nombre seleccionado para la plataforma corresponde a: *DomoticTech*. La primera parte del nombre *Domotic* corresponde a la palabra “Domótica” en inglés, y la última parte, *Tech* se refiere a “Tecnología”. Por lo tanto, el nombre dado a la plataforma puede ser entendido como “Tecnología domótica”.

LOGOTIPO DE LA PLATAFORMA

El logotipo identificará la plataforma a través del uso de símbolo que lo caracterice. Este logotipo estará presente en cualquier interacción entre el usuario y la interfaz. El logotipo corresponde a una representación gráfica del sistema de control y éste debe ser sintetizado de forma que sea fácilmente identificado por el usuario, prescindiendo de información innecesaria. Inicialmente, para efectos del diseño del logotipo que identifique la plataforma del sistema domótico, se procedió a realizar un boceto del mismo, tal y como se aprecia en la figura 3.34. En las figuras que le preceden (Figuras 3.35 y 3.36) se observará el acabado final del logotipo.

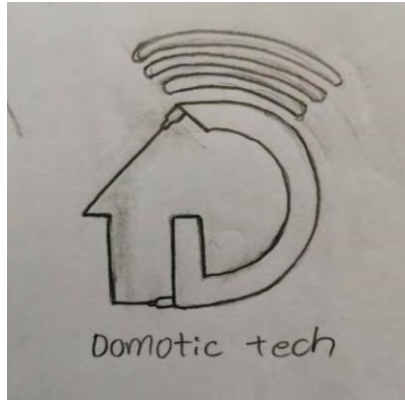


Figura 3.34 Boceto final del logotipo de la plataforma



Figura 3.35 Logotipo de la plataforma



Figura 3.36 Aplicación logotipo de la plataforma

3.6 ETAPA 3: CODIFICACIÓN

Esta etapa consiste en transformar el diseño en un lenguaje de programación. La codificación del diseño de la etapa anterior será un procedimiento en el que se comenzará a crear las estructuras de programación necesarias para la creación del software, el cual será la aplicación para los dispositivos electrónicos. Este procedimiento puede realizarse con cualquier lenguaje de programación y constituye uno de los pasos finales que se efectúan para la creación de aplicación en la ingeniería de software.

3.6.1 ELECCIÓN DE LOS LENGUAJES DE CONDIFICACIÓN

Para la elección de los lenguajes de codificación que traducirán el diseño de la etapa anterior a un lenguaje de máquina, se tomó en cuenta diferentes criterios como lo es el área de aplicación del sistema diseñado, la complejidad algorítmica, consideraciones de rendimiento, complejidad de las estructuras de datos y el entorno en que se ejecutará el sistema. A continuación, se especificará los lenguajes que cumplieron con el criterio para ser utilizados en la codificación.

- Lenguaje de etiquetas HTML5
- Lenguaje de clases CSS3
- Lenguaje de programación JavaScript
- Lenguaje de consulta SQL
- Lenguaje de programación PHP
- Lenguaje de programación basado en C++

En la Figura 3.38 puede observarse un esquema en donde se señala los lenguajes de codificación seleccionados de acuerdo a la aplicación del sistema y los criterios antes mencionados.

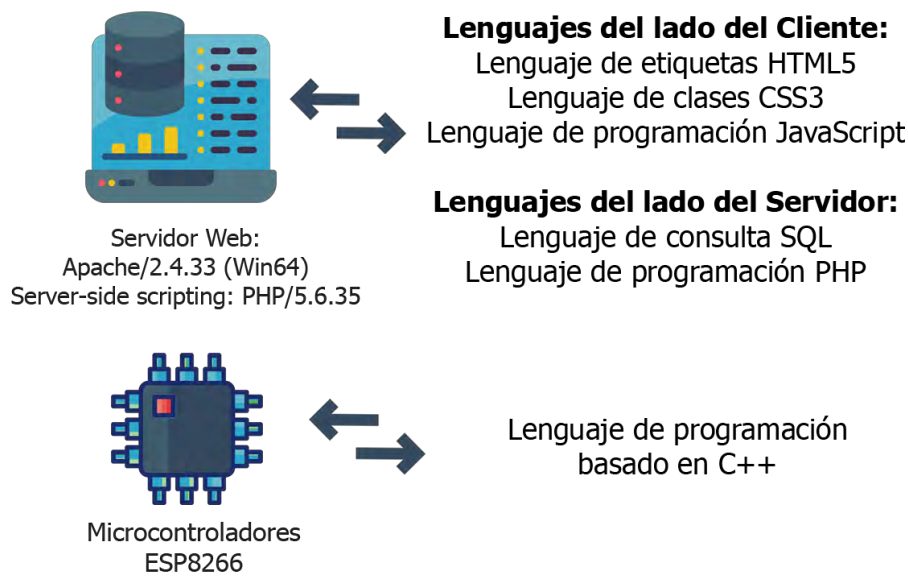


Figura 3.37 Esquema de los lenguajes de codificación del software

3.6.2 ACTIVIDADES DE LA CODIFICACIÓN

Siguiendo los lineamientos de la etapa de diseño, y en base a los lenguajes de codificación seleccionados se procede a:

a) Escritura del código fuente: se realizó la codificación de los diferentes programas necesarios para la implementación del sistema domótico. Para el caso de los programas asociados con los microcontroladores, cuya codificación se efectuó a través de un lenguaje de programación basado C++, el proceso de traducción del diseño finalizó cuando el compilador aceptó el código fuente como entrada y produjo los resultados esperados.

Como puede observarse en el flujograma de la Figura 3.38, donde se ejemplifica este proceso de programación de los microcontroladores ESP8266, véase que este inicia con la configuración inicial del hardware y se establecen los parámetros de configuración WIFI predeterminados. Si el microcontrolador no puede conectarse a la red domótica, entonces habilita un punto de acceso en el que permite al usuario agregar nuevos parámetros de red y el microcontrolador se mantendrá en este estado hasta que sea capaz de conectarse a la red domótica. Si se establece la conexión, entonces los nuevos parámetros de red son guardados.

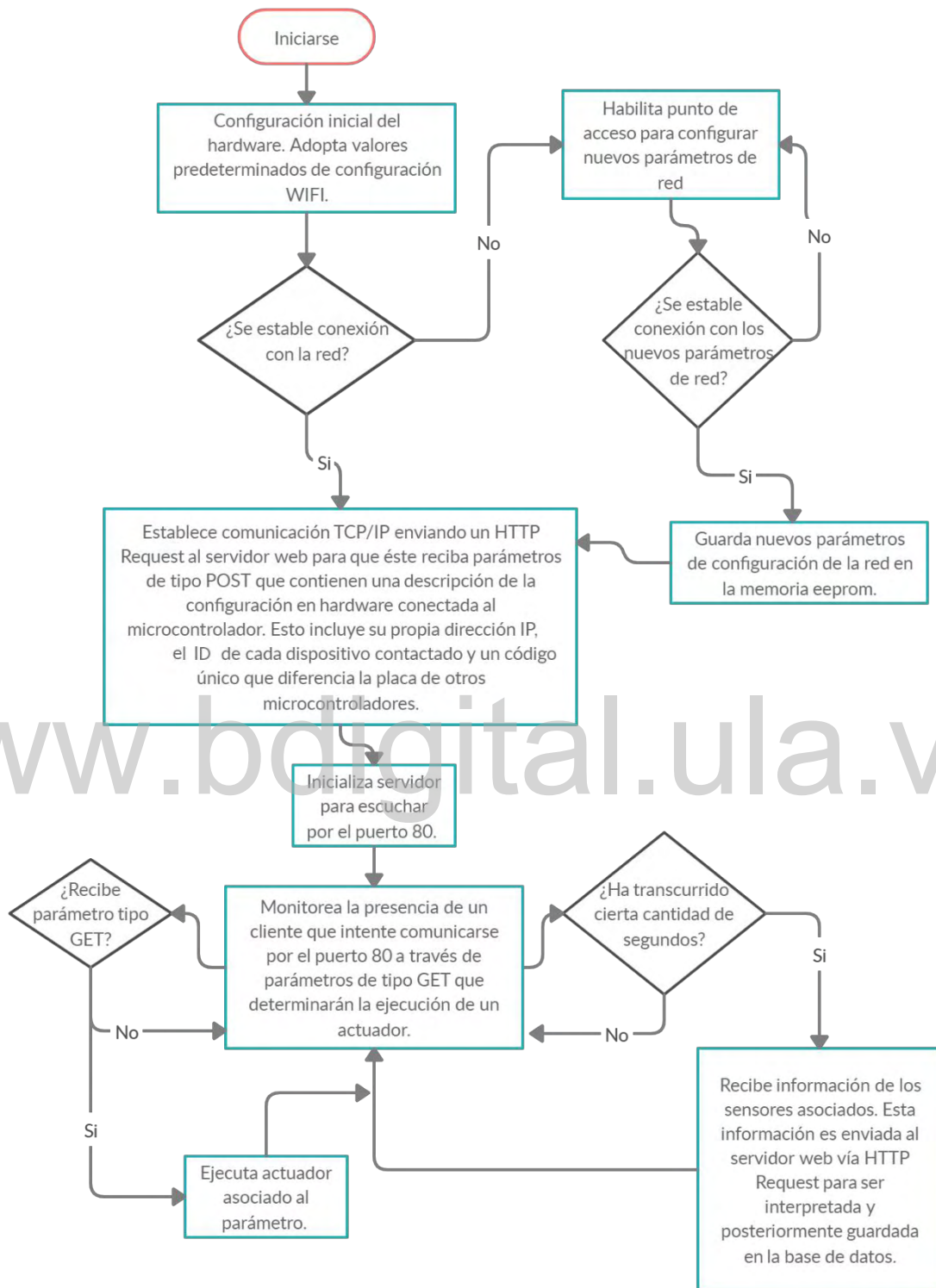


Figura 3.38 Diagrama de flujo para la programación de los ESP8266

Una vez que los microcontroladores están conectados a la red domótica, ellos asignan un id a cada dispositivo (sensor o actuador) conectados y establecen una comunicación TCP/IP con el servidor web en la que, a través de un HTTP *request* de tipo POST, envían esta información al servidor domótico para ser guardada en la base de datos, proporcionando así una descripción de la configuración en hardware conectado a los nodos. Esta información se refiere a la dirección ip de los nodos, el id de cada dispositivo, y un identificador único de la placa ESP8266. Este proceso es descrito con mayor detalle en la sección 3.6.3.

Una vez que la información es enviada, los nodos habilitan el modo servidor para escuchar por el puerto 80, monitoreando constantemente las peticiones recibidas por parte de los clientes desde el servidor domótico a través de parámetros de tipo GET. Estos parámetros determinan la ejecución de un actuador, por lo que si son recibidos, se ejecuta el actuador asociado al parámetro. Adicionalmente, cada cierta cantidad de segundos, dependiendo del tipo de sensor, los valores de sensado son enviados al servidor web vía HTTP *request* para ser interpretados y guardados en la base de datos. Esta información es enviada de forma similar a la descrita en el apartado 3.6.3.

En el caso la programación del software asociado a la interfaz del programa, se emplearon lenguajes de programación que no requieren ser compilados o interpretados como PHP y JavaScript. Un poco de este proceso se detalla en el apartado 3.6.5, donde se detalla el proceso de la adquisición de datos por parte del servidor web al recibir el HTTP *request* de los microcontroladores.

b) Realizar pruebas de funcionalidad: Al obtener una versión terminada del sistema de control, se efectuaron diferentes pruebas funcionales que aseguran que las entradas definidas producen los resultados esperados, concluyendo que los componentes del sistema funcionan correctamente, aunque aún no haya sido probado en un ambiente domótico de implementación.

3.6.3 ASIGNACIÓN DE UN IDENTIFICADOR ID A CADA DISPOSITIVO

En el código de programación de los microcontroladores ESP8266 cada dispositivo, ya sea de entrada o de salida, tendrá asignado una identificación única o **id**. A continuación, en la tabla 2 y 3, se muestra el identificador que se le asignó a cada dispositivo dentro del código de programación. Obsérvese que el nodo o microcontrolador posee un identificador único conocido como **chipid**, el cual está dado por el fabricante, mientras que el resto de los identificadores de los sensores y actuadores están dados por el programador.

Tabla 2. Asignación de identificadores en el nodo 1

Dispositivo	Identificador (Id)
ESP8266 (nodo)	1255589
Dispositivos de entrada (sensores)	
fc-37	000.000.0.001
hc-sr04 (1)	000.000.0.002
fc-51	000.000.0.003
ky-028	000.000.0.004
max6675	000.000.0.005
hc-sr04 (2)	000.000.0.006
Dispositivos de salida (actuadores)	
Led (1)	000.000.0.007
Led (2)	000.000.0.008

Tabla 3. Asignación de identificadores en el nodo 2

Dispositivo	Identificador (Id)
ESP8266 (nodo)	12770691
Dispositivos de entrada (sensores)	
ky-018	000.000.0.009
ky038	000.000.0.013
dht11	000.000.0.014
Dispositivos de salida (actuadores)	
28byj-48	000.000.0.010
sg-90 (1)	000.000.0.012
Relé	000.000.0.015
<i>Buzzer</i>	000.000.0.016
sg-90 (2)	000.000.0.017

Cuando el nodo se conecta a la red domótica, éste envía al servidor toda la información de los dispositivos conectados al mismo a través de una petición HTTP, tal y como se explicará en el siguiente apartado.

3.6.4 HTTP REQUEST DE LOS ESP8266

Dentro del código de programación de los microcontroladores ESP8266 se encuentra una función en común llamada **enviardatos**, esta función se encarga de definir la trama de la petición HTTP que será la responsable de establecer comunicación desde los nodos (ESP8266) hacia el servidor. Esta función es utilizada únicamente para el envío de información en esta dirección. Ahora bien, para la recepción de información en los microcontroladores, el servidor se encarga de enviar los datos a la dirección *ip* de los nodos a través de una petición HTTP de tipo GET. En la Figura 3.39 se puede apreciar que la petición HTTP creada por los microcontroladores, la cual específicamente se encuentra contenida dentro del recuadro y la cual contempla los siguientes aspectos:

Línea de Inicio:

1. Se especifica el método HTTP, que en este caso se corresponde al método POST el cual indica que hay datos que serán enviados hacia el servidor.
2. El objetivo de la petición, en este caso, la *URL* que indica la carpeta y el archivo PHP al cual será enviado la petición.
3. La versión de HTTP que define la estructura del mensaje (HTTP/1.1)

```
String enviardatos(String datos) {
  String linea = "error";
  WiFiClient client;
  strhost.toCharArray(host, 49);
  if (!client.connect(host, 80)) {
    return linea;
  }

  client.print(String("POST ") + strurl + " HTTP/1.1" + "\r\n" +
    "Host: " + strhost + "\r\n" +
    "Accept: */*" + "\r\n" +
    "Content-Length: " + datos.length() + "\r\n" +
    "Content-Type: application/x-www-form-urlencoded" + "\r\n" +
    "\r\n" + datos);
  delay(10);
}
```

Figura 3.39 Fragmento de código de programación de los ESP8266

Cabecera de la petición:

- 1) *Host*: indica la dirección *ip* o nombre de dominio del servidor al que se le envía la información. En este caso la variable **strhost** contiene la *ip*.
- 2) *Accept*: especifica las extensiones multipropósito de correo de internet aceptadas en la petición.

Cuerpo de la petición:

- 1) *Content-length*: indica la longitud del mensaje. En el código se observa que se utiliza la longitud de la variable **datos**, la cual almacena el mensaje a enviar.
- 2) *Content-type*: indica las extensiones multipropósito de correo de internet del contenido y seguidamente se incluye el contenido del mensaje.

Un ejemplo del contenido almacenado por la variable **datos** se muestra en la Figura 3.40, siendo ésta la información que un nodo envía al servidor a través de la función **enviardatos** cuando el nodo es encendido o reseteado, proporcionando información al servidor que incluye el **id** de cada dispositivo conectado al nodo, así como el **chipid** o identificador único del nodo y la dirección *ip* asignada al mismo. Toda esta información será recibida por el servidor y se almacenará posteriormente en la base de datos en la tabla llamada **configuración**, tal y como se explica en el apartado 3.5.10. Cabe destacar que, de manera similar y empleando la misma función **enviardatos**, el nodo enviará datos al servidor relacionado con las lecturas obtenidas por los diferentes sensores que componen el nodo.

```
enviardatos("iniciar_chipid=" + chipid + "&1ky-018=000.000.0.009" + "&128byj-48=000.000.0.010"  
+ "&1hc-sr501=000.000.0.011" + "&1sg-90=000.000.0.012" + "&1ky-038=000.000.0.013" +  
"&1dht11=000.000.0.014" + "&1rele=000.000.0.015" + "&1buzzer=000.000.0.016" +  
"&2sg-90=000.000.0.017" + "&ip_esp8266=" + ip_esp);
```

Figura 3.40 Ejemplo de datos enviados al servidor

3.6.5 RECEPCIÓN DE DATOS DE LOS ESP8266

El servidor envía información a los nodos a través de un HTTP *request* empleando el método GET, para ello, tal y como se observa en la siguiente figura 3.41, se hace una consulta SQL en la base de datos para determinar la *ip* del

microcontrolador o nodo al que se le enviará la información (recuadro azul), usando como criterio de búsqueda el identificador o **id** del microcontrolador. Una vez efectuada la búsqueda, el servidor crea la ruta de envío de datos (**\$url**) como se observa en el recuadro amarillo, usando la dirección ip obtenida en la consulta, y añadiendo como variable GET los datos a enviar (**\$data**). En el último recuadro de la figura se observa la estructura de la petición HTTP efectuada desde el servidor hacia el nodo en cuestión, la cual se encuentra estructurada de forma similar a como se explicó anteriormente, con la diferencia de que, en este caso, se empleó el lenguaje de servidor PHP. La variable **\$resultado** guarda el **id** con el que se identifica el actuador a ser accionado, por lo que, cuando el mensaje es recibido por el nodo, este podrá identificar de manera unívoca el dispositivo en cuestión.

```

mysqli_select_db($conex,"sevidor_domotico");
$query_datos_esp="SELECT * FROM esp8266 WHERE chipid = '". $_POST['chipid']."' ";
$datos_esp = mysqli_query($conex, $query_datos_esp);
$row_datos_esp = mysqli_fetch_assoc($datos_esp);
$totalRows_datos_esp= mysqli_num_rows($datos_esp);

$url="http://".$_row_datos_esp['ip_chipid']."/recibir?". $resultado."=".$condi;

$data = http_build_query( array( 'variable' => '0' ) );

$options = array(
    'http' => array(
        'header' => "Content-type: application/x-www-form-urlencoded",
        'method' => 'GET',
        'content' => $data,
    ),
);

$context = stream_context_create( $options );

$result = file_get_contents( $url, false, $context );

```

Figura 3.41 Envío de datos desde el servidor a los nodos

Ahora bien, los microcontroladores reciben esta información a través de una función llamada **recibirdatos**, esto se observa en la Figura 3.42, la cual muestra un fragmento de las funciones empleadas para la recepción de datos. Estas funciones son empleadas para el control de los actuadores del sistema, ya que los nodos leen las variables GET recibidas desde el servidor y su contenido es asignado de acuerdo al **id** de cada actuador, una vez guardada la información recibida, los nodos estarán listos para ejecutar los actuadores correspondientes.

```
void recibirdatos() {
motor=server.arg("000_000_0_010");
servo_1=server.arg("000_000_0_012");
servo_2=server.arg("000_000_0_017");
rele=server.arg("000_000_0_015");
buzzer_1=server.arg("000_000_0_016");
```

```
void recibirdatos() {
rele = server.arg("000_000_0_007");
led = server.arg("000_000_0_008");
```

Figura 3.42 Fragmento de código de programación para la recepción de datos 1

Básicamente, la función **recibirdatos**, es invocada constantemente por el nodo para asegurarse de recibir los datos provenientes del servidor, tal y como se observa en el recuadro morado de la figura 3.43. Esta función es configurada al encenderse o resetearse el dispositivo, tal y como se observa en el recuadro azul.

```
server.on("/recibir", recibirdatos);
server.begin();
```

```
server.handleClient();
```

Figura 3.43 Fragmento de código de programación para la recepción de datos 2

3.7 ETAPA 4: PRUEBAS

Una vez que la plataforma de control domótica se ha terminado de codificar, la siguiente fase del modelo metodológico corresponde a la etapa de pruebas. En esta etapa el sistema de control domótico debe ser instalado en el ambiente de pruebas, esto quiere decir que, el hardware asociado al sistema estará censando y controlado un ambiente determinado, ya que en esta fase metodológica se busca comprobar que el sistema diseñado y codificado funciona correctamente cumpliendo con los requerimientos anteriormente especificados. En primer lugar, se procedió a hacer un montaje del circuito electrónico necesario para la operatividad de los diferentes sensores, actuadores y microcontroladores del sistema domótico y se comprobó que tantos los sensores, actuadores, así como los microcontroladores, estuviesen funcionando correctamente.

Para esta prueba, se emplearon dos microcontroladores ESP8266 y todos los dispositivos anteriormente descritos en esta investigación. El esquema empleado para el montaje en el ambiente de pruebas se observa en la Figura 3.44 y 3.45 respectivamente, cada uno de estos esquemas representa un nodo en el sistema de control domótico. Una vez efectuado el montaje, los resultados de dicha prueba indicarán si se consiguió un correcto funcionamiento del sistema de control diseñado. Cabe destacar que, tal y como se muestra en los presentes esquemas, la conexión de cada dispositivo al nodo o microcontrolador ESP8266 es cableada, pero la comunicación del nodo con el servidor es completamente inalámbrica, así que no se requerirá más conexión cableada que la mostrada en los esquemas.

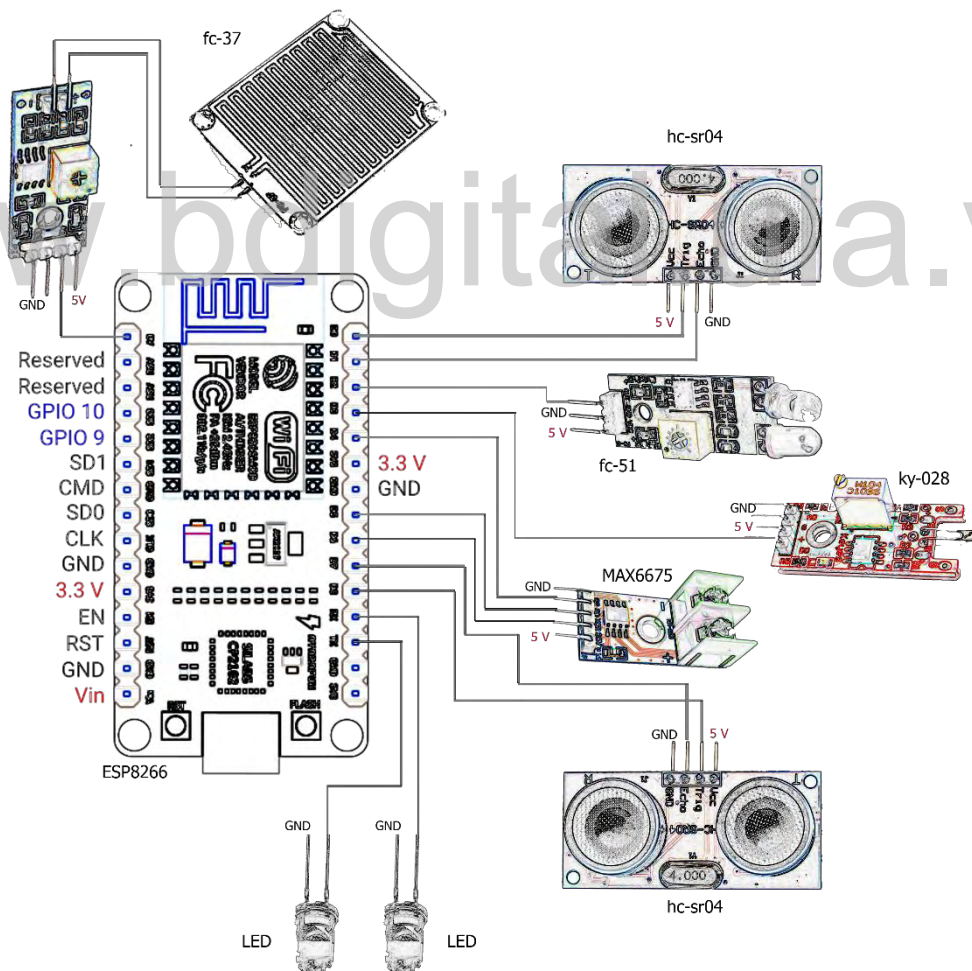


Figura 3.44 Esquema de circuito electrónico de prueba 1

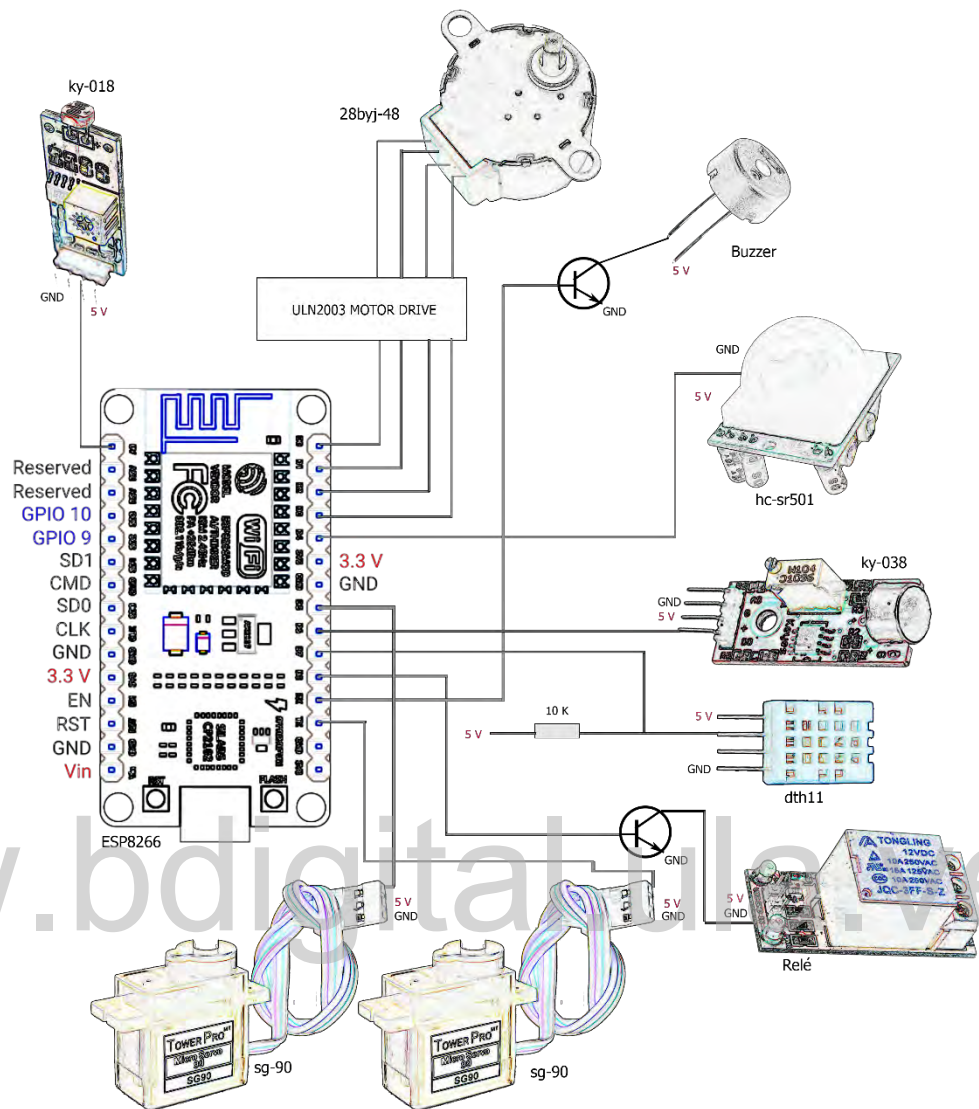


Figura 3.45 Esquema de circuito electrónico de prueba 2

Se abrió una cuenta en la plataforma y se creó un proyecto de automatización. En este punto, se verificó que cada uno de los sensores conectados y vinculados a la plataforma arrojaran valores de sensado que fuesen visualizados en la plataforma a través de gráficos. Se comprobó que cada uno de los actuadores funcionasen correctamente, ya sea al ser accionados por el usuario desde la plataforma o al ser programados para accionarse de acuerdo a un horario o valor dado por un sensor del sistema.

La quinta y última etapa, correspondiente a la etapa de implementación, será descrita en el siguiente capítulo.

CAPITULO IV: IMPLEMENTACIÓN

En esta sección, se efectuará una descripción de la interfaz gráfica de usuario de la plataforma diseñada, así como su respectiva implementación y funcionamiento en un entorno domótico.

4.1 LOGIN

A continuación, se observa la interfaz gráfica diseñada para el *login* de la plataforma de control domótico *DomoticTech*.



Figura 4.1 Interfaz de login

4.2 CREAR PROYECTO DE AUTOMATIZACIÓN

La plataforma para el sistema de control domótico diseñada le permitirá al usuario crear diferentes proyectos. Cada proyecto corresponde a un entorno de automatización, lo que supone que un mismo usuario podrá ser capaz de controlar entornos separados desde su misma cuenta, por ejemplo, controlar y monitorizar su casa y también su oficina, entornos geográficamente separados, pero que son agrupados en una misma plataforma. Para crear un proyecto el usuario deberá seguir los pasos indicados por el sistema.

Crear Proyecto Configuración de un servidor Domótico

A continuación deberá llenar el siguiente formulario para la configuración inicial de su nuevo servidor domótico. Se le indicará que ingrese las características básicas del entorno que desea automatizar a través del servidor. Debe hacer click en "Siguiete" cada vez que finalice un paso.

1 Paso 1 Configurar Proyecto

2 Paso 2 Configurar ambientes

3 Paso 3 Cargar Plano

4 Paso 4 Finalizar Configuración

Nombre del Proyecto

Descripción del Proyecto (Opcional)
Opcionalmente puede agregar una breve descripción de su proyecto.

Figura 4.2 Creación de proyecto

4.2.1 Primer paso: configurar proyecto

En este paso el usuario ingresará el nombre que desea asignar al proyecto, así como una breve descripción del mismo, éste último campo es opcional. Como ejemplo, se creará un proyecto llamado “Mi hogar”.

4.2.2 Segundo paso: configurar ambientes

En este paso, el usuario podrá ser capaz de ingresar los ambientes que conforman el entorno domótico que está creando. Estos ambientes pueden ser: salas, habitaciones, baños, exteriores, cubículos, cocinas, comedores, entre otros.

1 Paso 1 Configurar Proyecto

2 Paso 2 Configurar ambientes

3 Paso 3 Cargar Plano

4 Paso 4 Finalizar Configuración

Nombre del ambiente

Aquí podrá configurar los ambientes o entornos pertenecientes al proyecto que desea automatizar. El sistema le dará la opción de agregar tantos ambientes como usted desee. Por ejemplo: habitación, exterior, baño, cocina, cubículo, jardín, salón, etc.

Figura 4.3 Segundo paso para crear proyecto

4.2.3 Tercer Paso: Cargar plano

En el tercer paso, el usuario deberá cargar un plano del ambiente que desea automatizar. Convenientemente, éste debería ser un plano en el que se pueda apreciar los diferentes ambientes que componen el entorno domótico.

4.2.4 Cuarto paso: finalizar configuración

Una vez que el usuario considere que ya ha terminado de ingresar todos los datos correspondientes al proyecto y ya no desee incluir más ambientes, entonces deberá hacer clic en “finalizar”. Una vez que el usuario haya finalizado, si es la primera vez que ha creado un proyecto, el sistema le solicitará que seleccione dicho proyecto como proyecto principal o proyecto a utilizar, tal y como se aprecia en la siguiente figura.



Figura 4.4 Selección de proyecto a utilizar

Si el usuario tuviese más proyectos asociados a su cuenta, entonces aparecerá la lista de todos los proyectos creados, permitiendo que el usuario seleccione el proyecto a utilizar. Normalmente esta ventana aparece cada vez que el usuario ingresa a la plataforma, pudiendo decidir con qué proyecto trabajar. Si en algún momento el usuario cambia de opinión y desea cambiar de proyecto, puede hacerlo en cualquier momento sin necesidad de salir y volver a ingresar a la plataforma. Este procedimiento se especificará más adelante.

4.3 PANTALLA PRINCIPAL

Una vez que se ha seleccionado el proyecto a utilizar, entonces el sistema mostrará dicho proyecto en la página de inicio o página principal como se observa a continuación. El usuario puede acceder a la pantalla de inicio en cualquier momento al hacer clic en la opción “inicio” del menú principal.



Figura 4.5 Pantalla de inicio

El recuadro mostrado en la esquina superior derecha corresponde a información que la plataforma proporciona al usuario cuando ha creado un proyecto por primera vez, de manera que, si el usuario no está familiarizado con la plataforma o no desea leer la documentación asociada a la misma, podrá hacer uso de la plataforma sin dificultades. Estos recuadros pueden ser cerrados por el usuario en cualquier momento y desaparecen después de una cantidad razonable de tiempo. Obsérvese también que el plano mostrado en la pantalla principal corresponde al plano que el usuario haya subido durante la creación del proyecto.

4.4 MENÚ DE LA PLATAFORMA

Cuando se ha ingresado en la plataforma, se observa que el sistema consta de un menú principal ubicado de forma lateral a la izquierda de la pantalla. Este

menú surgió a través del prototipo de pantalla propuesto en la fase de diseño del sistema de control. La siguiente figura especifica las funciones que cumple cada botón del menú y a continuación se mostrará a profundidad las características asociadas a cada sección comprendida por el menú principal.



Figura 4.6 Menú principal de la plataforma

4.5 ÍTEMS O DISPOSITIVOS DE AUTOMATIZACIÓN

Al hacer clic en la opción “ítems o dispositivos” del menú principal, el usuario accederá a la sección encargada de personalizar los dispositivos de automatización que se utilizarán en los diferentes proyectos del usuario. La plataforma trae un total de 14 dispositivos predeterminados y tiene la opción de hacer uso de estos ítems predeterminados para cualquiera de sus proyectos, o puede personalizarlos de manera que su experiencia en el uso de la plataforma sea más amigable. La personalización de los dispositivos consiste en crear un dispositivo, ya sea de entrada o de salida, darles un nombre y una descripción de acuerdo a las preferencias del usuario, así como asignarle una imagen o ícono para representarlo.



Figura 4.7 Ítems o dispositivos de automatización predeterminados

Configuración de Items Dispositivos de automatización

A continuación podrá crear y personalizar la cantidad de items que desee y posteriormente configurarlos. Por favor complete el formulario. Podrá asociar estos items a cualquier proyecto.

Item

Tipo:

Descripción

Figura 4.8 Configuración o personalización de ítems

En este punto, el usuario deberá asignar un nombre para el ítem que pretende crear o personalizar, luego deberá decidir si se trata de un dispositivo de entrada o de salida. Si el usuario elige el dispositivo como una entrada, entonces el sistema

habilitará una lista de sensores, mientras que si, por otro lado, el usuario seleccionar el dispositivo como una salida, entonces el sistema habilitará una lista de actuadores. A continuación, se observa un ejemplo de personalización del ítem, al cual se le asignará el nombre de “detector de fuego” y se seleccionará como dispositivo de entrada.

Al ser un dispositivo de entrada, el sistema habilitará una lista de sensores disponibles y se escogerá la opción del sensor termistor entre la lista de sensores, ya que este sensor es capaz de detectar la presencia del fuego.

The screenshot shows a web interface titled "Configuración de Items" with a subtitle "Dispositivos de automatización". Below the title, there is a paragraph: "A continuación podrá crear y personalizar la cantidad de items que desee y posteriormente configurarlos. Por favor complete el formulario. Podrá asociar estos items a cualquier proyecto." The form contains the following fields:

- Item:** A text input field containing "Detector de fuego".
- Tipo:** A dropdown menu set to "Entrada".
- Sensor:** A dropdown menu set to "Seleccionar".
- Descripción:** A text input field, currently empty.
- Selección:** A list of sensor options with radio buttons. The "Sensor termistor KY-028" option is selected and highlighted in blue.

At the bottom of the form, there is a grid of icons representing various sensors and actuators, including a light bulb, a plant, a water drop, a clock, a key, a robot arm, a bar chart, a tablet, and a trophy.

Figura 4.9 Personalizando ítem de entrada 1

En este punto, el usuario podrá escribir una descripción del ítem si así lo desea, y seleccionará un ícono de entre la lista proporcionada por el sistema, que le permitirá representar su ítem o dispositivo de automatización, en el caso del ejemplo mostrado, se seleccionó un ícono que representa la detección del fuego. Para terminar, el usuario debe hacer clic en “generar ítem”.



Figura 4.10 Personalizando ítem de entrada 3

Una vez que el ítem o dispositivo de automatización haya sido creado, el sistema lo mostrará en la lista de dispositivos agregados como se observa a continuación. El usuario tendrá la opción de eliminarlo en cualquier momento haciendo clic en el ícono representado por un cesto de basura.



Figura 4.11 Lista de dispositivos agregados

El usuario podrá agregar o personalizar la cantidad de ítems que desee. A manera de ejemplificar esta, se crearon un total de 7 dispositivos de automatización personalizados, los cuales aparecen en el listado de dispositivos agregados como se observa a continuación.

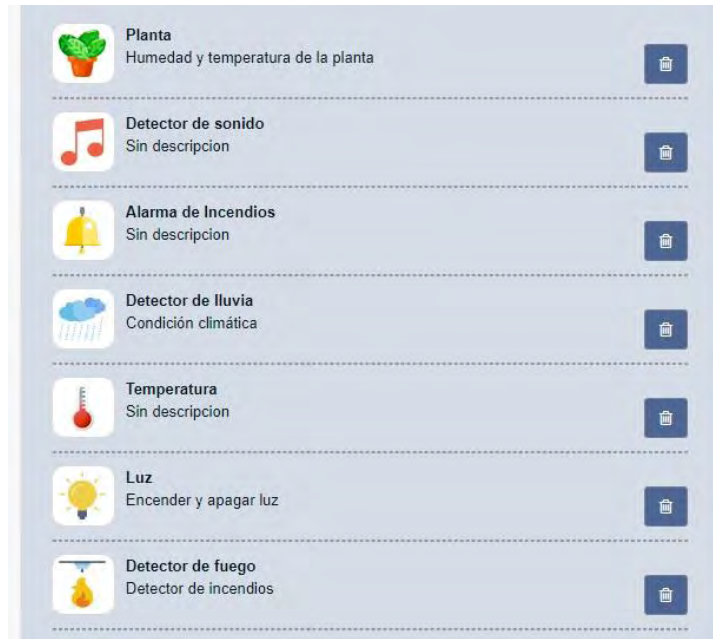


Figura 4.12 Lista de dispositivos agregados 3

4.6 PROYECTOS

La segunda sección del menú principal corresponde a “Proyectos”. En esta sección el usuario podrá acceder a la lista de proyectos creados y le será posible configurarlos o eliminarlos.



Figura 4.13 Proyectos creados

Ahora bien, en caso de que el usuario deseara crear más proyectos, podrá hacer clic en “agregar” y entonces el sistema le permitirá crear un nuevo proyecto, para ello, se repiten los mismos cuatro pasos anteriormente especificados. Si, por ejemplo, se creara otro proyecto llamado “Mi apartamento”, la interfaz se vería de la siguiente manera. El usuario podrá crear tantos proyectos como desee.



Figura 4.14 Proyectos creados 2

Por otro lado, el usuario tiene la posibilidad de eliminar los proyectos creados de acuerdo a sus preferencias. Para ello, debe hacer clic en el icono de borrar, el cual está representado por una papelera o bote de basura. Al hacer seleccionar este icono, el sistema le pedirá que confirme si realmente desea borrar el proyecto seleccionado.



Figura 4.15 Interfaz para borrar proyecto

Adicionalmente, el usuario también cuenta con un botón de configuración asociado al proyecto, el cual le permitirá ejecutar un conjunto de acciones que le permitirán establecer una configuración para su entorno de automatización.

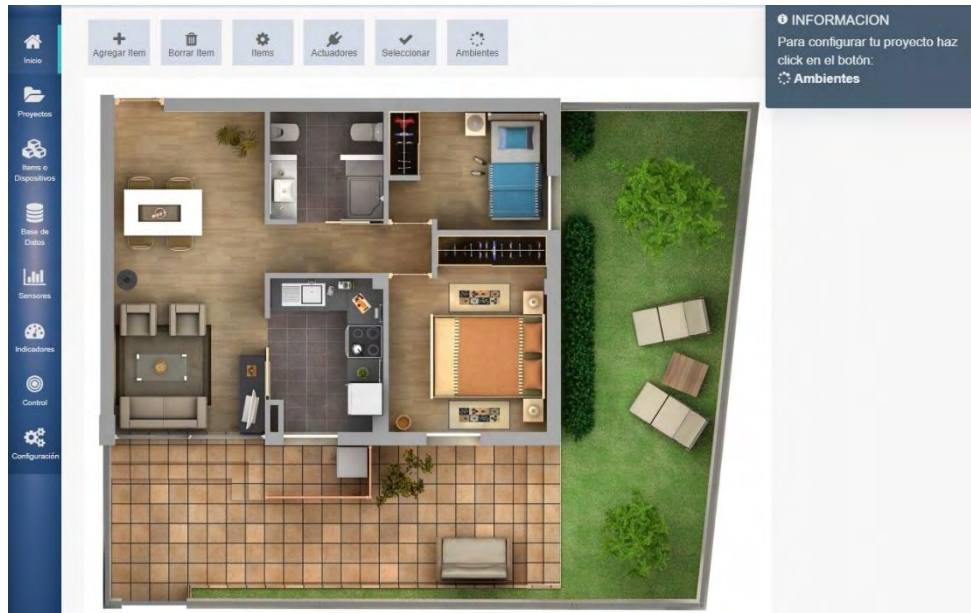


Figura 4.16 Interfaz para configuración de proyectos.

Obsérvese que en la parte superior de la pantalla aparece un menú horizontal de color gris con diferentes botones destinados a la personalización y/o configuración del proyecto seleccionado. A lo largo del presente capítulo se describirá la función de cada uno de ellos.



Figura 4.17 Sub menú para la configuración de proyectos

4.7 AMBIENTES

Si el usuario selecciona en la barra de menú horizontal la función “Ambientes” tal y como el sistema le sugiere, podrá configurar los ambientes asociados al entorno domótico. El sistema le mostrará la lista de ambientes agregados y podrá configurar cada uno de ellos:



Figura 4.18 Configuración de ambientes

Al hacer clic en el icono de configuración, el sistema le pedirá que cargue una imagen que represente el ambiente que ha seleccionado de su entorno doméstico. La imagen que representa esta opción en el sistema se puede visualizar en la siguiente figura.



Figura 4.19 Agregar imagen de ambiente

Una vez que haya cargado la imagen, el ambiente en cuestión estará configurado y podrá hacer lo mismo con el resto de los ambientes. Un ejemplo de ambiente configurado es el siguiente.



Figura 4.20 Ambiente configurado

Cuando el usuario termine de configurar los ambientes de su entorno domótico, al regresar a la pantalla de configuración de proyectos, podrá observar que ahora aparece otro menú vertical ubicado en la parte lateral derecha que contiene el nombre de todos los ambientes configurados.



Figura 4.21 Sección de pantalla de configuración de proyectos

Si, por ejemplo, el usuario decide hacer clic en el ambiente “Habitación”, el sistema le mostrará la representación de dicho ambiente, tal y como se observa en la siguiente imagen.

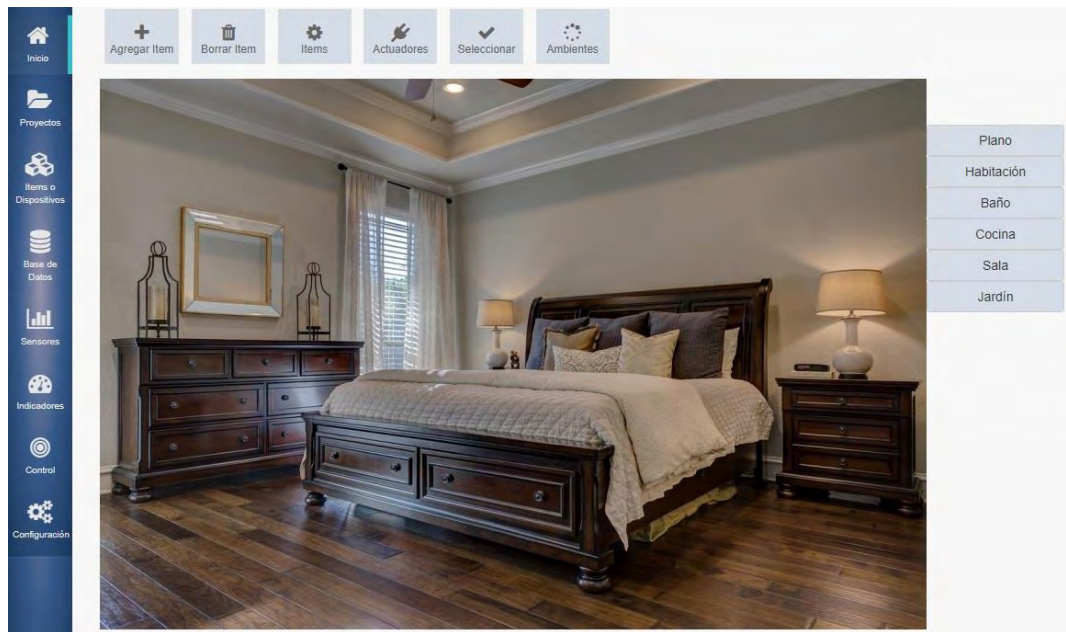


Figura 4.22 Pantalla de configuración de proyectos para ambiente habitación

Por otro lado, si el usuario hace clic en el ambiente “baño”, aparecerá entonces la representación de dicho ambiente dada por la imagen que el usuario subió a la hora de configurar el ambiente en cuestión.

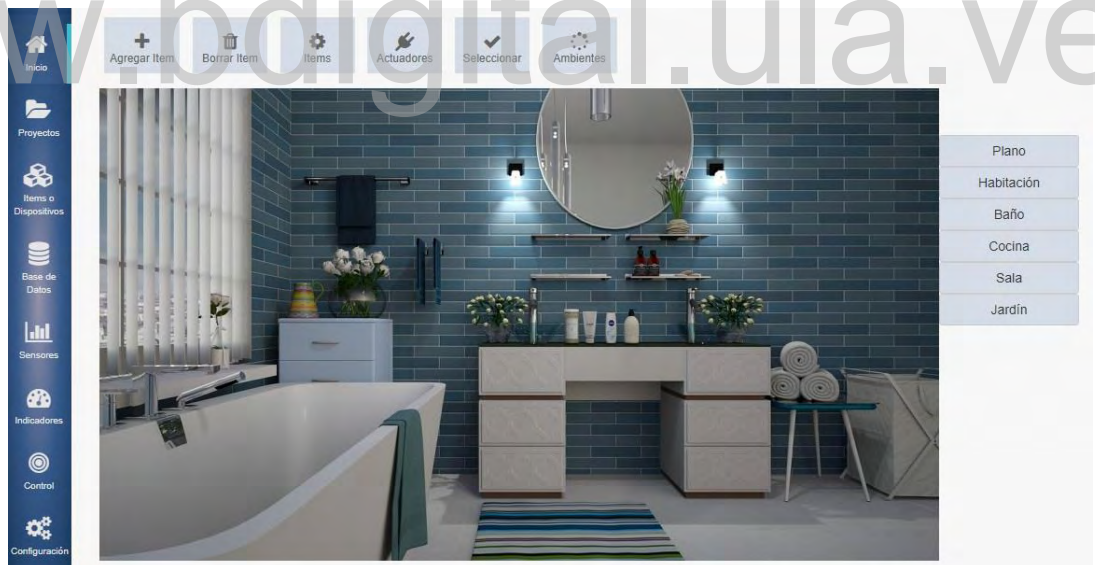


Figura 4.23 Pantalla de configuración de proyectos para ambiente baño

4.8 AGREGAR ITEMS

Para asociar un ítem al proyecto que se está configurando, el usuario podrá acceder a esta función a través del menú superior de la pantalla, haciendo clic en el

icono “agregar ítem”. Una vez accionado este botón, el sistema dirigirá al usuario a una ventana que le permitirá seleccionar el dispositivo o ítem que desea asociar al proyecto. La lista de dispositivos a asociar estará comprendida por los dispositivos predeterminados por el sistema más los dispositivos que el usuario haya personalizado previamente. A continuación, se observa la lista de dispositivos de automatización proporcionada por el sistema en el ejemplo en cuestión, el cual consta de los 14 dispositivos predeterminados más los 7 personalizados. El usuario deberá seleccionar el ítem que desea asociar, además de indicar cuál será el ambiente en el que se encontrará ubicado.

Configuración Avanzada de Proyectos Ambientes de automatización

A continuación podrá agregar ítems o dispositivos de automatización al proyecto seleccionado. Por favor complete el formulario.

Seleccione el ítem que desea agregar a su proyecto e indique en que ambiente está ubicado

<input type="radio"/> dht11 Sensor de temperatura y humedad	<input type="radio"/> hc-sr04 Sensor de ultrasonido
<input type="radio"/> hc-sr501 Sensor de movimiento PIR	<input type="radio"/> ky-028 Sensor termistor
<input type="radio"/> ky-038 Sensor de sonido	<input type="radio"/> max6675 Termopar tipo K
<input type="radio"/> fc-37 Sensor de lluvia	<input type="radio"/> sg-90 Servomotor
<input type="radio"/> 28byj-48 Motor a pasos	<input type="radio"/> Relé Relé electromecánico
<input type="radio"/> Led Luz led	<input type="radio"/> Buzzer Alarma buzzer
<input type="radio"/> fc-51 Sensor infrarrojo	<input type="radio"/> ky-018 Sensor fotoresistivo
<input type="radio"/> Detector de fuego Detector de incendios	<input type="radio"/> Luz Encender y apagar luz
<input type="radio"/> Temperatura Sin descripción	<input type="radio"/> Detector de lluvia Condición climática
<input type="radio"/> Alarma de incendios Sin descripción	<input type="radio"/> Detector de sonido Sin descripción
<input type="radio"/> Planta Humedad y temperatura de la planta	

Ambiente:

Figura 4.24 Lista de dispositivos para ser asociados a un proyecto

A efectos de ejemplificar esta función, se seleccionará el dispositivo personalizado anteriormente llamado “luz”, el cual corresponde a un relé electromecánico, y como ambiente, se escogerá que sea ubicado en la habitación del entorno domótico en cuestión. Luego se hará clic en “configurar” y el sistema permitirá que el usuario seleccione la ubicación del dispositivo tanto en el plano principal el entorno domótico, así como en la imagen que representa el ambiente en el que se encuentra ubicado. Para establecer la ubicación del dispositivo seleccionado en el plano principal, arrastre un recuadro que representa la ubicación del ítem en el plano, y una vez posicionado en el lugar seleccionado a criterio del usuario, podrá hacer clic en “configurar ubicación” y la ubicación del dispositivo se habrá efectuado satisfactoriamente. Adicionalmente, el sistema le pedirá al usuario que también ubique el mismo dispositivo dentro de la imagen correspondiente al ambiente en el que se encuentra asociado.



Figura 4.25 Configurar ubicación de dispositivo en el plano principal

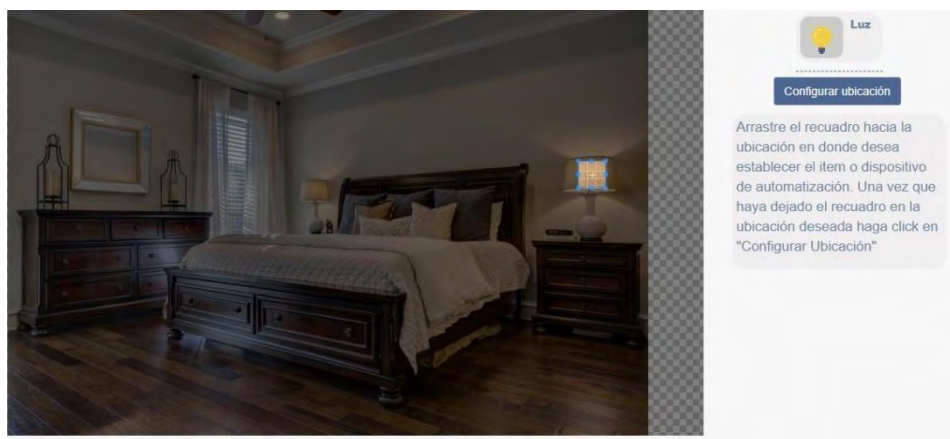


Figura 4.26 Configurar ubicación de dispositivo en el ambiente

Así pues, como en el ejemplo mostrado, el usuario ubicó la luz en la mesita de noche de la habitación, entonces para continuar con el ejemplo, obsérvese la ubicación del ítem en la imagen que representa dicha habitación. Se arrastró el recuadro que representa la ubicación del icono “luz” en la mesita de noche. Habiendo configurado exitosamente la ubicación del dispositivo, se puede observar en la figura 4.30 que, al regresar a la pantalla de configuración de proyectos, aparece el dispositivo en la ubicación en la que fue establecido en el plano principal. Y en 4.31 al seleccionar en el menú lateral derecho el ambiente “habitación”, el cual corresponde el ítem previamente configurado, entonces se observará también la ubicación del dispositivo en la imagen.



Figura 4.27 Pantalla de configuración de proyecto con un ítem asociado

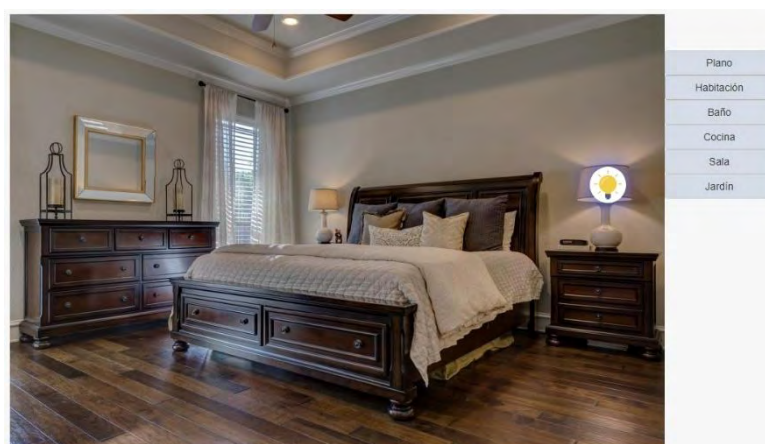


Figura 4.28 Pantalla de configuración de proyecto con un ítem asociado en entorno

De esta manera, la plataforma le permitirá al usuario agregar tantos dispositivos como disponibilidad de hardware haya. A continuación, se observará la pantalla de configuración con varios dispositivos de automatización asociados.



Figura 4.29 Pantalla de configuración de proyecto con varios dispositivos asociados



Figura 4.30 Pantalla de configuración de proyecto con varios dispositivos asociados en habitación

4.9 CONFIGURACIÓN

Para acceder a la configuración del microcontrolador al sistema, el usuario debería dirigirse a la opción “configuración” en el menú principal. En esa pantalla debe observarse la ventana mostrada a continuación. Si el usuario desea añadir un microcontrolador a su cuenta y posteriormente vincularlo a un proyecto, deberá hacer clic en “agregar”.



Figura 4.31 Pantalla para añadir microcontrolador al sistema

A continuación, el sistema le indicará al usuario los pasos a seguir para poder establecer una comunicación entre el servidor y el microcontrolador a añadir. Esta configuración consta de tres pasos, los cuales corresponden a configurar el microcontrolador para que se conecte a la red domótica, seguidamente identificar dicho dispositivo con el código ChipID única establecido por el fabricante y que el usuario podrá consultar cuando visualice el hardware, y en el último paso, el usuario podrá finalizar este proceso para permitir añadir otros microcontroladores al sistema de control domótico.

Una vez agregado, el microcontrolador debería aparecer en la lista de hardware asociado. Para efectos del ejemplo en cuestión, se ha añadido otro microcontrolador a la cuenta como se muestra en la Figura 4.32. Obsérvese que ninguno de ellos ha sido vinculado a un proyecto todavía.



Figura 4.32 Hardware asociado al sistema

Seguidamente, si el usuario quisiera vincular uno o ambos microcontroladores a un proyecto, entonces deberá hacer clic en el icono de configuración ubicado a la derecha de cada microcontrolador. Al seleccionar el proyecto al cual desea vincular el microcontrolador, el usuario debe hacer clic en “aceptar” y el hardware se encontrará vinculado al proyecto seleccionado. Ahora la lista de hardware asociado indicará esta vinculación



Figura 4.33 Asociación de ESP8266 a un proyecto

4.10 CONFIGURACIÓN DE ITEMS

En el menú superior horizontal de la pantalla de configuración de proyectos se encuentra el botón llamado “ítems”. Al accionar este botón el sistema le permite al usuario configurar los ítems asociados al proyecto. Esta configuración podrá efectuarse una vez que el usuario haya configurado los microcontroladores y

posteriormente los haya asociado al proyecto. Para efectuar esta configuración previa, el usuario debería dirigirse a la opción “configuración” en el menú principal, la cual fue expuesta anteriormente. Ahora bien, suponiendo que el usuario haya configurado satisfactoriamente el microcontrolador, existen dos formas por las cuales puede acceder a esta configuración: a través el menú horizontal de la pantalla de configuración antes mencionado, o en la pantalla principal al hacer clic en un dispositivo no configurado.

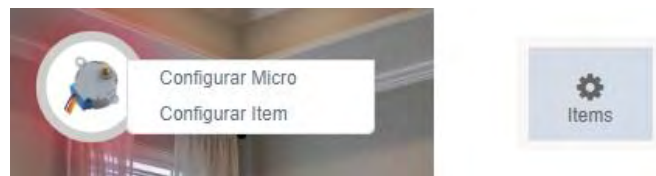


Figura 4.34 Métodos para acceder a la configuración del ítem

Si el usuario se dirige a la pantalla de inicio, observará los dispositivos asociados. Al posicionar el cursor sobre cada uno de ellos, el icono se hará más grande e indicará si ha sido configurado o no.



Figura 4.35 Pantalla de configuración de proyecto con varios dispositivos asociados en sala de estar

Al hacer clic sobre el icono que indica que aún no ha sido configurado, se despliega un sub menú al lado del ícono que nos permite acceder a la configuración del mismo o a la del microcontrolador.



Figura 4.36 Sub menú para ítem no configurado

Cuando el usuario haya accedido a la pantalla de configuración de ítems, podrá observar la lista de ítems asociados y un formulario de selección en el que tendrá la oportunidad de decidir a qué hardware vinculará el ítem o dispositivo de automatización.

Inicialmente, todos los dispositivos que no hayan sido configurados aparecen sin configuración. Cuando el usuario haga clic en cualquiera de estos campos, se desplegarán las diferentes opciones disponibles en hardware asociado a los microcontroladores vinculados al proyecto en cuestión. Si, por ejemplo, para el dispositivo a configurar existen uno o más dispositivos en hardware disponibles, entonces el usuario tendrá la opción de decidir cual utilizar. En la imagen que se muestra a continuación, se observa una lista de todos los dispositivos que fueron vinculados al proyecto. Si hay suficiente disponibilidad en hardware, el usuario podrá configurarlos todos, pero si por el contrario hay más ítems que dispositivos en hardware, entonces el usuario no podrá configurarlos todos.

En un caso como este, el usuario tendrá que agregar un nuevo microcontrolador que proporcione más disponibilidad en hardware para ser utilizado, una vez agregado y vinculado al proyecto, el usuario podrá configurar los dispositivos faltantes.

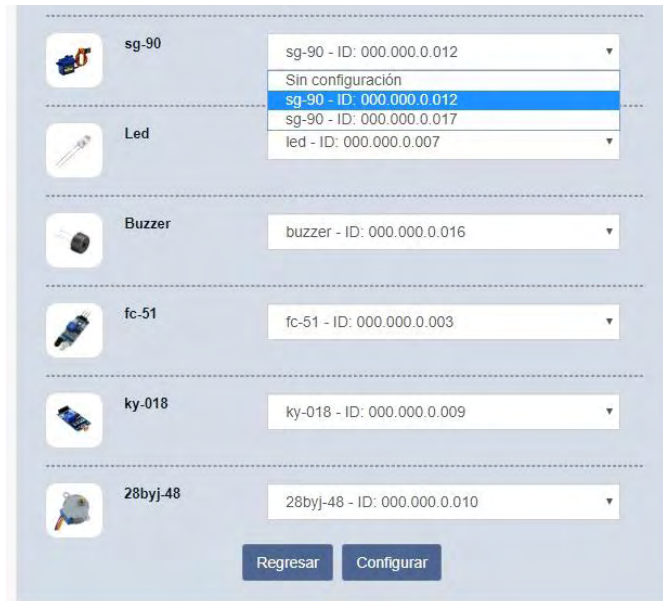


Figura 4.37 Selección de hardware para dispositivos asociados

Si los dispositivos no son configurados, entonces no podrán ser accionados o establecer un sensado, según corresponda.



Figura 4.38 Dispositivos configurados y no configurados

4.11 ELIMINAR ITEM ASOCIADO

Si el usuario desea eliminar un ítem que haya sido asociado a un proyecto, independientemente de si éste ha sido configurado o no, podrá hacerlo al acceder a la pantalla de configuración de proyectos y hacer clic en el menú horizontal superior

en la opción “borrar ítem”. El usuario podrá eliminar el dispositivo que desee haciendo clic en el icono representado por un cesto de basura. Debe tomar en cuenta que, lo que se elimina en este caso es la asociación del ítem al proyecto en cuestión. El ítem permanecerá en la sección de ítems para ser utilizado o asociado en cualquier proyecto. Así que, si deseara volver a utilizarlo, puede asociarlo nuevamente.



Figura 4.39 Eliminar ítems asociados a un proyecto

4.12 SENSORES

Al acceder a la sección de “sensores” en el menú principal ubicado en el lateral izquierdo de la pantalla, el usuario podrá observar diferentes gráficas que corresponden a las lecturas de los diferentes sensores que haya configurado. Si un dispositivo de entrada o sensor no ha sido configurado, entonces no aparecerá su gráfica en esta pantalla. Esta parte de la interfaz resulta de gran utilidad para el

usuario ya que le permitirá monitorizar los diferentes sensores que operan en el entorno domótico seleccionado.

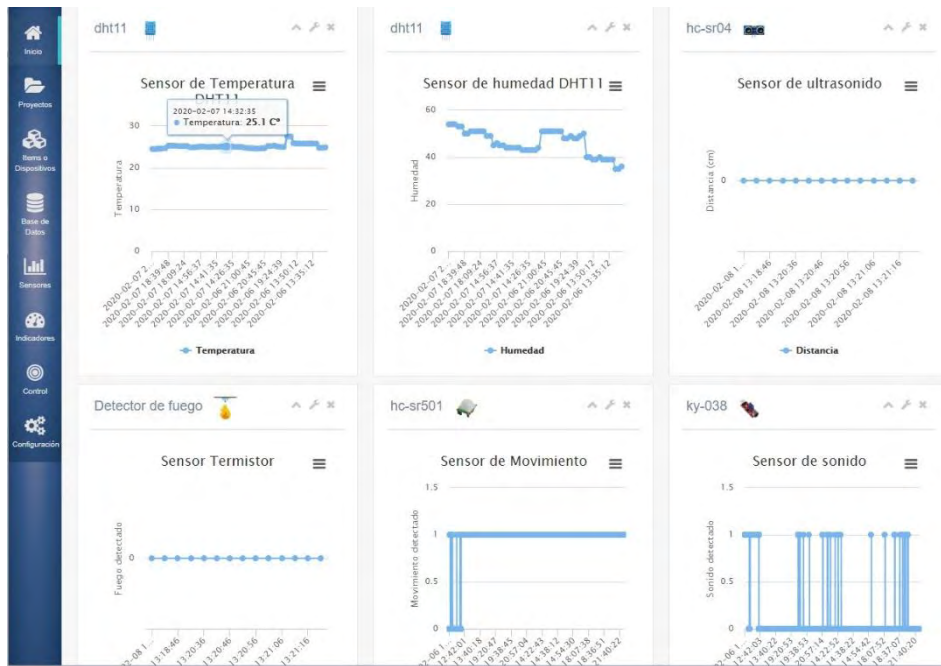


Figura 4.40 Gráfica de sensores

www.bdigital.ula.ve

4.13 BASE DE DATOS

En el menú principal el usuario conseguirá la opción de “Base de datos”, la cual corresponde a una sección de consulta de los valores arrojados por los sensores configurados para los diferentes proyectos de automatización creados.



Figura 4.41 Consultar base de datos del sistema

Como se visualizó en la figura anterior, el usuario podrá consultar la base de datos correspondiente a cada uno de sus proyectos. Así pues, deberá seleccionar el proyecto cuyos datos desea consultar. Una vez seleccionado, se muestra una lista de dispositivos de entrada o sensores disponibles o asociados al proyecto seleccionado tal y como se observa en la Figura 4.42.



Figura 4.42 Consulta en la base de datos de un proyecto seleccionado

En caso de que el usuario tenga todos los sensores configurados, podrá acceder a sus respectivos datos. Si, por el contrario, hubiese algún dispositivo que todavía no fue configurado, entonces no podrá acceder a la información. Si el usuario seleccionase, por ejemplo, el sensor DHT11, entonces podrá visualizar los datos proporcionados por el mismo, como se aprecia en la Figura 4.43.

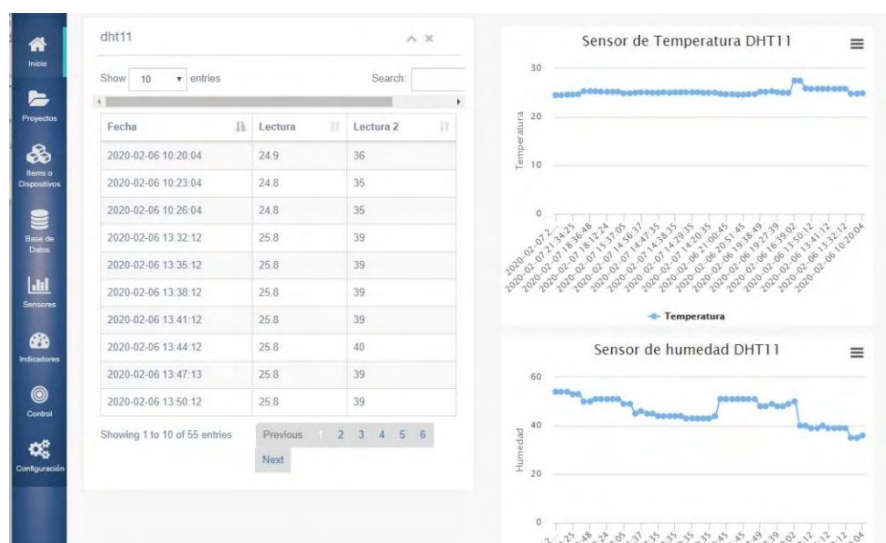


Figura 4.43 Pantalla de consulta de base de datos para sensor dht11

4.14 CONTROL

Si el usuario ha vinculado dispositivos de salida o actuadores a su proyecto, entonces al dirigirse a la sección “Control” del menú principal, podrá encontrar una interfaz de control de dispositivos como se muestra a continuación. Si los ítems aún no han sido configurados, el sistema restringirá que el usuario pueda manipularlos o accionarlos.

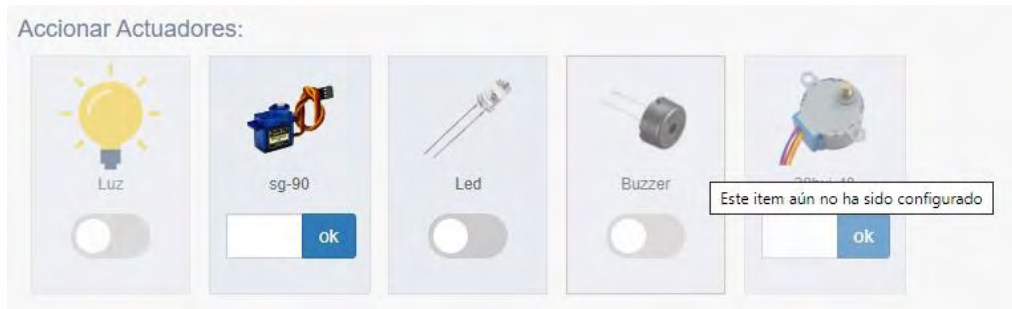


Figura 4.44 Interfaz para accionar actuadores

Una vez que el usuario configure todos sus ítems de salida o actuadores, entonces podrá acceder a ellos y accionarlos en cualquier momento de acuerdo a sus necesidades. Para accionar un dispositivo de encendido y apagado, como lo puede ser un relé, un led o un *buzzer*, el usuario deberá accionar el botón correspondiente a dicho dispositivo y al hacerlo, se accionará simultáneamente el dispositivo en hardware asociado. Para el caso de dispositivos como los servos motores o el motor a pasos, el actuador debe recibir información acerca de cuántos grados debe girar el mismo. Por ejemplo, el servo motor aceptará una lectura que esté comprendida entre los 0° y 180°, mientras que el motor a pasos podrá ser accionado desde los 0° hasta los 360°, según lo decida el usuario.

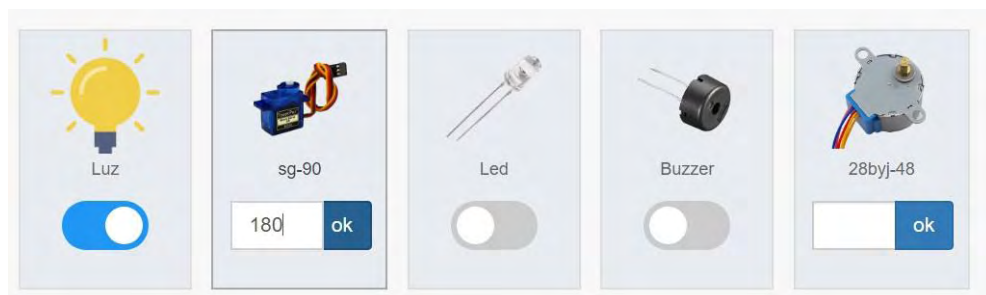


Figura 4.45 Ejecutar dispositivos actuadores.

Otra forma de accionar los dispositivos puede ser desde la pantalla de inicio o pantalla principal de la plataforma. Si el dispositivo está configurado y se trata de un actuado de sólo encendido y apagado, entonces simplemente haciendo clic sobre él, el usuario conseguirá accionarlo. En las siguientes imágenes se muestra como este tipo de dispositivos puede ser accionado. El aro color azul indica que se encuentra encendido, mientras que el aro color gris indica que se encuentra apagado.

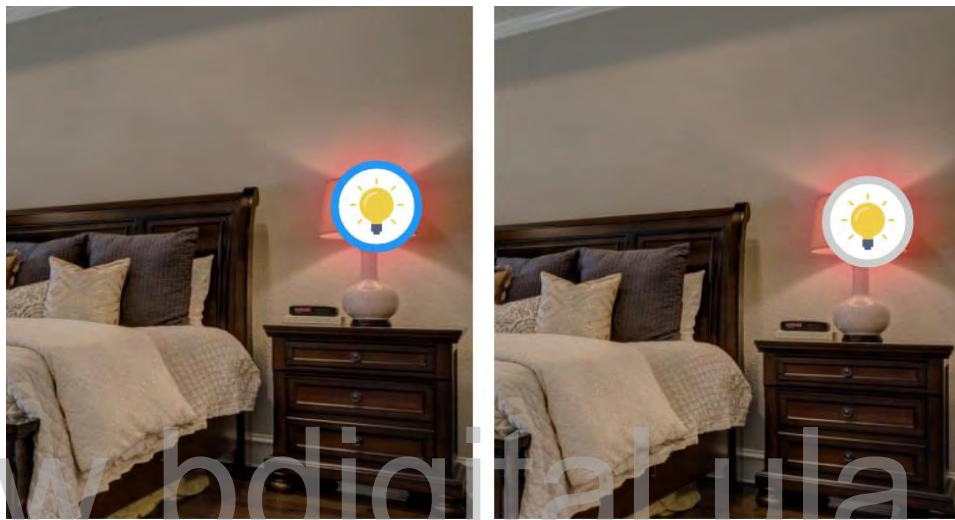


Figura 4.46 Control de actuador desde pantalla principal

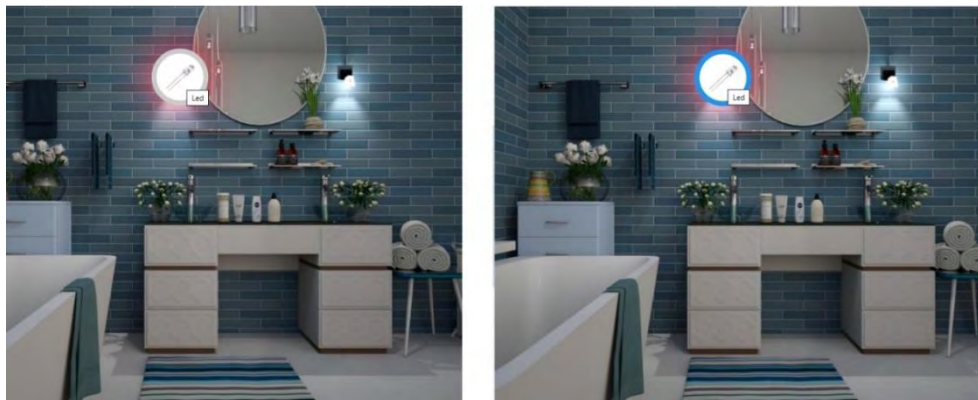


Figura 4.47 Control de actuador desde pantalla principal 2

Para los dispositivos de salida como los servos motores y el motor a pasos, debido a que su accionamiento no está basado en un encendido y apagado, al hacer clic sobre el dispositivo el sistema le dará la opción de controlar el dispositivo.

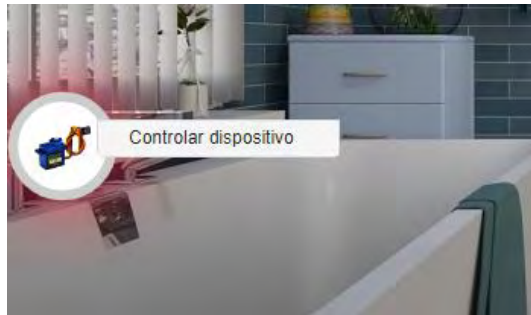


Figura 4.48 Accionar sg-90 o 28by-48 desde pantalla principal

4.15 PROGRAMAR ACTUADORES

Al acceder a la sección “Control” del menú principal, el usuario podrá encontrar una interfaz de control de dispositivos, así como la opción de programar actuadores. En este punto, el usuario tiene la opción de decidir bajo qué condiciones pueden ser accionados los actuadores sin que él debe ejecutar la acción por sí mismo. Al hacer clic en la opción “configurar”, el sistema desplegará una ventana que permitirá al usuario programar el dispositivo de salida que seleccione y la condición bajo la cual se accionará.



Figura 4.49 Accionar dispositivos de salida

Existen dos tipos de condiciones para las cuales se pueden programar los actuadores: debido a la lectura de un sensor o en un horario establecido. Si el usuario elige la condición “de acuerdo a la lectura de un sensor”, entonces el sistema habilitará otro campo en el formulario que le permitirá seleccionar el sensor que determinará la acción del actuador.

Como se observa en la imagen a continuación, el nuevo campo desplegado mostrará una lista de los sensores configurados para el proyecto en cuestión. Por otro lado, si el usuario decide que desea que un dispositivo de salida se accione todos los días a la misma hora de acuerdo a sus necesidades específicas, entonces podrá programarlo seleccionando la condición “de acuerdo a un horario” y cuando el sistema habilite un campo podrá especificar la hora a la cual desea que el actuador seleccionado sea accionado.



Figura 4.50 Transición de ventanas al seleccionar condición

Una vez que se ha programado un, esta programación se muestra en la interfaz. Adicionalmente, el usuario podrá borrar o eliminar la programación efectuada en cualquier momento.

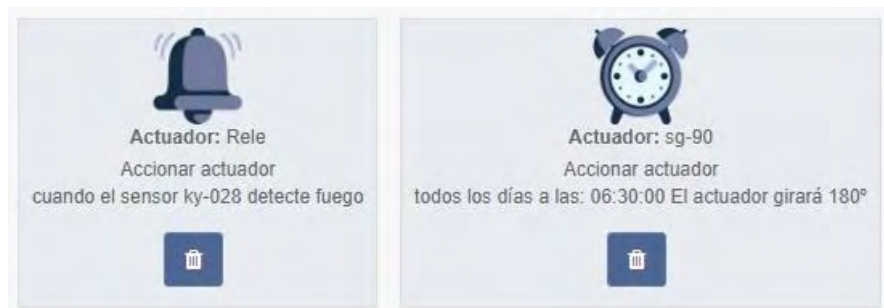


Figura 4.51 Programación para accionar dispositivos de salida guardada

CONCLUSIONES

Este trabajo de grado se enfocó en el diseño e implementación de un sistema de control domótico basado en un software de código abierto el cual permitió la integración de diferentes tecnologías de dispositivos de automatización que empleaban diferentes protocolos de comunicación, dando como resultado una plataforma capaz de controlar y monitorizar ambientes domóticos. Para concluir, las aportaciones más considerables de este proyecto son:

- El diseño de un sistema de control domótico que incorpora mecanismos y dispositivos de automatización capaces de comunicarse entre ellos y con los usuarios a través de una interfaz que coordina y controla todos los dispositivos electrónicos.
- La creación de una interfaz de usuario amigable e intuitiva, la cual se basó en PHP, MySQL, JavaScript, HTML5 y CSS3, permitiendo que el usuario goce de una interacción flexible y haga uso de la interfaz sin poseer conocimientos de ingeniería.
- La integración de diferentes elementos que conforman un sistema domótico y que emplean diferentes protocolos de comunicación, estableciéndose así los mecanismos de comunicación necesarios entre la central domótica, los sensores y actuadores.
- La creación de un sistema de control cuya adquisición de datos opera de forma inalámbrica a través del protocolo de comunicación TCP/IP.

Finalmente, el sistema de control domótico diseñado e implementado se basó en una plataforma de código abierto cuya implementación en hardware es accesible y de bajo costo, ofreciendo soluciones en automatismo, confort, reducción del trabajo doméstico, aumento de la seguridad, así como la racionalización de los diferentes consumos y la optimización de los recursos.

RECOMENDACIONES

En base a los resultados obtenidos en la presente investigación, para efectos de asegurar una mejora en términos de implementación del sistema de control domótico desarrollado, se recomienda lo siguiente:

- En lugar de implementar la plataforma en un servidor local, se recomienda efectuar la implementación un servidor remoto que provea un alojamiento web o hosting para hospedar la plataforma web; y proporcione a su vez, un servidor de nombres de dominio (DNS) que permita conectar el nombre de dominio del sistema domótico con una dirección de IP específica. De esta manera, se garantiza que el servidor esté establecido en una dirección IP estática, tenga siempre acceso a internet, y el usuario pueda conectarse al servidor desde cualquier parte del mundo.
- Al favorecer los recursos de la plataforma implementándola en un servidor remoto, se recomienda minimizar los riesgos en seguridad informática, implementando protocolos, estándares, herramientas y reglas concebidas para resguardar la integridad del software, la base de datos y la infraestructura del sistema de control.
- Se recomienda adquirir un certificado de seguridad SSL (capa de puertos seguros) en su versión más actual de seguridad en capas de transporte (TLS) cuya tecnología estandarizada permita cifrar el tráfico de datos entre el servidor web y el usuario, permitiendo así la confidencialidad del dato/mensaje y de los códigos de autenticación empleados en el servidor domótico, garantizando de esta manera la integridad, confidencialidad y autenticidad de los datos.
- Agregar más funcionalidades a la plataforma para favorecer aún más la experiencia del usuario.

REFERENCIAS

- [1] M. Jiménez, *La casa Inteligente: un lugar amable para vivir*. Madrid: U-Gob, 2019. p. 41.
- [2] B. Sniderman, G. Gorman, J. Holdowsky, J. Mariani y B. Dalton, *El diseño de las cosas: construyendo en la conectividad del IoT*. Deloitte University Express, 2016.
- [3] M. Raynor y M. Cotteleer, *Cuanto más cambian las cosas: la creación de valor, la captura de valor, y la Internet de las cosas*. Deloitte Review, 2015 pp. 51-52.
- [4] K. Rose, S. Eldridge y L. Chapin. *La internet de las cosas – Una breve reseña. Para entender mejor los problemas y desafíos de un mundo más conectado*. Internet Society (ISOC), 2015.
- [5] J. Huidobro, *La domótica como solución de futuro. Concepto, campo de aplicación y beneficios*. Madrid: fundación de la energía de la comunidad de Madrid, 2007. pp. 15-16.
- [6] Nayar Systems, *IoT & Elevators: la divulgación del conocimiento*. Castellón: Nayar Systems, 2019. pp. 26-58.
- [7] J. Martín, *Instalaciones Domóticas. Electricidad y electrónica*. Madrid: Editorial Editex S. A, 2010. pp. 11-80.
- [8] H. Martín y F. Sáez, *Domótica: Un enfoque socio técnico. Fundación Rogelio Segovia para el Desarrollo de las Telecomunicaciones*. Madrid, 2006. pp. 50-110.
- [9] L. Molina, *Instalaciones Domóticas*. Madrid: Editorial Mcgraw Hill, 2012.
- [10] Fundación de la Innovación Bankinter, *El Internet de las Cosas en un mundo conectado de objetos inteligentes*. Madrid: Fundación de la Innovación Bankinter, 2011. p. 66.
- [11] L. Redolfi, *Domótica*. 1a edición. Ciudad Autónoma de Buenos Aires: Fox Andina, 2013. pp. 52-69.
- [12] M. Horro, “Programa en Java de un entorno de control para un sistema domótico”. Trabajo de Grado en Ingeniería Telemática, Universidad Carlos III de Madrid,

Madrid, 2008. [En línea] Disponible en: https://e-archivo.uc3m.es/bitstream/handle/10016/7011/PFC_Carlos_Horro_Marcos.pdf?sequence=1&isAllowed=y

[13] A. Durán, “Instalación Domótica de una vivienda unifamiliar”. Trabajo de grado de Ingeniería Industrial, Universidad Pontificia Comillas, Madrid, 2009. [En línea] Disponible en: <https://www.iit.comillas.edu/pfc/resumenes/4a4601f904610.pdf>

[14] J. Nuñez, “Diseño e implementación de un entorno de control para un sistema domótico”. Trabajo de Grado en Ingeniería eléctrica, Universidad de Los Andes, Facultad de Ingeniería, Mérida, 2014. [En línea] Disponible en: https://mega.nz/folder/E1kUGQzJ#rr_t1d4nzWL2Du82MceWXg

[15] J. Quinde, J. Israel, B. Zhingre y P. Buele, “Diseño de software para el control y monitoreo domótico en una minicomputadora basado en protocolo TCP/IP”. Trabajo de grado en Ingeniería electrónica, Universidad Politécnica Salesiana, Cuenca, 2015. [En línea] Disponible en: <http://dspace.ups.edu.ec/handle/123456789/7601>

[16] V. Sol, “*Sistema para la monitorización y asistencia en el hogar usando openhab y robótica móvil*”. Trabajo de grado en Ingeniería Informática. Universidad de Málaga, Málaga, 2016. [En línea]. Disponible en: <https://riuma.uma.es/xmlui/bitstream/handle/10630/12985/Victor%20del%20Sol%20ToledanoMemoria.pdf?sequence=1&isAllowed=y>

[17] A. García, “Plataforma domótica basada en Raspberry Pi y el protocolo MQTT”. Trabajo de grado en Ingeniería de tecnologías de comunicación, Universidad de Granada, Granada, 2017. [En línea]. Disponible en: http://wpd.ugr.es/~jorgenavarro/thesis/2017_TFG_AlejandroGarciaSoria.pdf

[18] P. Serna, “Vivienda inteligente: proyecto domótico de bajo coste para una vivienda unifamiliar”. Trabajo de grado en Ingeniería en tecnologías Industriales, Escuela Técnica Superior de Ingenieros Industriales y de Telecomunicación, Universidad de Cantabria, Cantabria, 2018. [En línea]. Disponible en: <https://repositorio.unican.es/xmlui/bitstream/handle/10902/13034/405612.pdf?sequence=1&isAllowed=y>

- [19] L. Velasco, “*Interfaz de Usuario en dispositivos Domóticos*”. Trabajo de Grado en Ingeniería eléctrica, Universidad de Los Andes, Facultad de Ingeniería, Mérida, 2019.
- [20] CommScope Inc, *Conectividad para Edificios Inteligentes: dando forma a los negocios siempre conectados del futuro*. Carolina del Norte: NASDAQ: COMM, 2018. pp. 5-47.
- [21] J. Briceño, *Transmisión de datos*. 3ra ed. Mérida: Universidad de Los Andes, Facultad de Ingeniería, Departamento de publicaciones, 2005. pp. 14-380.
- [22] J. Barceló, J. Íñigo, R. Martí, E. Peig y X. Perramon, *Redes de Computadoras*. Barcelona: Fundació per a la Universitat Oberta de Catalunya, 2004. pp. 39-488.
- [23] S. Fernández, O. Entrecanales y R. Piñeiro. *Sistemas Telemáticos*. Madrid: McGraw-Hill Interamericana de España S.L, 2005. pp. 93 – 102.
- [24] W. Stallings, *Comunicaciones y Redes de Computadores*. 7ma ed. Madrid: Pearson Educación, S.A, 2004. pp. 14-495.
- [25] División de Sistemas de la Universidad Católica de Los Ángeles, *Metodología de desarrollo de Software versión 001*. Chimbote: Rectorado, 2017. pp. 5-39.
- [26] A. Oppel y R. Sheldon, *Fundamentos de SQL*. 3a edición. Ciudad de México: McGRAW-HILL Interamericana Editores, S.A, 2009. pp. 4-15.
- [27] J. Gil, *Guía del curso “PHP+SQL”*. Valladolid: Grupo Universitario de Informática (GUI) de la Universidad de Valladolid, 2003. pp. 2-20.
- [28] I. Gilfillan, *La biblia de mySQL*. Edición Ilustrada. Madrid: Anaya Multimedia S.A, 2003. pp. 40-41.
- [29] J. Luján, *HTML5, CSS Y JAVASCRIPT. Crea tu web y apps con el estándar de desarrollo*. Madrid: RC Libros, 2016. pp. 2-3.
- [30] J. Gauchat, *El gran libro de HTML5, CSS3 y Javascript*. 1era edición. Barcelona: Marcombo S.A, 2012. pp. 1-87.
- [31] J. Collell, *CSS3 y Javascript avanzado*. Catalunya: UOF Universitat oberta de Catalunya, 2013. pp. 7-26.

- [32] A. Cobo, P. Gómez, D. Pérez y R. Rocha, *PHP y MySQL Tecnologías para el desarrollo de aplicaciones web*. España: Ediciones Díaz de Santos, 2005. p. 23.
- [33] J. Cegarra, *Metodología de la Investigación Científica y Tecnológica*. Madrid: Ediciones Díaz de Santos, 2011. p. 42.
- [34] G. Fidias, *El proyecto de Investigación*. 3a Edición. Caracas: Editorial Episteme S.A /Oriol Ediciones, 2012. p. 67.
- [35] H. Cerda, *Los elementos de la Investigación*. Bogotá: Magisterio, 2011. p.330.
- [36] Punto Flotante S.A, *HC-SR501 PIR Sensor infrarrojo de movimiento Datasheet*, 2017 [En línea] Disponible en: <https://puntoflotante.net/Manual-DEL-USUARIO-SENSOR-DE-MOVIMIENTO-PIR-HC-SR501.pdf>
- [37] Mouser Electronics. *Ultrasonic Ranging Module HC - SR04 Datasheet*, 2019. [en línea] Disponible en: <https://www.mouser.com/datasheet/2/813/HCSR04-1022824.pdf>
- [38] Integrated Products, *MAX6675 Cold-Junction-Compensated K-Thermocouple-to-Digital Converter (0°C to +1024°C) Datasheet*, 2014. [En línea] Disponible en: <https://datasheets.maximintegrated.com/en/ds/MAX6675.pdf>
- [39] Robosoft Systems, *IR FC-51 Sensor – Single Datasheet*, 2019. [En línea] Disponible en: http://www.dmf.unisalento.it/~denunzio/allow_listing/ARDUINO/FC51.pdf
- [40] Velleman, *Sensor de sonido KY-038 Datasheet*, 2017. [En línea] Disponible en: <https://www.robotshop.com/media/files/pdf/sound-sensor-module-arduinoatasheet.pdf>
- [41] Vaisala, *Sensor de lluvia FC-37 Datasheet*, 2015. [En línea] Disponible en: https://urolakostapk.files.wordpress.com/2016/10/y1-83-rain-detector-datasheet_low.pdf
- [42] Joy-IT, *KY-028 Temperature Sensor module (Thermistor) Datasheet*, 2017. [En línea] Disponible en: <https://datasheetspdf.com/parts/KY-028.pdf?id=1402039>
- [43] Mouser Electronics, *DHT11 Humidity & Temperature Sensor Datasheet*, 2019. [En línea] Disponible en: <https://www.mouser.com>

com/ datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-143054.pdf

- [44] Joy-IT, *KY-018 Photoresistor module Datasheet*, 2017. [En línea] Disponible en: <https://datasheetspdf.com/parts/KY018.pdf?id=1402029MOVIMIENTO-PIR-HC-SR501.pdf>
- [45] Tower Pro, *SG90 9 g Micro Servo Datasheet*, 2019. [En línea] Disponible en: <https://datasheet4u.com/datasheet-parts/SG90-atasheet.php?id=791970>
- [46] Kiatronics, *28BYJ-48 – Stepper Motor Datasheet*, 2019. [En línea] Disponible en: <http://robocraft.ru/files/datasheet/28BYJ-48.pdf>
- [47] FEC Future Electronic Corporation, *Relay Modules Datasheet*, 2019. [En línea] Disponible en: https://www.fecegypt.com/uploads/dataSheet/1522335719_relay%20module.pdf
- [48] STMicroelectronics, *ULN2001A-ULN2003A Datasheet*, 2019 [En línea] Disponible en: <https://www.st.com/resource/en/datasheet/uln2001.pdf>
- [49] Espressif Systems, *Microcontroller ESP8266EX Datasheet Version 4.3*, 2019. [En línea]. Disponible en: https://www.espressif.com/sites/default/files/documentation/0a-ESP8266ex_datasheet_en.pdf