

UNIVERSIDAD MAYOR DE SAN SIMÓN
FACULTAD DE CIENCIAS Y TECNOLOGÍA
CARRERA DE INGENIERÍA DE SISTEMAS

Sistema de Control Domotico de Vivienda Unifamiliar.

Alfredo Solis Quiroga

Octubre, 2017



UNIVERSIDAD MAYOR DE SAN SIMÓN
FACULTAD DE CIENCIAS Y TECNOLOGÍA
CARRERA DE INGENIERÍA DE SISTEMAS



Sistema de Control Domotico de Vivienda Unifamiliar.

Alfredo Solis Quiroga

Proyecto de Grado, para Optar al Diploma Académico en la Carrera de Licenciatura en
Ingeniería de Sistemas.

Profesor guía: Lic. Carmen Rosa Garcia Perez

Profesor Co-guía: Lic. Carmen Rosa Garcia Perez

Octubre, 2017

Dedicatoria

A mi madre y mi hermana con mucho amor y cariño le dedico todo mi esfuerzo y trabajo puesto para la culminación de este proyecto de grado. A mi familia, quienes a lo largo de mi vida han velado por mi bienestar y educación siendo mi apoyo en todo momento. Depositando su entera confianza en cada reto que se me presentaba sin dudar ni un solo momento en mi inteligencia y capacidad. Es por ello que soy lo que soy ahora.

Agradecimientos

Doy gracias primordialmente a Dios por darme la inteligencia, sabiduría, paciencia, entendimiento y la capacidad para ejercer este proyecto. A mis padres por todo su apoyo, comprensión y confianza. A mi tutora Lic. García Pérez Carmen Rosa por guiarme en todo momento hasta la culminación del proyecto. A mis compañeros de trabajo por el compromiso y empeño que le pusieron cada uno de nosotros para sacar adelante y ejercer este proyecto. A mis docentes a quienes les debo gran parte de mis conocimientos, gracias a su paciencia y enseñanza. y finalmente un eterno agradecimiento a esta prestigiosa universidad la cual abrió sus puertas a jóvenes, preparando para un futuro competitivo y formando como personas de bien.

Resumen

Agregar un texto resumen de su trabajo de titulación, este debe ser de entre 100 y 200 palabras.
Sugerencia: El resumen del trabajo de titulación debe ser lo último que se escriba.

Palabras claves— Palabra clave 1, Palabra clave 2, Palabra clave 3, Palabra clave 4, Palabra clave 5.

Índice general

1	INTRODUCCIÓN	1
1.1	Antecedentes	1
1.2	Análisis del problema	1
1.2.1	Árbol de problema	2
1.2.2	Descripción del Problema	2
1.3	Propuesta de solución	3
1.3.1	Árbol de medios y fines	3
1.4	Objetivo general	3
1.5	Objetivos específicos	4
1.6	Justificación	4
1.7	Alcance	4
2	DOMÓTICA	5
2.1	Introducción	5
2.2	Aspectos principales	5
2.2.1	En el ámbito del ahorro energético	5
2.2.2	En el ámbito del nivel de confort	6
2.2.3	En el ámbito de la protección patrimonial (seguridad)	6
2.2.4	En el ámbito de las comunicaciones	7
2.3	Estructura de una Instalación Domótica	7
2.3.1	Actuación los Sistemas de Domótica	8
2.4	Arquitectura de un Sistema de Domótica	9
2.5	Medios de Transmisión / Bus	9
2.6	Sistemas Domóticos	11
2.6.1	Los Protocolos de Domótica	12
2.6.2	Tecnologías y estándares	13
2.6.3	Funcionamiento De Un Sistema Domótico	13
3	ARDUINO Y NODEJS	15
3.1	Introducción	15
3.2	Programar un Arduino	15
3.3	NodeJS	16

3.3.1	Gestor De Paquetes	16
3.3.2	Marcos de Desarrollo	17
3.3.3	Componentes para comunicación en tiempo real	17
3.4	Comunicación serial	17
4	VIVIENDA UNIFAMILIAR	20
4.1	Introducción	20
4.1.1	Familia	20
4.1.2	Vivienda y Hogar	22
4.2	Definición	22
4.3	Tipos de viviendas	22
4.4	Integrantes de una Vivienda Unifamiliar	23
4.4.1	Roles de los Integrantes	24
4.5	Recursos de una vivienda	24
5	DESARROLLO DEL TRABAJO	26
5.1	Introducción	26
5.2	Proceso de desarrollo de la metodología	26
5.3	Sensores y Armado del Prototipo	27
5.3.1	Análisis	27
5.3.2	Diseño	30
5.3.3	Codificación	30
5.3.4	Pruebas	33
5.4	Montar servidor NodeJS y Establecer Comunicaciones	34
5.4.1	Análisis	34
5.4.2	Diseño	35
5.4.3	Codificación	35
5.4.4	Pruebas	37
5.5	Control y monitoreo Web	38
5.5.1	Análisis	38
5.5.2	Diseño	39
6	Resultados.	40
6.1	Ejemplo de titulo 2.	40
6.1.1	Ejemplo de titulo 3.	40
6.2	Discusión de resultados.	40
7	Conclusiones	41
	Bibliografía	41
A	Ejemplo de apéndice o anexo	43

A.1	Sub título primer apéndice o anexo	43
A.1.1	Sub sección del primer apéndice o anexo	43

Índice de tablas

Tabla 5.1	Planificación de tareas primer módulo (Elaboración Propia).	28
Tabla 5.2	Planificación de tareas Segundo Módulo (Elaboración Propia).	34
Tabla 5.3	Planificación de tareas tercer módulo (Elaboración Propia).	39

Índice de figuras

Figura 1.1	Árbol de Problemas (Elaboración Propia)	2
Figura 1.2	Árbol de Solucion (Elaboración Propia)	3
Figura 2.1	Dispositivos de sistema domotico (Elaboración Propia)	8
Figura 2.2	Arquitectura domótica(Elaboración Propia)	10
Figura 2.3	Estandares Internacionalesde Domotica. (Elaboración Propia)	14
Figura 3.1	Estructura de un Programa Arduino. (Elaboración Propia)	16
Figura 3.2	Estructura de Directorios Generada con Express. (Elaboración Propia)	18
Figura 4.1	Tipología De Hogares, Encuesta De Hogares 2015. (Instituto Nacional de Estadística)	21
Figura 4.2	Tipos de Vivienda Unifamiliar(Elaboración Propia)	23
Figura 4.3	Integrantes de una Familia. (Elaboración Propia)	25
Figura 5.1	Librería DHT22 para sensor de temperatura en Arduino IDE. (Elaboración Propia)	28
Figura 5.2	Tipos de Vivienda Unifamiliar(Elaboración Propia)	29
Figura 5.3	Diseño en Fritzing.(Elaboración Propia)	30
Figura 5.4	Diseño del Case.(Elaboración Propia)	31
Figura 5.5	Pruebas de Caja Negra y Blanca.(Elaboración Propia)	34
Figura 5.6	Diseño del Esquema de Comunicación Entre los Módulos.	35
Figura 5.7	Diseño del Árbol Directorios.	36

Figura 5.8	Servidor en funcionamiento.	38
Figura 5.9	Diseño de las vistas.	39

Capítulo 1

INTRODUCCIÓN

1.1. Antecedentes

La casa siempre ha sido esencial para los seres humanos desde los inicios de los tiempos hasta la actualidad. En el transcurso de esta se incorporaron varios elementos para que la vivienda se ha más confortable, porque para muchos al retornar de una jornada de trabajo, se sientan tranquilos al regresar a su hogar. También para los que se encuentran dentro la casa, se sientan seguros al interactuar con la misma.

Como la tecnología va abarcando en varias áreas, también lo está haciendo con nuestra vivienda, por que las personas buscan tener más control de los recursos de la vivienda y poder interactuar con la misma, es la esencia humana que nos impulsa a tener el control de todo.

Aprovechar de una manera más efectiva la energía que se usa en nuestro hogar , que la luz del jardín, sala , pasillo no esté encendida hasta medio día cuando no sea necesaria , ya sea porque al salir se nos olvidó apagarla o porque la dejamos prendida para tratar de engañar que hay alguien en casa, podemos optimizar su uso evitando derroches.

Con la edad avanzada, accidentes, o enfermedades se ven disminuidas nuestras capacidades, imposibilitando o haciendo difícil vivir con autonomía, provocando que las personas tengan dificultades para vivir hasta en su propio hogar, con un hogar inteligente esta dependencia se disminuye en gran medida.

1.2. Análisis del problema

Para la mejor interpretación y comprensión del problema, se analizara el árbol de problemas que es la representación, más relevante de una lista de problemas causas-efecto que se obtuvo al realizar entrevistas verbales a las personas. De este modo se pudo tener a primera mano las necesidades que las personas deseen.

1.2.1. Árbol de problema

El árbol de problemas está compuesto principalmente de tres partes con las cuales se trabajara de manera más comprensible:

La primera es el problema central que se identificó de la lista de problemas causa-efecto, como la más relevante de entre todas y está al centro del árbol. La segunda son las causas, estos aportan en gran magnitud al problema central y se encuentran en la parte inferior del árbol. Y por último los efectos, que son generados por los problemas y se encuentran en la parte superior del árbol. Toda esta descripción se la puede apreciar en la Figura 1.1 respectivamente.

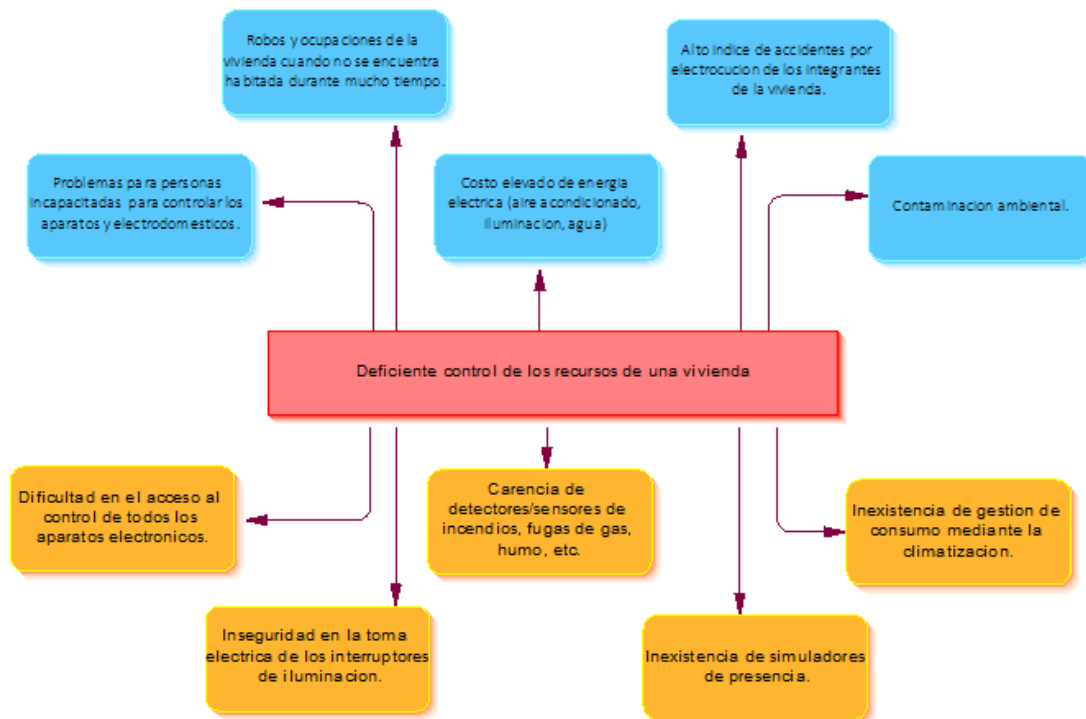


Figura 1.1: Árbol de Problemas (Elaboración Propia)

1.2.2. Descripción del Problema

El problema central donde se enfocara el proyecto, es la exagerada utilización de los recursos energéticos de una vivienda. También está la dificultad de las personas de la tercera edad y niños, difícilmente pueden acceder al control de estos electrodomésticos, y si lo hiciera, lo lograría con el riesgo de dañar su integridad física.

1.3. Propuesta de solución

Para resolver todos los problemas vistos anteriormente en el árbol de problemas se hará que cada una de ellas sea solucionada indirectamente. Para eso se verá el árbol de medios y fines.

1.3.1. Árbol de medios y fines

El árbol de medios y fines no es más que el árbol de problemas pero transformada desde una vista positiva de los problemas ya vistos. De esta manera las causas serán objetivos a cumplir, el problema central será el objetivo meta y los efectos serán los resultados a obtener, como se puede apreciar en la Figura 1.2 para su mejor comprensión.

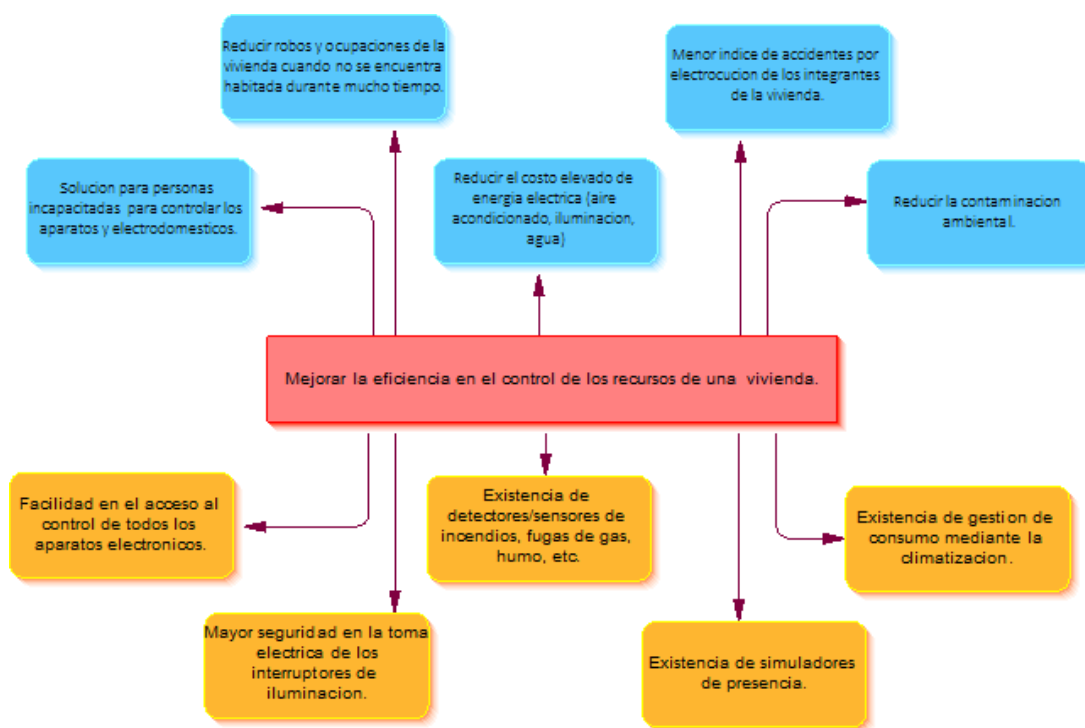


Figura 1.2: Árbol de Solucion (Elaboración Propia)

1.4. Objetivo general

Implementación de un sistema domótico para el control de los recursos domésticos en vivienda unifamiliar.

1.5. Objetivos específicos

Los objetivos específicos que contribuirán a desarrollar el *objetivo general* del trabajo son los siguientes:

- Monitoreo de la vivienda mediante sensores de presencia, temperatura, humedad y lluvia.
- Facilidad en el acceso al control en tiempo real de todos los aparatos eléctricos.
- Aplicar de NodeJS para montar un servidor con el cual se realizara el control en tiempo real de los recursos de la vivienda.
- Implementación de un simulador de presencia para la seguridad de la vivienda.

1.6. Justificación

Con la realización del sistema de control domótico para los habitantes de la vivienda se le asegura un aumento del confort, de la seguridad, del ahorro energético y de las facilidades de comunicación, haciendo uso de las nuevas tecnologías de comunicación y teléfonos inteligentes.

1.7. Alcance

Dirige los recursos de una vivienda como el control de encendido y apagado de las luces también monitorea la vivienda como ser los datos de temperatura, humedad y gas. Todo esto se realiza haciendo uso del sistema de Control que interactúa a través de una página web que está conectada mediante una red local o global como el internet estando dentro de la vivienda o de manera remota. Controlando en tiempo real, para que los usuarios puedan ver los cambios al momento de realizarse.

Para el mejor control de los dispositivos de la vivienda se hará uso de las placas de micro-controladores Arduino conjuntamente de los sensores de temperatura, humedad, lluvia y detectores de gas y humo.

No se tendrán cuentas de usuario ni restricciones para el acceso al sistema, no se garantiza el estado de los electrodomésticos y otros recursos por el manejo inapropiado del sistema.

Capítulo 2

DOMÓTICA

2.1. Introducción

El concepto domótica se refiere a la automatización y control (encendido / apagado, apertura / cierre y regulación) de aparatos y sistemas de instalaciones eléctricas y electrotécnicos (iluminación, climatización, persianas y toldos, puertas y ventanas motorizados, el riego, etc.) de forma centralizada y/o remota. El objetivo del uso de la domótica es el aumento del confort, el ahorro energético y la mejora de la seguridad personal y patrimonial en la vivienda. ([Morales, 2011](#))

2.2. Aspectos principales

Los servicios que ofrece la domótica se pueden agrupar en cuatro aspectos principales: ([Moya and Tejedor, 2004](#))

- En el ámbito del ahorro energético.
- En el ámbito del nivel de confort.
- En el ámbito de la protección patrimonial.
- En el ámbito de las comunicaciones.

2.2.1. En el ámbito del ahorro energético

El ahorro energético es un pilar fundamental en la domótica, ya que con la automatización de diferentes recursos dentro la vivienda se asegura el consumo justo de la energía eléctrica. Para este propósito se tienen los siguientes puntos más sobresalientes:

- Climatización: Aspectos que se relacionan con el clima y estaciones del año.

Programación: La libertad de poder realizar el control de la manera que uno lo prefiera.

Zonificación: Adecuarse a una estación del año.

- Gestión eléctrica:

Racionalización de cargas eléctricas: Desconexión de equipos de uso no prioritario.

Gestión de tarifas: Derivando el funcionamiento de algunos aparatos a horas de tarifa reducida.

Uso de energías renovables: energía eólica y la energía termosolar.

2.2.2. En el ámbito del nivel de confort

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

- Iluminación: La mayor parte de los recursos en una vivienda son las luces.

Automatización del apagado/ encendido en cada punto de luz.

Regulación de la iluminación según el nivel de luminosidad ambiente.

- Automatización de todos los distintos sistemas dotándolos de control eficiente y de fácil manejo:

Sistemas de telefonía/energía eléctrica.

Redes de gas/agua.

Equipos electrónicos.

- Integración del portero al teléfono, o del video-portero al televisor:

Recepción de personas en la puerta de la vivienda.

- Control vía Internet:

Aplicaciones para teléfonos inteligentes.

- Gestión Multimedia y del recreo electrónico:

Manipulación de los equipos de sonido y cine en casa.

- Generación de macros y programas de forma sencilla para el usuario:

Sistemas de control domótico.

2.2.3. En el ámbito de la protección patrimonial (seguridad)

Consiste en una red de seguridad encargada de proteger tanto los Bienes Patrimoniales y la seguridad personal.

- Simulación de presencia:

Escenarios para simular presencia en la vivienda.

- Detección de conatos de incendio, fugas de gas, escapes de agua.

Actuadores de cierre y apagado:

Notificación de los problemas a los habitantes de la vivienda.

- Alerta médica. Tele-asistencia:

Marcado automático a emergencias.

- Cerrar persianas en forma puntual y seguro:

De acuerdo al clima o definidos por el habitante de la vivienda.

2.2.4. En el ámbito de las comunicaciones

Este aspecto se observa la relación del sistema domótico con el habitante de la vivienda, la forma de comunicarse fuera o en el interior, recibir las señales y poder realizar acciones.

- Ubicuidad en el control tanto externo como interno.
- Transmisión de alarmas.
- Intercomunicaciones.

2.3. Estructura de una Instalación Domótica

La amplitud de una solución de domótica puede variar desde un único dispositivo, que realiza una sola acción, hasta amplios sistemas que controlan prácticamente todas las instalaciones dentro de la vivienda(CasaDomo.com, 2017). Los distintos dispositivos de los sistemas de domótica se pueden clasificar en los siguientes grupos:

- Controlador: Los controladores son los dispositivos que gestionan el sistema según la programación y la información que reciben. Puede haber un solo controlador, o varios distribuidos por el sistema.
- Actuador: El actuador es un dispositivo capaz de ejecutar y/o recibir una orden del controlador y realizar una acción sobre un aparato o sistema (encendido/apagado, subida/bajada, apertura/cierre, etc.).
- Sensor: El sensor es el dispositivo que monitoriza el entorno captando información que transmite al sistema (sensores de agua, gas, humo, temperatura, viento, humedad, lluvia, iluminación, etc.).
- Bus: Es bus es el medio de transmisión que transporta la información entre los distintos

dispositivos por un cableado propio, por la red de otros sistemas (red eléctrica, red telefónica, red de datos) o de forma inalámbrica.

- **Interface:** Los interfaces refiere a los dispositivos (pantallas, móvil, Internet, conectores) y los formatos (binario, audio) en que se muestra la información del sistema para los usuarios (u otros sistemas) y donde los mismos pueden interactuar con el sistema.

Es preciso destacar que todos los dispositivos del sistema de domótica no tienen que estar físicamente separados, sino varias funcionalidades pueden estar combinadas en un equipo. Por ejemplo en la Figura 2.1 se muestra un equipo de Central de Domótica, que puede ser compuesto por un controlador, actuadores, sensores y varios interfaces.

2.3.1. Actuación los Sistemas de Domótica

Los sistemas de domótica actúan sobre, e interactúan con, los aparatos y sistemas eléctricos de la vivienda según:

- El programa y su configuración.
- La información recogida por los sensores del sistema.
- La información proporcionada por otros sistemas interconectados.
- La interacción directa por parte de los usuarios.

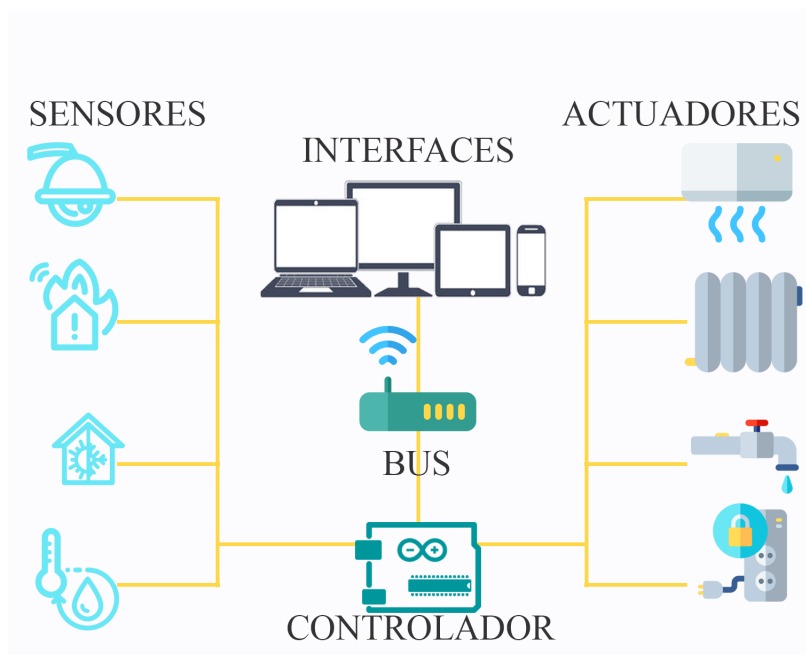


Figura 2.1: Dispositivos de sistema domotico (Elaboración Propia)

2.4. Arquitectura de un Sistema de Domótica

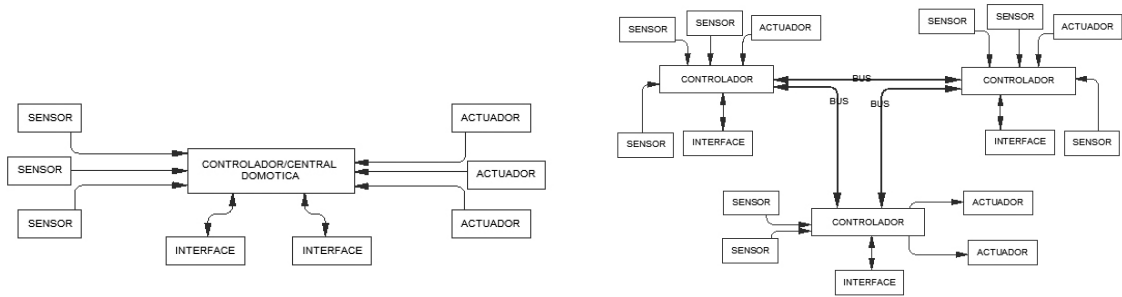
La arquitectura de los sistemas de domótica hace referencia a la estructura de su red. La clasificación se realiza en base de donde reside la “inteligencia” del sistema domótico([Serrano, 2009](#)). Las principales arquitecturas son:

- **Arquitectura Centralizada.**- En un sistema de domótica de arquitectura centralizada, un controlador centralizado, envía la información a los actuadores e interfaces según el programa, la configuración y la información que recibe de los sensores, sistemas interconectados y usuarios. Como se aprecia en la Figura 2.2a.
- **Arquitectura Descentralizada** – En un sistema de domótica de Arquitectura Descentralizada, observe la Figura 2.2b hay varios controladores, interconectados por un bus, que envía información entre ellos y a los actuadores e interfaces conectados a los controladores, según el programa, la configuración y la información que recibe de los sensores, sistemas interconectados y usuarios.
- **Arquitectura Distribuida** - En un sistema de domótica de arquitectura distribuida, en la Figura 2.2c, se puede apreciar que cada sensor y actuador es también un controlador capaz de actuar y enviar información al sistema según el programa, la configuración, la información que capta por si mismo y la que recibe de los otros dispositivos del sistema.
- **Arquitectura Híbrida / Mixta** – En un sistema de domótica de arquitectura híbrida (también denominado arquitectura mixta) se combinan las arquitecturas de los sistemas centralizadas, descentralizadas y distribuidas. Observe la Figura 2.2d que a la vez que puede disponer de un controlador central o varios controladores descentralizados, los dispositivos de interfaces, sensores y actuadores pueden también ser controladores (como en un sistema “distribuido”) y procesar la información según el programa, la configuración, la información que capta por si mismo, y tanto actuar como enviarla a otros dispositivos de la red, sin que necesariamente pasa por otro controlador.

2.5. Medios de Transmisión / Bus

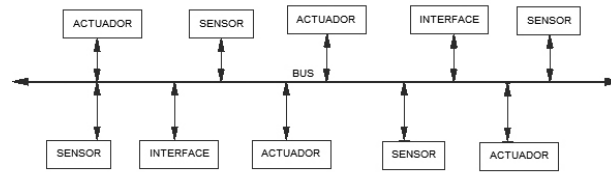
El medio de transmisión de la información, interconexión y control, entre los distintos dispositivos de los sistemas de domótica puede ser de varios tipos ([CasaDomo.com, 2017](#)). Los principales medios de transmisión son:

- **Cableado Propio** – La transmisión por un cableado propio es el medio más común para los sistemas de domótica, principalmente son del tipo: par apantallado, par trenzado (1 a 4 pares), coaxial o fibra óptica.
- **Cableado Compartido** – Varios soluciones utilizan cables compartidos y/o redes existentes para la transmisión de su información, por ejemplo la red eléctrica (corrientes portadoras),

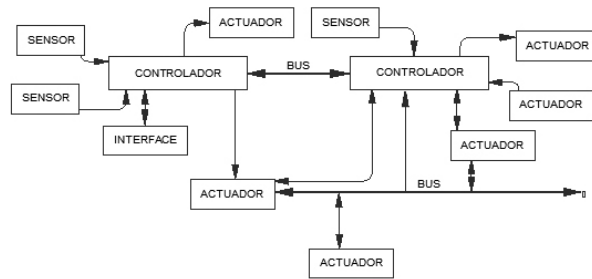


(a) Arquitectura domótica centralizada.

(b) Arquitectura domótica descentralizada.



(c) Arquitectura domótica distribuida.



(d) Arquitectura domótica mixta.

Figura 2.2: Arquitectura domótica(Elaboración Propia)

la red telefónica o la red de datos.

- Inalámbrica – Muchos sistemas de domótica utilizan soluciones de transmisión inalámbrica entre los distintos dispositivos, principalmente tecnologías de radiofrecuencia o infrarrojo.

Cuando el medio de transmisión está utilizado para transmitir información entre dispositivos con la función de “controlador” también se denomina “Bus”. El bus también se utiliza muchas veces para alimentar a los dispositivos conectados a él (por ejemplo European Installation Bus – EIB).

2.6. Sistemas Domóticos

Existen diferentes tecnologías o estándares de domótica que se aplican para la implementación de dispositivos y la red de comunicación, entre los que cabe señalar:

- **Sistemas Domóticos Ad Hoc** Estos sistemas están pensados para aplicaciones determinadas y su configuración es muy limitada, como por ejemplo el control de intensidad de una luminaria, el control de riego por temporizador, el encendido de luminarias activadas por sensor de movimiento y así un número de posibles ejemplos para casos concretos. Pero, en definitiva serán pocos elementos, unos cercanos a otros que posiblemente podamos encontrar en estructuras centralizadas con conexión en estrella basadas en bus. La funcionalidad de los sistemas está limitada a la programación establecida por fábrica, dejando poco margen de configuración por parte del usuario, tan solo tendrá libertad para definir unos pocos parámetros, por ejemplo, en un temporizador de regadío, el horario de encendido y de apagado. Debido a que están pensados para aplicaciones determinadas encontraremos en el mercado los elementos Ad Hoc especializados en algún tipo de aplicación: seguridad, control de clima, alarmas, confort, ahorro energético, etc. Una característica importante es que los elementos no pueden comunicarse entre sí, sin embargo, es viable insertar elementos ad hoc, mediante adaptadores, en un sistema domótico que utilice una de las tres restantes técnicas, pudiendo ser controlado([Serrano, 2009](#)).
- **Sistemas Domóticos sobre Red Eléctrica de Baja Tensión** En este tipo de instalaciones domóticas, en Redes Eléctricas de Baja Tensión, también conocidas como Power Line (PL), el medio de transmisión es el cableado de la red eléctrica de baja tensión (220 VAC). La transmisión es digital para lo cual se requiere que la onda sinusoidal sea lo más limpia posible, como máximo tenga una distorsión del 10 % sobre la tensión eficaz de 220 VAC y en frecuencia ± 0.5 Hz. sobre los 50 Hz. Al emitir una señal de radiofrecuencia el sistema ha de estar homologado para cumplir las normas que eviten que dicha señal interfiera en la señal del suministro eléctrico. Además, al ser aditiva dicha interferencia, proveniente de los diversos sistemas domóticos en paralelo, hay que atender a este detalle a la hora de la planificación del proyecto domótico. Aunque normalmente con los sistemas homologados y un buen diseño, en general, los sistemas PL introducen poca interferencia de radiofrecuencia. Otro detalle a tener en cuenta en los sistemas domóticos PL es la impedancia. Ésta

genera una disminución de la tensión como consecuencia del decremento de la impedancia, causada por el aumento de la capacidad electrostática (C). Estos pequeños cambios resistivos son detectados por los sistemas domóticos y deben adaptarse dinámicamente a ellos. Con estos puntos a tener en cuenta la aplicación domótica PL no se debe utilizar, por normativa, en los casos siguientes: a la hora de monitorizar equipos médicos, conectar varios edificios, en redes eléctricas donde estén conectadas maquinarias que sobrepasen los límites de interferencia radioeléctrica (generalmente motores de potencia), cuando exista transformadores en la red eléctrica, si la línea de red eléctrica es utilizada por sistemas que utilicen la banda de 105,6 kHz. – 115,2 kHz. y finalmente, cuando no se asegure los límites 220c +/- 10 % y 50 Hz. +/- 0,5 Hz(Serrano, 2009).

- Técnica sobre Radio Frecuencia Los elementos empleados para esta técnica utilizan como medio de transmisión el radioeléctrico. Cada sensor y actuador lleva integrado un dispositivo transmisor y receptor. Un solo receptor (o varios, si existen obstáculos insalvables o la vivienda es amplia) es el que recibe las señales de los sensores para procesarla y emitirla a los actuadores. La comodidad de esta técnica es que no hace falta ninguna obra de acometida para la instalación. Incluso pueden adherirse a cristales (como mamparas) haciendo al sistema muy versátil. La frecuencia utilizada para la transmisión aún no está estandarizada, por ejemplo, Jung, utiliza 433 MHz con una potencia más baja que la empleada en la telefonía móvil. Dicha frecuencia permite una transmisión de 1000 bits/s. La modulación utilizada es ASK (Amplitude Shift Keying). Un “1” ó “0” se asocia a un nivel distinto de señal que se modula con la portadora de 433 MHz. El alcance dependerá de los obstáculos que se encuentren, normalmente cuando el espacio es diáfano la distancia es de 300 m. decreciendo a 50 m. cuando existen obstáculos(Serrano, 2009).

2.6.1. Los Protocolos de Domótica

Los protocolos de comunicación son los procedimientos utilizados por los sistemas de domótica para la comunicación entre todos los dispositivos con la capacidad de “controlador”. Existen una gran variedad de protocolos, algunos específicamente desarrollados para la domótica y otros protocolos con su origen en otros sectores, pero adaptados para los sistemas de domótica. Los protocolos pueden ser del tipo estándar abierto (uso libre para todos), estándar bajo licencia (abierto para todos bajo licencia) o propietario (uso exclusivo del fabricante o los fabricantes propietarios)(Moya and Tejedor, 2004).

No se puede entender la domótica, sin conocer el protocolo de comunicaciones, como lenguaje de comunicación del Sistema Domótico. A través del protocolo se comunican los diversos dispositivos que componen la red domótica, de los cuales sobresalen los protocolos:

- Proprietarios o cerrados.- Son protocolos específicos de una marca en particular y que solo son usados por dicha marca. Pueden ser variantes de Protocolos Estándar. Son protocolos cerrados de manera que solo el fabricante puede realizar mejoras y fabricar dispositivos

que “hablen” el mismo idioma. Esto protege los derechos del fabricante, pero limita la aparición de continuas evoluciones en los sistemas domóticos, con lo que, a medida que los sistemas con protocolo estándar se van desarrollando, van ganando cuota de mercado a los sistemas de protocolo propietario. Otro problema que tienen es: la vida útil del sistema domótico, en un sistema propietario que depende en gran medida de la vida de la empresa y de la política que siga, si la empresa desaparece, el sistema desaparece y las instalaciones se quedan sin soporte ni recambios.

- Estándar o Abiertos.- Son protocolos definidos entre varias compañías con el fin de unificar criterios. Se los conoce como abiertos (open systems), es decir, que no existen patentes sobre el protocolo de manera que cualquier fabricante puede desarrollar aplicaciones y productos que lleven implícito el protocolo de comunicación. En un sistema estándar, si una empresa desaparece o deja de sacar productos al mercado, no afecta demasiado ya que hay otros productos en el mercado que cubren ese aspecto.

2.6.2. Tecnologías y estándares

Algunas tecnologías siguen presentes hasta el momento y se aplican en los sistemas domóticos actuales ([Herrera Quintero, 2005](#)). Los protocolos estándar para aplicaciones domóticas más extendidos en la actualidad son: KNX, Lonworks y X10 como se observa en la Figura 2.3.

- X-10: EE.UU finales de los 70.
- EHS (European Home System): 1992 Unión Europea.
- EIB (European Installation Bus).
- BatiBUS.
- KONNEX: EHS + EIB + BatiBUS.
- BIODOM: desarrollado por españoles.
- BACnet (de Building Automation and Control Networks).

2.6.3. Funcionamiento De Un Sistema Domótico

La configuración de un sistema domótico está íntimamente ligada a los procedimientos de transmisión de información que posibilitan el diálogo entre dichos periféricos y la unidad central. Los terminales (radiadores de calefacción, electrodomésticos, puntos de luz, etc.). Suelen ser equipos convencionales a los que se aporta una inteligencia o capacidad de comunicación a través de una interfaz. Los elementos de campo comprenden todo el conjunto de sensores que permiten convertir una magnitud física en señal eléctrica, y los actuadores u órganos de mando, capaces de transformar una señal eléctrica en una acción sobre el entorno físico.



Figura 2.3: Estándares Internacionales de Domotica. (Elaboración Propia)

Todos los elementos de campo envían y reciben señales a través de una red de comunicaciones (bus domótico), para comunicarse entre ellos y con la unidad central encargada de gestionar los intercambios de información. Estas señales de control están codificadas de una determinada forma (protocolos de comunicación), por lo que se necesitan unos elementos que pasen las señales bus y, a su vez, de señales bus a señales de salida a los actuadores (relés, interruptores, etc.). Estos elementos se suelen denominar de diferentes formas: módulos de entrada/salida, acopladores, interfaces, etc. (Moya and Tejedor, 2004).

Capítulo 3

ARDUINO Y NODEJS

3.1. Introducción

Arduino es una plataforma de hardware libre, basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios. El hardware consiste en una placa con un microcontrolador Atmel AVR y puertos de entrada/salida. Los microcontroladores más usados son el Atmega168, Atmega328, Atmega1280, ATmega8 por su sencillez y bajo coste que permiten el desarrollo de múltiples diseños. Por otro lado el software consiste en un entorno de desarrollo que implementa el lenguaje de programación Processing/Wiring y el cargador de arranque que es ejecutado en la placa. Arduino puede tomar información del entorno a través de sus entradas analógicas y digitales, puede controlar luces, motores y otros actuadores. El microcontrolador en la placa Arduino se programa mediante el lenguaje de programación Arduino (basado en Wiring) y el entorno de desarrollo Arduino (basado en Processing)([McRoberts, 2011](#)).

3.2. Programar un Arduino

Para programar un Arduino es necesario descargar e instalar el Arduino IDE, esta herramienta nos permite programar y subir nuestros archivos al dispositivo, el lenguaje de programación utilizado es compatible con C/C++ (a simple vista no es estrictamente C/C++ pero al momento de compilar el IDE se encarga de agregar los headers necesarios para que funcione en el dispositivo).

La estructura básica del lenguaje de programación de Arduino es bastante simple y se compone de al menos dos partes. Estas dos partes necesarias, o funciones, encierran bloques que contienen declaraciones, estamentos o instrucciones. El `setup()` es la parte encargada de recoger la configuración y `loop()` es la que contiene el programa que se ejecutará cíclicamente (de ahí el término loop –bucle–). Ambas funciones son necesarias para que el programa trabaje, en la Figura 3.1 se muestra la sintaxis.

La función de configuración (setup) debe contener la declaración de las variables. Es la primera función a ejecutar en el programa, se ejecuta sólo una vez, y se utiliza para configurar o inicializar pinMode (modo de trabajo de las E/S), configuración de la comunicación en serie y la inicialización de variables. La función bucle (loop) siguiente contiene el código que se ejecutara continuamente (lectura de entradas, activación de salidas, etc.) Esta función es el núcleo de todos los programas de Arduino y la que realiza la mayor parte del trabajo.

```
void setup() //Primera Parte
{
  //Aquí se realiza todo el código de inicialización y configuración
  estamentos;
}
void loop() //Segunda Parte
{
  //Aquí se realiza toda la funcionalidad principal de la aplicación
  estamentos;
}
```

Figura 3.1: Estructura de un Programa Arduino. (Elaboración Propia)

3.3. NodeJS

Nodejs es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor basado en el lenguaje de programación ECMAScript, asíncrono, con I/O de datos en una arquitectura orientada a eventos por lo tanto asíncrona, basado en el motor V8 de Google. V8 es el entorno de ejecución para JavaScript creado para Google Chrome, está escrito en C++ y compila el código fuente JavaScript en código de máquina en lugar de interpretarlo en tiempo real al lado del servidor. Nodejs contiene libuv para manejar eventos asíncronos. Libuv es una capa de abstracción de funcionalidades de redes y sistemas de archivo en sistemas Windows y sistemas basados en POSIX como Linux, Mac OS X y Unix. Además de la alta velocidad de ejecución de Javascript, la verdadera magia detrás de Nodejs es algo que se llama Bucle de Eventos (Event Loop - espera y envía eventos o mensajes). Para escalar grandes volúmenes de clientes, todas las operaciones intensivas I/O en Nodejs se llevan a cabo de forma asíncrona.

3.3.1. Gestor De Paquetes

Se han construido miles de bibliotecas de código abierto para Node.js, la mayoría de las cuales están alojadas la web de NPM(Node Package Manager). Es el gestor de paquetes usado por Node para publicar, buscar, instalar y desarrollar aplicaciones. Permite compartir y reutilizar paquetes de código para ensamblarlos en nuevas y/o mejores funcionalidades para:

- Archivos del sistema I/O.
- Redes: DNS, HTTP, TCP, TLS/SSL, UDP.
- Datos binarios (buffers).

- Funciones criptográficas.
- Flujos de datos.

Estos módulos utilizan una API diseñada para reducir la complejidad de la escritura de aplicaciones de servidor. Aunque NPM también puede gestionar paquetes del lado del cliente (front-end) está más enfocado a la gestión de módulos del servidor. El gestor de paquetes NPM, no obstante, es un poquito distinto a otros gestores de paquetes que podemos conocer, porque los instala localmente en los proyectos. Es decir, al descargarse un módulo, se agrega a un proyecto local, que es el que lo tendrá disponible para incluir. Aunque cabe decir que también existe la posibilidad de instalar los paquetes de manera global en nuestro sistema.

3.3.2. Marcos de Desarrollo

Son entornos de trabajo que disponen de las librerías necesarias para automatizar y facilitar el desarrollo de aplicaciones. El scaffolding es una técnica admitida por algunos marcos modelo-vista-controlador, en los cuales el programador puede especificar cómo se puede usar la base de datos de la aplicación. El compilador o marco utiliza esta especificación, junto con plantillas de código predefinidas, para generar el código final que la aplicación puede usar para crear, leer, actualizar y eliminar entradas de la base de datos, tratando de manera efectiva las plantillas como un andamio sobre el cual construir una aplicación más poderosa(Quintana et al., 2004).

- Generador de aplicaciones Express.- Es una herramienta del generador de aplicaciones, express-generator para crear rápidamente un esqueleto de aplicación, disponible en el gestor de paquetes. Observe la Figura la cual es representa la estructura de directorios generada.

3.3.3. Componentes para comunicación en tiempo real

Socketio es una librería que nos permite manejar eventos en tiempo real mediante una conexión TCP y todo ello en JavaScript. Con el cual tenemos a nuestra disposición un nuevo protocolo que permite la interacción entre el cliente y el servidor, facilitando la transmisión de datos en tiempo real en ambas direcciones, es una comunicación bidireccional (el emisor envía un mensaje por medio de un canal al receptor, quien lo recibe y envía la retroalimentación)(Rai, 2013).

3.4. Comunicación serial

La comunicación serial es un protocolo muy común para comunicación entre dispositivos que se incluye de manera estándar en prácticamente cualquier computadora. La comunicación serial es también un protocolo común utilizado por varios dispositivos para instrumentación; existen varios dispositivos compatibles con GPIB que incluyen un puerto RS-232. Además, la comunicación serial puede ser utilizada para adquisición de datos si se usa en conjunto con un dispositivo



Figura 3.2: Estructura de Directorios Generada con Express. (Elaboración Propia)

remoto de muestreo(Tamayo, 2009).

El concepto de comunicación serial es sencillo. El puerto serial envía y recibe bytes de información un bit a la vez. Aun y cuando esto es más lento que la comunicación en paralelo, que permite la transmisión de un byte completo por vez, este método de comunicación es más sencillo y puede alcanzar mayores distancias. Por ejemplo, la especificación IEEE 488 para la comunicación en paralelo determina que el largo del cable para el equipo no puede ser mayor a 20 metros, con no más de 2 metros entre cualesquier dos dispositivos; por el otro lado, utilizando comunicación serial el largo del cable puede llegar a los 1200 metros(Tamayo, 2009).

Típicamente, la comunicación serial se utiliza para transmitir datos en formato ASCII. Para realizar la comunicación se utilizan 3 líneas de transmisión: (1) Tierra (o referencia), (2) Transmitir, (3) Recibir. Debido a que la transmisión es asincrónica, es posible enviar datos por un línea mientras se reciben datos por otra. Existen otras líneas disponibles para realizar handshaking, o intercambio de pulsos de sincronización, pero no son requeridas. Las características más importantes de la comunicación serial son la velocidad de transmisión, los bits de datos, los bits de parada, y la paridad(Instruments, 2006). Para que dos puertos se puedan comunicar, es necesario que las siguientes características sean iguales:

- Velocidad de transmisión (baud rate): Indica el número de bits por segundo que se transfieren, y se mide en baudios (bauds). Por ejemplo, 300 baudios representa 300 bits por segundo. Cuando se hace referencia a los ciclos de reloj se está hablando de la velocidad de

transmisión. Por ejemplo, si el protocolo hace una llamada a 4800 ciclos de reloj, entonces el reloj está corriendo a 4800 Hz, lo que significa que el puerto serial está muestreando las líneas de transmisión a 4800 Hz. Las velocidades de transmisión más comunes para las líneas telefónicas son de 14400, 28800, y 33600. Es posible tener velocidades más altas, pero se reduciría la distancia máxima posible entre los dispositivos. Las altas velocidades se utilizan cuando los dispositivos se encuentran uno junto al otro, como es el caso de dispositivos GPIB.

- Bits de datos: Se refiere a la cantidad de bits en la transmisión. Cuando la computadora envía un paquete de información, el tamaño de ese paquete no necesariamente será de 8 bits. Las cantidades más comunes de bits por paquete son 5, 7 y 8 bits. El número de bits que se envía depende en el tipo de información que se transfiere. Por ejemplo, el ASCII estándar tiene un rango de 0 a 127, es decir, utiliza 7 bits; para ASCII extendido es de 0 a 255, lo que utiliza 8 bits. Si el tipo de datos que se está transfiriendo es texto simple (ASCII estándar), entonces es suficiente con utilizar 7 bits por paquete para la comunicación. Un paquete se refiere a una transferencia de byte, incluyendo los bits de inicio/parada, bits de datos, y paridad. Debido a que el número actual de bits depende en el protocolo que se seleccione, el término paquete se usará para referirse a todos los casos.
- Bits de parada: Usado para indicar el fin de la comunicación de un solo paquete. Los valores típicos son 1, 1.5 o 2 bits. Debido a la manera como se transfiere la información a través de las líneas de comunicación y que cada dispositivo tiene su propio reloj, es posible que los dos dispositivos no estén sincronizados. Por lo tanto, los bits de parada no sólo indican el fin de la transmisión sino además dan un margen de tolerancia para esa diferencia de los relojes. Mientras más bits de parada se usen, mayor será la tolerancia a la sincronía de los relojes, sin embargo la transmisión será más lenta.
- Paridad: Es una forma sencilla de verificar si hay errores en la transmisión serial. Existen cuatro tipos de paridad: par, impar, marcada y espaciada. La opción de no usar paridad alguna también está disponible. Para paridad par e impar, el puerto serial fijará el bit de paridad (el último bit después de los bits de datos) a un valor para asegurarse que la transmisión tenga un número par o impar de bits en estado alto lógico. Por ejemplo, si la información a transmitir es 011 y la paridad es par, el bit de paridad sería 0 para mantener el número de bits en estado alto lógico como par. Si la paridad seleccionada fuera impar, entonces el bit de paridad sería 1, para tener 3 bits en estado alto lógico. La paridad marcada y espaciada en realidad no verifican el estado de los bits de datos; simplemente fija el bit de paridad en estado lógico alto para la marcada, y en estado lógico bajo para la espaciada. Esto permite al dispositivo receptor conocer de antemano el estado de un bit, lo que serviría para determinar si hay ruido que esté afectando de manera negativa la transmisión de los datos, o si los relojes de los dispositivos no están sincronizados.

Capítulo 4

VIVIENDA UNIFAMILIAR

4.1. Introducción

Hoy en día el crecimiento poblacional de Bolivia hace notar que muchas familias viven en la misma vivienda, lo cual rompe el concepto de vivienda unifamiliar. Si no se tiene clara el concepto de una vivienda unifamiliar, veremos algunos conceptos que lo definen como ser la familia y vivienda.

4.1.1. Familia

La familia es un grupo de personas formado por individuos que se unen, primordialmente, por relaciones de filiación o de pareja. El Diccionario de la Lengua Española la define, entre otras cosas, como un grupo de personas emparentadas entre sí que viven juntas, lo que lleva implícito los conceptos de parentesco y convivencia, aunque existan otros modos, como la adopción. También definida, como un conjunto de ascendientes, descendientes y demás personas relacionadas entre sí por parentesco de sangre o legal.

Según la Declaración Universal de los Derechos Humanos, es el elemento natural, universal y fundamental de la sociedad, tiene derecho a la protección de la sociedad y del Estado.

Los lazos principales que definen una familia son de dos tipos: vínculos de afinidad derivados del establecimiento de un vínculo reconocido socialmente, como el matrimonio que, en algunas sociedades, sólo permite la unión entre dos personas mientras que en otras es posible la poligamia, y vínculos de consanguinidad, como la filiación entre padres e hijos o los lazos que se establecen entre los hermanos que descienden de un mismo padre. También puede diferenciarse la familia según el grado de parentesco entre sus miembros.

No hay consenso sobre una definición universal de la familia. La familia nuclear, fundada en la unión entre hombre y mujer, es el modelo principal de familia como tal, y la estructura difundida mayormente en la actualidad. Las formas de vida familiar son muy diversas, dependiendo de

factores sociales, culturales, económicos y afectivos. La familia, como cualquier institución social, tiende a adaptarse al contexto de una sociedad. Las familias están clasificadas en los siguientes tipos:

- Familia nuclear: formada por la madre, el padre y uno o más hijos.
- Familia extensa: abuelos, tíos, primos y otros parientes consanguíneos o afines.
- Familia monoparental: en la que el hijo o hijos viven con un solo progenitor (ya sea la madre o el padre).
- Familia ensamblada: Una familia ensamblada o familia reconstituida o familia mixta es una familia en la cual uno o ambos miembros de la actual pareja tiene uno o varios hijos de uniones anteriores.
- Familia homoparental: Se considera familia homoparental aquella donde una pareja de hombres o de mujeres se convierten en progenitores de uno o más niños.
- La familia de padres separados: en la que los padres se niegan a vivir juntos; no son pareja pero deben seguir cumpliendo su rol de padres ante los hijos por muy distantes que estos se encuentren.

En el Estado Plurinacional de Bolivia, 45,5 % de los hogares está conformado por el jefe de hogar, cónyuge e hijo(s) (hogar nuclear completo), mientras que 10,9 % está integrado por el jefe/a de hogar sin cónyuge, con la presencia de hijo(s) (hogar monoparental), Según el Instituto Nacional de Estadística (INE). Observe el Figura 4.1 representa la tipología de los hogares en bolivia encuesta del año 2015.

A nivel nacional, existen aproximadamente 3.012.000 hogares, de los cuales 2.036.000 se encuentran en el área urbana y 976.000 en el área rural, según la Encuesta de Hogares 2015. El tamaño medio del hogar es de 3,6 personas sin considerar a trabajadoras del hogar (Bolivia, 2012).

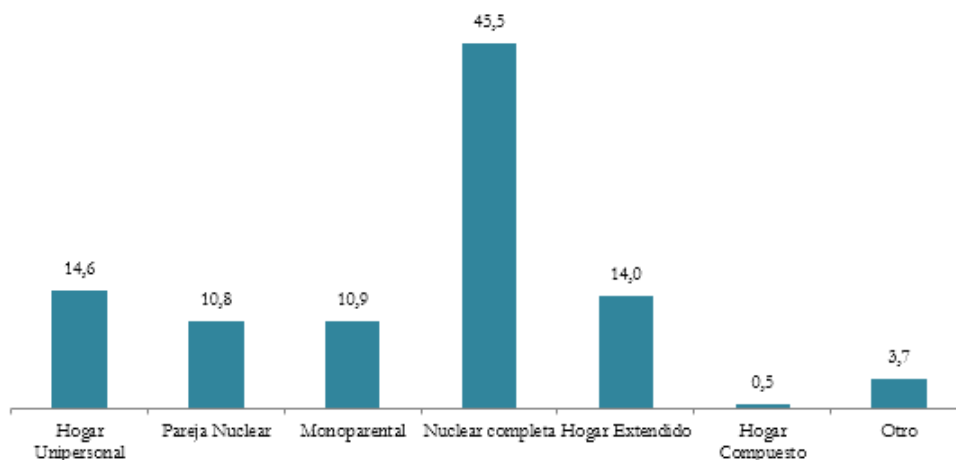


Figura 4.1: Tipología De Hogares, Encuesta De Hogares 2015. (Instituto Nacional de Estadística)

4.1.2. Vivienda y Hogar

La vivienda, es un edificio cuya principal función es ofrecer refugio y habitación a las personas, protegiéndolas de las inclemencias climáticas, el frío de la noche, del calor de algunos días y de otras amenazas naturales. La primera función de la vivienda es proporcionar un espacio seguro y confortable para resguardarse. El clima condiciona en gran medida tanto la forma de la vivienda como los materiales con que se construye y hasta las funciones que se desarrollan en su interior.

La palabra hogar se usa para designar a un lugar donde un individuo o grupo habita, creando en ellos la sensación de seguridad y calma. En esta sensación se diferencia del concepto de casa, que sencillamente se refiere a la vivienda física. La palabra hogar proviene del lugar donde se encendía el fuego, a cuyo alrededor se reunía la familia para calentarse y alimentarse. Se aplica también a todas aquellas instituciones residenciales que buscan crear un ambiente hogareño, por ejemplo: hogares de retiros, hogares de crianza, etc.

4.2. Definición

La definición de este tipo de casa puede variar entre jurisdicciones u organismos estadísticos, y sabiendo los puntos anteriores previamente vistos lo definimos como:

Una sola familia (hogar, casa o vivienda) significa que el edificio está generalmente ocupado por un solo hogar o la familia, y se compone de una sola unidad de vivienda o habitación. Se excluye, sin embargo, cualquier alojamiento a corto plazo (hoteles, moteles, casas de huéspedes), alojamiento de alquiler a gran escala (internados, apartamentos) y de las viviendas colectivas.

La mayoría de las viviendas unifamiliares se construyen en lotes más grandes que la propia estructura, añadiendo un área que rodea la casa, que comúnmente se llama un patio o un jardín, con un garaje y entrada.

4.3. Tipos de viviendas

Las comparaciones respecto al tipo de unidad de vivienda y su organización interior, su agrupamiento y densidad tienen implicancias en los estilos de vida de los usuarios. Las tipologías arquitectónicas, determinan las características de cada una de estas viviendas:

- Unifamiliar aislada o exenta: Es aquel edificio habitado por una única familia que no está en contacto físico con otras edificaciones como se aprecia en la Figura 4.2a. Normalmente están rodeadas por todos sus lados por un terreno perteneciente a la vivienda, en el que se suele instalar un jardín, la vivienda puede tener uno, varios o todos sus lados alineados con las calles públicas.
- Unifamiliar pareada: En este caso, se construyen dos viviendas unifamiliares que exterior-

mente están en contacto, aunque en su distribución interior son totalmente independientes, teniendo cada una de ellas su propio acceso desde la vía pública en la Figura 4.2b se tiene un ejemplo.

- Unifamiliar adosada: Similar a la pareada, pero esta vez cada vivienda está en contacto con otras dos (una a cada lado). Este tipo de viviendas se suelen caracterizar por tener una planta estrecha y alargada y por la presencia de ventanas únicamente en los extremos de la casa como se observa en la Figura 4.2c donde hay dos columnas de viviendas.



(a) Vivienda unifamiliar aislada.



(b) Vivienda unifamiliar pareada.



(c) Vivienda unifamiliar adosadas.

Figura 4.2: Tipos de Vivienda Unifamiliar (Elaboración Propia)

4.4. Integrantes de una Vivienda Unifamiliar

Familia conviviente formada por los miembros de un único núcleo familiar, el grupo formado por los miembros de una pareja y/o sus hijos como se observa en la Figura 4.3. Las definiciones más amplias consideran en un núcleo familiar tanto a los grupos formados por dos adultos emparejados, con o sin hijos, como a los formados por un adulto con uno o varios hijos. Algunas definiciones más restrictivas la reducen a los casos en los que están presentes los dos progenitores.

4.4.1. Roles de los Integrantes

En el estudio de las familias, conocer cuál es la estructura familiar es uno de los pasos iniciales. Conocer el rol que cumplen cada uno de los integrantes de la familia será importante para tener una mejor visión del funcionamiento familiar.

La familia está organizada según el orden jerárquico en que se disponen sus miembros, donde cada posición le confiere obligaciones y prerrogativas delimitadas por reglas concretas las cuales contribuyen a vincular el funcionamiento de la familia con los fines del grupo familiar. En la familia sus miembros pueden ubicarse en el puesto de padre, madre, hijo, hermano, abuelo. A continuación se hace mención de los roles más sobresalientes:

- Rol Materno.- Los psicoanalistas están de acuerdo en la concepción clínica de lo que constituye “un buen ejercicio maternal”. La madre debe constituirse en un “medio aprovisionador total” del niño y esta provisión consiste en algo más que la mera satisfacción de necesidades fisiológicas. La madre debe realizar todo lo que el niño es incapaz de hacer por sí mismo: alimentación, vestido, higiene y transporte, añadiendo a la atención maternal un contenido afectivo seguro; es un hecho emocional que se integra y unifica con el hecho físico.

- Rol Paterno.- La presencia de la figura paterna, está relacionada con la misión del padre en el seno de la familia, y en particular, respecto a la relación que ha de establecer con el hijo. La misión quedaría enmarcada dentro de las siguientes características:

Ser modelo de identificación para el hijo/hija.

Establecer un tipo particular de liderazgo en el interior de la familia.

Desarrollar una concreta acción formativa en la vida del hijo (seguridad, valores, autoridad, disciplina, identidad personal).

- Rol de Hermano.- Los hermanos y hermanas mayores a menudo actúan como modelo y profesores para sus hermanos menores. En estudios se han demostrado que los niños pequeños observan cuidadosamente a sus hermanos o hermanas mayores, con frecuencia cogen sus juguetes que han abandonado o imitan sus acciones. Los hermanos que no se llevan mucha diferencia de edad, a menudo tienen intereses similares, les gustan las mismas cosas y parecen entenderse mutuamente.

4.5. Recursos de una vivienda



Figura 4.3: Integrantes de una Familia. (Elaboración Propia)

Capítulo 5

DESARROLLO DEL TRABAJO

5.1. Introducción

El Problema que la mayoría de los desarrolladores enfrenta es el de como iniciar el desarrollo, ya teniendo las herramientas a utilizar y todo el cronograma planificado. Por ese motivo se describirá un proceso al cual realice unos ajustes para adecuarse a mis expectativas.

Para el desarrollo del sistema se optó por la utilización de un proceso de desarrollo denominado “proceso iterativo incremental” el cual se adecua de mejor manera al modo de trabajo para este tipo de proyecto.

5.2. Proceso de desarrollo de la metodología

Se realiza un análisis desde tres puntos de vista para entender mejor la estructura del sistema. La primera parte costa de un sistema de sensores y actuadores que están para el monitoreo de la vivienda. La segunda parte es el servidor y las comunicaciones entre el sistema de sensores/actuadores y el cliente final, ya sea haciendo uso de una red local o una global como la internet. La 3° y última parte que es la que el usuario final ve y hace uso de manera que le sea más fácil la interacción con los recursos de la vivienda. Es el front-end que hace posible todo esto, ya que como el lenguaje web es multiplataforma hace más sencillo a los usuarios finales el no tener q instalar o depender de un sistema operativo en concreto para poder utilizar el sistema, gracias a esto los usuarios pueden acceder desde sus PC, portátiles con distintos tipos de sistemas operativos y mejor aún desde sus Smartphone o tabletas.

El sistema de control domótica es realizado mediante 3 iteraciones los cuales son:

5.3. Sensores y Armado del Prototipo

La primera iteración consiste en la arquitectura del hardware, la recolección de datos mediante sensores y el envío de instrucciones a los activadores, dependiendo de las condiciones impuestas previamente. Los sensores almacenan la información en variables para luego enviarlos al servidor en forma de una cadena.

5.3.1. Análisis

Analizando algunos escenarios para este módulo se vieron integrar las herramientas, dispositivos y tareas a consideradas para esta iteración:

- Herramientas.- Se hace el uso del IDE Arduino para la codificación y cargado del programa a la placa controladora. Para el ensamblado del prototipo es necesario:

Kit para el armado del PBC.

Conectores tipo hembra-macho.

Conectores tipo macho-macho.

Acrílico de 4mm de espesor para el diseño del case.

Adaptador de AC de voltaje 220 con salida de 5v.

Resistencias de 220ohm.

- Dispositivos.- Para el prototipo se vio conveniente el uso de los siguientes dispositivos:

Sensor de temperatura y humedad (DHT22).

Sensor de detección de gas (MQ3).

Sensor de lluvia (YL-83).

Sensor de presencia (PIR).

Módulo relé de 8 canales (para el manejo de corriente alterna).

Modulo enchufe hembra.

Led RGB de cátodo común.

- Tareas.- se realiza un cuadro de actividades con un orden de prioridades, las cuales se detallan en la Tabla donde la primera columna pertenece a las actividades, la segunda la prioridad que tiene estas y la tercera para la descripción detallada de la misma.

Pruebas década sensor con las librerías necesarias para el correcto funcionamiento esperando las salidas requeridas para el manejo de la información recolectada, A continuación se verán las pruebas con los sensores de temperatura y humedad DHT22, sensor de gas MQ2, sensor de

Iteracion I: Hadware		
Actividad	Prioridad	Descripcion
Preparación	1	Diagramación y armado de partes.
Pruebas individuales	2	Pruebas de cada sensor y actuador .
Prueba conjunta	2	Pruebas de todos los sensores en un solo código Arduino.
Compra de hardware	1	Sensores y adquisición de equipo de electrónica.
Compactar hardware	3	Fijar los componente de manera que sea portátil.
Construcción de case	1	Construcción de la caja para ensamblar los componentes.
Ensamblado	1	Ensamblado de sensores/actuadores y todo los componentes necesarios.

Tabla 5.1: Planificación de tareas primer módulo (Elaboración Propia).

presencia infrarroja PIR y el sensor de lluvia LY-83.

Temperatura y humedad: Este sensor recolecta los cambios variantes en la temperatura y humedad del ambiente en el que se encuentra, se lo sitúa en habitaciones, sala de estar, dentro la vivienda para el monitoreo de cada sección de la misma. Es necesario la importación de dos librerías para el funcionamiento (DHT y Andafruitsensor) como se muestra en la Figura 5.1. El

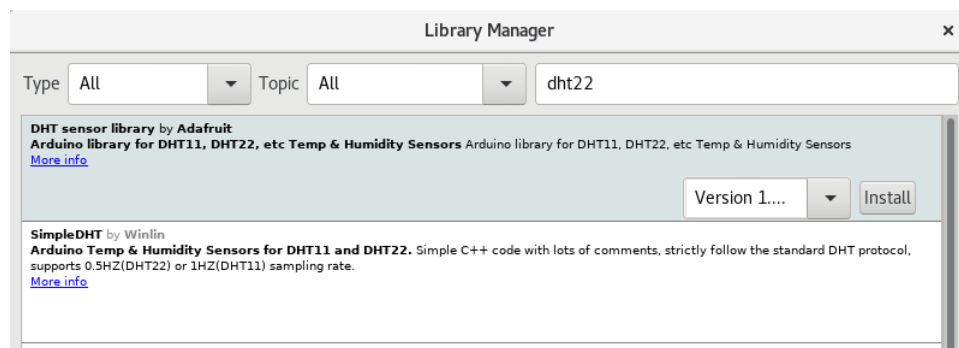


Figura 5.1: Librería DHT22 para sensor de temperatura en Arduino IDE. (Elaboración Propia)

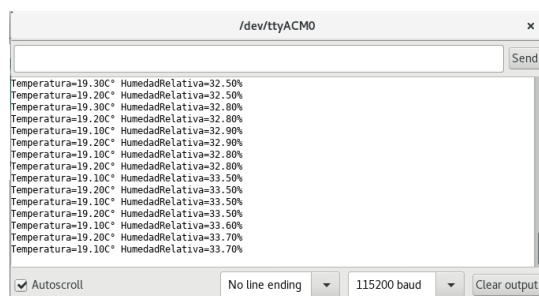
código de lectura del sensor es por el pin 2 de manera digital, la librería nos facilita la conversión de los datos de entrada digital a la conversión de °C y la lectura del valor de la humedad relativa. El resultado se lo puede observar en la Figura 5.2a, la cual muestra los valores que manda el sensor lo cual es visualizado por el monitor serial con los calores para temperatura= 26.40°C y humedad del 34.60 % respectivamente.

Sensor de Gas MQ2: Este sensor se lo sitúa en la cocina de mayor preferencia, de esta manera tener el monitoreo de esta área. La información que envía el sensor se recolecta por el pin analógico A1 con valores de 1-1023 las cuales son mapeadas a valores de 0 a 100 para tener

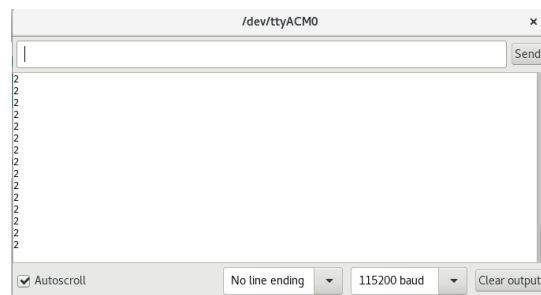
un mejor comprendiendo y relacionar a condiciones normales de los gases. La cual define que es dañino y peligroso si supera el valor de 30 y aceptable menor a esta. En la Figura 5.2b se muestra estados normales de concentración de gas con un valor de 2.

Sensor de presencia PIR: Este sensor de preferencia situado en los pasillos para la activación de las luces, o en las puertas interiores, el sensor recoleta ondas de choque que varías en su entorno alrededor de 4-5 metros de distancia en un diámetro de aproximado 360°. Estas variaciones de ondas son regulables por el sensor para ser más sensible o caso contrario, la variación de lectura de las ondas son de 4s. En la Figura 5.2c se puede apreciar la lectura del cambio de estado del sensor el cual modifica un actuador Encendido/apagado respectivamente, en este caso cambia de un estado de apagado a encendido pasar una mano cerca del sensor.

Sensor de Lluvia YL-83: de preferencia situarlo en la parte exterior de la vivienda, para recolectar información acerca de las lluvias. También puede actuar como sensor de fugas de agua. LA información es recolectada por el pin analógico A0, la cual está en un rango de valores de 0-1023 los cuales se mapean para que sea interpretada de mejor manera en 0-100. En la Figura 5.2d se muestra un cambio en los valores recolectados ya que se lo aplico agua en forma de gotas sobre el sensor, el cual reacciono de manera esperada.



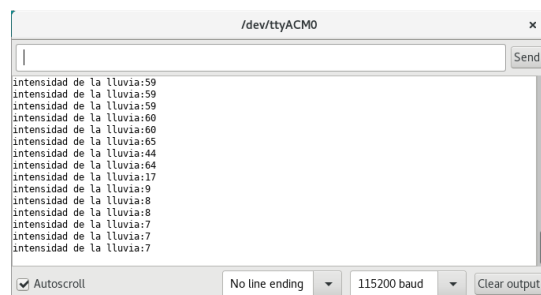
(a) Recolección de datos del sensor de Temperatura.



(b) Recolección de datos del sensor de Gas.



(c) Recolección de datos del sensor de Proximidad.

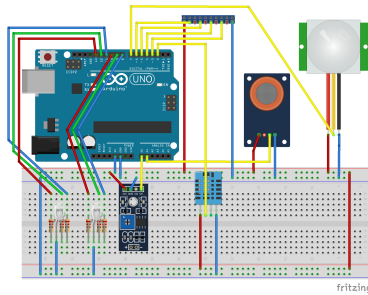


(d) Recolección de datos del sensor de Lluvia.

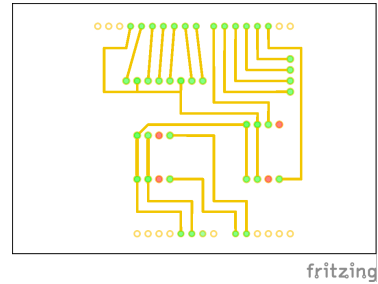
Figura 5.2: Tipos de Vivienda Unifamiliar(Elaboración Propia)

5.3.2. Diseño

Se realiza el diseño para el armado del prototipo, como también la integración con la siguiente iteración. El diseño se realiza mediante uso del framework fritzing, para poder tener una interpretación más certera, vease la Figura 5.3a.



(a) Diseño de los componentes.



(b) Diseño de Placa de Prototipos.

Figura 5.3: Diseño en Fritzing.(Elaboración Propia)

Una vez terminado el diseño, se genera una placa de circuito como se ve en la Figura 5.3b , para el mejorar el encapsulado de los sensores y actuadores, para el que control y los demás hardware estén mejor situados.

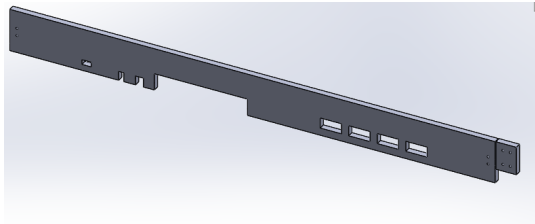
Se realiza el diseño del case donde se encuentran los sensores y actuadores como también la placa controladora y la Raspberry. El diseño está realizada en Solidwork para corte láser en acrílico, teniendo como base una dimensión de 20x23 cm. y los lados con cortes para los sensores y las placas, y una pata superior con perforaciones para los sensores de gas y presencia. Una vez terminado el diseño, se genera una placa de circuito como se ve en la Figura 5.3b , para el mejorar el encapsulado de los sensores y actuadores, para el que control y los demás hardware estén mejor situados.

5.3.3. Codificación

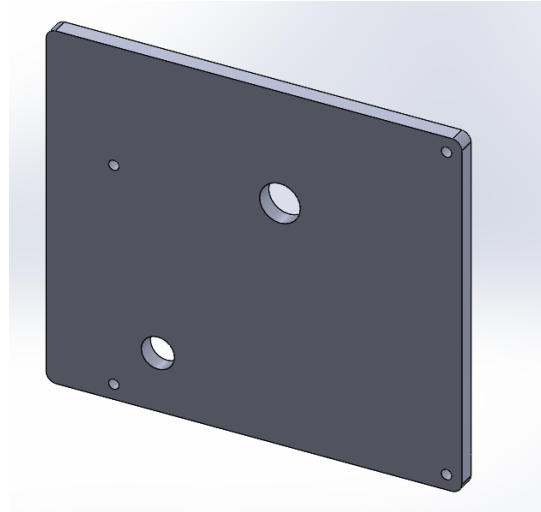
El archivo principal para este módulo es “domotica.ino”. La cual está realizado con el IDE Arduino, para poder apreciar de mejor manera ver el código completo en anexos. Se importan las librerías necesarias de los sensores (DHT22)

```
1    #include "DHT.h"           //cargamos la libreria DHT
2
```

Luego se declara las variables y la inicialización de las mismas como entradas digitales o analógicas, como los pines de salida. Todo esto en el método setup().



(a) Contorno del case.



(b) Tapa del case.

Figura 5.4: Diseño del Case.(Elaboración Propia)

```

1      Serial.begin(115200);      // Se inicia la comunicacion serial
2      dht.begin();              //Se inicia el sensor de temperatura
y humedad
3      //Temperatura
4      pinMode(LED_RED, OUTPUT); //se declara como salida los pines
para LED_RED 13
5      pinMode(LED_BLUE, OUTPUT); //se declara como salida los pines
para LED_BLUE 11
6      //lluvia
7      pinMode(analogInPin, INPUT); //se declara como entrada en
sensor pin en A0
8      //PIR
9      pinMode(pinSensorPIR, INPUT); //se declara como entrada en
sensor pin en pin 7
10     pinMode(pinActivadorPresc, OUTPUT); //se declara como salida en
pin12
11     digitalWrite(pinActivadorPresc, LOW); //se inicia como apagado
el pin12
12     //gas
13     pinMode(Rele, OUTPUT); //se declara como salida los pines
para rele 3
14     pinMode(buzzer, OUTPUT); //se declara como salida los pines
para buzzer 4

```

```

15     pinMode(agua, OUTPUT); //se declara como salida los pines
    para agua 5
16     // init LEDS
17     pinMode(led_rojo,OUTPUT); //se declara como salida los pines
    para led_pin 10
18     pinMode(led_verde,OUTPUT); //se declara como salida los pines
    para pwmPin 9
19     pinMode(led_azul,OUTPUT); //se declara como salida los pines
    para pin 8
20

```

Y las funciones más importantes en el control y manejo de información recolectado por los sensores son las siguientes que están dentro de la función loop():

```

1     //DHT Sensor Temp&Hum
2     float h = dht.readHumidity();      //Se lee la humedad
3     float t = dht.readTemperature();  //Se lee la temperatura
4     //PIR
5     estadoPIR= digitalRead(pinSensorPIR);    // se lee del sensor PIR
6     //GAS
7     sensorGas = analogRead(pinSensorGas);    //se lee del sensor de
    gas
8     sensorGas = map(sensorGas, 0, 1023, 0, 100); //mapeo de 0-100 de
    0-1023
9

```

La obtención de instrucciones que se recibe por medio del servidor es de forma alfanumérica (a1), donde la letra es la clave que hace referencia a un actuador seguida de un valor, la cual es la instrucción que realizara en este caso el encendido (1) y apagado (0).

```

1     // lee una letra y que recoja el valor
2     if(Serial.available()>0){
3     lectura = Serial.read();
4     switch (lectura) {
5     case 'a':
6         lectura = Serial.parseInt();
7         digitalWrite(led_azul,lectura);
8         break;
9     default:

```

```

10     break;
11 }
12

```

Los datos recolectados por los sensores son enviados en forma de cadena al servidor mediante comunicación serial:

```

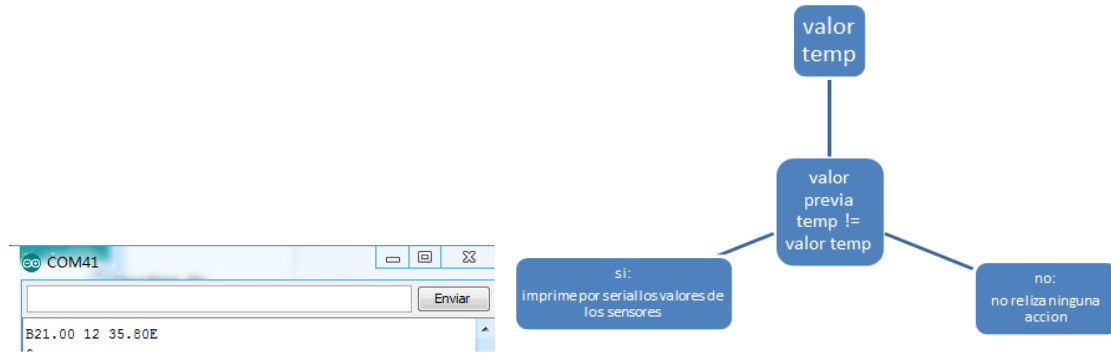
1     // mandamos por serial una sola cadena con todos los datos de
    los sensores
2     if(prevValueTemp != sensorTemperatura){
3         Serial.print("B");           // inicio de caracter
4         Serial.print(sensorTemperatura); //envia valor de temp
5         Serial.print(" ");           //separador de cadenas para
    el split
6         Serial.print(sensorGas);      //envia valor GAS
7         Serial.print(" ");           //separador de cadenas para
    el split
8         Serial.print(sensorH);        //envia valor humedad
9         Serial.println("E");          // fin del caracter
10        prevValueTemp = sensorTemperatura;
11    }
12

```

5.3.4. Pruebas

Para validar el correcto funcionamiento y la calidad en esta iteración se realizan las pruebas de caja blanca y caja negra:

- Pruebas de Caja Negra.- Se realizaron las pruebas tanto en cada sensor de forma individual como de forma conjunta para poder recolectar los datos y enviarlos por comunicación serial. En la Figura 5.5a se muestra los valores “B21.00 12 35.80E” estos valores corresponden a la temperatura, concentración de gas y la humedad respectivamente. El envío de datos ocurre siempre y cuando hay una variación en el valor de los sensores leídos anteriormente comparados con la lectura actual.
- Pruebas de Caja Blanca.- Como se vi anteriormente uno de las funciones importantes es el envío de datos por comunicación serial que se realiza mediante el siguiente método de forma correcta vease la Figura 5.5b.



(a) Salida de Datos por el Puerto Serial. (b) Flujo de datos de la Funcion de envio de datos.

Figura 5.5: Pruebas de Caja Negra y Blanca.(Elaboración Propia)

5.4. Montar servidor NodeJS y Establecer Comunicaciones

Esta es la segunda iteración que se encarga de establecer la comunicación entre la parte del hardware donde se encuentran los sensores/actuadores y la placa controladora. Mediante una comunicación serial es posible establecer la conexión con el servidor que se encuentra en una Raspberry. Cabe recalcar que la comunicación serial está ligada al hardware y no pueden estar separadas más de 2m de distancia de la placa controladora. Para la realización de esta iteración se vio necesario seguir las siguientes acciones:

5.4.1. Análisis

Se realiza la planificación de las tareas de este módulo como la recolección de requerimientos y generar plan de trabajo como se muestra en la Tabla .

Iteracion II: Servidor		
Actividad	Prioridad	Descripcion
Preparación	1	Configuración del router.
Configurar Raspberry	2	Instalación de NodeJS y dependencias.
Scaffold	1	Creación de la estructura del servidor.
Comunicación serial	1	Recolección/envió de datos por serial con el Arduino.
Ip publica	3	Definir la ip publica para el acceso al servidor desde internet.
Control de tiempos	1	Control de tiempos de respuesta desde un cliente al hardware.
Acceso remoto	2	Configuración de servicio VNC y ssh.

Tabla 5.2: Planificación de tareas Segundo Módulo (Elaboración Propia).

5.4.2. Diseño

La parte del sistema de los sensores recolecta toda la información necesaria y los envía por comunicación serial protocolo rc3220 hacia el servidor de node. El cual recolecta esa información y actualiza todos los datos de los sensores y los muestras a todos los usuarios conectados mediante el modulo socket.io, este último es para aplicaciones en tiempo real, la cual hace posible que cada cambio realizado por un usuario sea visible al mismo tiempo por otros que están conectados al mismo tiempo o ver los cambios al ingresar al sistema. Para que el sistema de los sensores tenga mayor certeza en los tiempos es necesario que se encuentren en una misma red local tanto el servidor como el sistema de sensores.



Figura 5.6: Diseño del Esquema de Comunicación Entre los Módulos.

Fuente:(Elaboración Propia)

El diseño para una infraestructura de aplicaciones web de NodeJS se realiza mediante herramienta de generador de aplicaciones express para crear rápidamente un esqueleto de aplicación. La aplicación generada tiene la siguiente estructura de directorios:

5.4.3. Codificación

Los módulos principales para la comunicación entre el Arduino y el servidor, como también el servidor y los usuarios finales mediante la plataforma web se requiere de estos dos módulos: serialport y socket.io. Para esto se instala el módulo utilizando el gestor de paquetes de Nodejs

```
1      $ npm install serialport
2      $ npm install socket.io
3
```

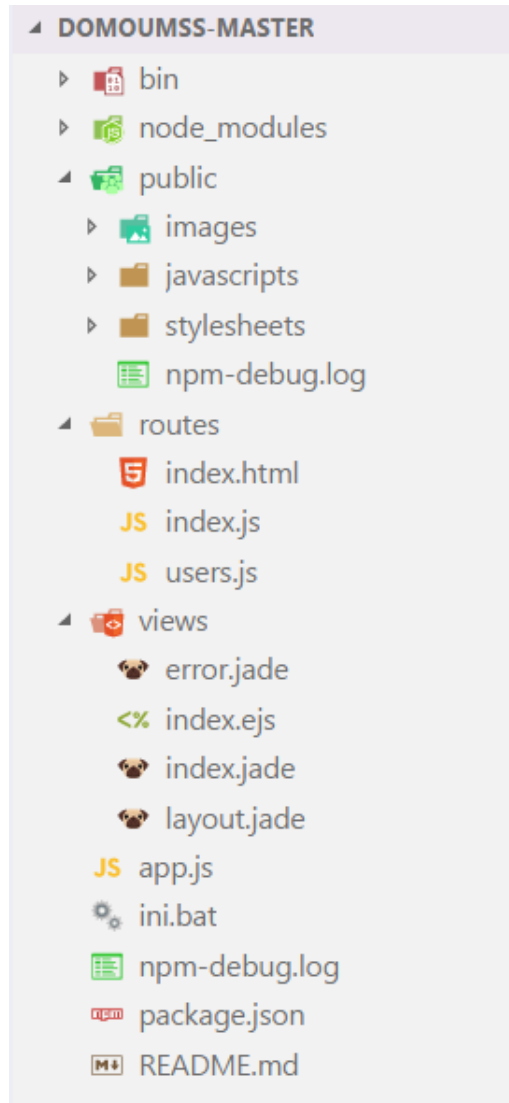


Figura 5.7: Diseño del Árbol Directorios.

Fuente(Elaboración Propia)

Siguiendo en el archivo JavaScript `app.js` se añade el código para integrar la comunicación serial con la placa Arduino.

```
1    var SerialPort = require("serialport").SerialPort;  
2    var socketio = require('socket.io').listen(4000);  
3
```

Se inicializa la configuración para el puerto serial con los valores: nombre del puerto que en caso de la Raspberry es `/dev/ttyACM0`, para sistemas operativos Windows son los puertos COM. También el baudrate, la cual debe de ser igual que la configuración del Arduino a 115200, son

los valores necesarios para establecer la comunicación entre el servidor y el Arduino.

```

1      var SerialPort = require("serialport").SerialPort;
2      var socketio = require('socket.io').listen(4000); //comunicacion
      Serial- Configuracion de puertos
3 var serialPort;
4 var portName = '/dev/ttyACM0'; //dev/ttyAMA0
5 function serialListener()
6 {
7     var receivedData = "";
8     serialPort = new SerialPort(portName, {
9         baudrate: 115200,
10         // defaults for Arduino serial communication
11         dataBits: 8,
12         parity: 'none',
13         stopBits: 1,
14         flowControl: false
15     });
16

```

Para recibir los valores enviados por el Arduino, se realiza con el siguiente método:

```

1      serialPort.on("open", function () {
2          serialPort.on('data', function(data) {
3              });
4          });
5

```

Para enviar de acciones a la placa Arduino por comunicación serial:

```

1      serialPort.write('a1');
2

```

5.4.4. Pruebas

Se realizaron las pruebas de los sensores para poder recolectar los datos y enviarlos por comunicación serial. A la vez recibir instrucciones para activar los actuadores.

Iteración III: Monitoreo-Frontend		
Actividad	Prioridad	Descripción
Preparación	1	Diseño de mockups.
Control encendido/apagado	1	Acciones de encendido/apagado desde la web.
Comunicación con el servidor	1	Usar sockets con el servidor.
Cámaras de vigilancia	2	Vistas para las cámaras.
Monitoreo	1	Vistas de los datos de los sensores.
Grafica	3	representacion de datos en grafica.
Simulacion de prescencia	2	Acciones programadas de encendido/apagado durante el dia y la noche.

Tabla 5.3: Planificación de tareas tercer módulo (Elaboración Propia).

5.5.2. Diseño

Se realiza el diseño mediante uso de mockups, para poder tener una interpretación más certera. La primera ventana donde se muestra los valores de los sensores de temperatura, humedad y la concentración de gas. La segunda ventana para las acciones de encendido y apagado de las luces, en la tercera ventana una gráfica del comportamiento de la temperatura y la última ventana la sección de vigilancia mediante cámaras.

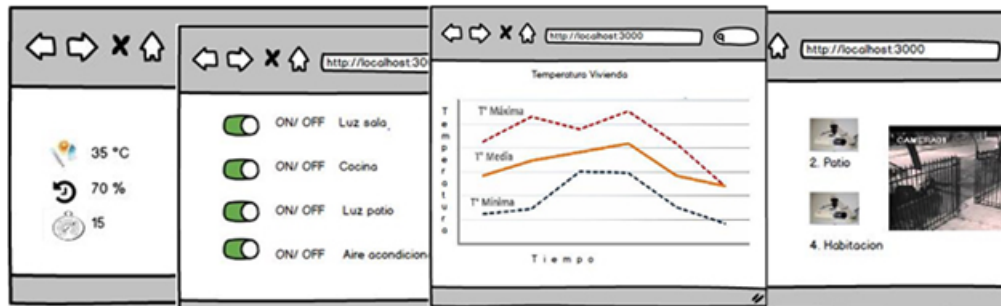


Figura 5.9: Diseño de las vistas.

Fuente: Elaboración Propia

5.5.3. Codificación

El archivo principal para este módulo es "index.jade". La cual forma parte de la estructura de un proyecto generado por express-generator, para poder apreciar de mejor manera ver el código completo en anexos.

También se configura los clientes para secreto, lo cual permite que los cambios de lectura de los sensores sea en tiempo real, como también las acciones del encendido/apagado de luces sean

visibles.

La Figura ?? Muestra ... (Explicar las figuras. Notar que la figura fue citada con referencia cruzada y la numeración es automática)

Capítulo 6

Resultados.

6.1. Ejemplo de titulo 2.

6.1.1. Ejemplo de titulo 3.

Ejemplo de titulo 4 (no numerado).

Ejemplo de título 5:

6.2. Discusión de resultados.

Capítulo 7

Conclusiones

Bibliografía

- Bolivia, I. (2012). Instituto nacional de estadística. *Censo de Población y Vivienda 2001*. 21
- CasaDomo.com (2017). Todo sobre edificios inteligentes @MISC. 7, 9
- De Solminihac, H. (2005). *Procesos y técnicas de construcción*. Ediciones UC.
- Herrera Quintero, L. F. (2005). Viviendas inteligentes (domótica). *Ingeniería e Investigación*, 25(2):47–52. 13
- Instruments, N. (2006). Comunicación serial: Conceptos generales. 18
- McRoberts, M. (2011). Arduino básico. *São Paulo: Novatec*. 15
- Morales, G. (2011). La domótica como herramienta para un mejor confort, seguridad y ahorro energético. *Ciencia e Ingeniería*, pages 39–42. 5
- Moya, J. M. H. and Tejedor, R. J. M. (2004). *Domótica: edificios inteligentes*. Creaciones copyright. 5, 12, 14
- Quintana, C., Reiser, B. J., Davis, E. A., Krajcik, J., Fretz, E., Duncan, R. G., Kyza, E., Edelson, D., and Soloway, E. (2004). A scaffolding design framework for software to support science inquiry. *The journal of the learning sciences*, 13(3):337–386. 17
- Rai, R. (2013). *Socket. IO Real-time Web Application Development*. Packt Publishing Ltd. 17
- Serrano, P. J. M. (2009). Domótica y aplicaciones para el hogar @MISC. 9, 11, 12
- Tamayo, A. (2009). Comunicación serial. *Colombia.[En línea] Disponible en: <http://galaxi0.wordpress.com/el-puerto-serial/>[Consulta: Julio, 2012]*. 18

Apéndice A

Ejemplo de apéndice o anexo

A.1. Sub título primer apéndice o anexo

A.1.1. Sub sección del primer apéndice o anexo