



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CARRERA DE INGENIERÍA EN ELECTRÓNICA

Y REDES DE COMUNICACIÓN

**TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERÍA EN ELECTRÓNICA Y REDES DE COMUNICACIÓN**

**TEMA: PROTOTIPO DE INFRAESTRUCTURA DE COMUNICACIÓN PARA
NOTIFICACIONES GENERADAS POR EL CONSUMO ELÉCTRICO EN UN
HOGAR.**

AUTOR: RICARDO JOSÉ MADERA ROSERO

DIRECTOR: MSC. FABIÁN GEOVANNY CUZME RODRÍGUEZ

IBARRA- ECUADOR

2019



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO			
CÉDULA DE IDENTIDAD:	0401696539		
APELLIDOS Y NOMBRES:	Ricardo José Madera Rosero		
DIRECCIÓN:	Ibarra		
EMAIL:	rjmaderar@utn.edu.ec		
TELÉFONO FIJO:	062615460	TELÉFONO MÓVIL:	0995420717

DATOS DE LA OBRA	
TÍTULO:	PROTOTIPO DE INFRAESTRUCTURA DE COMUNICACIÓN PARA NOTIFICACIONES GENERADAS POR EL CONSUMO ELÉCTRICO EN UN HOGAR.
AUTOR (ES):	Ricardo José Madera Rosero
FECHA: DD/MM/AAAA	08 de marzo de 2019
SOLO PARA TRABAJOS DE GRADO	
PROGRAMA:	<input checked="" type="checkbox"/> PREGRADO <input type="checkbox"/> POSGRADO
TÍTULO POR EL QUE OPTA:	Ingeniero en Electrónica y Redes de Comunicación
ASESOR /DIRECTOR:	Ing. Fabián Geovanny Cuzme Rodríguez, MSc.

CONSTANCIA

El autor manifiesta que la obra objeto de la presente autorización es original y se la desarrolló sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es el titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra a los 08 días del mes de marzo de 2019

EL AUTOR:



Ricardo José Madera Rosero

040169653



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN

MAGISTER FABIÁN CUZME, DIRECTOR DEL PRESENTE TRABAJO DE TITULACIÓN CERTIFICA:

Que, el presente trabajo de Titulación "PROTOTIPO DE INFRAESTRUCTURA DE COMUNICACIÓN PARA NOTIFICACIONES GENERADAS POR EL CONSUMO ELÉCTRICO EN UN HOGAR". Ha sido desarrollado por el señor Ricardo José Madera Rosero, bajo mi supervisión.

Es todo en cuanto puedo certificar en honor a la verdad.



MSc. Fabián Cuzme

DIRECTOR

UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

DEDICATORIA

A mis padres, por ser un referente de virtudes y buenos valores, por ser un ejemplo de trabajo y superación, quienes han guiado el rumbo de mi vida por el mejor camino y me han apoyado en todo momento.

Ricardo José Madera Rosero

UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

AGRADECIMIENTO

Mis más sinceros agradecimientos:

A mis familiares, por el enorme apoyo incondicional brindado a lo largo de este proceso.

A mis amigos, que de una u otra forma animaron y motivaron a terminar este trabajo de investigación.

A “La Plaga” por hacer más llevadera la vida universitaria y demostrar una y mil veces que no importa que tan bajo se pueda caer, siempre levantarse con ganas de superación.

A la Universidad Técnica del Norte, por haber sido un segundo hogar durante todo este tiempo, especialmente a la Carrera de Electrónica y Redes de Comunicación.

A mis profesores, por compartir sus conocimientos y experiencias en virtud de brindar una formación integral a sus estudiantes, de manera especial al Ing. Fabián Cuzme MSc, por ser guía en el desarrollo de este trabajo de titulación.

A mis asesores Ing. Mauricio Domínguez MSc e Ing. Daniel Jaramillo MSc, por su invaluable contribución en la realización del presente proyecto.

Ricardo José Madera Rosero

ÍNDICE

IDENTIFICACIÓN DE LA OBRA	II
CONSTANCIA	¡Error! Marcador no definido.
CERTIFICACIÓN	¡Error! Marcador no definido.
DEDICATORIA	V
AGRADECIMIENTO	VI
ÍNDICE.....	VII
ÍNDICE DE FIGURAS	XI
ÍNDICE DE TABLAS	XIII
RESUMEN	XIV
ABSTRACT	XV
CAPÍTULO I.....	1
1.1 Tema.....	1
1.2 Problema.....	1
1.2.1 Planteamiento del problema.	3
1.3 Objetivos.....	3
1.3.1 Objetivo general.	3
1.3.2 Objetivos específicos.	4
1.4 Alcance	4
1.5 Justificación.....	6
CAPÍTULO II.....	7
2.1 Redes eléctricas inteligentes (<i>Smart Grids</i>).....	7
2.1.1 Protocolos de las redes eléctricas inteligentes	8
2.1.2 Arquitectura del sistema multicapas de una smart grid	15
2.1.3 Aplicaciones en redes locales	18
2.1.5 Tecnologías de comunicación para una red eléctrica inteligente	18
2.1.5 Tecnologías de comunicación para una HAN/BAN/IAN	20
2.2 Consumo eléctrico	23
2.2.1 Consumo eléctrico en modo de espera	23
2.2.2 Aparatos electrónicos que más consumen energía en modo de espera.	24
2.2.3 Aparatos eléctricos que más consumen energía en modo de funcionamiento.	24
2.2.4 Smart Meters y Smart Grids en otros países.	25
2.3 IoT (Internet de las cosas)	28
2.3.1 Modelos de comunicación de la Internet de las Cosas	29
2.4 Sensores de corriente	32

2.4.1 Sensor de efecto Hall	33
2.4.2 Bobina Rogowski	33
2.4.3 Resistencia shunt	34
2.4.4 Transformador de corriente	34
2.5 Computación en la Nube	35
2.5.1 Modelos de implementación de <i>Cloud Computing</i>	36
2.5.2 Modelos de Servicio de <i>Cloud Computing</i>	37
CAPÍTULO III	39
3.1 Metodología.....	39
3.1.1 Modelo en V	39
3.1.2 Estándar ISO / IEC / IEEE 29148: 2011	41
3.2 Análisis	41
3.2.1 Situación Actual.....	41
3.2.2 Encuesta	42
3.2.3 Descripción general del sistema propuesto	45
3.3 Requerimientos del sistema propuesto	46
3.3.1 Requerimientos de Stakeholders	47
3.3.2 Requerimientos del Sistema	47
3.3.3 Requerimientos de Arquitectura	48
3.3.4 Requerimientos de Software	49
3.4 Selección de Hardware y Software.....	51
3.4.1 Selección de Hardware	51
3.4.2 Selección de Software	55
3.5 Recursos.....	61
3.5.1 Recursos Humanos	61
3.5.2 Recursos Económicos	62
3.5.3 Recursos Tecnológicos.....	62
3.6 Diseño del sistema	64
3.6.1 Diagrama de bloques del sistema	64
3.6.2 Diagrama de flujo del sistema.....	65
3.6.3 Diagrama de conexión de los componentes electrónicos	67
3.6.4 Servidor XAMPP.....	68
3.6.5 Pruebas iniciales	69
3.6.6 Diseño del circuito impreso	74
CAPÍTULO IV	76
4.1 Implementación.....	76

4.1.1 Elaboración de los circuitos impresos	77
4.2 Pruebas del módulo electrónico	79
4.2.1 Portabilidad	79
4.2.2 Microcontrolador	79
4.2.3 ESP-01S (WiFi)	80
4.2.4 Funcionamiento continuo y fidelidad de las mediciones de consumo eléctrico ...	81
4.3 Pruebas de la Plataforma de Administración y Gestión	82
4.3.1 Lecturas del gestor de bases de datos	82
4.3.2 Interfaz de acceso a la página web	83
4.3.3 Página Web para control de la plataforma de administración y gestión	84
4.3.4 Acciones de Alerta	87
4.4 Mediciones del contador eléctrico	87
4.4.1 Mediciones sin implementar el prototipo	87
4.4.2 Mediciones de consumo eléctrico con la implementación del prototipo	88
4.5 Vida útil del prototipo.....	89
4.6 Costo de componentes electrónicos importados.....	90
4.7 Relación costo beneficio del prototipo	91
CONCLUSIONES.....	94
RECOMENDACIONES	96
REFERENCIAS	97
ANEXOS	101
Anexo A. Modelo de encuesta y tabulación de resultados	101
De la encuesta presentada y efectuada a 39 familias, se obtienen los siguientes resultados.	102
Anexo B. Instalación de librerías para Arduino y ESP-01S.....	105
Instalación de Librería WiFiESP8266	105
Instalación de Librería PubSubClient.h	107
Anexo C. Desarrollo de la Plataforma de Administración y Gestión en AWS.	108
Anexo D. Acceso por SSH a la instancia creada.....	112
Anexo E. Acceso mediante escritorio remoto a la instancia.	115
Anexo F. Instalación, inicio y pruebas con el Broker Mosquitto	119
Anexo G. Instalación de Servidor XAMMP	120
Anexo H. Creación de base de datos con el gestor phpMyAdmin	125
Anexo J. Mediciones con y sin implementación de prototipo.....	130
Anexo K. Programación	133
Programación de Arduino Nano	133

Programación de WiFi ESP-01S	141
Scripts de RHEL	144
Anexo L. Manual de Usuario	155
Anexo M. Manual de Administrador.....	159

ÍNDICE DE FIGURAS

Figura 1. Topología PRIME	9
Figura 2. Arquitectura Meters and More	10
Figura 3. Modelo de capas de DLMS/COSEM.....	11
Figura 4. Protocolo G3.PLC y modelo OSI.....	12
Figura 5. Resumen de protocolos de Smart Grids	14
Figura 6: Estándares IEC/IEEE para Smart Grids.....	14
Figura 7. Arquitectura del sistema multicapas de una red inteligente.....	15
Figura 8. Tecnologías de almacenamiento de energía.....	16
Figura 9. Requerimientos de velocidad de datos y rango de cobertura para una smart grid.....	17
Figura 10. Ejemplo de comunicación dispositivo a dispositivo	29
Figura 11. Ejemplo de comunicación dispositivo a la nube	30
Figura 12. Ejemplo de comunicación dispositivo a puerta de enlace.....	31
Figura 13. Ejemplo de comunicación back-end	32
Figura 14. Sensor de efecto Hall	33
Figura 15. Bobina Rogowsky	34
Figura 16. Sensor Shunt.....	34
Figura 17. Representación esquemática del transformador.....	35
Figura 18. Modelos de servicios ofrecidos en Cloud Computing.	38
Figura 19. Esquema del modelo en V.....	40
Figura 20. Protocolos de comunicación entre los subsistemas: módulo electrónico y plataforma de administración y gestión.....	46
Figura 21. Arduino Nano.....	52
Figura 22. Módulo wifi ESP01S.....	54
Figura 23. Sensor de corriente ACS712	55
Figura 24. Modelo de publicación y suscripción de MQTT para los sensores de IoT ...	61
Figura 25. Diagrama de bloques del sistema.....	65
Figura 26. Diagrama de flujo del sistema.....	66
Figura 27. Conexión de los componentes electrónicos	68
Figura 28. Valores enviados por consola mediante protocolo MQTT	70
Figura 29. Valores '0' y '3' enviados por MQTT	71
Figura 30. Valores '0' y '2' enviados por MQTT	72
Figura 31. Valores '1' y '3' enviados por MQTT	72
Figura 32. Valores '1' y '2' enviados por MQTT	72
Figura 33. Pruebas de envío de datos de los sensores de corriente ASC-712	73
Figura 34. Librería Amchart con línea de tiempo	74
Figura 35. Diagrama de conexiones del circuito impreso	74
Figura 36. Diagrama de pistas del circuito impreso	75
Figura 37. a) Tomacorriente doble para cajetín. b) Caja metálica	77
Figura 38. a) Colocación de la placa impresa en percloruro férrico. b) Pistas de material conductor	77
Figura 39. Elementos electrónicos soldados a la placa de cobre.....	78
Figura 40. Módulo electrónico terminado	78

Figura 41. Espacio de almacenamiento y memoria dinámica ocupados en el microcontrolador.	79
Figura 42. Espacio de almacenamiento y memoria dinámica ocupados en el módulo WiFi.	80
Figura 43. Comparación de la medición de sensor ASC712 y multímetro.	81
Figura 44. Valores recibidos por parte del sensor 1 y almacenados en la base de datos	82
Figura 45. Valores recibidos por parte del sensor 2 y almacenados en la base de datos	83
Figura 46. Base de datos para usuarios y contraseñas.	83
Figura 47. Interfaz de login	84
Figura 48. Página Web de monitoreo de consumo eléctrico	85
Figura 49. Potencia nominal del Horno Microondas.....	85
Figura 50. Menú de ingreso de nuevos equipos, potencias nominales, frecuencia de medición y horario de apagado.....	86
Figura 51. Interfaz de flujo de energía eléctrica.....	86
Figura 52. Registro de eventos de equipos asociados al sistema realizado	86
Figura 53. Mensaje de alerta de la página web.....	87
Figura 54. Correo electrónico de alerta	87
Figura 55. Consumo eléctrico durante un mes, sin implementar el prototipo.....	88
Figura 56. Consumo eléctrico durante un mes, con implementación del prototipo.	88

ÍNDICE DE TABLAS

Tabla 1. Tecnologías de comunicación de las Smart Grids.....	19
Tabla 2. Consumo en Standby	24
Tabla 3. Consumo en funcionamiento.....	25
Tabla 4. Requerimiento del sistema y abreviaturas	47
Tabla 5. Requerimientos de Stakeholders	47
Tabla 6. Requerimientos del Sistema	48
Tabla 7. Requerimientos de Arquitectura.....	48
Tabla 8. Requerimientos de Software y abreviaturas	49
Tabla 9. Requerimientos de Infraestructura como Servicio	49
Tabla 10. Requerimientos del Sistema Operativo	50
Tabla 11. Requerimientos del Protocolo de IoT.....	50
Tabla 12. Selección de tarjeta programable.....	52
Tabla 13. Selección del módulo WiFi	53
Tabla 14. Selección del sensor de corriente	54
Tabla 15. Selección de la Infraestructura como Servicio	56
Tabla 16. Comparación de compromisos y descuentos entre proveedores de servicio de nube.	57
Tabla 17. Selección del Sistema Operativo	58
Tabla 18. Comparación entre los sistemas operativos CentOS y RHEL.....	58
Tabla 19. Selección del Protocolo de IoT.....	59
Tabla 20. Recursos Humanos presentes en el proyecto.....	62
Tabla 21. Recursos Económicos presentes en el proyecto.	62
Tabla 22. Recursos Tecnológicos presentes en el proyecto.	63
Tabla 23. Significado del acrónimo XAMPP	69
Tabla 24. Pruebas iniciales del prototipo.....	69
Tabla 25. Estado de los relevadores 1 y 2 controlados por MQTT.....	71
Tabla 26. Vida útil de los componentes principales del módulo electrónico.....	89
Tabla 27. Registro de precios de componentes electrónicos importados de China.....	90

ÍNDICE DE ECUACIONES

Ecuación 1. Determinación del tamaño de muestra para una encuesta	43
---	----

RESUMEN

Las redes eléctricas inteligentes también conocidas como Smart Grids, integran tecnologías de información y comunicación a su infraestructura, de este modo se obtiene una comunicación bidireccional entre la empresa proveedora del servicio y los consumidores. A nivel del hogar también es posible llevar un control del consumo energético, integrando tecnologías de medición, envío de datos, y servicios de computación.

En la presente investigación se documenta la realización de un prototipo de infraestructura de comunicación que toma, procesa y muestra información acerca del consumo eléctrico de los equipos presentes en un hogar y permite automatizar eventos, como encendido y apagado de los equipos, y envío de notificaciones sobre consumo eléctrico irregular. Por tanto, la finalidad de este proyecto es el ahorro energético mediante la utilización responsable del suministro de energía eléctrica y se enmarca en la tendencia actual denominada Smart Grids.

El prototipo está formado por dos subsistemas: un Módulo Electrónico encargado del muestreo y envío de datos, conexión con la Internet, ejecución de actuadores de acuerdo a horarios programados, entre otras funciones; y una Plataforma de Administración y Gestión que muestra la información enviada por el primer subsistema y permite la interacción de ambos mediante una página web. Finalmente se muestran las pruebas realizadas y los resultados obtenidos.

ABSTRACT

Smart Grids integrate information and communication technologies into their infrastructure, thus obtaining bidirectional communication between the service provider and consumers. At the household level it is also possible to control energy consumption, integrating measurement technologies, data transmission, and computer services.

The present investigation documents the realization of a communication infrastructure prototype that takes, processes and displays information about the electrical consumption of the equipment present in a home and allows to automate certain events, such as switching the equipment on and off, and sending notifications on irregular electrical consumption. Therefore, the purpose of this project is the energy saving through a responsible consumption of electricity supply, and it's part of the current trend called Smart Grids.

The prototype consists of two subsystems: An Electronic Module responsible for sampling and sending data, connection to the Internet, execution of actuators according to programmed schedules, among others; and an Administration and Management Platform that shows the information sent by the first subsystem and allows the interaction of both through a web page. Finally, the tests performed, and the results obtained are shown.

CAPÍTULO I

ANTECEDENTES

Este capítulo contiene las bases del desarrollo del proyecto, qué busca conseguir y sobre qué está sustentado, al igual que sus limitaciones.

1.1 Tema.

PROTOTIPO DE INFRAESTRUCTURA DE COMUNICACIÓN PARA NOTIFICACIONES GENERADAS POR EL CONSUMO ELÉCTRICO EN UN HOGAR.

1.2 Problema

La actividad humana contribuye día a día al incremento del efecto invernadero, lo cual incide gravemente en el cambio climático a nivel mundial, y a su vez puede afectar peligrosamente a los ecosistemas naturales y a la sociedad. (Aragón, 2015). Yvo de Boer, ex secretario de la Convención Marco de las Naciones Unidas sobre el Cambio Climático manifestó: “La eficiencia energética es el medio más promisorio para reducir las emisiones de gases de efecto invernadero a corto plazo”, dicha eficiencia consiste en lograr un aprovechamiento óptimo de la energía, manteniendo el mismo nivel de calidad de vida. Asimismo, se considera que el consumo energético será el principal problema de la humanidad en los próximos 50 años, así lo afirmaba el Premio Nóbel Richard E. Smalle, ya en 2003. (Huang, 2013).

En un hogar se puede llegar a desperdiciar entre un 7% y 11% de energía eléctrica, éste fenómeno ocurre debido a que la mayoría de electrodomésticos y dispositivos electrónicos funcionan con corriente continua de bajo voltaje, sin embargo, la corriente suministrada en la red eléctrica es alterna de alto voltaje, por este motivo se requiere de un transformador para adquirir los niveles adecuados para cada dispositivo, y ya que el transformador funciona todo el tiempo, incluso si el equipo está apagado, se produce un consumo adicional e innecesario de electricidad, denominado consumo en modo de espera o en standby. Para evitar este desperdicio de energía eléctrica se debería desconectar los dispositivos cuando no estén siendo utilizados, no obstante, realizar esta tarea manualmente puede resultar tedioso para el usuario. (Arroyo, 2016).

En este sentido, se tiene otro problema común, que consiste en la utilización de electrodomésticos con más de 10 años de antigüedad, lo cual aumenta hasta en 20% el valor de la factura de energía eléctrica, ya que requieren mayor cantidad de electricidad para funcionar, aquel valor adicional depende del tiempo de uso y del mantenimiento brindado a los equipos. Esto se debe en gran parte, a que el usuario no es consciente del consumo eléctrico real de sus dispositivos, ya que no tiene forma de saberlo al no contar con un sistema que le permita monitorizar constantemente a los mismos. (Pérez, 2015).

En los hogares ecuatorianos, según el documento técnico de Información Ambiental en hogares del 2016, se registró un valor de consumo mensual promedio de electricidad de \$19.34, cifra que se ha incrementado en \$3.53 desde 2012. De igual manera el documento menciona un cambio de hábitos, en virtud de favorecer a la eficiencia en el uso de electricidad, como: desenchufar los cargadores al terminar la carga del equipo, utilizar dispositivos ahorradores de energía, apagar las luces al salir de la habitación, entre otros. (INEC, 2016).

De acuerdo con un estudio realizado por el Lawrence Berkely National Laboratory del Departamento de Energía de Estados Unidos, entre los equipos que más consumen electricidad en standby, se tiene a los equipos de sonido con alrededor de 14.4 vatios/h, y los hornos microondas con 3,8 vatios/h los cuales pueden subir hasta los 25 vatios/h con la puerta abierta. (Guardián, 2017). Por otro lado, las planchas para ropa tienen un alto consumo energético mientras están funcionando, y pueden sobrepasar los 1000 vatios/h si se dejan conectadas, dependiendo del modelo, así lo menciona el doctor en ingeniería nuclear Luis Juanicó. (Clarín, 2015).

1.2.1 Planteamiento del problema.

¿Cómo evitar el consumo eléctrico en modo de espera de manera automática?

¿Cómo verificar si un dispositivo eléctrico o electrónico está consumiendo una cantidad excesiva de energía eléctrica?

¿Qué tecnologías permiten implementar un módulo electrónico que realice las tareas anteriores y pueda ser controlado mediante una página web?

1.3 Objetivos.

1.3.1 Objetivo general.

Desarrollar un prototipo de infraestructura de comunicación para administración y gestión de datos generados por el consumo energético en un hogar que permita contribuir al uso eficiente de este recurso.

1.3.2 Objetivos específicos.

- Establecer requerimientos para la infraestructura de comunicación que permita tener una bidireccionalidad de datos entre la parte electrónica y la plataforma de administración y gestión.
- Diseñar un módulo electrónico que permita enviar información referente al consumo eléctrico hacia la plataforma de administración y gestión mediante una tecnología de comunicaciones y que controle el flujo de energía hacia un dispositivo.
- Diseñar una plataforma de almacenamiento para la visualización de los datos recolectados y gestión de los dispositivos involucrados.
- Realizar las pruebas de funcionamiento correspondientes de la infraestructura propuesta para administración, gestión y recolección de datos.

1.4 Alcance

Se definirán los requerimientos de la electrónica que mejor se adapten al desarrollo del prototipo en cuanto a hardware y software, tomando en cuenta la operatividad y adaptabilidad de los componentes presentes en el mercado con el sistema a realizar, mediante el estándar IEEE 29148.

El diseño del módulo de adquisición de datos procurará ser lo más compacto y económico posible de manera que sea funcional, y la administración y gestión se realizará en los tres aparatos mencionados en la sección del problema (equipo de sonido, microondas y plancha para ropa). El módulo constará de una placa microprocesadora de código abierto y de bajo consumo, la cual será alimentada mediante la red eléctrica del hogar, y se encargará de medir el consumo eléctrico generado en cada uno de los

conectores del módulo electrónico por medio de sensores de corriente, esta información será enviada mediante tecnología de comunicación inalámbrica hacia la plataforma diseñada, en donde se almacenará un registro del consumo del dispositivo conectado, y en caso de sobrepasar un umbral se generará una alerta. El intervalo de tiempo óptimo para el envío de datos se determinará en la experimentación de la investigación propuesta. Además, el prototipo estará programado de acuerdo a horarios definidos para cortar y reconectar el suministro de energía eléctrica de manera automática por medio de un relevador, para evitar el consumo energético innecesario por modo de standby. Para llevar un registro del consumo de energía eléctrica se utilizará un contador ubicado a continuación del medidor de la empresa eléctrica. De este modo se evidenciaría el consumo eléctrico con y sin la utilización del sistema diseñado, al cabo de un tiempo medido en semanas.

La plataforma de administración y gestión de datos se implementará en una infraestructura basada en la nube, ya que la *IaaS-Cloud Computing* permite una mayor eficiencia de costos y una implantación flexible, de este modo se proporciona acceso a servicios informáticos, sin depender de los sistemas físicos o de su ubicación real, únicamente se requiere acceso a Internet. (Relica, 2014). Para esto, se analizarán las ofertas de servidores con mayor presencia en la nube, así como sus características para determinar el que mejor se adapte a los requerimientos del sistema propuesto.

La plataforma será accesible por medio de una interfaz web que permita llevar un control adecuado de los datos generados por el módulo electrónico y únicamente alertará sobre un consumo excesivo de energía en el componente que se encuentre conectado al dispositivo de monitoreo, más no indicará la causa exacta del mismo (expiración de tiempo de vida útil, falla en algún componente del aparato, instalación eléctrica incorrecta, entre otras causas.).

Se realizarán pruebas de funcionamiento del sistema en un entorno controlado, integrando el módulo electrónico y la plataforma de administración y gestión.

1.5 Justificación

En la actualidad, la arquitectura de la información basada en Internet facilita el intercambio de bienes y servicios entre los equipos conectados a la red. De esto se deriva un paradigma conocido como IoT (Internet de las cosas), lo cual resulta de una combinación de sensores y actuadores que proporcionan y reciben información, mediante redes bidireccionales que transmiten los datos para ser utilizados por múltiples servicios para los usuarios. (Salazar & Silvestre, 2016)

Se gestiona mejor lo que somos capaces de medir y visualizar, así lo menciona Mary Burgoon, Gerente de Desarrollo de Mercado de CPG Industry. Actualmente la tecnología hace posible que los dispositivos pueden proveer información acerca de su consumo eléctrico, de esta manera, los usuarios se vuelven un ente proactivo en cuanto al uso adecuado de este recurso. (Burgoon, 2017).

Este proyecto busca brindar una solución adecuada en beneficio de la eficiencia energética, al proveer herramientas que permitan controlar el consumo eléctrico de determinados equipos en un hogar, y a su vez, evite el desperdicio de energía, favoreciendo así, la economía del usuario y la conservación del planeta.

CAPÍTULO II

FUNDAMENTO TEORICO

Este capítulo contiene información bibliográfica de los conceptos más relevantes que intervienen en el desarrollo del presente proyecto. Se revisan fuentes certificadas como artículos, libros, periódicos, noticieros, entre otros; y se presentan de manera organizada, concisa y debidamente referenciada, de acuerdo con los temas y subtemas a tratar.

2.1 Redes eléctricas inteligentes (*Smart Grids*)

La electricidad, pasó de ser una primicia tecnológica que despertaba curiosidad y admiración entre los habitantes de las grandes urbes a convertirse, medio siglo después, en una fuente de energía multifacética (luz, fuerza, calor), que se identificaba cada vez más con el progreso y la vida moderna. La electricidad era la tecnología de aplicación general por excelencia, como ninguna otra hasta el advenimiento de la informática en las décadas posteriores. (Tafunell, 2018)

Una red eléctrica inteligente se define, según la Unión Europea, como: “una red eléctrica que puede integrar de manera eficiente el comportamiento y las acciones de todos los usuarios conectados a ella (generadores, consumidores o ambos), para garantizar un sistema de energía económicamente eficiente, sostenible con bajas pérdidas, y altos niveles de calidad y seguridad del suministro”.

Por otro lado, el Departamento de Energía de los Estados Unidos (DOE) realizó un análisis de los sistemas de una red eléctrica y determinó las áreas donde se deben efectuar

mejoras para considerarse como una red eléctrica inteligente, en este sentido (Falvo, Martirano, Sbordone, & Bocci, 2013) propone lo siguiente:

- **Confiabilidad:** Reducción de la frecuencia de interrupción, calidad adecuada de alimentación eléctrica y un mejor servicio al cliente.
- **Seguridad:** Reducida vulnerabilidad a ataques y eventos naturales.
- **Economía:** Tendencia a la baja sobre los futuros precios de la electricidad.
- **Eficiencia:** Conservación de energía por los usuarios, reducidas pérdidas en el sistema, reducciones en operaciones, mantenimiento y gastos de capital.
- **Respeto al medio ambiente:** Habilitación de energías renovables intermitentes.
- **Seguridad:** Protección para los trabajadores y el público.

2.1.1 Protocolos de las redes eléctricas inteligentes

Los protocolos existentes acerca de redes eléctricas inteligentes no son tan abundantes como en otros entornos de la industria, gracias a la colaboración de distribuidores de energía, fabricantes y desarrolladores, y su estandarización. Entre los protocolos más relevantes según (CERTSI, 2017) se tiene: PRIME, Meters and More, DLSP/COSEM, G3-PLC, OSGP.

PRIME (Por sus siglas en inglés PowerLine Intelligent Metering Evolution), este protocolo se encarga de implementar funciones correspondientes a capa física y capa de enlace de acuerdo con el modelo OSI.

En la capa física, el protocolo PRIME utiliza tecnología PLC (Power Line Communication), en la banda de frecuencia llamada CENELEN-A que va de 3 a 95 KHz,

pero puede extenderse hasta los 500KHz en la versión PRIME 1.4 (última versión), se utiliza una modulación OFDM (Multiplexación por División de Frecuencias Ortogonales).

En la capa de enlace se conforma una estructura de red en árbol, con dos tipos de nodos para la red: Base y Servicio. De acuerdo con la Figura 1, existe solamente un Nodo Base en cada subred, el cual actúa como ente principal en la comunicación, y donde los demás nodos se asocian paulatinamente. Mientras que el Nodo de Servicio, inicialmente se encuentra desconectado, éste se encarga de mantener la conexión en la subred y realizar enrutamiento hacia otros nodos.

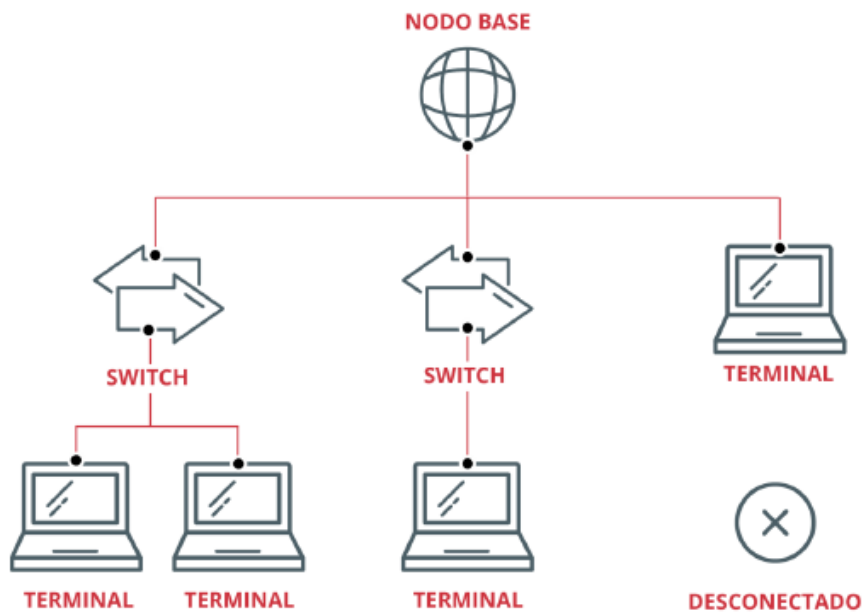


Figura 1. Topología PRIME
Fuente: (CERTSI, 2017)

METERS AND MORE, es un protocolo propietario desarrollado por la empresa energética italiana ENEL; La Figura 2 muestra la cobertura sobre la pila del modelo OSI y su aplicación sobre diversos medios de transmisión. Este protocolo añade varias características de seguridad en su diseño:

- Cifrado mediante claves AES de 128 bits.
- Autenticación en base a claves simétricas.
- Protección frente a ataques de retransmisión.
- Comprobación de integridad de mensaje.
- Claves individuales para cada contador, con control de acceso (lectura/escritura).
- Protección extremo-extremo.

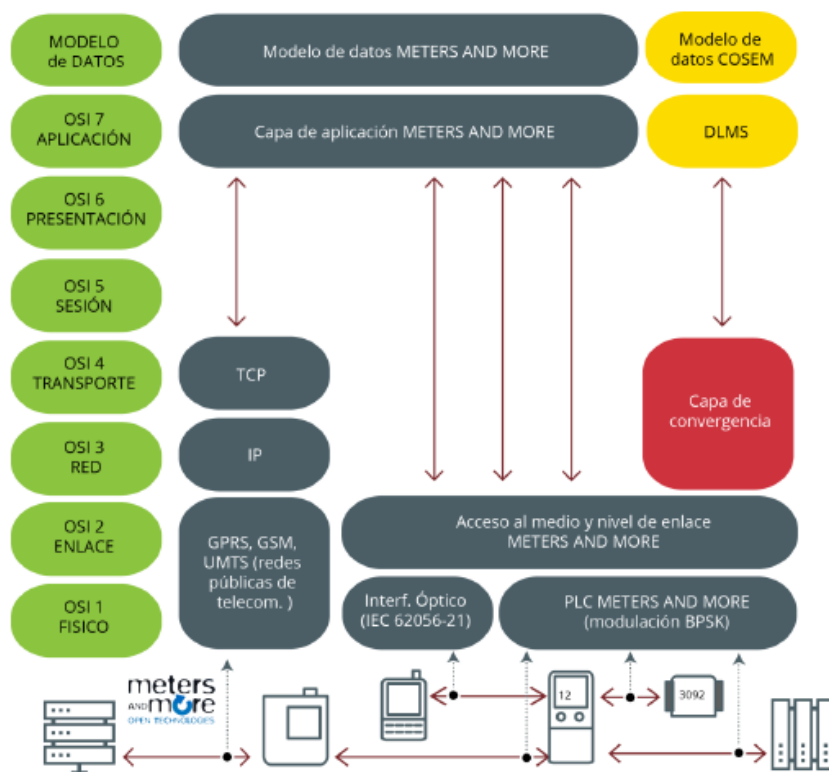


Figura 2. Arquitectura Meters and More
Fuente: (CERTSI, 2017)

DLMS/COSEM, es un protocolo a nivel de capa aplicación, es decir, desde la capa 4 a la 7 de acuerdo con el modelo OSI, tal como se muestra en la Figura 3. Este protocolo es regulado por la norma IEC 62056. (CERTSI, 2017) y fue desarrollado para operar conjuntamente con el protocolo PRIME, así como con el protocolo Meters and More. Su nombre está formado por dos grupos de siglas:

DLSM: “Device Language Message Specificaiton”, un modelo abstracto de entidades de comunicación.

COSEM: “COMpanion Specification for Energy Metering”, fija los parámetros, basados en estándares, para intercambiar información con los contadores de energía.



Figura 3. Modelo de capas de DLMS/COSEM
Fuente: (CERTSI, 2017)

G3-PLC, es un protocolo abierto e internacional para redes inteligentes desarrollado por Sagem, ERDF y Maxim. Opera por debajo de los 500KHz y comprende las capas: física y de enlace (MAC), donde es soportado por OFDM y 6LoWPAN como capa de adaptación para paquetes IPv6, como se muestra en la Figura 4. Este protocolo es utilizado especialmente en infraestructuras con nodos a gran escala, y entre sus características más importantes se encuentran:

- Robustez y amplio rango de frecuencias de comunicación que proporcionan una gran ventaja a la hora de instalar dispositivos inteligentes que envíen datos a los concentradores.
- Su diseño permite una comunicación punto a punto mediante IPv6.

- Utiliza las bandas definidas por ARIB (Association of Radio Industries and Businesses), FCC (Comisión Federal de Comunicaciones) y CENELEC (Comité Europeo de Normalización Electrotécnica).
 - ARIB define la frecuencia en Asia de la banda para PLC, entre 10 y 450 kHz.
 - La sección 15 de FCC define la frecuencia entre 10 y 490 kHz, para PLC en Norte América
 - CENELEC EN50065-1 define el rango para PLC en Europa, en bandas de baja frecuencia (bandas A, B, C, D).

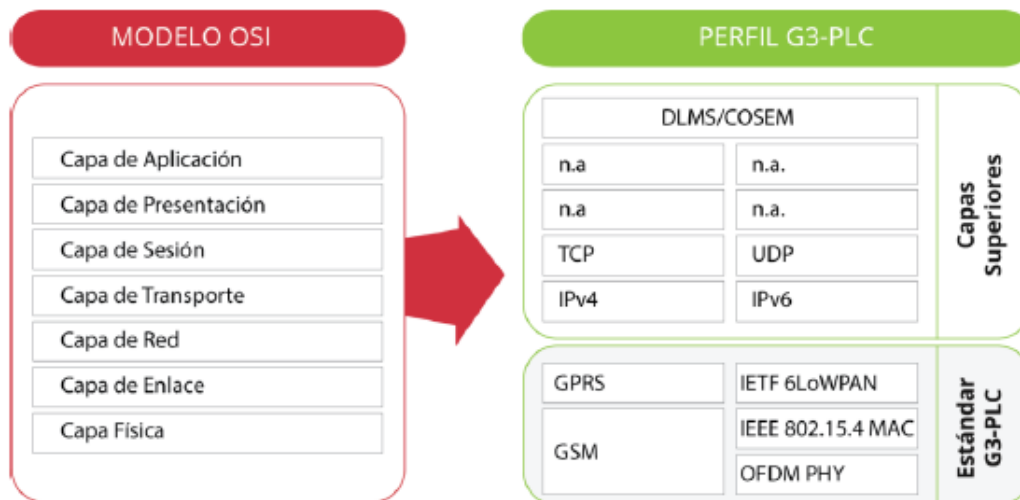


Figura 4. Protocolo G3.PLC y modelo OSI
Fuente: (CERTSI, 2017)

OSGP, (protocolo abierto de Smart Grid). Fue desarrollado por OSGP Alliance, y se aplica en varios países, en proyectos de medición inteligente a gran escala. Es un protocolo ampliamente utilizado y actualmente existen más de 100 millones de dispositivos desplegados alrededor del mundo. OSGP trabaja en una frecuencia que va desde 9 kHz hasta 95 kHz, y provee una comunicación segura entre nodos de control y medidores, mediante una capa de control independiente del medio. (CERTSI, 2017)

OSGP está basado en los siguientes estándares abiertos: ETSI GS OSG 001 (aborda lo referente a Capa de aplicación), ISO/IEC 14908-1 (cubre Capa de transporte) y ETSI TS 103 908 (cubre Capa física).

OSGP incluye medidas de seguridad para proteger la privacidad de sus usuarios, utilizando cifrado de datos. De acuerdo con lo señalado por (CERTSI, 2017), OSGP presenta cuatro características de seguridad, que se detallan a continuación:

- Algoritmo RC4: Sistema de cifrado que convierte el texto plano en texto cifrado bit a bit. El algoritmo RC4 en OSGP es similar al algoritmo utilizado en WEP y tiene debilidades similares.
- Función de respuesta: Usa un mensaje de autenticación como función de respuesta.
- Secure Broadcast: Utiliza este mecanismo para enviar actualizaciones de firmware.
- Claves: Utiliza claves de sesión para el cifrado de mensajes y una clave maestra para autenticarse.

La Figura 5 muestra un resumen comparativo de los protocolos de redes eléctricas inteligentes (Smart Grids) mencionados anteriormente.

	Protocolo	PRIME	DLMS/COSEM	Meters and More	G3-PLC	OSGP
Aspectos generales	Tipo de estándar	Abierto	Abierto	Propietario	Abierto	Abierto
	Medio de transmisión	PLC	Ethernet	PLC Ethernet Serie	PLC Ethernet	PLC Ethernet
	Región de uso	España	España	Italia España	Francia	Norte de Europa
	Compatibilidad	DLMS/COSEM	PRIME M&M G3-PLC OSGP	DLMS/COSEM	DLMS/COSEM	DLMS/COSEM G3-PLC
Seguridad	Cifrado	Perfiles 1 y 2	Niveles Low y High	SI	SI	SI
	Autenticación	Perfiles 1 y 2	Niveles Low y High	SI	SI	SI
Capas implementadas por el protocolo (nivel OSI)	1	X		X	X	X
	2	X		X	X	X
	3			X	X	X
	4		X	X	X	X
	5		X		X	X
	6		X		X	X
	7		X	X	X	X
Recomendaciones de seguridad		Utilizar el perfil de seguridad 1 o 2	Utilizar High Level security Sobre TCP/IP realizar filtrado en el puerto 4059	En despliegues conjuntos con DLMS/COSEM aplicar la seguridad en ambos protocolos	Utilizar autenticación vía RADIUS	Utilizar medidas de cifrado adicionales

Figura 5. Resumen de protocolos de Smart Grids
Fuente: (CERTSI, 2017)

Respecto a esto (Raddatz, 2017) menciona que la Comisión Electrotécnica Internacional (IEC) y el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) han provisto varios estándares que se muestran en la Figura 6, los cuales se aplican en varias regiones del mundo, tales como: Europa, Asia, África, Estados Unidos, América del Sur, entre otros.

IEC/IEEE and other Standards	Client/Master Deployment	Server/Slave Deployment	Communication Medium	Distribution
IEC 60870-5-104	Control centre and grid management system. Also used for local IED integration in substation	Substation and smart grid devices: Communication with control centre	Ethernet TCP/IP	Europe, Asia, Middle East, Africa
IEC 60870-5-101	Mainly in substation for legacy IED integration		Serial RS232	Mainly replaced by IEC 60870-5-104
IEC 60870-5-103	Substation for protection IED integration		Serial RS485	Mainly replaced by IEC 61850
DNP3/IEEE 1815	Control centre and grid management system. Also used for local IED integration in substation	Substation and smart grid device: Communication with control centre or substation control system	Serial RS232/485 and Ethernet TCP/IP	USA, South America, Australia, United Kingdom
IEC 61850	Substation and smart grid devices for local IED integration. Increasingly in smart grid management system for grid management	Protection relays, remote terminal units, smart grid controllers, energy meters: Communication and management link to control centre or substation control system	Ethernet TCP/IP and Multicast Ethernet packets	Europe, Asia, Middle East, Africa, Canada and increasingly in Australia and South America
Modbus	Mainly in substation for legacy IED integration		Serial RS232/485 and Ethernet TCP/IP	Worldwide
M-Bus (based on IEC 60870-5)	Electrical meter integration		M-Bus serial and M-Bus wireless	Worldwide

Figura 6: Estándares IEC/IEEE para Smart Grids
Fuente: (Raddatz, 2017)

2.1.2 Arquitectura del sistema multicapas de una smart grid

Este tipo de red se considera una plataforma interactiva, que contiene una capa de sistema de energía, una capa de control, una capa de comunicación, una capa de seguridad y una capa de aplicación, tal como se muestra en la Figura 7. Esta arquitectura describe la interacción entre los distintos protocolos a utilizar en el despliegue de una red inteligente.

Smart Metering and Grid Applications				Customer Applications				Application Layer
Authentication, Access Control, Integrity Protection, Encryption, Privacy								Security Layer
Cellular, WiMAX, Fiber Optic			PLC, DSL, Coaxial Cable, RF Mesh			Home Plug, ZigBee, WiFi, Z-Wave		Communication Layer
WAN			NAN/FAN			HAN/BAN/IAN		
PMUs	Cap Banks	Reclosers	Swithes	Sensors	Transformers	Meters	Storage	Power Control Layer
Power Transmission/Generation			Power Distribution			Customer		Power System Layer

Figura 7. Arquitectura del sistema multicapas de una red inteligente
Fuente: (Kuzlu, Pipattanasomporn, & Rahman, 2014)

A continuación, según (Kuzlu, Pipattanasomporn, & Rahman, 2014), se describe de manera general la función que realiza cada una de las capas:

Capa de sistema de energía

Uno de los propósitos de las *Smart grids* es la capacidad de aprovechar al máximo la generación de energía mediante fuentes renovables, y a su vez reducir la dependencia de los combustibles fósiles, sin embargo, no se puede prescindir totalmente de estos combustibles debido a las fluctuaciones de las condiciones ambientales (viento, sol,

lluvias, sequías, entre otras). Es importante contar con un sistema de almacenamiento de energía local, para combatir la intermitencia de la generación energética de fuentes renovables. Respecto a esto, en la Figura 8 se indican las tecnologías de almacenamiento de energía disponibles.

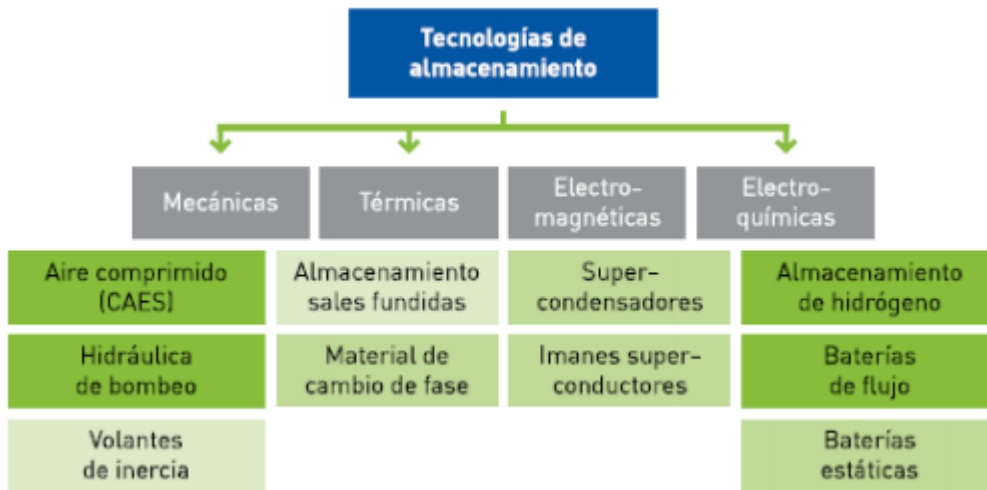


Figura 8. Tecnologías de almacenamiento de energía
Fuente: (IDAE, 2011)

Capa de control

Esta capa se encarga de la monitorización de la red inteligente, control y funciones de gestión. Controla interrupciones en el servicio, posee sistemas de medición automática remota, sistemas de control supervisado y adquisición de datos (SCADA), además detecta fallas en alimentadores de distribución y eventos climáticos en las líneas de transmisión,

Capa de comunicación

La capa de comunicación es uno de los elementos más críticos que da soporte a las aplicaciones de red inteligente. En la Figura 9, se muestra la clasificación de acuerdo con

la velocidad de datos y rango de cobertura, de este modo se tiene tres tipos de entornos de redes:

- Red de área de hogar (HAN), red de área de edificio (BAN) y red de área industrial (IAN).
- Red de área de barrio (NAN) y red de área de campo (FAN).
- Red de área amplia (WAN).

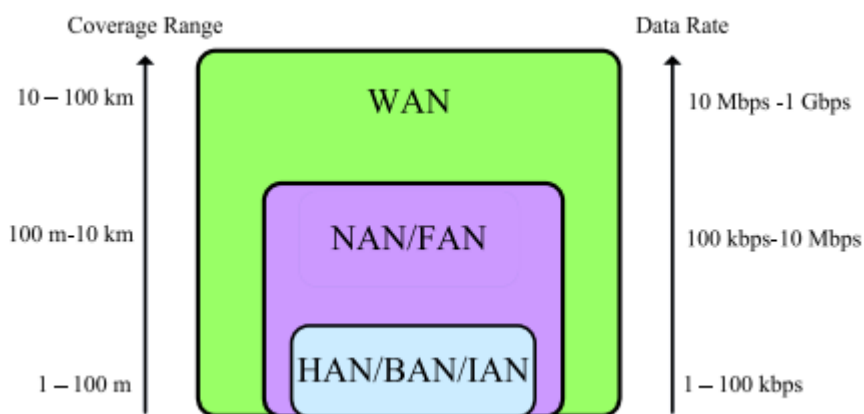


Figura 9. Requerimientos de velocidad de datos y rango de cobertura para una smart grid.
Fuente: (Kuzlu, Pipattanasomporn, & Rahman, 2014)

Capa de seguridad

Una *Smart grid* es capaz de detectar y anunciar un fallo potencial incluso antes de que suceda, facilitando una reconfiguración inmediata y evitando cortes del servicio eléctrico. También deben contar con un sistema de protección sofisticado, de modo que pueda mitigar ataques como: manipulación de datos, acceso no autorizado, malware, entre otros.

Capa de aplicación

Ofrece aplicaciones de redes inteligentes a clientes y empresas, tales como: facturación, control de respuesta de demanda, control de averías, monitoreo de carga, mercados energéticos en tiempo real y una gama de servicios adicionales.

2.1.3 Aplicaciones en redes locales

Las redes locales HAN, BAN, IAN, se encuentran en el extremo del cliente en la arquitectura de red. De acuerdo con (Kuzlu, Pipattanasomporn, & Rahman, 2014), estas redes dan soporte a las comunicaciones entre dispositivos eléctricos y electrónicos, vehículos eléctricos, y otros equipos en las instalaciones del cliente.

HAN proporciona comunicaciones para electrodomésticos y equipos que son capaces de enviar y recibir señales para un medidor inteligente, pantallas *in-home* (IHDS) y/o sistemas de gestión de energía en el hogar (HEM). Se tienen aplicaciones como: automatización residencial, ajuste óptimo del termostato para zonas térmicas del hogar, control y gestión de cargas eléctricas.

BAN e IAN se utilizan para clientes comerciales e industriales, se enfocan en la automatización de edificios, calefacción, ventilación y aire acondicionado (HVAC), entre otras aplicaciones de gestión energética industrial.

2.1.5 Tecnologías de comunicación para una red eléctrica inteligente

Un sistema de comunicaciones es el componente fundamental en una infraestructura de red eléctrica inteligente. Con la integración de tecnologías avanzadas y aplicaciones destinadas a una *smart grid*, se genera una gran cantidad de datos que posteriormente serán analizados. Por lo tanto, es muy importante elegir una infraestructura de comunicaciones adecuada para manejar estos datos y ofrecer un servicio confiable, seguro y rentable en todo el sistema.

Existen varias tecnologías de comunicaciones soportadas por dos medios de comunicación: cableado e inalámbrico, mediante los cuales se realiza la transmisión de

datos entre contadores inteligentes y centrales eléctricas. En ocasiones, las tecnologías inalámbricas tienen ciertas ventajas sobre las tecnologías cableadas, como bajos costos en infraestructura y facilidad de conexiones en áreas difíciles de alcanzar. No obstante, la naturaleza de este medio de transmisión puede causar interferencias en la señal. Por otro lado, las soluciones cableadas no presentan problemas de interferencia y su operación no depende de baterías, como las soluciones inalámbricas comúnmente lo hacen, según menciona (Tsampasis, Bargiotas, Elias, & Sarakis, 2016). Se tiene una visión general de las tecnologías de comunicación de redes eléctricas inteligentes en la Tabla 1.

Tabla 1. *Tecnologías de comunicación de las Smart Grids*

Tecnología	Estándar/protocolo	Velocidad de transmisión max.	Rango de cobertura max.	Red		
				HAN/BAN/IAN	NAN/FAN	WAN
Tecnologías cableadas						
Fibra óptica	PON	155 Mbps-2.5 Gbps	60 km			X
	WDM	40 Gbps	100 km			
	SONET/SDH	10 Gbps	100 km			
DSL	ADSL	1-8 Mbps	5 km		X	
	HDSL	2 Mbps	3.6 km			
	VDSL	15-100 Mbps	1.5 km			
Coaxial	DOCSIS	172 Mbps	28 km		X	
Cable						
PLC	Homeplug	14-200 Mbps	200 m	X		
	Narrowland	10-500 kbps	3 km		X	
Ethernet	802.3x	10 Mbps - 10 Gbps	100 m	X	X	
Tecnologías inalámbricas						
Z-Wave	Z-Wave	40 kbps	30 m	X		
Bluetooth	802.15.1	721 kbps	100 m	X		
ZigBee	ZigBee	250 kbps	100 m	X	X	
	ZigBee Pro	250 kbps	1600 m			

WiFi	802.11x	2-600 Mbps	100 m	X	X	
WiMAX	802.16	75 Mbps	50 m		X	X
Wireless Mesh	RF mesh, 802.11, 802.15	Depende del protocolo	Depende del despliegue	X	X	
Celular	2G	14.4 kbps	50 km		X	X
	2.5G	144 kbps				
	3G	2 Mbps				
	3.5G	14 Mbps				
	4G	100 Mbps				
Satellite	Satellite Internet	1 Mbps	100-6000 km			X

Fuente: (Kuzlu, Pipattanasomporn, & Rahman, 2014)

2.1.5 Tecnologías de comunicación para una HAN/BAN/IAN

Entre las tecnologías de comunicación para redes locales (HAN, BAN, IAN), según (Ananthkrishnan, 2015) se tienen las siguientes:

Comunicaciones de Línea de Potencia (PLC)

Las comunicaciones mediante PLC incluyen sistemas de envío y recepción de datos por el mismo medio de transmisión de energía eléctrica hacia residencias, edificios u otras instalaciones. Los sistemas PLC permiten una gran variedad de aplicaciones, por ejemplo: lectura automática de medidores, control de cargas, domótica; estos sistemas pueden implementarse de varias maneras, dependiendo de las regulaciones locales, características de la red eléctrica, entre otras. Algunos estándares de PLC desarrollados son: IEEE 1901, Powerline Intelligent Metering Evolution (PRIME), HomePlug AV, Especificaciones ITU-T G.hn (G.9960 y G.9961).

Ethernet

Ethernet es una tecnología de comunicación cableada, basada en el estándar IEEE 802.3, según menciona (Kuzlu, Pipattanasomporn, & Rahman, 2015), generalmente se usa en conexiones de redes de área local (LAN). Una vez que la red es instalada, es difícil hacer cambios. Además, los dispositivos se conectan a la red media cable par trenzado, a través de un hub o switch en una topología de estrella.

Z-Wave

Esta tecnología de comunicación inalámbrica, de bajo consumo y bajo costo es conveniente para comunicaciones de corto alcance, está diseñada para aplicaciones de control remoto en entornos comerciales y residenciales. Z-Wave admite redes en malla, aunque con una baja velocidad de transmisión de datos. (Kuzlu, Pipattanasomporn, & Rahman, 2015)

Bluetooth

Bluetooth está basado en el estándar IEEE 802.15.1, y consiste en una red inalámbrica de área personal. Es generalmente utilizado en dispositivos personales portables, esta tecnología no tiene una capa de seguridad muy fuerte en comparación con otros estándares de comunicación inalámbrica, inclusive puede sufrir interferencias con WiFi. (Kuzlu, Pipattanasomporn, & Rahman, 2015)

ZigBee

ZigBee está basado en el estándar IEEE 802.15, según (Baimel, Tapuchi, & Baimel, 2016), es una tecnología de comunicaciones inalámbrica con un consumo de energía, velocidad de datos, complejidad y dificultad de implementación, relativamente bajos. Es una tecnología ideal para iluminación inteligente, monitoreo de energía, automatización en el hogar, entre otras aplicaciones de corto alcance. ZigBee permite conectar hasta 60000 dispositivos a su red, sin embargo, puede ser susceptible a interferencias con otros aparatos que comparten el mismo medio de transmisión.

WiFi

Es la tecnología inalámbrica más popular en una red de datos para instalaciones del cliente y su estándar es el IEEE 802.11, de acuerdo a (Kuzlu, Pipattanasomporn, & Rahman, 2015) provee confiabilidad, seguridad y comunicaciones de alta velocidad, y opera en las bandas de frecuencia no licenciadas ISM (Industry, Scientific and Medical) de 2,4GHz, 3.5GHz y 5GHz. Su consumo de energía suele ser superior a otras tecnologías inalámbricas de corto alcance como ZigBee y Z-Wave.

Wireless Mesh

En esta red, cada nodo opera como host y también como enrutador, reenviando paquetes hacia otros nodos que pueden no estar dentro de la transmisión inalámbrica directa, este tipo de red establece y mantiene automáticamente la conectividad entre sus nodos, así lo afirma (Tsampasis, Bargiotas, Elias, & Sarakis, 2016).

La red en malla es una solución con una organización dinámica y escalable, lo cual mejora el rendimiento de la red y extiende su cobertura. Se debe tener en cuenta la aplicación de técnicas de cifrado por motivos de seguridad, ya que la información de medición pasa por cada punto de acceso.

2.2 Consumo eléctrico

La energía eléctrica es fundamental para los hogares, puesto que se utiliza para la iluminación y para el funcionamiento de los diferentes aparatos eléctricos (TV, ordenador, aspiradora, entre otros.).

El consumo eléctrico (medido en kW/h) se obtiene a través del medidor o contador de la luz, el cual utiliza la potencia contratada de electricidad y el tiempo de funcionamiento de los aparatos eléctricos e iluminación. Este consumo se ve reflejado en las facturas mensuales del hogar.

2.2.1 Consumo eléctrico en modo de espera

De acuerdo con (Wei, Zhang, Jin, Yin, & Wu, 2014), el consumo de electricidad en modo de espera o standby se refiere a la potencia consumida por aparatos eléctricos que no pueden ser completamente apagados sin ser desconectados, tales como televisores, hornos microondas, aparatos de aire acondicionado, entre otros. Generalmente este tipo de dispositivos consumen energía durante todo el día, de manera desapercibida. Esta potencia es consumida por las fuentes de alimentación, que convierten corriente alterna (CA) a corriente continua (CC), y los sensores y circuitos electrónicos que reciben la señal a distancia y se mantienen energizados. El incremento del número de electrodomésticos hace que la potencia de consumo en modo de espera ya no sea insignificante con respecto

a la energía eléctrica total consumida en el hogar, se estima que en un hogar promedio, la pérdida de energía por el concepto antes mencionado va del 4% al 14%. (Wei, Zhang, Jin, Yin, & Wu, 2014).

2.2.2 Aparatos electrónicos que más consumen energía en modo de espera.

La Agencia Internacional de la Energía en EE.UU, mediante el *Lawrence Berkely National Laboratory* realizó un estudio sobre el consumo de energía eléctrica de aparatos que se mantienen enchufados, a pesar de encontrarse apagados (consumo eléctrico en modo de espera). Éstos pueden consumir alrededor de 1,6 KW diarios, según la (BBC, 2018), los principales aparatos se muestran en la Tabla 2.

Tabla 2. *Consumo en Standby*

DISPOSITIVO ELECTRÓNICO	CONSUMO EN W/h
Decodificador de cable	7
Laptop	8.9 a 15.7
Equipo de sonido	14.4
Computadora	21.1 a 73.9
Consola de Videojuegos	1 a 23.3
Cafetera	1
Televisor	6.6
Microondas	3.08 a 25.79
Teléfono inalámbrico	2.9
Cargador de teléfono celular	< 1

Fuente: (BBC, 2018)

2.2.3 Aparatos eléctricos que más consumen energía en modo de funcionamiento.

El doctor en ingeniería nuclear Luis Juanicó, especialista en diseños avanzados y evaluaciones económicas, explicó cuáles son los electrodomésticos que más energía consumen (Clarín, 2015), éstos electrodomésticos y sus valores de consumo se listan en la Tabla 3.

Tabla 3. Consumo en funcionamiento

DISPOSITIVO ELÉCTRICO/ELECTRÓNICO	CONSUMO EN W/h
Horno eléctrico	3000 a 10000
Lavarropas	300 (agua fría); 2500 a 3000 (agua caliente)
Lavavajillas	3000
Termotanque eléctrico	2500 a 3000
Aire acondicionado	2000
Plancha	800 a 2300
Freidora eléctrica	2000
Cafetera y jarra eléctricas	900
Wafflera	800
Tostadora	800

Fuente: (BBC, 2018)

2.2.4 Smart Meters y Smart Grids en otros países.

Para realizar una implementación adecuada de Smart Meters y Smart Grids, es necesario definir objetivos claros en planes estratégicos de política pública. La revista (ELECTRICIDAD, 2016) describe algunos países como Alemania, Estados Unidos, Italia, entre otros; como los que más experiencia han adquirido y que deben ser considerados en un proceso de estudio para avanzar en las redes eléctricas inteligentes (Smart Grids) y así permitir una gestión activa del consumo de energía eléctrica.

Según (ELECTRICIDAD, 2016), el especialista de la Agencia Alemana de Cooperación Internacional GIZ, Matthias Grandel, menciona que la instalación de medidores inteligentes en el país germano forma parte del plan de la Unión Europea, donde su objetivo es que el 80% de los consumidores del bloque dispongan de estos dispositivos para el 2020.

Por otra parte, refiriéndose a Estados Unidos, José Gaviria, presidente y CEO de Primestone, empresa que diseña software para los medidores eléctricos, afirma que en

este país la instalación se realizó para “controlar la demanda en invierno y verano para lo cual se gestiona el consumo”.

El primer gran desafío al pensar en la instalación en este país fue la eficiencia energética para gestionar el consumo residencial y de la industria. Además, Estados Unidos cuenta con los recursos de redes de telecomunicaciones para el envío de información, debido a su gran despliegue de fibra óptica. (ELECTRICIDAD, 2016)

Asimismo en Italia, según el estudio “Energía Inteligente: Experiencias, costos, beneficios, lecciones para Chile”, realizado por Fraunhofer Chile para la Corporación de Fomento de la Producción (Corfo), menciona que Italia dispone de más de 34 millones de medidores inteligentes bidireccionales instalados, que permiten la lectura remota del consumo de energía residencial y habilitan en tiempo real, procedimientos contractuales como inicio o término del servicio. (ELECTRICIDAD, 2016)

En lo referente a China, el documento entregado a Corfo (ELECTRICIDAD, 2016), señala que un plan piloto enfocado en la implementación de medidores inteligentes y redes residenciales en una ciudad determinada, podría pagarse en 6 o 7 años, y crearía significativos beneficios ambientales a través de ahorros de energía y reducción de emisiones de CO₂.

Del mismo modo se menciona a Corea del Sur, donde el informe de Fraunhofer Chile dice que:

Con el financiamiento conjunto del gobierno y el sector privado, Kepco (empresa eléctrica) está llevando a cabo un proyecto de demostración de Smart Grid muy ambicioso en la isla de Jeju, donde se incorpora infraestructura avanzada de

medidores, almacenamiento de energía y redes de monitoreo y telemetría.
(ELECTRICIDAD, 2016, p.1)

De modo similar, en Reino Unido se destaca el proyecto Energy Demand Research Project (EDRP) que detectó las mejores prácticas y tecnologías para fomentar la participación de los usuarios en la gestión de la demanda del servicio eléctrico, de manera que se reduzca el consumo y el coste de la energía. Se instalaron contadores inteligentes en 60.000 hogares, los cuales permiten obtener informes detallados del consumo eléctrico. (ELECTRICIDAD, 2016)

En cuanto a Brasil, México y Colombia; (ELECTRICIDAD, 2016) considera el punto de vista de Arturo Banderas, gerente de ventas de Latinoamérica de medidores inteligentes de Siemens, donde resalta la experiencia latinoamericana. En este sentido, Banderas señala que Brasil se mantiene como líder en medición inteligente, con un plan de inversión de US \$36.000 millones hasta 2020. Le siguen México que ya tiene un proyecto nacional de medición inteligente; y Colombia, con sus primeros despliegues de plataformas de gestión. El ejecutivo indica que Siemens ha instalado en estos mercados una plataforma que actualmente gestiona 72 millones de medidores inteligentes, esto incluye despliegues masivos en Brasil, México y Colombia; capacitado con interfaces abiertas y con adaptadores desarrollados para integrar los principales sistemas del mercado.

Entre las aplicaciones que tienen estos dispositivos en dichos países, Banderas menciona la protección de pérdidas, monitoreo de carga de los equipos en la red, simulación y planeamiento, seguimiento de la calidad de la energía, balance energético, prepago, gestión de interrupciones, e interacción con plataformas de control como Scada (Supervisory Control and Data Acquisition), EMS (Energy Management System), DMS

(Distribution Management System), ADMS (Active Distribution Management System), y Microgrids.

Por otro lado, en Ecuador, de acuerdo al (INER, 2015) y (CENACE, 2015) una de las iniciativas que impulsa el Gobierno Nacional es el cambio de la matriz energética, dando apertura a la introducción, desarrollo e impulso de nuevas tecnologías que aporten con el medio ambiente, por lo cual a través del Ministerio de Electricidad y Energía Renovable-MEER lleva adelante desde octubre del 2015 el “Análisis para la Implementación de Redes Inteligentes en Ecuador” mediante la ejecución del Programa de Redes Inteligentes-REDIE, el cual tiene como objetivo habilitar la operación en tiempo real de las redes y hacer uso eficiente de la energía, a través de la introducción de nuevas tecnologías que aseguren una alta relación beneficio-costos, mejorar la calidad de generación con un mínimo costo y maximizar el uso de la infraestructura eléctrica.

2.3 IoT (Internet de las cosas)

(Rose, Eldridge, & Chapin, 2015) hace una reseña sobre la “Internet de las Cosas”, término que se utilizó por primera vez en 1999 por el británico Kevin Ashton, el cual describía un sistema donde los objetos del mundo físico se conectaban a Internet a través de sensores.

En este sentido, la Internet de las cosas considera escenarios donde la conectividad de red y la capacidad de cómputo se aplican a sensores y artículos de uso diario, permitiendo a estos dispositivos, generar, intercambiar y consumir datos con una intervención humana reducida. Este concepto conlleva una importancia técnica, social y económica, donde las proyecciones de la IoT sobre la Internet y la economía son sorprendentes; se estima que

para el año 2025 existirán cien mil millones dispositivos conectados, lo cual representará 11 billones de dólares norteamericanos.

2.3.1 Modelos de comunicación de la Internet de las Cosas

Es necesario conocer cómo se conectan y comunican los dispositivos de la IoT, desde el punto de vista de operatividad. Según (Rose, Eldridge, & Chapin, 2015), se tienen cuatro modelos de comunicación, documentados en la RFC 7452 por parte del Comité de Arquitectura de Internet, en 2015; siendo estos modelos los que se describen a continuación:

Modelo de Comunicación ‘dispositivo a dispositivo’

Este modelo de comunicación, tal como se muestra en la Figura 10, representa dos o más dispositivos que se comunican directamente entre ellos, es decir, sin utilizar un servidor de aplicaciones como intermediario. Generalmente se utilizan protocolos como Bluetooth, ZigBee o Z-Wave.

De manera que, se utiliza comúnmente en sistemas de automatización a nivel del hogar, donde se envían pequeños paquetes de datos con una baja tasa de transmisión. Se tiene dispositivos interconectados tales como: interruptores, bombillas de luz, cerraduras, termostatos, entre otros.



Figura 10. Ejemplo de comunicación dispositivo a dispositivo
Fuente: (Rose, Eldridge, & Chapin, 2015)

Modelo de Comunicación ‘dispositivo a la nube’

La Figura 11 representa este modelo de comunicación, en donde el dispositivo IoT se comunica directamente con un servicio en la nube para intercambiar datos y controlar el tráfico de mensajes. Utiliza mecanismos como Wi-Fi o Ethernet para disponer una conexión entre el dispositivo y la red IP, que luego se enlazaría con el servicio en la nube.

En la práctica, este modelo de comunicación es utilizado por varios dispositivos electrónicos, como: *SmartTV* de Samsung, *Learning Thermostat* de Nest Labs, entre otros. Se caracteriza por transmitir información a una base de datos en la nube, donde se efectúa un análisis de consumo de energía del dispositivo y un acceso remoto al mismo, entre otras funciones.

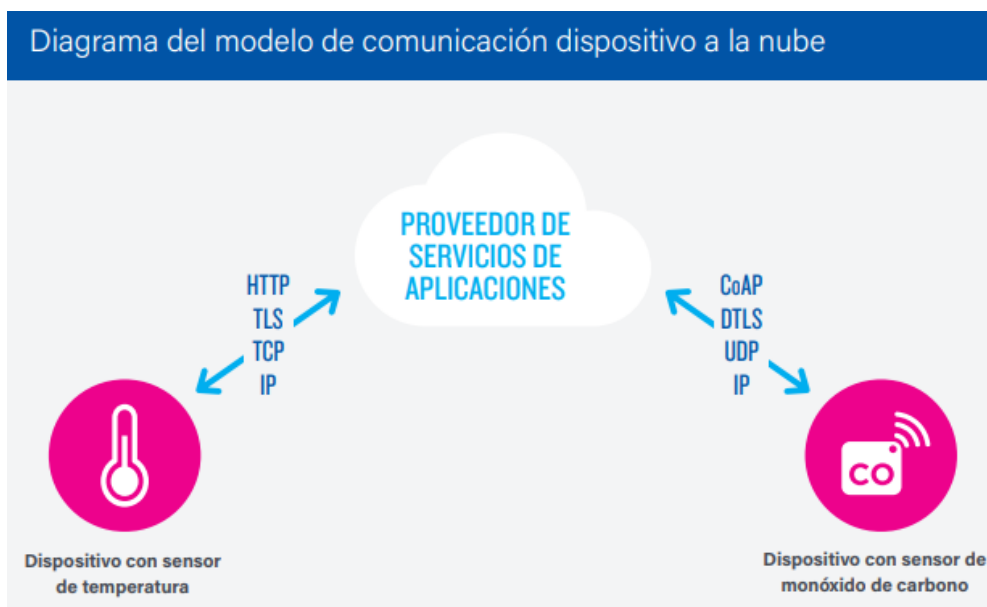


Figura 11. Ejemplo de comunicación dispositivo a la nube
Fuente: (Rose, Eldridge, & Chapin, 2015)

Modelo de Comunicación ‘dispositivo a puerta de enlace’

Este modelo de comunicación también se conoce como modelo dispositivo a puerta de enlace de capa aplicación (ALG). Un dispositivo se conecta a través de un servicio

ALG para acceder a un servicio en la nube. Es decir, hay un software de aplicación que hace de intermediario entre el dispositivo y la aplicación en la nube, y además proporciona mecanismos de seguridad y traducción de protocolos o datos, como se muestra en la Figura 12.

En consecuencia, en muchos casos un teléfono inteligente realiza las funciones de una puerta de enlace, el cual se comunica con un dispositivo mediante una aplicación para luego transmitir datos a la nube. Este modelo se utiliza, por ejemplo, para llevar registros de la actividad física de una persona.

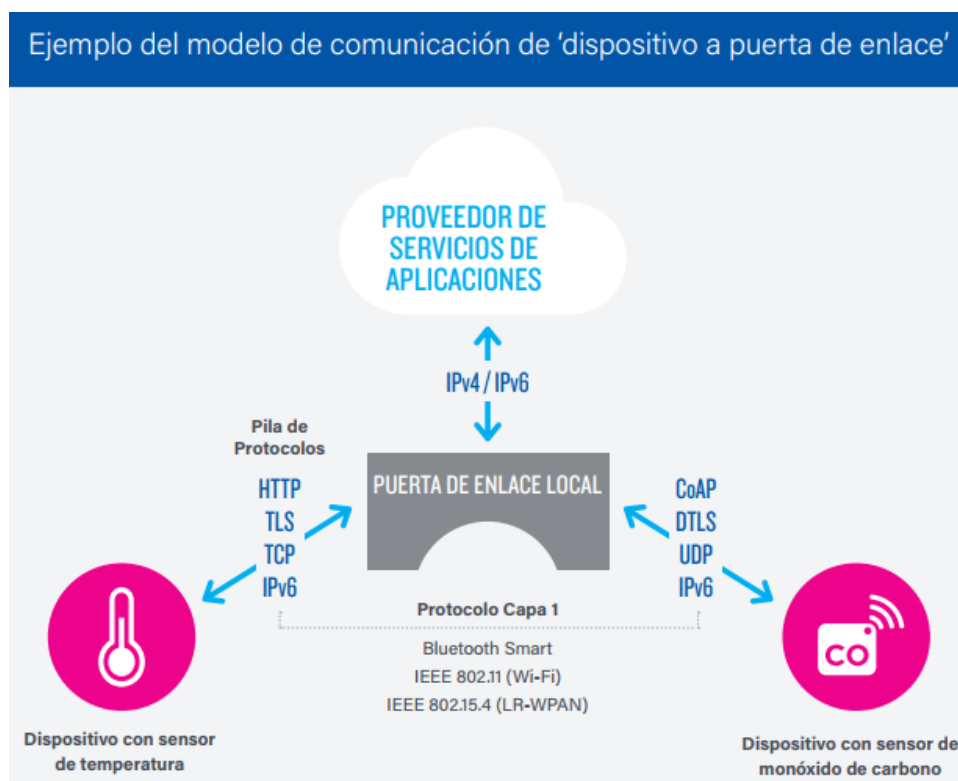


Figura 12. Ejemplo de comunicación dispositivo a puerta de enlace
Fuente: (Rose, Eldridge, & Chapin, 2015)

Modelo de Comunicación 'Back-End'

Este modelo de comunicación permite a los usuarios exportar y analizar información de objetos inteligentes en un servicio en la nube, conjuntamente con datos provenientes de otras fuentes. En otras palabras, el usuario permite a terceras personas hacer uso de los

datos obtenidos por sus sensores. Esta arquitectura se considera una extensión del modelo ‘dispositivo a la nube’, como se evidencia en la Figura 13.

Por consiguiente, en un ambiente corporativo, un usuario a cargo de un conjunto de oficinas puede solicitar datos sobre el consumo de energía y otros servicios provenientes de varios sensores de la IoT. Una arquitectura de *back-end* eficaz permite el acceso y análisis, en la nube, de toda la gama de dispositivos instalados en el sitio. (Rose, Eldridge, & Chapin, 2015)

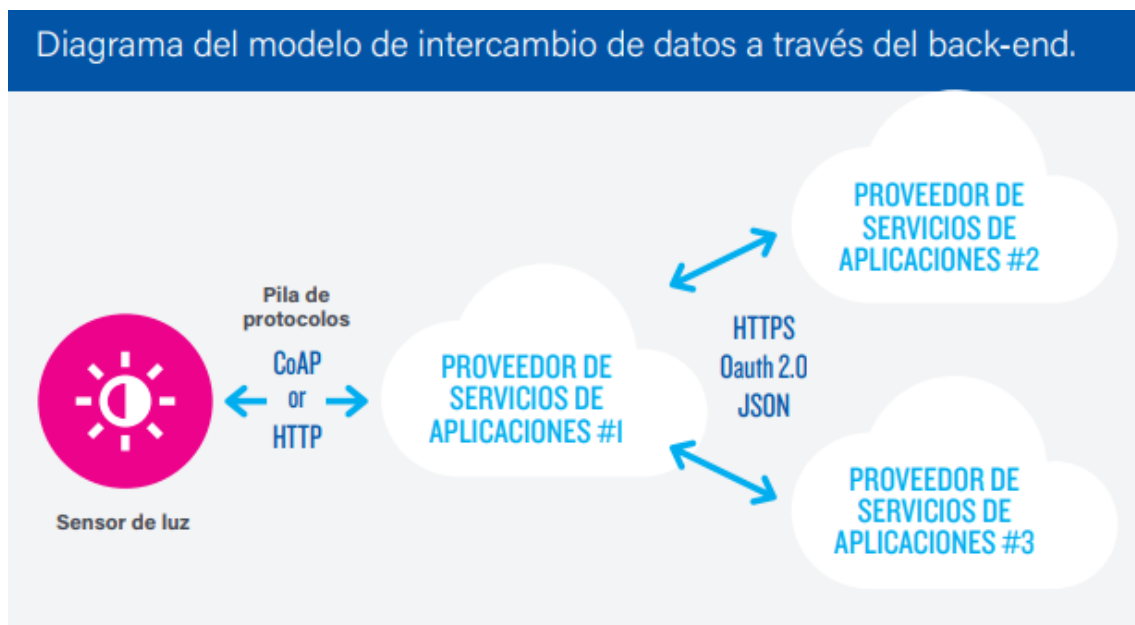


Figura 13. Ejemplo de comunicación back-end
Fuente: (Rose, Eldridge, & Chapin, 2015)

2.4 Sensores de corriente

Estos sensores se utilizan para monitorizar corriente alterna o continua en un circuito eléctrico. En (Vargas, y otros, 2015) se mencionan 4 tipos de sensores: Sensor de efecto Hall, Bobina Rogowski, Shunt de corriente de baja resistencia, y Transformador de corriente.

2.4.1 Sensor de efecto Hall

El efecto Hall resulta de la fuerza que se ejerce sobre una carga eléctrica que se encuentra en movimiento, y está sometida a un campo eléctrico y un campo magnético. El físico E. Hall, en 1879 descubrió que, si un conductor sobre el cual circulaba corriente era ubicado en un campo magnético perpendicular a la misma, se medía una pequeña diferencia de potencial. Entonces con la creación del campo magnético por una corriente que circula por un conductor, puede medirse la magnitud de la corriente en el conductor o bobina. Este sensor se muestra en la Figura 14.

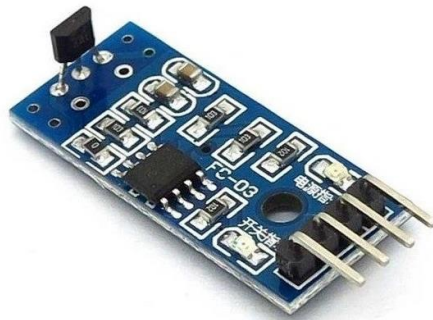


Figura 14. Sensor de efecto Hall
Fuente: (Moviltronics, 2018)

2.4.2 Bobina Rogowski

Esta bobina, mostrada en la Figura 15, posee un núcleo de aire, y es capaz de manejar altas corrientes. Teóricamente el límite superior del a bobina es el voltaje de ruptura del aire (0.4-3.0 mV/m). Este tipo de sensor se aplica en interruptores con reconexión automática de media detención, ya que, debido a su gran precisión, se mejora considerablemente la protección, midiendo la corriente de fase y residual.



Figura 15. Bobina Rogowsky
Fuente: (Vargas, y otros, 2015)

2.4.3 Resistencia shunt

Los sensores Shunt de corriente, mostrados en la Figura 16, son los de menor valor, ofrecen una lectura sencilla y con una buena precisión. Este tipo de sensor es generalmente el más utilizado para la medición de corriente. Cuando se requieren mediciones de alta precisión se debe tener en cuenta la inductancia del sensor Shunt (relación entre flujo magnético e intensidad de corriente eléctrica).

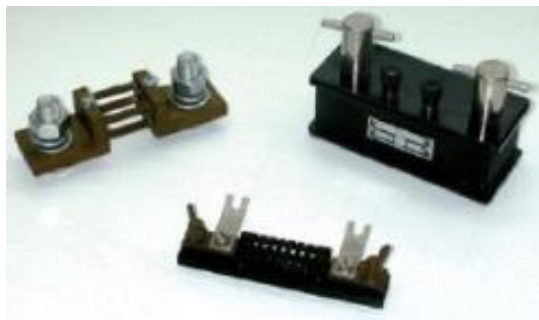


Figura 16. Sensor Shunt
Fuente: (Vargas, y otros, 2015)

2.4.4 Transformador de corriente

Un transformador es un aparato electromagnético que permite aumentar o disminuir el voltaje de un circuito eléctrico, o pasar de corriente alterna a continua por

medio del fenómeno de la inducción electromagnética. La Figura 17 representa su constitución, donde se observa dos bobinas devanadas sobre un núcleo cerrado.

De este modo, se mantiene la cantidad de energía que fluye a través de un transformador, si aumenta el voltaje de la señal, se reduce la corriente, y viceversa. Es decir, la energía de salida de un transformador es igual a la energía de entrada, considerando pequeñas pérdidas.

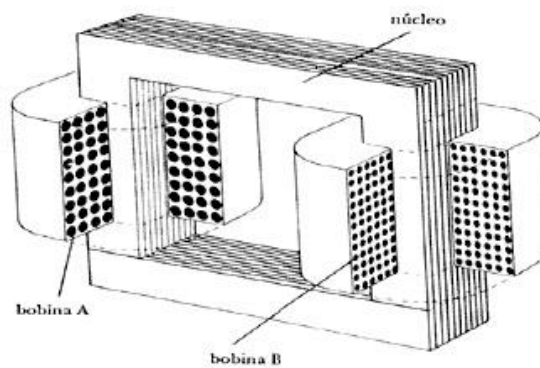


Figura 17. Representación esquemática del transformador
Fuente: (Vargas, y otros, 2015)

2.5 Computación en la Nube

La computación en la nube o *cloud computing*, es considerada según (Alarcón, Guisao, & Guzmán, 2017), como una nueva era en la computación, la cual surge con la finalidad de solucionar múltiples dificultades respecto a los productos y la industria del software, como los elevados costos de adquisición en recursos de TI (Tecnologías de la Información) y el mantenimiento continuo, entre otros.

Este modelo de computación permite un acceso por demanda a la red, donde se tiene un conjunto de recursos informáticos configurables, tales como: capacidad de almacenamiento, levantamiento de servidores, diseño de redes, aplicaciones y servicios.

2.5.1 Modelos de implementación de *Cloud Computing*

Se tiene cuatro modelos de implementación principales de Cloud Computing mencionados por (Alarcón, Guisao, & Guzmán, 2017) y reconocidos por el Instituto Nacional de Estándares y Tecnología (NIST), de los EE. UU; los cuales son: nube pública, nube privada, nube comunitaria y nube híbrida.

En este sentido, (Mukundha & Vidyamadhuri, 2017) describe los modelos de implementación de *Cloud Computing* de la siguiente manera:

Nube pública

Este modelo de implementación ofrece aplicaciones y servicios para uso público por medio de una red, estas nubes pueden ser utilizadas por cualquier persona u organización. Las nubes públicas más conocidas: son Amazon Web Services, Windows Azure, Google AppEngine e IBM Blue Cloud.

Nube privada

Este modelo de implementación es manejado por una única organización y puede ser administrado por la misma organización, o por otra entidad proveedora del servicio. Esta clase de nube es preferida por empresarios con necesidades dinámicas, y que requieren un control directo sobre el ambiente de trabajo. Algunas de las nubes privadas disponibles son: Amazon Private Cloud, SUSE Open Stack Private Cloud, entre otras.

Nube comunitaria

Este tipo de nube es manejada por un grupo de organizaciones, quienes comparten instancias comunes como seguridad o almacenamiento y pueden ser alojadas y administradas de manera interna y externa. Este modelo de servicio se denomina una

infraestructura de multi-inquilino. Algunas nubes comunitarias destacadas son QTS DataCenters y Salesforce.

Nube híbrida

Este tipo de implementación combina dos o más nubes de las descritas anteriormente (pública, privada o comunitaria). En este modelo se considera la privacidad de la organización y la disponibilidad para múltiples inquilinos, junto con la capacidad de comunicarse con otras organizaciones. Por ejemplo, una entidad puede almacenar los datos sensibles de los clientes en una nube privada, hacer uso de una nube comunitaria para conectarse con otras organizaciones semejantes, y utilizar una nube pública para enlazar una herramienta empresarial a una aplicación de software. Algunas nubes híbridas son SoftLayer.com y RackSpace.

2.5.2 Modelos de Servicio de *Cloud Computing*

Los modelos de servicio en la nube, mostrados en la Figura 18, se describen como aquellos que se encuentran disponibles para los clientes. (Gorelik, 2013) menciona tres tipos: Software como Servicio (SaaS), Plataforma como servicio (PaaS), e Infraestructura como Servicio (IaaS), estos modelos pueden tener sinergias entre sí y ser interdependientes.

Software como Servicio (SaaS)

Este modelo de servicio proporciona soluciones de software *online* ya desarrolladas, el proveedor de SaaS tiene el control completo del software de aplicación. Algunos ejemplos incluyen correo en línea, sistemas de gestión de proyectos, plataformas de redes sociales, entre otros. Generalmente SaaS ofrece aplicaciones en línea ya desarrolladas, por ejemplo, Office365 y Facebook.

Plataforma como Servicio (PaaS)

Este modelo de servicio representa una plataforma de aplicación preconstruida para el cliente, donde este último no necesita invertir tiempo en crear una infraestructura dedicada para sus aplicaciones. PaaS puede escalar automáticamente y proveer componentes de infraestructura de acuerdo con los requerimientos de la aplicación. Generalmente las soluciones basadas en PaaS proporcionan una interfaz de programación de aplicaciones (API), que incluye un conjunto de funciones para el desarrollo de soluciones. Entre los proveedores de servicio más populares se tiene Google AppEngine y Amazon Web Services.

Infraestructura como Servicio (IaaS)

Este modelo de servicio provee componentes de infraestructura para sus clientes, es decir, máquinas virtuales, redes, almacenamiento, firewalls, balanceadores de cargas, entre otros. IaaS proporciona a sus clientes un acceso directo al S.O. de una máquina virtual, o al panel de configuración de un firewall o balanceador de carga.

Entre los proveedores más grandes de IaaS se encuentran Amazon Web Services y Microsoft Azure.

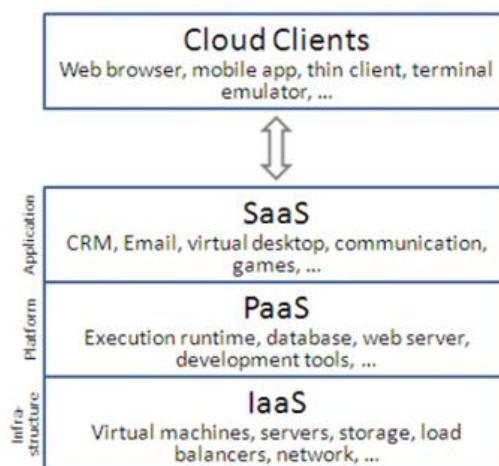


Figura 18. Modelos de servicios ofrecidos en Cloud Computing.
Fuente: (Gorelik, 2013)

CAPÍTULO III

DISEÑO DEL PROTOTIPO

En este capítulo se detalla el diseño del prototipo mediante la metodología de Modelo en V, y la selección de los componentes de hardware y software mediante el estándar IEEE 29148. Se diseña el prototipo electrónico de envío de datos sobre el consumo eléctrico y la plataforma de almacenamiento de la información accesible a través de una página web.

3.1 Metodología

Es necesario llevar un proceso en el desarrollo del presente trabajo de investigación, de manera que puedan cumplirse sus objetivos. Esto permite seguir una serie de pasos y a su vez evaluar a cada uno de ellos, para facilitar la corrección de errores y abarcar todos los aspectos que engloba el proyecto.

3.1.1 Modelo en V

La metodología por utilizar se conoce como “Modelo en V” y está representada en la Figura 19 por (IIIA-CSIC, 2018). Es una de las muchas variantes surgidas del modelo en cascada y resuelve en parte el problema de la simplicidad de este modelo, que sólo muestra una salida por proceso, y el hecho de que esta salida no puede ser reinsertada como parte de la entrada de etapas anteriores. El modelo representa en forma de V, las relaciones temporales entre las distintas fases del ciclo de desarrollo de un proyecto.

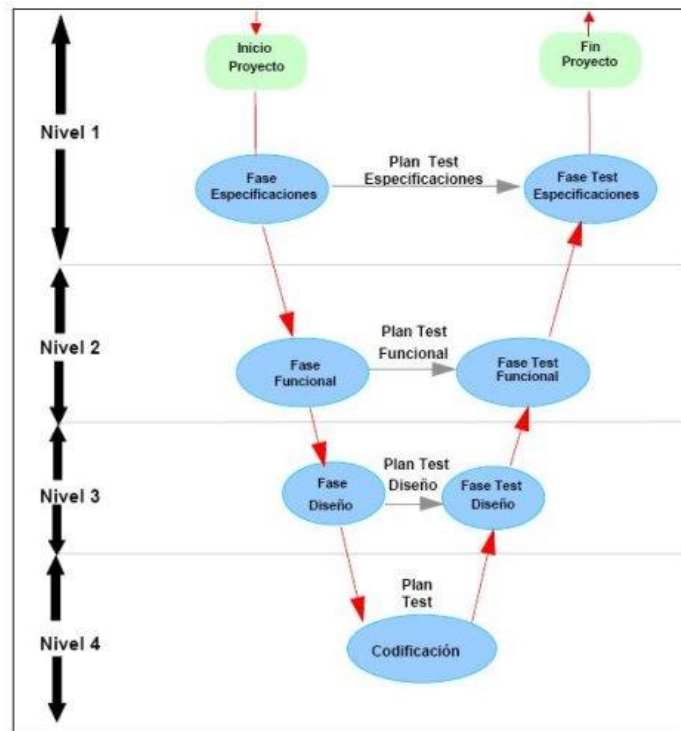


Figura 19. Esquema del modelo en V
Fuente: (IIIA-CSIC, 2018)

El nivel 1 del modelo en V, está orientado al “cliente”. Se tienen los dos extremos del proyecto, el inicio y el fin. Se compone del análisis de requisitos y especificaciones, en consecuencia, se traduce en un documento de requisitos y especificaciones.

El nivel 2 del modelo en V, se dedica a las características funcionales del sistema propuesto. Se caracteriza únicamente con aquellas funciones que son directa o indirectamente visibles por el usuario final, se traduce en un documento de análisis funcional.

El nivel 3 del modelo en V, define los componentes hardware y software del sistema final, a cuyo conjunto se denomina arquitectura del sistema.

El nivel 4 del modelo en V, es la fase de implementación, en la que se desarrollan los elementos unitarios o módulos del proyecto.

3.1.2 Estándar ISO / IEC / IEEE 29148: 2011

Este estándar definido por (ISO, IEC, & IEEE, 2011), contiene disposiciones para los procesos y productos relacionados con la ingeniería de requisitos para sistemas, productos y servicios de software a lo largo del ciclo de vida. Precisa la construcción de un buen requisito, proporciona atributos y características de los mismos, y analiza la aplicación iterativa y recursiva de los procesos de requisitos a lo largo del ciclo de vida. Se definen elementos de información aplicables a la ingeniería de requisitos y su contenido.

3.2 Análisis

En este apartado se determinan características del problema, las cuales permitan comprender sus condiciones actuales sobre una población, esta información es relevante para obtener requerimientos del sistema. Para cumplir este objetivo se analiza información sobre la toma de mediciones de consumo eléctrico y se propone una solución a nivel de hogar, además se utiliza la técnica de investigación denominada encuesta, cuyo modelo y tabulación de resultados se encuentran en el Anexo A, la encuesta es aplicada a varias familias del barrio San Roque de la ciudad de Mira, provincia del Carchi, que se ubica en las coordenadas 0°33'06.6"N (longitud) y 78°02'14.2"W (latitud).

3.2.1 Situación Actual

Ecuador al ser un país en vías de desarrollo aún tiene mucho camino que recorrer en lo referente a *smart grids*, sin embargo, la Secretaría Nacional de Planificación y Desarrollo propone proyectos innovadores como el cambio de la matriz energética donde exista mayor participación de fuentes renovables, ejecutando obras de hidroeléctricas. A la par, también se busca avanzar en la infraestructura de medición avanzada AMI, que permite el flujo de información bidireccional entre el cliente y la empresa proveedora del

servicio eléctrico, para tener lecturas de consumo más exactas y en menos tiempo, realizar cortes y reconexiones de manera remota, lo cual implica un ahorro económico para los usuarios y las empresas. (García & Ruiz, 2014). Aunque, (Telégrafo, 2012) menciona que treinta y cinco mil contadores inteligentes fueron instalados en Guayaquil en 2012, aún se utiliza contadores análogos y digitales en gran parte del Ecuador.

La infraestructura AMI constituye uno de los pilares fundamentales de una red eléctrica inteligente, ya que comprende la etapa de consumo por parte del usuario final, sin embargo, su alcance llega solamente hasta el contador eléctrico inteligente (smart meter) ubicado en la parte exterior del hogar, el cual registra el consumo eléctrico global de cada mes. Debido a esto, el control del consumo eléctrico responsable dentro de la vivienda es responsabilidad exclusiva de sus propietarios, ya que existen situaciones donde se genera un uso inadecuado de este recurso, al no contar con un medidor inteligente para los equipos eléctricos y electrónicos presentes en el hogar.

3.2.2 Encuesta

La encuesta busca recopilar datos acerca de los hábitos de consumo eléctrico de los habitantes de esta localidad, tales como: equipos eléctricos frecuentemente utilizados y sus horarios, mantenimiento periódico de sus dispositivos, detección de equipos encendidos innecesariamente, predisposición a adquirir un sistema como el que plantea este proyecto de investigación; esta información precisa detalles que ayudará a obtener las funcionalidades del sistema.

Puesto que se tiene una población definida de 90 familias en el barrio San Roque de la ciudad de Mira, de acuerdo con los registros proporcionados por el presidente del barrio. Según (Ochoa, 2013), para determinar el tamaño de la muestra se utiliza la Ecuación 1.

$$n = \frac{N \cdot Z^2 \cdot \sigma^2}{(N-1) \cdot e^2 + Z^2 \cdot \sigma^2}$$

Ecuación 1. Determinación del tamaño de muestra para una encuesta

Donde:

n = Tamaño de la muestra

N = Tamaño de la población

Z = Nivel de confianza deseado (dado por la distribución de Gauss), los valores más comunes de acuerdo con (Ochoa, 2013) son:

- Nivel de confianza 90%, entonces Z=1,64
- Nivel de confianza 95%, entonces Z=1,96
- Nivel de confianza 99%, entonces Z=2,58

Considerando que la población es muy uniforme ya que está ubicada en el mismo sector y tiene condiciones de vida similares entre cada familia, se puede reducir el tamaño de la muestra, por lo cual se toma el valor de Z = 1,64.

σ = Representa la desviación estándar. El valor de σ se obtiene de un estudio previo o de una estimación del investigador, generalmente se sitúa en 0,5 cuando no se conoce el valor, ya que según (Ochoa, 2013), esto representa el peor escenario, donde la población se divide en partes iguales; al utilizar este caso crítico se cubre todas las posibilidades de valores de desviación estándar.

e = Es el margen de error máximo permitido por el investigador. Se utiliza para este proyecto un valor de 0,1 ya que el nivel de confianza seleccionado anteriormente es de 90%.

Por lo tanto, remplazando los valores de $N = 90$; $Z = 1,64$; $\sigma = 0,5$; $e = 0,1$ en la Ecuación 1 se obtiene como resultado un valor de muestra: $n = 38,73$. Es decir, se aplicará la encuesta a 39 familias del total de la población establecida en 90 familias.

Es así como, luego de la aplicación, tabulación y análisis de resultados conseguidos con la encuesta y presentados en el Anexo A, se obtienen las siguientes conclusiones:

- De todos los electrodomésticos presentados, las personas utilizan seis de ellos con más frecuencia, entre los cuales se tiene: televisión (23%), equipo de sonido (17%), microondas (13%), plancha de ropa (13%), computadora (14%) y lavadora (18%).
- Las personas, en su mayoría (59%) han dejado encendido un dispositivo alguna vez, cuando no lo están utilizando, lo cual genera un consumo adicional de electricidad en el hogar, entre los dispositivos más comunes se tiene: TV (44%), equipo de sonido (22%), microondas (13%) y plancha de ropa (13%), cuyo tiempo de consumo innecesario varía entre una y veinticuatro horas, según la tabulación de las encuestas.
- La totalidad de encuestados afirma que nunca ha hecho mantenimiento a sus dispositivos eléctricos, porque mientras funcionaban no tenían necesidad de hacerlo, sin considerar que sus equipos podían deteriorarse poco a poco, hasta llegar a un punto de daño irreversible, y por ende un gasto económico al adquirir un equipo nuevo.
- La mayoría de las personas no presenta hábitos adecuados en cuanto a la preservación de sus electrodomésticos y la eficiencia energética, sin embargo, un 74% de los encuestados se mostró interesado en adquirir un dispositivo automático de control de energía, que permita llevar un registro adecuado del consumo eléctrico de sus equipos.

3.2.3 Descripción general del sistema propuesto

El sistema está formado por dos subsistemas: Un módulo electrónico de adquisición de datos y Una plataforma de administración y gestión vía Internet, la Figura 20 muestra los protocolos de comunicación que intervienen entre los dos subsistemas.

El módulo electrónico de adquisición de datos se ubica sobre un tomacorriente convencional y se encarga de medir la corriente eléctrica consumida por un dispositivo conectado al mismo, este módulo lleva un sensor de corriente que registra la variable eléctrica, y un microprocesador que envía los datos obtenidos por medio de wifi hacia la plataforma de administración y gestión; además un control de potencia que permita cortar y reactivar la energía eléctrica en el tomacorriente.

Mientras que, la plataforma de administración y gestión consiste en una máquina virtual alojada en la nube (cloud computing), la cual receptorá los datos enviados por parte del módulo electrónico y los mostrará en una página web para su monitoreo, en caso de sobrepasar cierto límite asignado, emite una alerta en la página web y envía un correo electrónico al usuario. De igual manera mediante una interfaz, es capaz de controlar el flujo eléctrico hacia el módulo electrónico y establecer parámetros iniciales de medición y automatización.

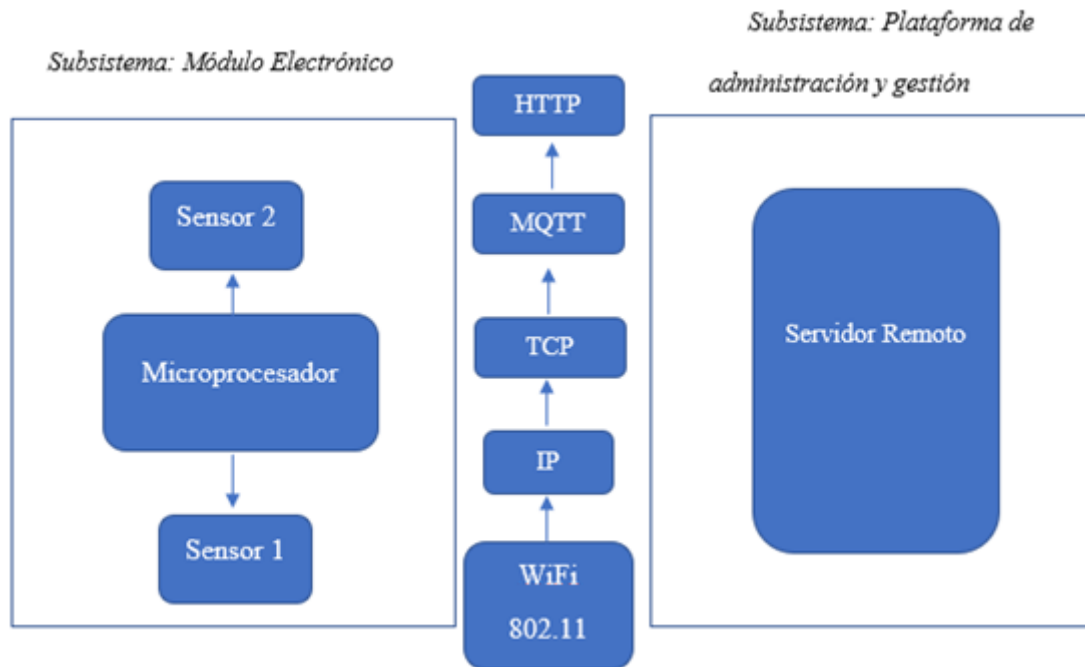


Figura 20. Protocolos de comunicación entre los subsistemas: módulo electrónico y plataforma de administración y gestión

3.3 Requerimientos del sistema propuesto

En esta sección se presenta los requerimientos del sistema, tomando en cuenta cuatro parámetros: requerimientos de stakeholders, requerimientos del sistema, requerimientos de arquitectura, y además se incluye un apartado de requerimientos de software para describir los requerimientos para la plataforma de administración y gestión. Estos requerimientos deben ser ejecutables y evaluables, para este efecto se asigna prioridades (alta, media y baja) a cada uno de los requerimientos, y se evalúan desde la perspectiva del diseñador del sistema, tal como menciona el estándar IEEE 29148. (ISO/IEC/IEEE, 2011),

Con la finalidad de llevar un manejo ordenado de la información, se utilizan abreviaturas para cada uno de los tipos de requerimientos, lo cual se detalla en la Tabla 4. Es importante señalar que los requerimientos de software se indican con otras

abreviaturas, ya que la plataforma de administración y gestión se considera como otro subsistema con diferentes requerimientos, mismos que se detallan en la sección 3.3.4 Requerimientos de Software.

Tabla 4. *Requerimiento del sistema y abreviaturas*

Abreviatura	Requerimiento
StRS	Requerimientos de Stakeholders
SyRS	Requerimientos del Sistema
SRSR	Requerimientos de Arquitectura

3.3.1 Requerimientos de Stakeholders

Estos requerimientos reflejan las necesidades de los usuarios y partes interesadas, junto con una valoración en cuanto a prioridad, esta información se observa en la Tabla 5.

Tabla 5. *Requerimientos de Stakeholders*

REQUERIMIENTOS DE STAKEHOLDERS					
Abreviatura	Requerimiento	Prioridad			Relación
		Alta	Media	Baja	
Requerimiento: StRS - Operacionales					
StRS1	Tamaño reducido del módulo electrónico	X			SRSR6
StRS2	Comunicación Inalámbrica	X			
StRS3	Dispositivo Portable		X		SyRS1
Requerimiento: StRS - Usuarios					
StRS4	Apariencia llamativa de la página Web		X		
StRS5	Información organizada y actualizada		X		
StRS6	Respuesta inmediata del módulo electrónico	X			

3.3.2 Requerimientos del Sistema

Estos requerimientos mencionan las funciones del sistema y lo relacionado con su funcionamiento, estos datos se registran en la Tabla 6.

Tabla 6. *Requerimientos del Sistema*

REQUERIMIENTOS DEL SISTEMA					
Abreviatura	Requerimiento	Prioridad			Relación
		Alta	Media	Baja	
Requerimiento: SyRS - Uso					
SyRS1	Facilidad de instalación		X		StRS3
SyRS2	Lectura de datos continua		X		
SyRS3	Prototipo duradero	X			
Requerimiento: SyRS - Rendimiento					
SyRS4	Funcionamiento automático	X			
Requerimiento: SyRS - Interfaces					
SyRS5	Conexión a Internet	X			
SyRS6	Pines análogos y digitales	X			SRS7
Requerimiento: SyRS - Físicos					
SyRS7	Tamaño reducido de módulos y placa	X			

3.3.3 Requerimientos de Arquitectura

Estos requerimientos, mostrados en la Tabla 7, hacen referencia a las necesidades del prototipo y a las características del software y hardware presentes en el sistema.

Tabla 7. *Requerimientos de Arquitectura*

REQUERIMIENTOS DE ARQUITECTURA					
Abreviatura	Requerimiento	Prioridad			Relación
		Alta	Media	Baja	
Requerimiento: SRS7 - Lógicos					
SRS71	Conexión con la nube	X			
SRS72	Puerto Serial	X			
SRS73	Convertor ADC	X			SyRS6
Requerimiento: SRS7 - Software para las placas					
SRS74	Librerías para los módulos	X			
SRS75	Lenguaje de programación de código abierto	X			
Requerimiento: SRS7 - Hardware					
SRS76	Tarjeta programable de tamaño reducido	X			StRS1
SRS77	Sensado de varios puertos	X			SyRS6
SRS78	Soporte para aplicaciones de potencia	X			

3.3.4 Requerimientos de Software

Este apartado analiza los requerimientos de la infraestructura como servicio que se utiliza para desarrollar la plataforma de administración y gestión, se determinan los requerimientos del sistema operativo, y los requerimientos del protocolo de transporte orientado a la Internet de la Cosas. De igual manera, se utilizan abreviaturas para cada uno de los requerimientos, como se muestra en la Tabla 8.

Tabla 8. *Requerimientos de Software y abreviaturas*

Abreviatura	Requerimiento
IAAS	Requerimientos de Infraestructura como Servicio
SO	Requerimientos del Sistema Operativo
IOT	Requerimientos del Protocolo de IoT

3.3.4.1 Requerimientos de la Infraestructura como Servicio (IaaS)

Los datos recogidos por el módulo electrónico serán enviados a un servidor alojado en la nube, para esto se necesita una plataforma que brinde los recursos necesarios para su implementación, tales como: creación de instancias, almacenamiento, capacidad de procesamiento, seguridad, escalabilidad, entre otros. De modo que, los requerimientos de IaaS se evalúan en la Tabla 9.

Tabla 9. *Requerimientos de Infraestructura como Servicio*

REQUERIMIENTOS DE INFRAESTRUCTURA COMO SERVICIO					
Abreviatura	Requerimiento	Prioridad			Relación
		Alta	Media	Baja	
IAAS1	Creación de instancias	X			
IAAS2	Espacio de almacenamiento	X			
IAAS3	Soporte para instalación de Sistemas Operativos	X			
IAAS4	Capacidad de escalamiento	X			
IAAS5	Seguridad integrada	X			
IAAS6	Documentación	X			
IAAS7	Versión gratuita		X		

3.3.4.2 Requerimientos del Sistema Operativo

Es necesario un sistema operativo que de soporte a aplicaciones de páginas web, bases de datos, y reciba información del módulo electrónico mediante protocolos de IoT, los requerimientos necesarios se evalúan en la Tabla 10.

Tabla 10. *Requerimientos del Sistema Operativo*

REQUERIMIENTOS DEL SISTEMA OPERATIVO					
Abreviatura	Requerimiento	Prioridad			Relación
		Alta	Media	Baja	
SO1	Sistema Estable	X			
SO2	Soporte para la placa programable	X			
SO3	Soporte para base de datos	X			
SO4	Soporte para páginas web	X			
SO5	Repositorio de librerías para IoT	X			
SO6	Open Source		X		

3.3.4.3 Requerimientos del Protocolo de IoT

Debido a que este proyecto propone un uso eficiente del recurso energético, se evalúa un protocolo de IoT de bajo consumo en cuanto a energía y ancho de banda, para luego ser implementado en el prototipo, los requerimientos muestran en la Tabla 11, junto a sus valoraciones.

Tabla 11. *Requerimientos del Protocolo de IoT*

REQUERIMIENTOS DEL PROTOCOLO DE IOT					
Abreviatura	Requerimiento	Prioridad			Relación
		Alta	Media	Baja	
IOT1	Bajo consumo energético	X			
IOT2	Bajo ancho de banda	X			
IOT3	Seguridad integrada	X			
IOT4	Compatible con el Sistema Operativo	X			
IOT5	Compatible con las placas programables	X			
IOT6	Librerías disponibles	X			

3.4 Selección de Hardware y Software

En esta sección se elige los componentes de hardware y software necesarios y adecuados para la realización del presente proyecto. En lo referente a hardware se tiene: tarjeta programable, encargada de realizar el procesamiento de los datos recibidos y tomar decisiones de acuerdo con su programación; módulo WiFi, que actúa como puente entre el módulo electrónico y el servidor remoto permitiendo la comunicación bidireccional; sensor de corriente, que registra el consumo de corriente de los equipo asociados y envía esta información a la tarjeta programable.

Por otra parte, la selección de software hace referencia a: infraestructura como servicio, que presenta los recursos de computación en la nube necesarios para desarrollar la plataforma de administración y gestión; sistema operativo, donde se instalan los paquetes de software que permiten levantar servicios de página web, bases de datos, conectividad, entre otros; protocolo de IoT, sobre el cual se envía información desde el módulo electrónico a la plataforma de administración y gestión, y viceversa.

Por consiguiente, para realizar la selección se determina con el valor “1” si el componente cumple con cierto requerimiento establecido en la sección 3.3 Requerimientos del Sistema propuesto, y “0” si no cumple, al final se suma las puntuaciones y el componente con el puntaje más alto es seleccionado, para esta selección se consideran los dispositivos con más presencia en el mercado.

3.4.1 Selección de Hardware

Se analiza los microcontroladores más comunes en el desarrollo de proyectos de IoT, entre los principales se tiene: PIC, AVR, Arduino Nano, Arduino Uno y Raspberry. A continuación, con los elementos antes mencionados se realiza la selección en la Tabla 12.

Tabla 12. Selección de tarjeta programable

Microcontrolador								
Tipo	Requerimiento							Total
	StRS6	SyRS6	SyRS7	SRSH2	SRSH3	SRSH4	SRSH7	
PIC	1	1	1	0	1	1	1	6
AVR	1	1	1	0	1	1	1	6
Arduino Uno	1	1	0	1	1	1	1	6
Arduino Nano	1	1	1	1	1	1	1	7
Raspberry	1	1	0	1	1	0	1	5

De los resultados obtenidos en el análisis de la Tabla 12, se determina la utilización de Arduino Nano como tarjeta programable, ya que ofrece la ventaja de tener un puerto serial para su programación y envío de datos, y ser de tamaño reducido.

Arduino Nano

Es una tarjeta programable que posee las mismas funcionalidades de Arduino Uno, pero en menor tamaño, su imagen se muestra en la Figura 21.

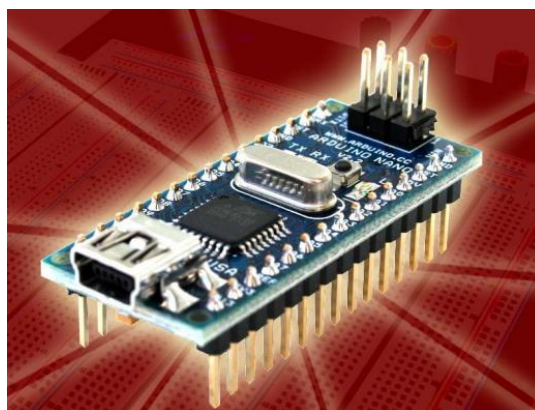


Figura 21. Arduino Nano
Fuente: (Arduino.cl, 2018)

Esta placa programable posee un microcontrolador ATmega328 cuyo voltaje de funcionamiento es 5V, tiene un número considerable de pines análogos y digitales (14 y 6 respectivamente) capaces de proporcionar una corriente de hasta 50mA cada uno; en cuanto a su memoria, tiene una EEPROM de 1KB y una memoria flash de 32KB; su

frecuencia de trabajo es de 16MHz y sus dimensiones son 1.85cm x 4.32cm, estas características se detallan en (Arduino.cl, 2018)

A continuación, en la Tabla 13, se realiza la selección del módulo WiFi, que se encargará de establecer la conexión con el servidor y la bidireccionalidad de comunicación de datos entre el módulo electrónico y la plataforma de administración y gestión. Los módulos WiFi a evaluarse son: NodeMCU, ESP WROOM 02, D1 mini pro y ESP01S.

Tabla 13. Selección del módulo WiFi

Tipo	Requerimiento					Total
	SyRS7	SRS1	SRS2	SRS4	SRS5	
NodeMCU	0	1	1	1	1	4
ESP WROOM 02	0	1	1	1	1	4
D1 mini pro	0	1	1	1	1	4
ESP01S	1	1	1	1	1	5

Se escoge el módulo WiFi ESP-01S, cuya ventaja es su tamaño reducido que se adapta perfectamente a los requerimientos del prototipo.

ESP01S (Wifi)

Es un módulo con wifi integrado, el cual se muestra en la Figura 22, se basa en el microcontrolador ESP8266, es un dispositivo versátil de bajo coste que establece un medio de comunicación entre las aplicaciones electrónicas y la Internet, como menciona (Llamas, 2017).

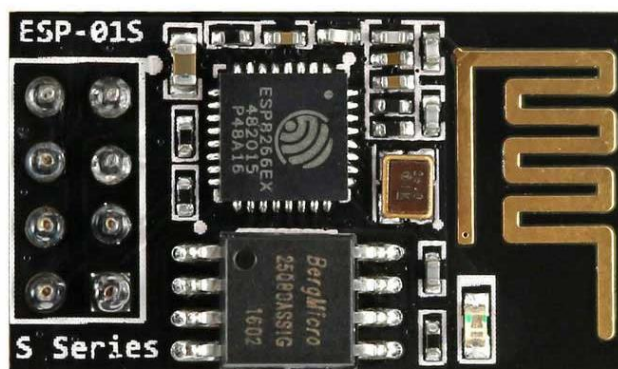


Figura 22. Módulo wifi ESP01S
Fuente: (Arrivals, 2018)

Entre sus principales características, se presenta su microcontrolador ESP8266 que funciona con un voltaje de 3 a 3.6V, utiliza el estándar 802.11 b/g/n y trabaja con la pila de protocolos TCP/IP, posee una frecuencia de funcionamiento de 2,4 GHz; y en cuanto a seguridad, soporta WPA y WPA2, este módulo tiene un consumo pico de 170mA, por lo cual necesita una fuente externa, ya que el Arduino Nano seleccionado anteriormente no puede proporcionar esa cantidad de corriente a través de sus pines (máximo 50mA), las características de este módulo se presentan en (Co.Ltd., 2015).

En cuanto a los sensores de corriente, se tiene al ACS-712 que es un sensor invasivo, es decir, tiene colocarse en el interior del circuito para realizar la medición; por otro lado el SCT-013 es un sensor no invasivo, que puede colocarse sobre los cables por donde circula la corriente que va a medirse, su selección se muestra en la Tabla 14.

Tabla 14. Selección del sensor de corriente

Tipo	Requerimiento						Total
	SyRS7	StRS6	SyRS2	SyRS3	SRSH5	SRSH8	
ACS-712	1	1	1	1	1	1	6
SCT-013	0	1	1	1	1	1	5

Se escoge el sensor ACS-712, cuya característica decisiva es su reducido tamaño, el hecho de ser un sensor invasivo no es una limitante, ya que el diseño del prototipo lo

considera dentro del circuito impreso que se realiza en la sección 3.6.6 Diseño del circuito impreso.

ACS712 (sensor de corriente)

Es un sensor de corriente por efecto hall fabricado por Allegro, el cual se muestra en la Figura 23, este sensor se considera una solución económica y precisa en la medición de corriente AC o DC, posee una señal analógica de bajo ruido y su voltaje de operación es de 5V, además su tiempo de respuesta es muy bajo, alrededor de 5us; y de acuerdo a su modelo, su capacidad de medición es de 5A, 20A y 30A, según menciona (Patagoniatec, 2018).



Figura 23. Sensor de corriente ACS712
Fuente: (Patagoniatec, 2018)

3.4.2 Selección de Software

En esta sección se analiza los requerimientos de software necesarios en el desarrollo de la plataforma de administración y gestión, para lo cual se tiene: proveedor de infraestructura como servicio, el cual ofrece una instancia en la nube donde se erige toda la estructura de este subsistema; sistema operativo, que es el software principal para la instalación, configuración, levantamiento de servicios y ejecución de programas; protocolo de IoT, encargado de establecer la comunicación entre los dos subsistemas y realizar el envío de datos a través de la Internet.

De igual manera, se determina con el valor “1” si el componente mencionado cumple con cierta característica, y “0” si no cumple. En la Tabla 15 se muestra la selección de la infraestructura como servicio (IaaS), los principales proveedores de IaaS son: Amazon Web Services (AWS), Microsoft Azure, IBM Cloud y Google Cloud Platform.

Tabla 15. Selección de la Infraestructura como Servicio

Infraestructura como Servicio									
Tipo	Requerimiento							Total	
	IAAS1	IAAS2	IAAS3	IAAS4	IAAS5	IAAS6	IAAS7		
Amazon Web Services	1	1	1	1	1	1	1	1	7
Microsoft Azure	1	1	1	1	1	1	1	1	7
IBM Cloud	1	1	1	1	1	0	1	1	6
Google Cloud Platform	1	1	1	1	1	1	1	1	7

En esta selección existen tres opciones con la misma puntuación: Amazon Web Services, Microsoft Azure y Google Cloud, estas plataformas se diferencian de IBM Cloud, debido a que tienen información más explicativa sobre su configuración y funcionamiento en la Internet.

Las tres opciones de nubes públicas con mayor puntaje ofrecen capacidades básicas muy similares en computación, almacenamiento y redes flexibles, de acuerdo con (Finnegan & Carey, 2018). Además, todas ellas comparten los elementos comunes de una nube pública: autoservicio y aprovisionamiento instantáneo, escalamiento automático, además de características de seguridad y gestión de identidades. En este sentido, en la Tabla 16 se realiza una comparación entre los servicios y algunas cláusulas de los proveedores de IaaS antes mencionados.

Tabla 16. Comparación de compromisos y descuentos entre proveedores de servicio de nube.

	AWS	Microsoft Azure	Google Cloud	Nube de IBM
Tipo de descuento	Instancias reservadas (RI)	Instancias reservadas (RI)	Descuento por uso sostenido (SUD) Descuento por uso comprometido (CUD)	Precios mensuales
Duración del compromiso	1 o 3 años	1 o 3 años	SUD: sin compromiso CUD: 1 o 3 años	Compromiso por mes
Rango de niveles de descuento	Hasta un 75%	Hasta 72%	SUD: hasta 30% CUD: 37% (1Y) o 55% (3Y)	Como 10%
Otros programas documentados	RI Volume Descuentos Spot instancias	Contrato de empresa de descuento de uso híbrido	VMs vacías	Negociado

Fuente: (Weins, 2017)

Además, cabe mencionar que los precios de los servicios ofrecidos por estas empresas son variables, esto se debe a los descuentos y promociones frecuentes en cada una de ellas. No obstante, es posible establecer una negociación directa con una determinada empresa, con la finalidad de establecer acuerdos de pago y servicios.

Para el desarrollo del presente proyecto se elige la versión gratuita de la nube de Amazon (AWS) por facilidad de contratación para el investigador, ya que a diferencia de las otras, acepta tarjeta de débito, es importante señalar que no se hará ningún recargo a menos que se sobrepase el límite gratuito asignado, el cual se indicará oportunamente. Sin embargo, el servidor de administración y gestión puede ser implementado en cualquiera de los proveedores de servicios de infraestructura antes mencionados.

A continuación, la selección del sistema operativo se indica en la Tabla 17, donde se evalúan los siguientes sistemas operativos: Windows Server, Ubuntu, Debian, Centos y Red Hat Enterprise Linux (RHEL), los cuales se utilizan frecuentemente en la instalación, configuración y levantamiento de servicios.

Tabla 17. Selección del Sistema Operativo

Sistema Operativo	Requerimiento						Total
	SO1	SO2	SO3	SO4	SO5	SO6	
Windows Server	1	1	1	1	1	0	5
Ubuntu	0	1	1	1	1	1	5
Debian	0	1	1	1	1	1	5
Centos	1	1	1	1	1	1	6
Red Hat Enterprise Linux	1	1	1	1	1	1	6

Al final de la selección se tiene una igualdad de puntuación entre dos sistemas operativos: CentOS y Red Hat Enterprise Linux (RHEL), debido a su mayor estabilidad respecto a los demás. Sin embargo, ambos sistemas operativos tienen características muy similares, como se muestra en la Tabla 18, ya que CentOS es una derivación de RHEL.

Finalmente, se escoge el sistema operativo RHEL, debido que es la imagen de SO que posee el proveedor de infraestructura como servicio Amazon Web Services.

Tabla 18. Comparación entre los sistemas operativos CentOS y RHEL

	CentOS	RHEL
License	FOSS – GLP anh others	Commercial -RedHat EULA
Branding	CentOS	RedHat
Security	SeLinux, NSS, Linux PAM, firewalld	SeLinux, NSS, Linux PAM, firewalld
Patches/fixes	SLA through Rogue Wave Software	SLA through Red Hat
Commercial support	24x7 support through Rogue Wave Software	24x7 support through Red Hat
Package management	Yum	Yum
Installer	ISO/Live CD	ISO
Enterprise management package	SpaceWalk/Catello	Red Hat Satellite 5 y 6
Clustering	Linux HA	Red Hat Cluster Suite (RHCS)
Bootloader	GRUB 2	GRUB 2
GUI	GNOME 3 /KDE SC 4.10	GNOME 3 /KDE SC 4.10
Service management	systemd	systemd
Storage management	LVM /SSM	LVM /SSM
Default file system	XFS	XFS
Current kernel	3.10.0.862.3.3.el7.x86_64	3.10.0.862.3.3.el7.x86_64

Virtualization	oVirt / KVM	Red Hat Virtualization Manager
VDI	SPICE	SPICE
Cloud images	Azure by Rogue Wave Software AWS Standard by Rogue Wave Software Google Cloud by CentOS.org	Azure by RedHat AWS by RedHat Google Cloud by RedHat

Fuente: (Wave, 2018)

En la Tabla 19 se selecciona el protocolo de IoT más adecuado para el tráfico de datos del proyecto, se evalúa HTTP, Websockets y MQTT, los cuales son más comunes en este tipo de aplicaciones, como lo menciona (Singh, 2017).

Tabla 19. Selección del Protocolo de IoT

Tipo	Requerimiento						Total
	IOT1	IOT2	IOT3	IOT4	IOT5	IOT6	
HTTP	0	0	1	1	1	0	3
Websockets	0	0	1	1	1	0	3
MQTT	1	1	1	1	1	1	6

En la selección de Protocolo de IoT, se tiene MQTT (Message Queue Telemetry Transport) con mayor puntuación sobre HTTP y Websockets, se evaluó de acuerdo con la comparación que realiza (Singh, 2017) entre los tres protocolos.

De acuerdo con (Yuan, 2017), MQTT está construido sobre la pila de protocolos TCP/IP y se ha convertido en un estándar para las comunicaciones de IoT, es un protocolo de red liviano y flexible con características adecuadas para los desarrolladores de aplicaciones, entre las cuales se tiene que:

- Al ser un protocolo liviano, permite implementarse sobre hardware de dispositivos altamente limitados y en redes con ancho de banda también limitado.

- Su flexibilidad posibilita que pueda soportar varios ambientes de aplicaciones para dispositivos y servicios de IoT.

MQTT utiliza un modelo de publicación y suscripción que define los tipos de entidades en la red, un intermediario de mensajes y un número de clientes. El intermediario es un servidor que recibe todos los mensajes de los clientes y luego los redirige hacia los clientes de destino.

Por consiguiente, para establecer la comunicación el cliente se conecta con el intermediario y se puede suscribir a cualquier "tema" de mensajes, esta conexión puede ser una conexión TCP/IP simple o una conexión TLS cifrada para mensajes confidenciales, el proceso de comunicación se explica a continuación:

1. El cliente publica el mensaje, sobre un tema, enviando el mensaje y el tema al intermediario.
2. Después, el intermediario redirige el mensaje a todos los clientes que están suscritos a ese tema.

Ya que los mensajes MQTT están organizados por temas, el desarrollador de aplicaciones es capaz de especificar que ciertos clientes sólo puedan interactuar con determinados mensajes. En la Figura 24 se representa un ejemplo, donde los sensores publican sus lecturas sobre del tema "sensor_data" y se suscriben al tema "config_change", de modo que las aplicaciones que guardan datos del sensor en una base de datos se suscriben al tema "sensor_data", y así puede recibir los valores enviados por los sensores y ajustar sus configuraciones como sensibilidad y frecuencia de muestreo; y a su vez publicar esos cambios en el tema "config_change" para que sean recibidos por los sensores respectivamente.

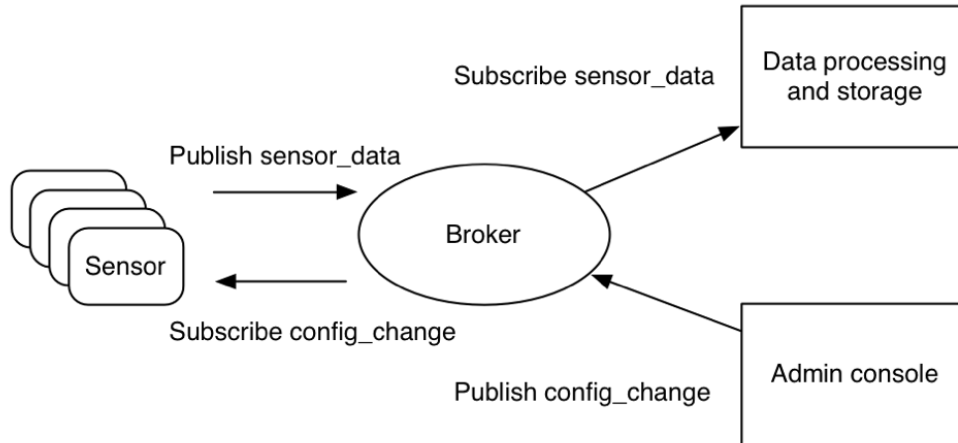


Figura 24. Modelo de publicación y suscripción de MQTT para los sensores de IoT
Fuente: (Yuan, 2017)

3.5 Recursos

En esta sección se detalla los tipos de recursos necesarios para la realización del presente proyecto de investigación, corresponde a los recursos: humanos, es decir, las personas que intervienen en el desarrollo de este proyecto; económicos, se refiere a los costos de los componentes eléctricos y electrónicos que se utilizan en la construcción del módulo electrónico; y tecnológicos, se refieren al software utilizado tanto en el módulo electrónico como en la plataforma de administración y gestión.

3.5.1 Recursos Humanos

Se menciona a las personas que intervienen en el presente proyecto de investigación, tales como los usuarios potenciales que fueron objeto de realización de la encuesta, el director del presente proyecto de investigación, los asesores y finalmente el desarrollador del proyecto, esta información se muestra en la Tabla 20.

Tabla 20. *Recursos Humanos presentes en el proyecto.*

RECURSOS HUMANOS	
FAMILIAS	Barrio San Roque de la Ciudad de Mira.
DIRECTOR	Ing. Fabián Cuzme. MSc.
ASESOR 1	Ing. Mauricio Domínguez. MSc.
ASESOR 2	Ing. Daniel Jaramillo. MSc.
DESARROLLADOR	Sr. Ricardo Madera.

3.5.2 Recursos Económicos

En la Tabla 21, se tiene una lista de los materiales necesarios en la realización del proyecto, sus costos unitarios y costos totales, de acuerdo con la página web de compras y ventas (mercadolibre, 2018).

Tabla 21. *Recursos Económicos presentes en el proyecto.*

RECURSOS ECONÓMICOS				
Cantidad	Elemento	Costo (USD)	Unitario	Costo Total (USD)
2	Arduino Nano	7.00		14.00
2	WiFi ESP-01S	4.50		9.00
3	ACS-712	7.50		22.50
2	RTC3231	3.50		7.00
1	Relevador 1 canal	2.00		2.00
1	Relevador 2 canales	3.75		3.75
2	Tomacorrientes	2.00		4.00
3	Enchufes	0.60		1.80
2	Cajas metálicas	3.00		6.00
1	Otros componentes electrónicos	10.00		10.00
TOTAL (USD)				80.05

Fuente: (mercadolibre, 2018)

3.5.3 Recursos Tecnológicos

Se refiere al software requerido para la ejecución del proyecto, tanto en el desarrollo y programación de los microcontroladores como en la infraestructura como servicio de AWS, esta información se detalla en la Tabla 22, y seguidamente se presentan las

características y funciones de cada uno. Debido a que se utiliza software libre, el costo de cada uno de ellos es cero.

Tabla 22. *Recursos Tecnológicos presentes en el proyecto.*

Software	Costo (USD)
IDE Arduino 1.8.5	0
Fritzing 0.9.3	0
Eagle 8.5.1	0
Brackets 1.8	0
PuTTY	0
AWS (Amazon Web Services)	0
TOTAL (USD)	0

3.5.3.1 Características y funciones de los recursos tecnológicos

IDE Arduino: Es un software de código abierto donde se puede escribir código de programación y cargarlo a la placa. El entorno está escrito en Java y basado en Processing, además posee una gran cantidad de librerías. (Arduino, Arduino, 2018)

Fritzing: Es una iniciativa de hardware de código abierto, donde se permite a los usuarios documentar sus prototipos, diseñar y fabricar pcbs profesionales. (Fritzing, 2017).

Eagle: Es un programa de automatización de diseño electrónico y permite que los desarrolladores de placas de circuito impreso conecten perfectamente diagramas esquemáticos, enrutamiento y ubicación de componentes. (Autodesk, 2018)

Brackets: Es un moderno editor de texto que facilita el diseño en el navegador. Está diseñado para desarrolladores de aplicaciones para usuario y diseñadores web, posee herramientas visuales enfocadas y soporte de preprocesador. (brackets.io, 2018)

PuTTY: Es un software de código abierto que está desarrollado y respaldado por un grupo de voluntarios, permite establecer conexiones SSH y telnet. (Putty, 2017)

AWS (Amazon Web Services): Proporciona una gran gama de servicios alojados en la nube pública, ofrece potencia de cómputo, almacenamiento de bases de datos, presentación de contenido, capacidad de escalamiento y crecimiento, entre otros. (AWS, 2018)

3.6 Diseño del sistema

Una vez determinados los requerimientos del sistema, y habiendo seleccionado los componentes de hardware y software a utilizarse, se procede al diseño del sistema. En esta sección se indica el funcionamiento del sistema por etapas, la interconexión de las placas programables y los módulos electrónicos, el diseño del circuito impreso, y la ejecución de las pruebas iniciales de comunicación del sistema para verificar su funcionamiento y realizar las correcciones necesarias.

3.6.1 Diagrama de bloques del sistema

El diagrama de bloques representado en la Figura 25, indica el funcionamiento del sistema, el cual inicia con la toma de datos a través de los sensores ACS712, luego son procesados en la placa Arduino, y enviados por medio del módulo WiFi hacia un servidor ubicado en la nube; este servidor mostrará la información requerida por el usuario, mediante una página web para su monitoreo.



Figura 25. Diagrama de bloques del sistema

3.6.2 Diagrama de flujo del sistema

La Figura 26 muestra de forma detallada los procesos y evaluaciones que va a realizar el sistema. En primera instancia se considera si el equipo a monitorizar está encendido mediante un sensado de puertos análogos, en caso de estar encendido se evalúa si se encuentra en el horario de apagado; si lo está, el tomacorriente se desactiva; por el contrario si no lo está, se toma los datos de los sensores ASC712 para conocer el consumo actual de energía eléctrica, y se envían mediante comunicación WiFi hacia el servidor remoto, el cual muestra la información en una página web, y a la vez determina si los valores recibidos se encuentran en un valor normal, en caso de no estarlo, se ejecutan acciones de alerta (mensaje de alerta en la página web y envío de correo electrónico). De igual manera, si el equipo a monitorizar no se encuentra encendido, se evalúa si se encuentra en horario de apagado, en este caso, el tomacorriente se desactiva; caso contrario termina el ciclo de procesos.

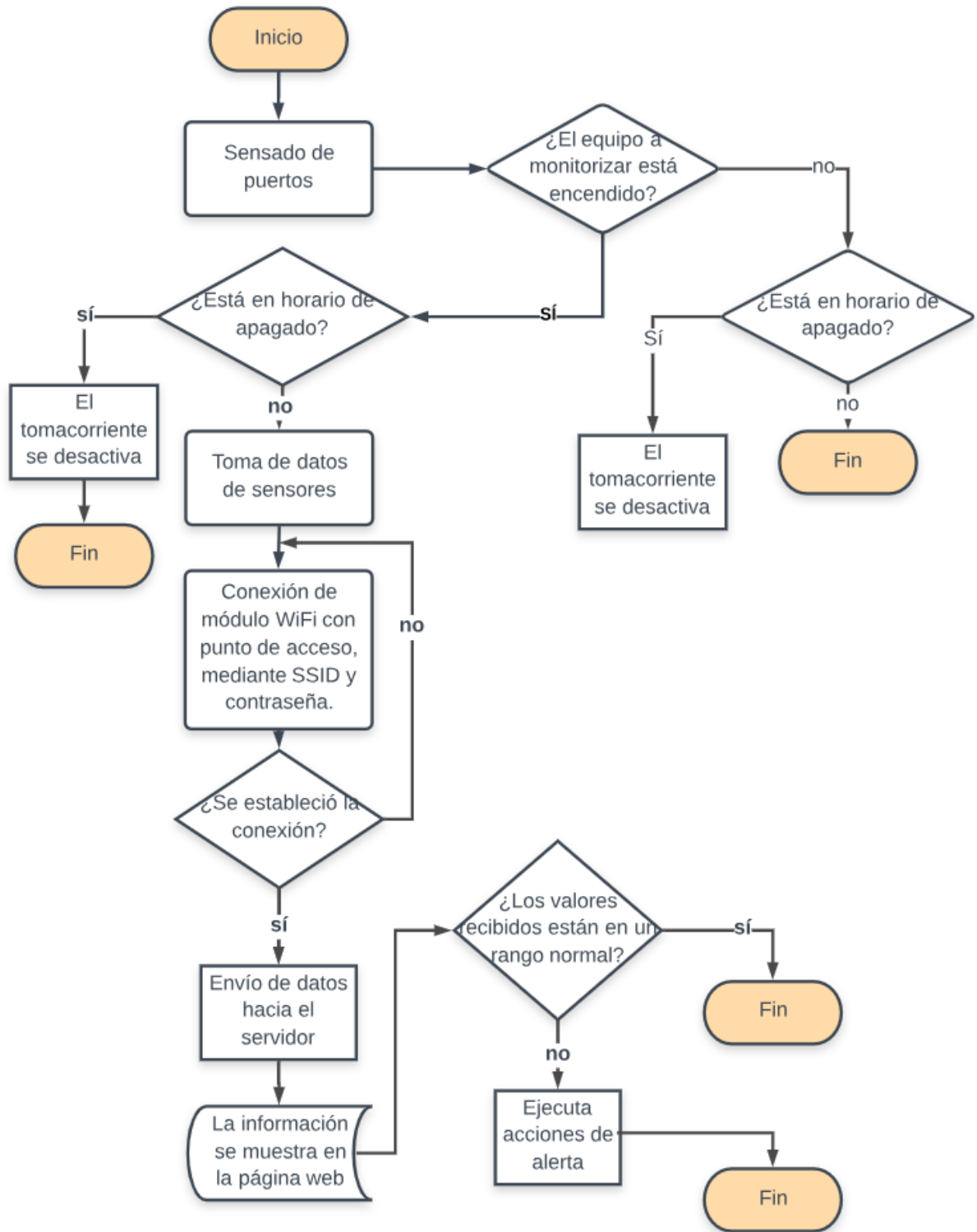


Figura 26. Diagrama de flujo del sistema

3.6.3 Diagrama de conexión de los componentes electrónicos

En la Figura 27 se muestra la interconexión de los elementos eléctricos y electrónicos, todo el circuito es energizado por el tomacorriente, para alimentar al Arduino se utiliza un conversor de corriente alterna a corriente continua de 5v, el módulo relevador se conecta directamente a los pines digitales D4 y D5 y en serie con los sensores de corriente, los cuales se conectan a los pines análogos A0 y A1 respectivamente. Es muy importante tener dos circuitos independientes en el tomacorriente, de modo que las mediciones de cada aparato conectado puedan ser diferenciadas.

En este sentido, el módulo WiFi se comunica mediante enlace serial con la placa controladora, y funciona como puente entre el módulo electrónico y la plataforma de administración y gestión. Cuando el prototipo electrónico no tenga que enviar datos, en horarios fuera de uso, entrará en un estado de bajo consumo, donde su tarifa eléctrica es prácticamente de cero, se agregará un pulsador en D3, en caso de necesitar activar el módulo electrónico en este estado; y otro pulsador en D2 para reiniciar el microcontrolador en caso de ser necesario.

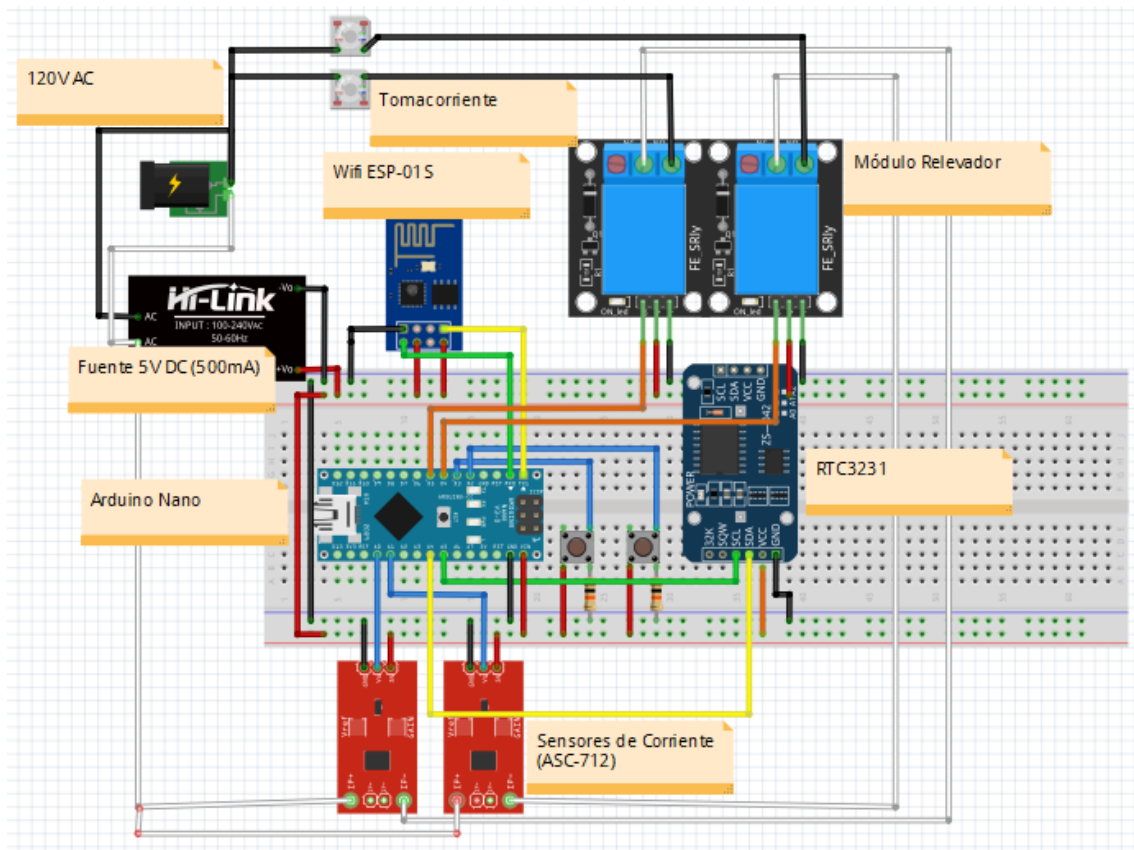


Figura 27. Conexión de los componentes electrónicos

3.6.4 Servidor XAMPP

Es un servidor independiente de código libre, que puede instalarse sobre cualquier sistema operativo. Un servidor XAMPP integra en una misma aplicación: servidor web, servidor de base de datos, intérpretes de lenguaje PHP y Perl, servidor de FTP, entre otros módulos, la instalación de este servidor se indica en el Anexo G.

El acrónimo XAMPP se forma de 5 siglas, las cuales se muestran en la Tabla 23, junto a su descripción.

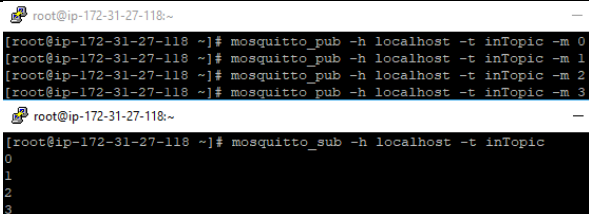


Tabla 23. Significado del acrónimo XAMPP

X	Cualquier sistema operativo, dependiendo si es Windows, Linux o MAC, se llamarían WAMPP, LAMPP y MAMPP respectivamente
A	Apache, es un servidor web de software libre con soporte para cualquier plataforma.
M	MySQL, es un sistema de gestión de base de datos relacional, multihilo y multiusuario, es decir varios procesos de ejecución y varios usuarios a la vez.
P	PHP, es un lenguaje de programación muy versátil y compatible con HTML, se utiliza en la creación de webs dinámicas
P	Perl, este lenguaje de programación toma características de C.

3.6.5 Pruebas iniciales

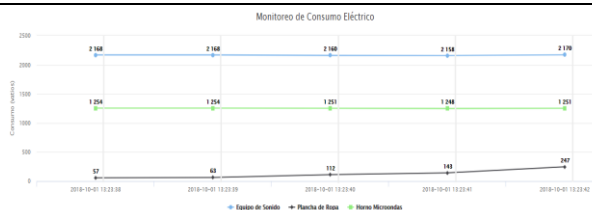
Durante el desarrollo del diseño del presente prototipo, se realizaron varias pruebas iniciales con la finalidad de verificar el funcionamiento de la parte principal del proyecto, que corresponde a la comunicación entre el módulo electrónico y el servidor remoto, estas pruebas se registran en la Tabla 24.

Tabla 24. Pruebas iniciales del prototipo

Prueba	Cumple	No cumple	Resultado obtenido																					
Envío de datos desde servidor MQTT	X		 <pre> root@ip-172-31-27-118:~ [root@ip-172-31-27-118 ~]# mosquitto_pub -h localhost -t inTopic -m 0 [root@ip-172-31-27-118 ~]# mosquitto_pub -h localhost -t inTopic -m 1 [root@ip-172-31-27-118 ~]# mosquitto_pub -h localhost -t inTopic -m 2 [root@ip-172-31-27-118 ~]# mosquitto_pub -h localhost -t inTopic -m 3 root@ip-172-31-27-118:~ [root@ip-172-31-27-118 ~]# mosquitto_sub -h localhost -t inTopic 0 1 2 3 </pre>																					
Respuesta de microcontrolador a los datos MQTT recibidos	X																							
Envío de datos de consumo eléctrico hacia la base de datos	X		 <table border="1"> <thead> <tr> <th>id</th> <th>valor</th> <th>tiempo</th> </tr> </thead> <tbody> <tr><td>1</td><td>2168</td><td>2018-10-01 13:23:38.000000</td></tr> <tr><td>2</td><td>2168</td><td>2018-10-01 13:23:39.000000</td></tr> <tr><td>1</td><td>57</td><td>2018-10-01 14:42:43.000000</td></tr> <tr><td>2</td><td>63</td><td>2018-10-01 14:42:44.000000</td></tr> <tr><td>1</td><td>1254</td><td>2018-10-01 15:36:24.000000</td></tr> <tr><td>2</td><td>1254</td><td>2018-10-01 15:36:25.000000</td></tr> </tbody> </table>	id	valor	tiempo	1	2168	2018-10-01 13:23:38.000000	2	2168	2018-10-01 13:23:39.000000	1	57	2018-10-01 14:42:43.000000	2	63	2018-10-01 14:42:44.000000	1	1254	2018-10-01 15:36:24.000000	2	1254	2018-10-01 15:36:25.000000
id	valor	tiempo																						
1	2168	2018-10-01 13:23:38.000000																						
2	2168	2018-10-01 13:23:39.000000																						
1	57	2018-10-01 14:42:43.000000																						
2	63	2018-10-01 14:42:44.000000																						
1	1254	2018-10-01 15:36:24.000000																						
2	1254	2018-10-01 15:36:25.000000																						

Visualización de gráfica con valores recibidos, en la página web

X



Con la finalidad de realizar las pruebas iniciales de funcionamiento, se implementó un servidor LAMP en una máquina local instalada en VMware, el sistema operativo fue CentOS 7, el cual es un sistema muy estable y bastante concurrente en el desarrollo de servidores, el levantamiento de la máquina virtual en la nube se encuentra en el Anexo C. Durante la creación de la instancia en la nube de Amazon se genera y descarga una llave privada que se usa posteriormente para acceder a la consola de la máquina virtual mediante un cliente SSH (PuTTY), este proceso se muestra en el Anexo D. Adicionalmente, de manera opcional se puede instalar un conjunto de paquetes para configurar un escritorio virtual, ya que el acceso a la instancia es por defecto mediante línea de comandos, los pasos se indican en el Anexo E, sin embargo, debe considerarse que esta opción consume muchos más recursos de procesamiento del servidor, por lo cual se decide trabajar únicamente mediante CLI.

Se realizaron pruebas del protocolo MQTT a través de la consola de comandos, se utilizó el bróker mosquitto, cuya instalación se encuentra en el Anexo F, este servidor publica mensajes en el tema “inTopic” con valores de 0, 1, 2 y 3, como se observa en la Figura 28. A su vez, el módulo WiFi ESP-01S, cuya instalación de librerías se encuentra en el Anexo B, se suscribe al mismo tema para recibir estos valores.

```
root@ip-172-31-8-107:~  
[root@ip-172-31-8-107 ~]# systemctl start mosquitto  
[root@ip-172-31-8-107 ~]# mosquitto_pub -t "inTopic" -m 1  
[root@ip-172-31-8-107 ~]# mosquitto_pub -t "inTopic" -m 3  
[root@ip-172-31-8-107 ~]# mosquitto_pub -t "inTopic" -m 0  
[root@ip-172-31-8-107 ~]# mosquitto_pub -t "inTopic" -m 2
```

Figura 28. Valores enviados por consola mediante protocolo MQTT

Una vez recibidos los valores del tema “inTopic”, el módulo WiFi se comunica por vía serial con la placa Arduino, y ésta a su vez controla los relevadores, los cuales actúan de acuerdo con el valor recibido, la Tabla 25 indica las acciones programadas para cada uno de los datos recibidos. La programación de Arduino Nano y Wifi ESP01S se indica en el anexo K.

Tabla 25. Estado de los relevadores 1 y 2 controlados por MQTT

Valores enviados mediante protocolo MQTT	Estado del Relevador 1	Estado del Relevador 2
0	Apagado	
1	Encendido	
2		Apagado
3		Encendido

La Figura 29 muestra el estado apagado del relevador 1, y el estado encendido del relevador 2, según los valores ‘0’ y ‘3’ enviados desde el servidor MQTT.

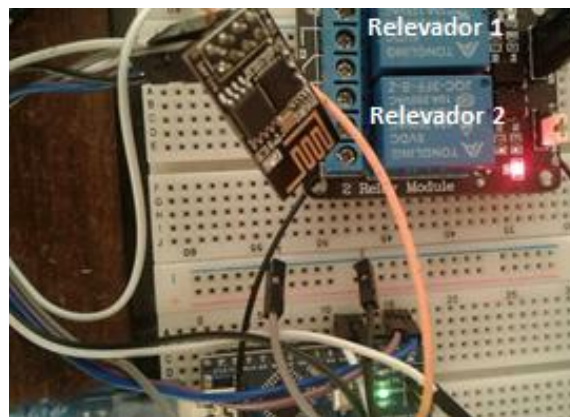


Figura 29. Valores ‘0’ y ‘3’ enviados por MQTT

La Figura 30 muestra el estado apagado del relevador 1, y el estado apagado del relevador 2, según los valores ‘0’ y ‘2’ enviados desde el servidor MQTT.

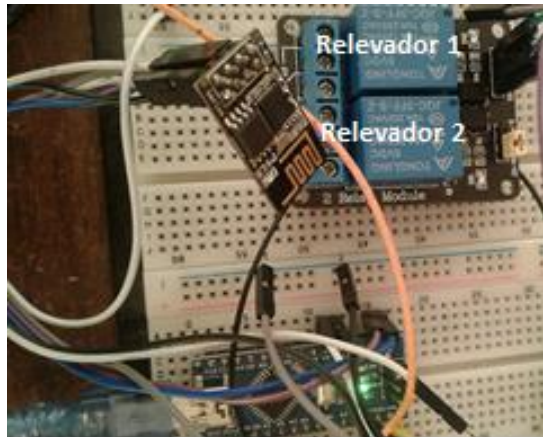


Figura 30. Valores '0' y '2' enviados por MQTT

La Figura 31 muestra el estado encendido del relevador 1, y el estado encendido del relevador 2, según los valores '1' y '3' enviados desde el servidor MQTT.

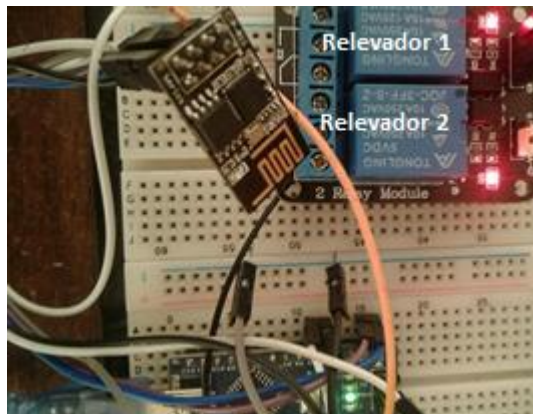


Figura 31. Valores '1' y '3' enviados por MQTT

La Figura 32 muestra el estado encendido del relevador 1, y el estado apagado del relevador 2, según los valores '1' y '2' enviados desde el servidor MQTT.

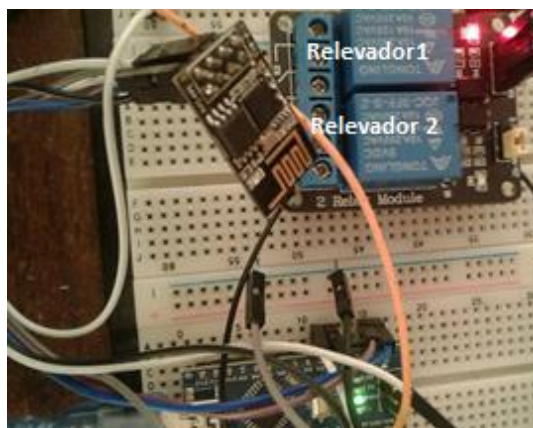


Figura 32. Valores '1' y '2' enviados por MQTT

Por otra parte, los sensores de corriente envían los datos, hacia la base de datos mysql, y son publicados en una página web mediante gráficas dinámicas. En la Figura 33 se visualiza el aspecto inicial de la página web, posteriormente se agregó una interfaz con botones para controlar el estado de los relevadores presentes en el módulo electrónico, y así controlar el encendido y apagado de los dispositivos conectados, además se incorporó botones para ingresar nuevos equipos, valores de consumo, frecuencia de medición y horarios de apagado y encendido del módulo electrónico, para guardar esta información es necesario la creación de una base de datos, la cual se muestra en el Anexo H, y los scripts del sistema operativo RHEL se encuentra en el anexo K.

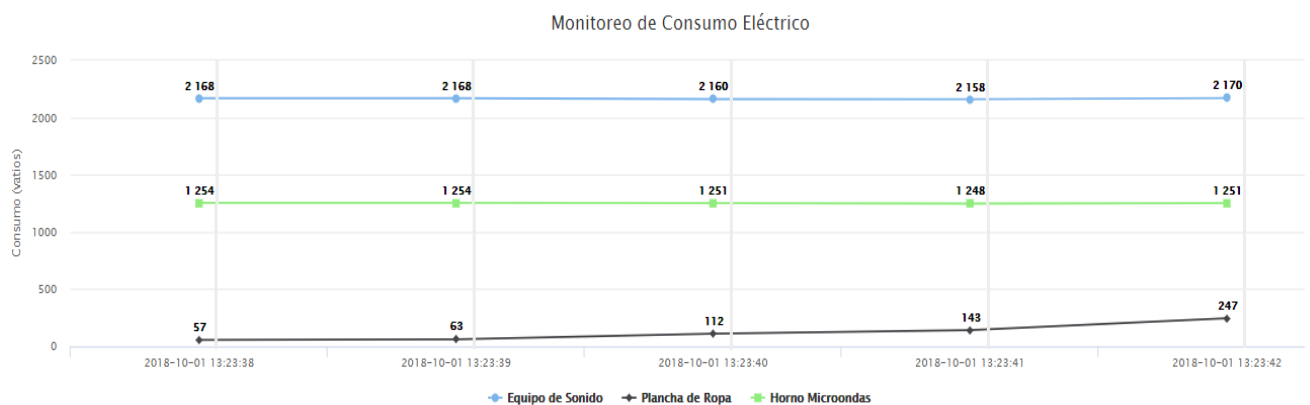


Figura 33. Pruebas de envío de datos de los sensores de corriente ASC-712

En las pruebas iniciales se utilizó el conjunto de librerías de Highchart para visualizar las gráficas de consumo eléctrico enviadas desde el módulo electrónico, sin embargo, debido a la necesidad de una línea temporal que permita filtrar datos de acuerdo a fechas, se decidió utilizar finalmente el conjunto de librerías de Amchart, cuyo diseño se muestra en la Figura 34, se puede ver en la parte superior una barra dinámica que permite ver registros en un intervalo de tiempo variable, los valores correspondientes a cada punto se van mostrando conforme el usuario desplaza el cursor a través de la gráfica. El acceso a la página web se realiza escribiendo la dirección plataforma.cf en el navegador, este DNS

se asigna mediante el sitio web gratuito www.freenom.com y los pasos a seguir se muestran en el Anexo I.

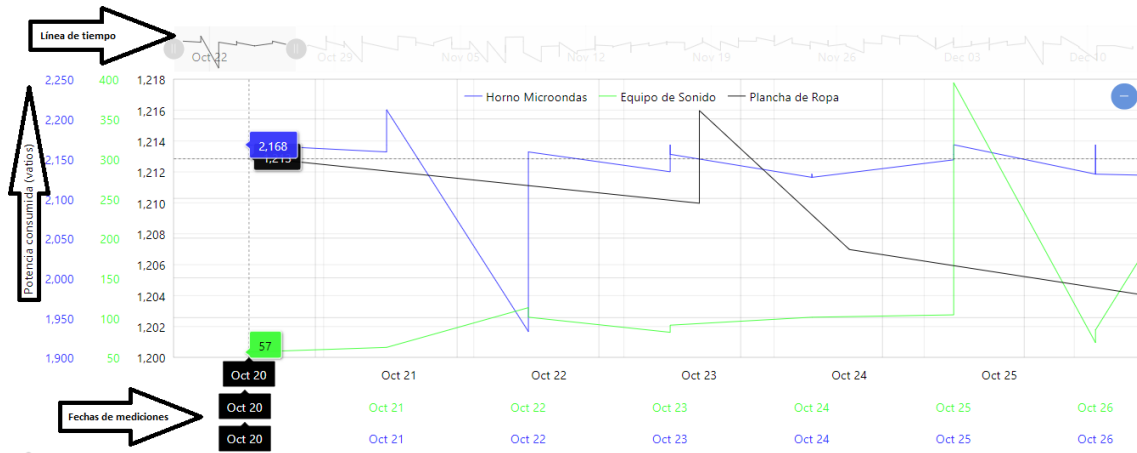


Figura 34. Librería Amchart con línea de tiempo

3.6.6 Diseño del circuito impreso

El diseño del circuito impreso se realizó en el programa Eagle, este programa posee una gran cantidad de librerías de Arduino y módulos compatibles, en la Figura 35 se muestra las conexiones de todos los componentes, se observa la placa programable Arduino Nano, 2 sensores de corriente ACS712, 1 relevador de 2 canales, 1 regulador 7833, 1 módulo WiFi ESP01S, y un módulo de reloj RTC.

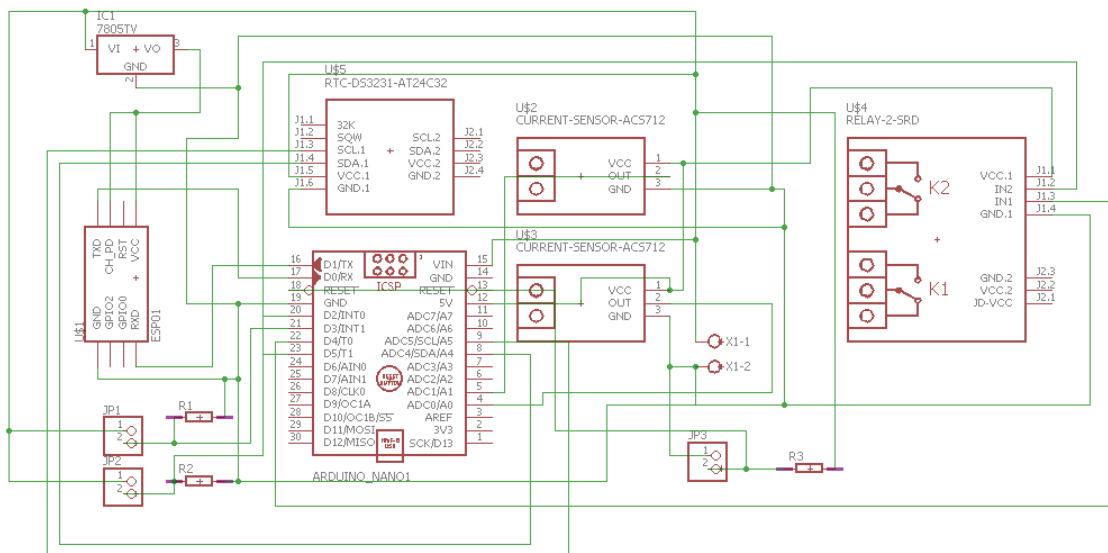


Figura 35. Diagrama de conexiones del circuito impreso

Una vez conectados los componentes, se procede a la realización de las pistas que serán impresas sobre la lámina de cobre y sobre las cuales se adhieren con estaño los componentes electrónicos, este diagrama se muestra en la Figura 36.

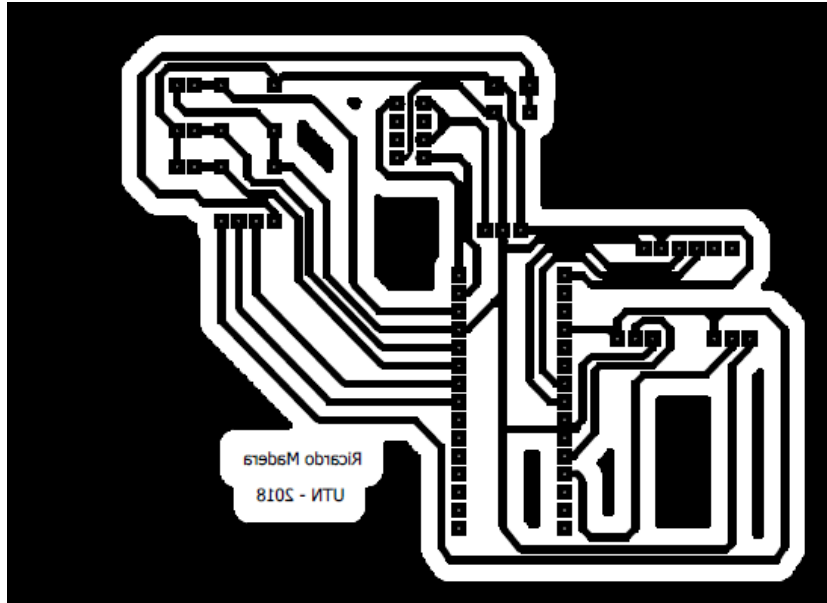


Figura 36. Diagrama de pistas del circuito impreso

CAPÍTULO IV

IMPLEMENTACIÓN Y PRUEBAS

En este capítulo se documentan las pruebas realizadas y resultados obtenidos luego de la implementación del prototipo, se evalúa el funcionamiento adecuado del sistema, tomando en cuenta, rendimiento de las placas programables, fiabilidad de los datos enviados, manejo adecuado de datos en el servidor, representación correcta de la información en la página web e interactividad funcional con el módulo electrónico, y finalmente se realiza un análisis de los costos de producción del prototipo electrónico.

4.1 Implementación

El módulo electrónico se coloca sobre un tomacorriente convencional, para lo cual se utiliza un tomacorriente doble para cajetín donde se van a conectar los dispositivos eléctricos, el cual se observa en la Figura 37a; y una caja metálica de 10cm x 8cm x 5cm donde se adapta el circuito electrónico y los demás componentes, la cual se muestra en la Figura 37b.

Debido a que el módulo electrónico puede gestionar como máximo a dos equipos, es necesario utilizar dos módulos electrónicos con la finalidad de gestionar a los tres equipos mencionados en el alcance del proyecto.



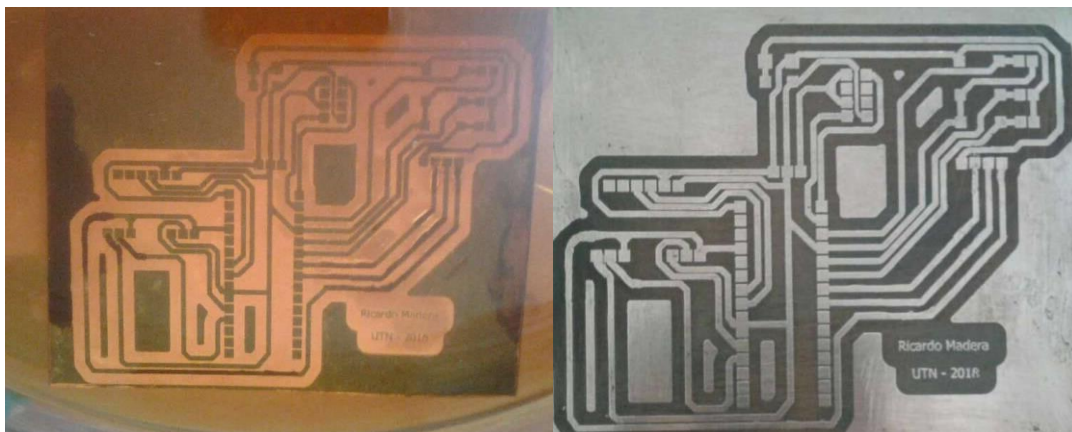
a)

b)

Figura 37. a) Tomacorriente doble para cajetín. b) Caja metálica

4.1.1 Elaboración de los circuitos impresos

Con el diseño del circuito, elaborado en el programa Eagle, se procede a copiarlo sobre una lámina de cobre, misma que es sumergida en una solución de percloruro férrico, como muestra la Figura 38a. Este ácido produce una rápida oxidación sobre el metal, haciendo que se diluya, pero sin afectar las zonas donde se encuentra la tinta. Al finalizar el proceso, solamente queda el cobre por donde estaban las pistas de tinta, como se muestra en la Figura 38b, de este modo ya se puede soldar los componentes electrónicos por el lado contrario.



a)

b)

Figura 38. a) Colocación de la placa impresa en percloruro férrico. b) Pistas de material conductor

Finalmente, en la Figura 39 se muestra la ubicación de los componentes electrónicos instalados sobre el circuito impreso.

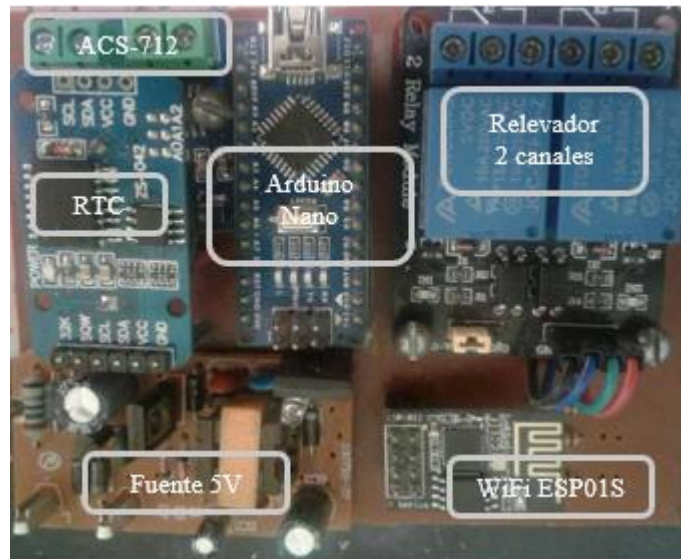


Figura 39. Elementos electrónicos soldados a la placa de cobre.

La Figura 40 muestra el módulo electrónico finalizado, el circuito electrónico se encuentra debidamente sellado en el interior de la caja, y el usuario puede conectar sus aparatos eléctricos al prototipo.



Figura 40. Módulo electrónico terminado

4.2 Pruebas del módulo electrónico

Se evaluarán parámetros como: portabilidad del módulo, capacidad de las tarjetas programables, saturación de datos, funcionamiento prolongado, precisión de las lecturas, entre otros.

4.2.1 Portabilidad

El módulo electrónico puede ser colocado prácticamente sobre cualquier tomacorriente del hogar para comenzar a operar inmediatamente, al ser un elemento que permanecerá inmóvil durante su funcionamiento, no representa mayores problemas en cuanto a movimientos bruscos o caídas, sin embargo, se debe tomar en cuenta que exista cobertura de la red WiFi en dicho punto. Además, en caso de necesitar colocarse en otro domicilio, se debe ingresar el SSID y contraseña de la nueva red WiFi en el programa del WiFi ESP-01S.

4.2.2 Microcontrolador

Las pruebas evalúan la capacidad del microcontrolador en lo referente al espacio de almacenamiento y memoria dinámica, esta información se despliega luego de compilar el programa en el IDE de Arduino, y sus valores se muestran en la Figura 41. Estas pruebas se aplican sobre el módulo electrónico encargado de gestionar dos equipos, ya que este se considera como caso crítico al compilar una programación más extensa que el módulo electrónico que gestiona un sólo equipo.

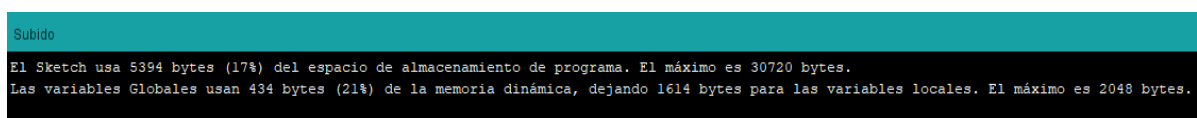


Figura 41. Espacio de almacenamiento y memoria dinámica ocupados en el microcontrolador.

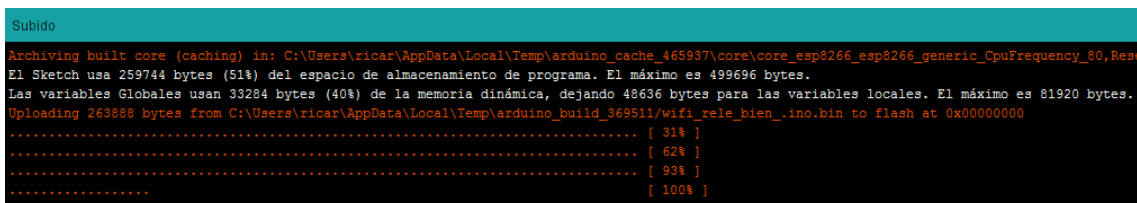
Como resultados de esta prueba se obtienen los recursos utilizados por el programa:

- 5394 bytes de espacio de almacenamiento, lo que corresponde al 17% de un total de 30720 bytes.
- 434 bytes de memoria dinámica, que corresponde al 21 % de un total de 2048 bytes.

En base a los valores mostrados en la Figura 41, los recursos utilizados por la programación realizada están muy por debajo de la capacidad máxima del Arduino Nano, por lo tanto, no existen problemas al compilar y ejecutar el código del programa.

4.2.3 ESP-01S (WiFi)

Al igual que en el Arduino Nano, se evalúa el módulo WiFi ESP-01S encargado de gestionar dos equipos, ya que es quien tendrá más carga de datos. En la Figura 42 se verifica la cantidad de recursos utilizados por el programa, estos valores se obtienen luego de la compilación realizada.



```
Subido
Archiving built core (caching) in: C:\Users\ricar\AppData\Local\Temp\arduino_cache_465937\core\core_esp8266_esp8266_generic_CpuFrequency_80,Res
El Sketch usa 259744 bytes (51%) del espacio de almacenamiento de programa. El máximo es 499696 bytes.
Las variables Globales usan 33284 bytes (40%) de la memoria dinámica, dejando 48636 bytes para las variables locales. El máximo es 81920 bytes.
Uploading 263888 bytes from C:\Users\ricar\AppData\Local\Temp\arduino_build_369511\wifi_rele_bien_ino.bin to flash at 0x00000000
..... [ 31% ]
..... [ 62% ]
..... [ 93% ]
..... [ 100% ]
```

Figura 42. Espacio de almacenamiento y memoria dinámica ocupados en el módulo WiFi.

Como resultados de esta prueba se obtienen los siguientes resultados:

- 259744 bytes de espacio de almacenamiento, lo que corresponde al 51% de un total de 499696 bytes.
- 33248 bytes de memoria dinámica, que corresponde al 40% de un total de 81920 bytes.

De la prueba mostrada en la Figura 42, se concluye que el módulo WiFi utiliza, alrededor de la mitad de su capacidad máxima, esto se debe a que se cargan dos librerías externas, una para la conexión WiFi (ESP8266WiFi.h) y otra para el protocolo de comunicación MQTT (PubSubClient.h).

4.2.4 Funcionamiento continuo y fidelidad de las mediciones de consumo eléctrico

Se evaluó el funcionamiento del módulo electrónico durante un tiempo prolongado, dejándose conectado durante un día, en ese tiempo la respuesta fue inmediata a la gestión realizada desde la página web. De igual manera se evaluó la fiabilidad de las mediciones de consumo de energía eléctrica, esto se realizó comparando los datos obtenidos con los sensores ASC-712, con datos medidos con un multímetro como se observa en la Figura 43. Para esta prueba se midió el consumo en amperios de una plancha de ropa, llegando a la conclusión de que varía muy poco su resultado.

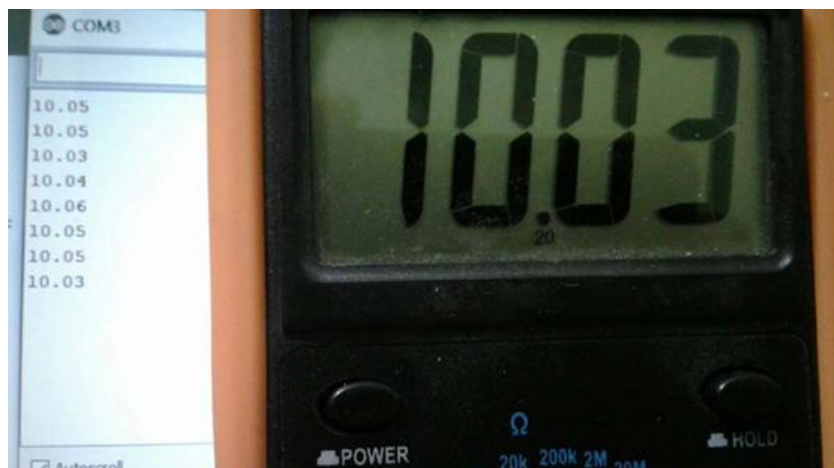


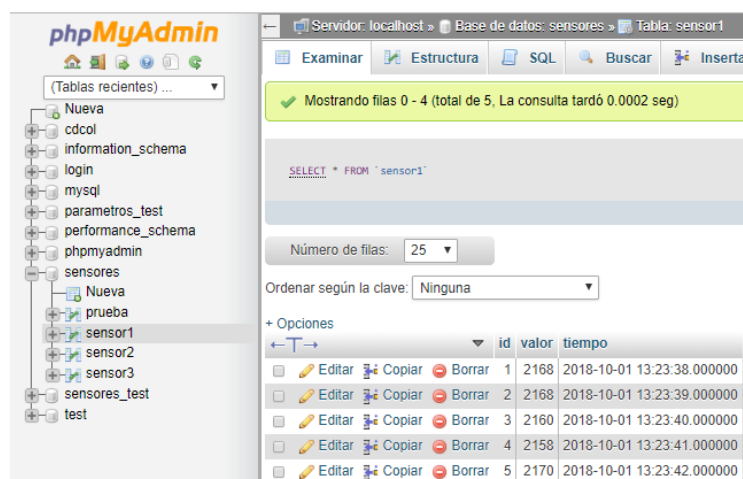
Figura 43. Comparación de la medición de sensor ASC712 y multímetro.

4.3 Pruebas de la Plataforma de Administración y Gestión

En el servidor remoto de administración y gestión implementado, se verificó que los datos lleguen y se muestren correctamente, se programaron alertas en caso de un consumo excesivo de energía eléctrica, el cual será parametrizado previamente, de acuerdo con la potencia nominal dada por el fabricante; además se creó un *login* para acceder a la página web y una interfaz para controlar el encendido y apagado del módulo electrónico. Finalmente, en la parte inferior de la página web se encuentra un historial de eventos de cada equipo ingresado.

4.3.1 Lecturas del gestor de bases de datos

Se evidencia que el servidor de base de datos reciba correctamente las lecturas provenientes del módulo electrónico. En la Figura 44 se muestra los valores enviados por parte del sensor 1, y en la Figura 45 los valores enviados por el sensor 2. Cada una de las tablas de los sensores tienen tres columnas; la primera se denomina *ID*, y representa el número de registro en la base de datos; la columna *valor* muestra el consumo eléctrico de un equipo, en vatios; y la columna tiempo muestra la hora y fecha en la cual fue recibido el dato.



The screenshot shows the phpMyAdmin interface. On the left is a tree view of databases and tables. The 'sensores' database is expanded, showing tables 'sensor1', 'sensor2', and 'sensor3'. The 'sensor1' table is selected. The main area shows a table view of the 'sensor1' table with 5 rows. The table has columns 'id', 'valor', and 'tiempo'. The data is as follows:

id	valor	tiempo
1	2168	2018-10-01 13:23:38.000000
2	2168	2018-10-01 13:23:39.000000
3	2160	2018-10-01 13:23:40.000000
4	2158	2018-10-01 13:23:41.000000
5	2170	2018-10-01 13:23:42.000000

Figura 44. Valores recibidos por parte del sensor 1 y almacenados en la base de datos

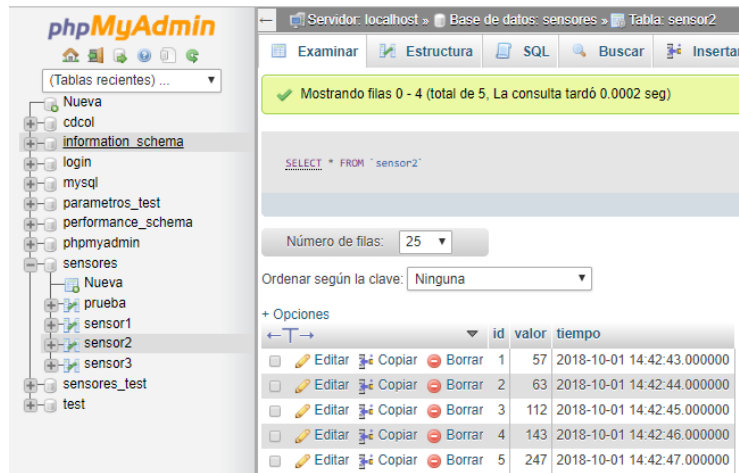


Figura 45. Valores recibidos por parte del sensor 2 y almacenados en la base de datos

4.3.2 Interfaz de acceso a la página web

Se considera una medida de seguridad para evitar el acceso no autorizado al sistema, ya que solo se puede ingresar escribiendo un usuario y contraseña válidos, los cuales se encuentran en la base de datos llamada *login*, como se muestra en la Figura 46. Una vez que se ha validado la información mencionada anteriormente, se concede el acceso al sistema de administración y gestión de los equipos eléctricos y electrónicos.

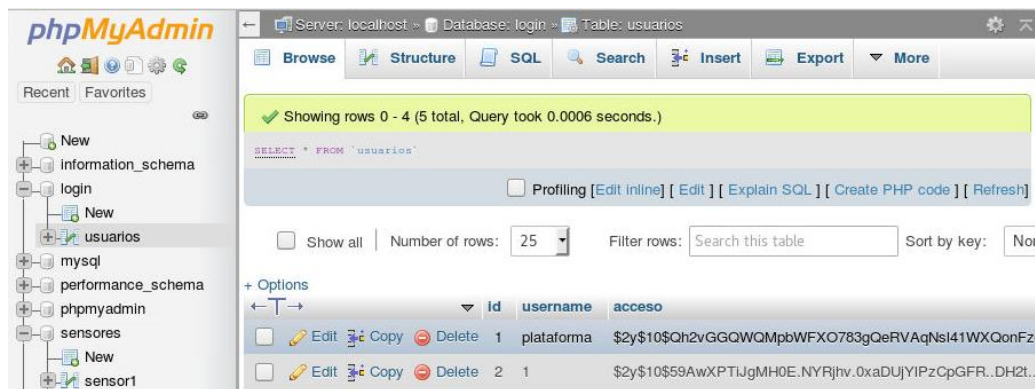


Figura 46. Base de datos para usuarios y contraseñas.

En la Figura 47 se observa el diseño de la interfaz de login y se verifica su adecuado funcionamiento, llenando los campos *usuario* y *contraseña* con la información almacenada en la tabla *usuarios*.

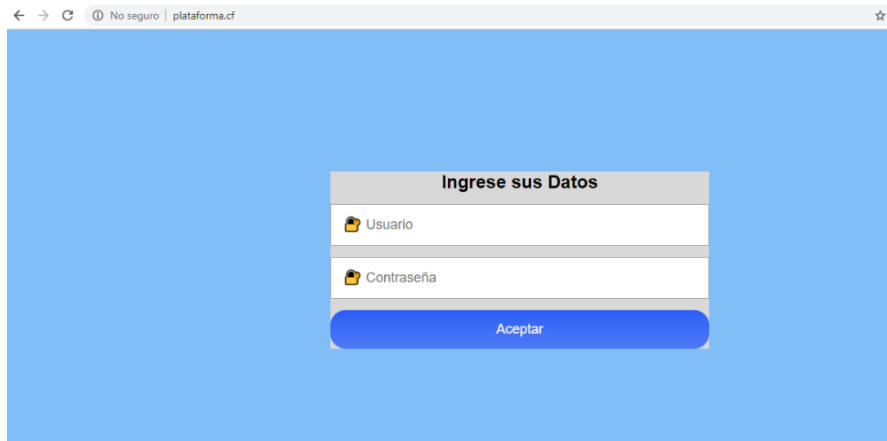


Figura 47. Interfaz de login

4.3.3 Página Web para control de la plataforma de administración y gestión

Una vez que se ha verificado el usuario y contraseña ingresados, se obtiene acceso a la página web del sistema, tal como se muestra en la Figura 48, donde se refleja la información almacenada en la base de datos, a manera de gráfica. En la parte izquierda se muestra diferentes escalas de consumo eléctrico, de acuerdo con los equipos monitoreados, mientras que en la parte inferior se muestran las escalas de tiempo, además en la parte superior se ubica la línea temporal, que permite ver intervalos específicos de consumo de electricidad. Mediante estas mediciones se logró identificar que el “Horno Microondas” estaba consumiendo una potencia mucho mayor a la normal que viene preestablecida por el fabricante, y se observa en la Figura 49 (1450W). Debido a que las mediciones indicadas en la página web son alrededor de 2200W, se estima un consumo eléctrico adicional de más del 60%.

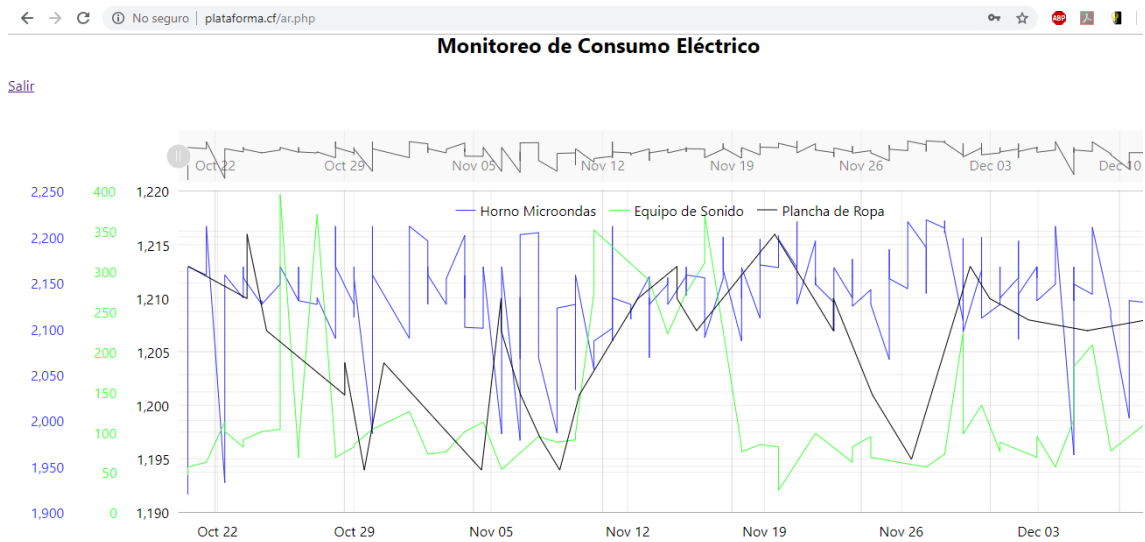


Figura 48. Página Web de monitoreo de consumo eléctrico

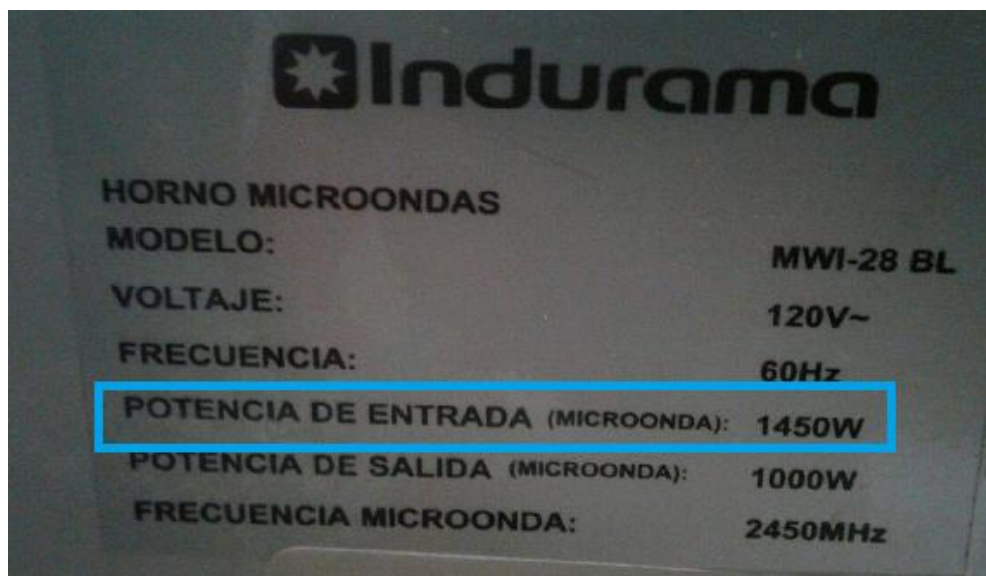


Figura 49. Potencia nominal del Horno Microondas

Al hacer click en el botón *Ingresar nuevos equipos*, ubicado en la parte superior izquierda de la página web, se despliega varios campos de texto, donde se ingresa los nuevos equipos que van a ser monitoreados, junto a sus potencias nominales, es decir, la potencia que debe consumir normalmente, la cual viene dada por el fabricante; y la frecuencia de toma de datos, que va desde un minuto hasta una hora entre mediciones.

Además, se puede establecer el tiempo de apagado del módulo electrónico, ingresando a la opción *Periodo de Apagado*, todos estos parámetros se indican en la Figura 50.

Ingresar nuevos equipos

Horno Microondas	1450	vatios	Frecuencia de medición (minutos)	1
Equipo de Sonido	410	vatios	Frecuencia de medición (minutos)	1
Plancha de Ropa	1200	vatios	Frecuencia de medición (minutos)	1

Aceptar

Periodo de apagado Desde 15:50 hasta 16:30

15 : 50 Hora de apagado

16 : 30 Hora de encendido

Aceptar

Figura 50. Menú de ingreso de nuevos equipos, potencias nominales, frecuencia de medición y horario de apagado

Mientras que, en la parte superior derecha de la página web se encuentra la interfaz que permite controlar el paso de energía eléctrica hacia los dispositivos monitoreados, como se observa en la Figura 51. Asimismo, en la parte inferior se tiene un registro histórico de alertas emitidas en cada uno de los equipos asociados al sistema realizado, como se muestra en la Figura 52.

SISTEMA DE MONITORIZACIÓN DE CONSUMO ELÉCTRICO

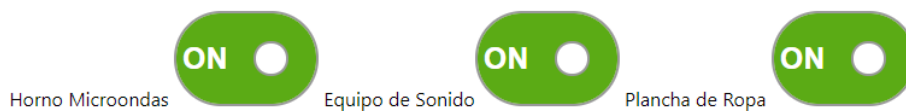


Figura 51. Interfaz de flujo de energía eléctrica

Horno Microondas : 0 alertas registradas. Equipo de Sonido : 0 alertas registradas. Plancha de Ropa : 0 alertas registradas.

Figura 52. Registro de eventos de equipos asociados al sistema realizado

4.3.4 Acciones de Alerta

Los mensajes de alerta de la plataforma de administración y gestión se despliegan cuando las mediciones sobrepasan la potencia nominal del aparato monitoreado. El sistema emite un mensaje de alerta en la página web del sistema de monitorización de consumo eléctrico, como muestra la Figura 53; y el envío de un correo electrónico programado, tal como se presenta en la Figura 54.

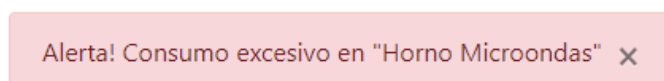


Figura 53. Mensaje de alerta de la página web



Figura 54. Correo electrónico de alerta

4.4 Mediciones del contador eléctrico

Se colocó un contador eléctrico dentro de la vivienda para medir el consumo de energía de todo el hogar, con la finalidad de evidenciar los resultados con y sin el prototipo implementado.

4.4.1 Mediciones sin implementar el prototipo.

Las mediciones de consumo eléctrico se realizaron durante un mes, donde los datos fueron registrados cada semana y se muestran en la Figura 55, estos valores pueden comprobarse en el Anexo J. Las mediciones fueron las siguientes:

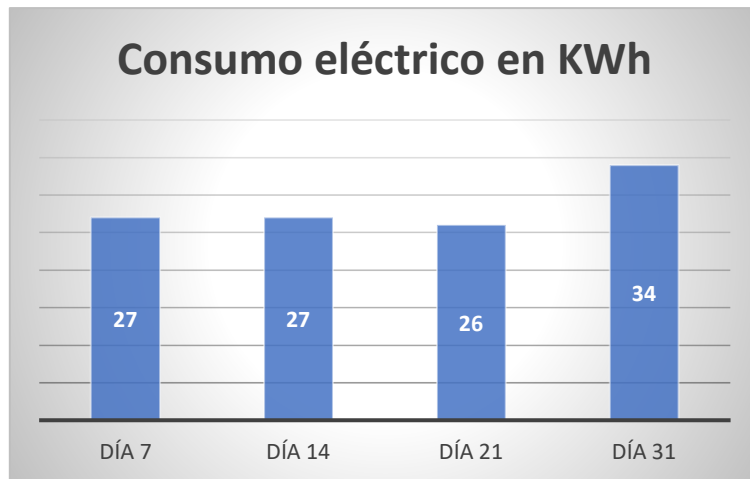


Figura 55. Consumo eléctrico durante un mes, sin implementar el prototipo.

4.4.2 Mediciones de consumo eléctrico con la implementación del prototipo

Las mediciones de consumo eléctrico se realizaron durante un mes, donde los datos fueron registrados cada semana y se muestran en la Figura 56, estos valores pueden comprobarse en el Anexo J. Las mediciones fueron las siguientes:

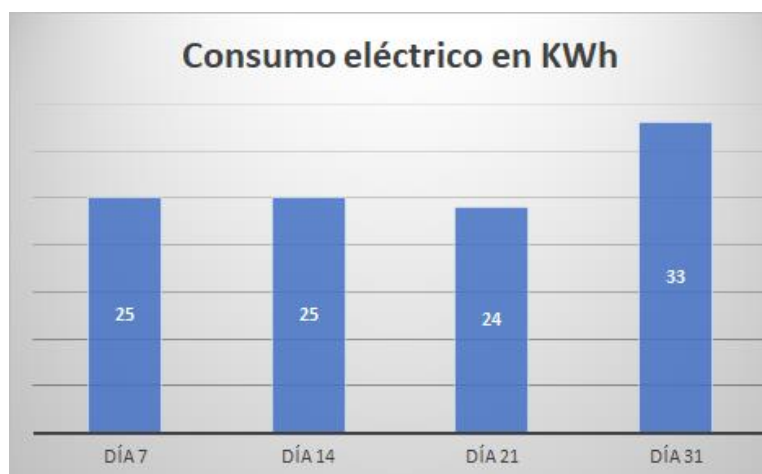


Figura 56. Consumo eléctrico durante un mes, con implementación del prototipo.

Realizando un análisis de las mediciones presentadas en las Figuras 55 y 56, se concluye que el consumo de energía eléctrica global se redujo de 114KWh a 107KWh en un mes, lo que representa un ahorro del 6.14%.

4.5 Vida útil del prototipo

La plataforma de administración y gestión, al ser un subsistema basado únicamente en software, no tiene un tiempo de vida útil definido, por lo tanto puede mantenerse operativa durante un largo periodo de tiempo.

El tiempo de vida útil del módulo electrónico se determina según la durabilidad de cada uno de sus componentes, esto se determina de acuerdo con los ciclos de escritura y borrado de los microcontroladores, por ejemplo, si un componente tiene 10000 *write/erase cycles* significa que puede ser regrabado 10000 veces antes de empezar a fallar. En cuanto a los relevadores, su vida útil se calcula según el número máximo de conmutaciones que puede realizar correctamente, estos valores vienen dados en los datasheet de cada uno de ellos.

Como se puede observar en la Tabla 26, cada componente tiene un número muy elevado de ciclos de escritura y borrado, además se considera la realización de mantenimiento al módulo electrónico cada cuatro meses, donde se evaluará el funcionamiento de cada uno de sus componentes. Debido a lo anterior, se concluye cualitativamente que es un prototipo de muy alta duración, se estima un tiempo de vida útil de aproximadamente 15 años, basado en los adelantos tecnológicos constantes que menciona (Rodriguez, 2015).

Tabla 26. *Vida útil de los componentes principales del módulo electrónico*

Componente electrónico	Vida útil
Atmega 328P (Arduino Nano)	Write/Erase Cycles: 10,000
Relay	Conmutations: 100000
ESP8266 (WiFi)	Write/Erase Cycles: 10000
DS3231 (RTC)	Write/Erase Cycles: 1000000

Fuente: (Arduino, 2018)

4.6 Costo de componentes electrónicos importados

A continuación, se hace un análisis del costo de los componentes electrónicos del prototipo, en base a los precios de una de las tiendas on-line más grandes del mundo llamada Aliexpress, sus productos son importados directamente desde China, lo cual implica un ahorro considerable en cada uno de ellos, aunque el tiempo de llegada del producto se sitúa entre uno y dos meses. Así, se evalúa los costos considerando importar una unidad, diez unidades, y cien unidades. De igual manera, se considera el impuesto de la Empresa Pública “Correos del Ecuador” (CDE E.P.) a las importaciones por comercio electrónico, que determina en la Resolución Nro. CDE-EP-CDE-EP-2017-0012-R del 15 de septiembre de 2017, que los envíos con un peso de hasta 2000 gramos deben pagar la cantidad de \$3,51 incluido IVA.

En ese sentido, se analiza el costo de producir una unidad de prototipo electrónico, tomando en cuenta los diferentes volúmenes de importación, en la Tabla 27 se clasifica cada uno de los componentes necesarios, considerando que algunos son comprados en el país, por lo que no se suma el impuesto CDE E.P.

Tabla 27. Registro de precios de componentes electrónicos importados de China

Cantidad	Componente electrónico	Precio incluido envío (1u) [USD]	Precio incluido envío (10u) [USD]	Precio incluido envío (100u) [USD]	Impuesto CDE E.P. [USD]
1	Arduino Nano	\$2.28	\$23.79	\$196.84	\$3.51
1	WiFi ESP-01S	\$1.79	\$15.26	\$121.12	\$3.51
1	RTC3231	\$1.06	\$15.02	\$125.46	\$3.51
2	ACS712	\$2.29	\$22.38	\$190.86	\$3.51
1	Relevador canales	2 \$1.04	\$14.82	\$123.46	\$3.51 (1u y 10u) \$10,53 (100u)
1	Otros	\$2.00	\$20.00	\$200.00	\$3.51
1	Tomacorriente doble	\$2.00	\$20.00	\$200.00	
2	Enchufe	\$0.60	\$6.00	\$60.00	
1	Caja metálica	\$3.00	\$30.00	\$300.00	

SUMA	\$18.95	\$192.65	\$1768.60
IMPUESTO CDE E.P.	\$21.06	\$21.06	\$28.08
TOTAL	\$40.01	\$213.71	1796.68

Fuente: (AliExpress, 2018)

Para obtener el costo de una unidad producida, se divide el valor total de producción para el número de unidades producidas, de este modo se obtiene los siguientes resultados:

- Al producir una unidad, el costo unitario es \$40.01
- Al producir diez unidades, el costo unitario es \$21.37
- Al producir cien unidades, el costo unitario es \$17.97

De lo expuesto anteriormente, se concluye que, a mayor cantidad de producción, el costo unitario de cada prototipo disminuye.

Considerando, de acuerdo con la encuesta realizada a 39 familias del barrio San Roque de la ciudad de Mira, provincia del Carchi, en promedio en un hogar se utiliza con mayor frecuencia los siguientes electrodomésticos: TV (23%), lavadora (18%), equipo de sonido (17%), computadora (14%), microondas (13%), plancha de ropa (13%). Según este análisis, se necesitarían tres módulos electrónicos para gestionar el consumo eléctrico de los dispositivos mencionados, ya que cada módulo electrónico puede gestionar hasta dos equipos.

En este caso, con la utilización de tres módulos electrónicos, tomando el valor unitario de la producción de cien unidades, se tiene un costo total de \$53.91.

4.7 Relación costo beneficio del prototipo

Según menciona (Comercio, 2018), el valor de la energía eléctrica en Ecuador es de 9,33 centavos por kWh, y un hogar en la sierra consume en promedio, alrededor de

150kWh al mes, lo que representa \$14 dólares mensuales de facturación. En las pruebas realizadas con la implementación del prototipo desarrollado, se obtuvo una reducción en el consumo eléctrico de 6.14% en un mes, tomando en cuenta que los dispositivos gestionados fueron: horno microondas, equipo de sonido y plancha de ropa.

Sin embargo, en las encuestas realizadas se determinó que aparte de los tres equipos tomados como modelo para las pruebas, se tienen otros que también son utilizados con frecuencia, estos son: televisión, computadora y lavadora. Para realizar una estimación del ahorro con todos los equipos se toman los valores de consumo stand-by de la Tabla 2 del Capítulo II, que corresponde a la fundamentación teórica.

Considerando los equipos usados en las pruebas, se tiene una potencia stand-by total de 14,4wh (equipo de sonido) + 3.08wh (microondas) = 17.48 wh. La plancha de ropa no se toma en cuenta, ya que no tiene consumo stand-by.

De la misma manera se toman los tres equipos resultantes de las encuestas, que también son frecuentemente utilizados, para un total de 8.9wh (computadora) + 6.6wh (TV) + 4.5wh (lavadora) = 20wh.

De este modo, si 17.48 wh representan un ahorro de electricidad de 6.14 %, realizando una regla de tres simple, se tiene un valor de 7.02% para 20wh. En total se esperaría un ahorro energético de alrededor de 13,16%, es decir se tendría un consumo total de 130.26 kWh, con un ahorro monetario de aproximadamente dos dólares mensuales. Este es el ahorro mínimo esperado, ya que no se toma en cuenta los equipos que se encuentren encendidos cuando no están siendo utilizados, y serían desconectados oportunamente por el prototipo de acuerdo con los horarios programados, produciendo un ahorro energético adicional.

Por lo tanto, dividiendo el costo de los módulos electrónicos para el ahorro eléctrico generado, este valor sería cubierto como máximo en dos años y tres meses. Contando, además con un sistema de prevención de averías en los equipos, en caso de encontrarse consumiendo una potencia mayor a la normal.

CONCLUSIONES

- La implementación del sistema de monitoreo de consumo eléctrico planteado permite visualizar datos acerca del consumo energético real de varios dispositivos eléctricos y electrónicos de un hogar, de esta manera el usuario puede discernir entre un valor de consumo de energía en rangos normales y un valor de consumo de energía excesivo debido a irregularidades presentes en sus equipos.
- La comunicación bidireccional entre módulo electrónico y plataforma de administración y gestión se realiza en tiempo real, debido a la adecuada integración de los componentes de hardware y software seleccionados mediante el Estándar IEEE 29148; y la utilización del protocolo de comunicación MQTT, ideal para aplicaciones de IoT, ya que utiliza muy pocos recursos de procesamiento y ancho de banda.
- Al implementar la plataforma de administración y gestión del sistema propuesto, en una instancia de una empresa proveedora de Infraestructura como Servicio, se puede prescindir del espacio físico y consumo de energía eléctrica que representa tener un servidor local, además es posible escalar y asignar recursos dinámicamente al servidor.
- Las pruebas de funcionamiento se realizaron durante dos meses, donde se consiguió un ahorro en el consumo eléctrico de alrededor del 6%, al contar con un sistema de desconexión automático de los dispositivos eléctricos y

electrónicos, de acuerdo con los horarios programados a través de la página web del servidor remoto.

- El costo de producción de un módulo electrónico puede reducirse en más del 50%, si se considera importar los sensores, actuadores y tarjetas programables directamente desde China, país donde se produce esta tecnología, por medio de la tienda online Aliexpress.

RECOMENDACIONES

- En diseños futuros se puede realizar un circuito más pequeño y eficiente, que se encuentre integrado directamente en la pared de la vivienda, en lugar de ser sobrepuesto como plantea este prototipo.
- Se puede realizar la parametrización de dispositivos desde la página web, de modo que se puedan agregar nuevos dispositivos para monitorizar, sin tener que editar el código fuente de los scripts.
- Se considera implementar un certificado digital de seguridad en la página web para tener un cifrado de la información transmitida.
- Si se desea aplicar el presente proyecto en entornos empresariales o industriales se deberá redefinir los requerimientos iniciales y realizar una selección minuciosa de los componentes electrónicos, tomando en cuenta todas sus características de tal manera que puedan adaptarse entre ellos y cumplir con las funcionalidades propuestas.
- En nuevos diseños se puede implementar otro tipo de tecnología de comunicación a parte de WiFi, por ejemplo, Zigbee.

REFERENCIAS

- Alarcón, C., Guisao, Y., & Guzmán, H. (2017). *Explorando Cloud Computing: Definiciones, Plataformas y Tendencias*. Medellín.
- AliExpress. (2018). Obtenido de <https://es.aliexpress.com/>
- Ananthakrishnan. (2015). *Routing Protocols for Power Line Communications (PLC)*.
- Aragón, D. d. (09 de septiembre de 2015). *Agencia Insular de Energía de Tenerife*. Obtenido de Guía sobre consumo energético de aparatos domésticos: http://www.agenergia.org/files/resourcesmodule/@random493ea37fa7d61/1228843832_Consumo_energ_Aptos_domesticos.pdf
- Arduino. (2018). *Arduino*. Obtenido de <https://www.arduino.cc/en/Main/Software>
- Arduino.cl. (2018). Obtenido de <http://arduino.cl/arduino-nano/>
- Arrivals, D. N. (2018). *Doogee New Arrivals*. Obtenido de <http://www.dx.com/p/esp-01s-esp8266-serial-wi-fi-wireless-transceiver-module-for-arduino-439163#.WsRDYi7wbIU>
- Arroyo, D. (25 de agosto de 2016). *El Telégrafo*. Obtenido de Consumos Fantasma Residencial: <http://www.eltelegrafo.com.ec/noticias/punto-de-vista/1/consumos-fantasma-residenciales>
- Autodesk. (2018). Obtenido de <https://www.autodesk.com/products/eagle/overview>
- AWS. (2018). *AWS*. Obtenido de <https://aws.amazon.com>
- Baimel, D., Tapuchi, S., & Baimel, N. (2016). *Smart Grid Communication Technologies-Overview, Research Challenges and Opportunities*. Israel.
- BBC. (02 de 04 de 2018). *América Noticias*. Obtenido de <http://www.americatv.com.pe/noticias/tecnologia/electrodomesticos-que-consumen-mas-energia-cuando-estan-apagados-n315659>
- brackets.io. (2018). Obtenido de <http://brackets.io/>
- Burgoon, M. (23 de enero de 2017). *Rockwell Automation*. Obtenido de http://www.rockwellautomation.com/es_CAR/news/blog/detail.page?pagetitle=M%C3%A1s-competitividad-gracias-a-la-concienciaci%C3%B3n-energ%C3%A9tica-%7C-Blog&content_type=blog&docid=d382f177782180973c62c3d8f885b573
- CENACE. (2015). Centro Nacional de Control de Energía_Planificación estratégica 2015-2017.
- CERTSI. (2017). *Guía de Seguridad en Protocolos Industriales Smart Grid*. España.
- Clarín. (30 de 12 de 2015). Obtenido de https://www.clarin.com/sociedad/energia-artefactos-consumen-electrodomesticos-ola_de_calor-cortes_de_luz_0_HkMUqCuv7g.html

- Clarín. (30 de 12 de 2015). *Clarín*. Obtenido de https://www.clarin.com/sociedad/energia-artefactos-consumen-electrodomesticos-ola_de_calor-cortes_de_luz_0_HkMUqCuv7g.html
- Co.Ltd., S. A. (2015). *ESP-01 WiFi Module-Datasheet*. Obtenido de <https://ecksteinimg.de/Datasheet/Ai-thinker%20ESP-01%20EN.pdf>
- ELECTRICIDAD, R. (25 de abril de 2016). *Electricidad. La revista energética de Chile*. Obtenido de <http://www.revistaei.cl/2016/04/25/experiencia-internacional-en-la-instalacion-de-medidores-inteligentes/#>
- Falvo, M., Martirano, L., Sbordone, D., & Bocci, E. (2013). *Technologies for Smart Grids: a brief review*. Roma: IEEE.
- Finnegan, M., & Carey, S. (03 de mayo de 2018). *COMPUTERWORLDUK*. Obtenido de <https://www.computerworlduk.com/it-vendors/microsoft-azure-vs-amazon-aws-public-cloud-comparison-which-cloud-is-best-for-enterprise-3624848/>
- Fritzing. (2017). *Fritzing*. Obtenido de <http://fritzing.org/home/>
- Gorelik, E. (2013). *Cloud Computing Models*. Cambridge: Massachusetts Institute of Technology.
- Guardián, E. (07 de 03 de 2017). *El Guardián. Grupo Informativo*. Obtenido de <https://elguardian.cr/entretenimiento/los-10-aparatos-electricos-que-mas-consumen-estando-apagados/>
- Güngör, V. C., Sahin, D., Kocak, T., Ergüt, S., Buccella, C., Cecati, C., & Hancke, G. P. (2011). *Smart Grid Technologies: Communication Technologies and Standards*.
- Huang, E. (mayo de 2013). *Generando ahorros a través de una gestión energética eficiente*. Obtenido de <http://www.sgs.com/~media/Global/Documents/White%20Papers/SGS%20SSC%20Energy%20Management%20White%20Paper%20May%202013%20ES%20Update%20A4%20EN%2013%2007.ashx>
- IDAE. (2011). *Evolución tecnológica y prospectiva de costes de las energías renovables*. Madrid.
- IIIA-CSIC. (2018). *Unidad de Desarrollo Tecnológico en Inteligencia Artificial*. Obtenido de Metodología de desarrollo de software. El Modelo en V o de Cuatro Niveles.: <http://www.iiia.csic.es/udt/es/blog/jrodriguez/2008/metodologia-desarrollo-sotware-modelo-en-v-o-cuatro-niveles>
- INEC. (2016). *Información Ambiental en hogares 2016*. Obtenido de http://www.ecuadorencifras.gob.ec/documentos/web-inec/Encuestas_Ambientales/Hogares/Hogares_2016/Documento%20tecnico.pdf
- INER. (2015). *Revista Científica del desarrollo energético. (Instituto Nacional de Eficiencia Energética y Energías Renovables). Energía Limpia*.

- ISO, IEC, & IEEE. (2011). *International Standard ISO/IEC/IEEE 29148*.
- ISO/IEC/IEEE. (2011). *INTERNATIONAL STANDARD ISO/IEC/IEEE 29148*. Switzerland .
- Kuzlu, M., Pipattanasomporn, M., & Rahman, S. (2014). *Communication network requirements for major smart grid applications in HAN, NAN and WAN*. Virginia: Elsevier.
- Kuzlu, M., Pipattanasomporn, M., & Rahman, S. (2015). *Review of Communication Technologies for Smart Homes/Building Applications*. Arlington: IEEE.
- Llamas, L. (21 de mayo de 2017). *Ingeniería, informática y diseño*. Obtenido de <https://www.luisllamas.es/arduino-wifi-esp8266-esp01/>
- mercadolibre. (2018). Obtenido de <https://www.mercadolibre.com.ec/>
- mercadolibre. (s.f.). Obtenido de https://articulo.mercadolibre.com.ve/MLV-517507747-toma-corriente-doble-con-cajetin-metal-y-tapa-2d-_JM
- Moviltronics. (2018). *Moviltronics*. Obtenido de <https://moviltronics.com.co/sensores/175-sensor-efecto-hall-44e.html>
- Mukundha, C., & Vidyamadhuri, K. (2017). *Cloud Computing Models: A Survey*. Hyderabad: Research India Publications.
- Ochoa, C. (11 de noviembre de 2013). *netquest*. Obtenido de <https://www.netquest.com/blog/es/que-tamano-de-muestra-necesito>
- Ouafi. (2018). *mercadolibre*. Obtenido de https://articulo.mercadolibre.com.ve/MLV-517507747-toma-corriente-doble-con-cajetin-metal-y-tapa-2d-_JM
- Patagoniatec. (2018). *Sensor De Corriente ACS712 5-20-30A*. Obtenido de <http://saber.patagoniatec.com/sensor-de-corriente-ac712-5-20-30a-ptec-arduino-pic/>
- Pérez, T. (04 de mayo de 2015). *sipse.com*. Obtenido de ¿Sabes cuánto consumen tus electrodomésticos viejos?: <https://sipse.com/novedades/electromesticos-viejos-suben-el-coste-de-luz-149966.html>
- Putty. (2017). *Putty*. Obtenido de <https://www.putty.org/>
- Raddatz, D. (02 de 2017). *Trends for Smart Grid Automation and Industry 4.0 Integration*. Obtenido de <http://pace.ee.uwa.edu.au/wp-content/uploads/2017/02/Smart-Grid-Communication-.pdf>
- Relica, Y. (2014). *ESPE*. Obtenido de ANÁLISIS Y EVALUACIÓN DE PARÁMETROS DE CALIDAD EN SERVICIOS CLOUD COMPUTING EN EL ECUADOR : <https://repositorio.espe.edu.ec/bitstream/21000/7956/1/AC-ET-ESPE-047745.pdf>
- Rico, E. (13 de junio de 2016). *Ermesh*. Obtenido de <http://www.ermesh.com/aprender-protocolo-mqtt-parte-1/>

- Rodriguez, M. (2015). *euroresidentes* . Obtenido de <https://www.euroresidentes.com/tecnologia/avances-tecnologicos/la-tecnologia-avanza-mas-rapido-cada-dia>
- Rose, K., Eldridge, S., & Chapin, L. (2015). *La Internet de las Cosas - Una breve reseña*. Internet Society (ISOC).
- Salazar, J., & Silvestre, S. (2016). *Universidad Politécnica de Cataluña*. Obtenido de Internet de las Cosas: https://upcommons.upc.edu/bitstream/handle/2117/100921/LM08_R_ES.pdf
- Singh, R. (26 de julio de 2017). *Linkedin*. Obtenido de Internet de las cosas: batalla de los protocolos (HTTP vs. Websockets vs. MQTT): <https://www.linkedin.com/pulse/internet-things-http-vs-websockets-mqtt-ronak-singh-cspo/>
- SketchUp. (2018). *SketchUp*. Obtenido de <https://www.sketchup.com/es>
- Tafunell, X. (2018). *LA REVOLUCIÓN ELÉCTRICA EN AMÉRICA LATINA*. Barcelona: Universitat Pompeu Fabra.
- TightVNC. (2017). *TightVNC Software*. Obtenido de <https://www.tightvnc.com>
- Tsampasis, E., Bargiotas, D., Elias, C., & Sarakis, L. (2016). *Communication challenges in Smart Grid*. Grecia: EDP Sciences.
- Vargas, I., Bravo, P., Bustamante, R., Castillo, B., Gonzales, C., Guzmán, F., . . . Rojas, A. (10 de julio de 2015). *DocSlide*. Obtenido de Sensores de Corriente: <https://docslide.com.br/documents/sensores-de-corriente.html>
- Wave, R. (25 de junio de 2018). *Rogue Wave Software*.
- Wei, Z., Zhang, Y., Jin, F., Yin, W., & Wu, L. (2014). *Standby Power Consumption Estimation For Energy Saving Service*.
- Weins, K. (22 de noviembre de 2017). *InfoWorld*. Obtenido de <https://www.infoworld.com/article/3237566/cloud-computing/cloud-pricing-comparison-aws-vs-azure-vs-google-vs-ibm.html>
- Yuan, M. (04 de 10 de 2017). *IBM*. Obtenido de <https://www.ibm.com/developerworks/ssa/library/iot-mqtt-why-good-for-iot/index.html>

SI _____

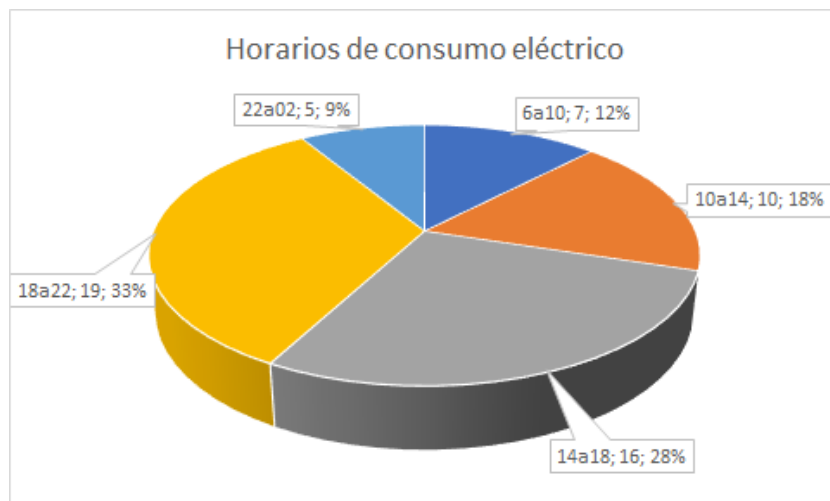
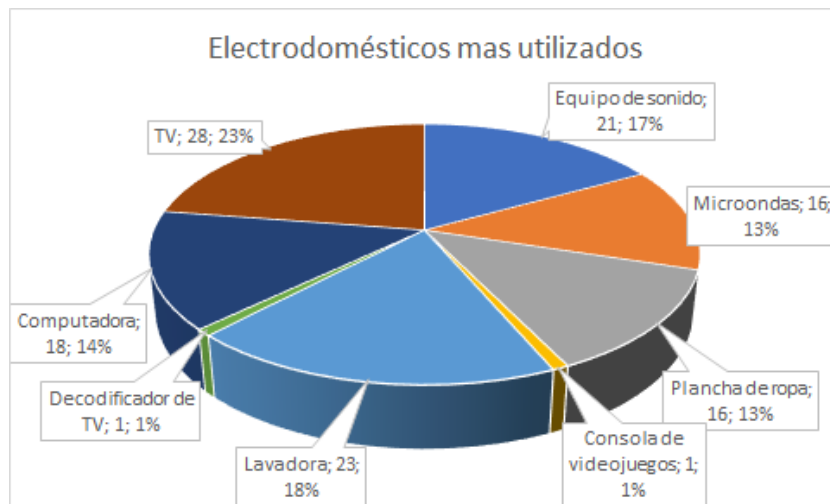
NO _____

6.- Estaría interesado/a en un dispositivo automático de control de energía para sus electrodomésticos

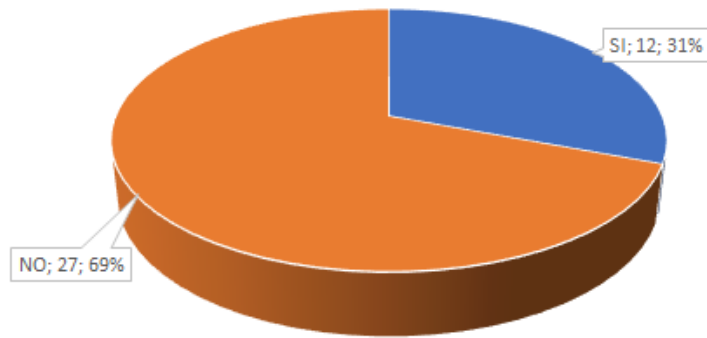
SI ____

NO__

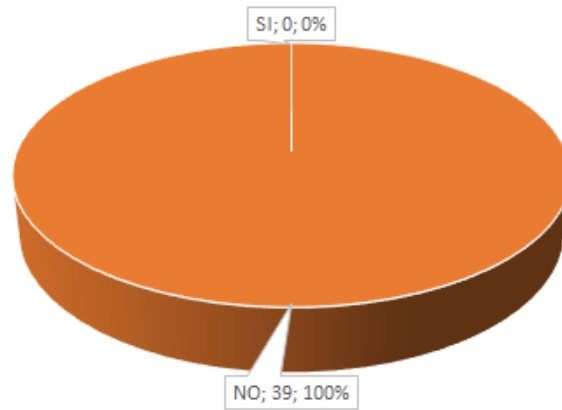
De la encuesta presentada y efectuada a 39 familias, se obtienen los siguientes resultados.



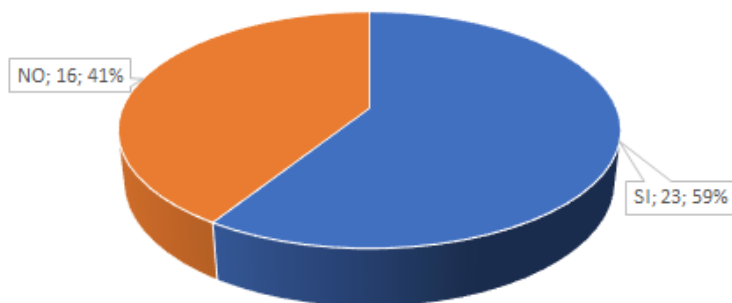
¿Desconecta sus dispositivos luego de usarlos?

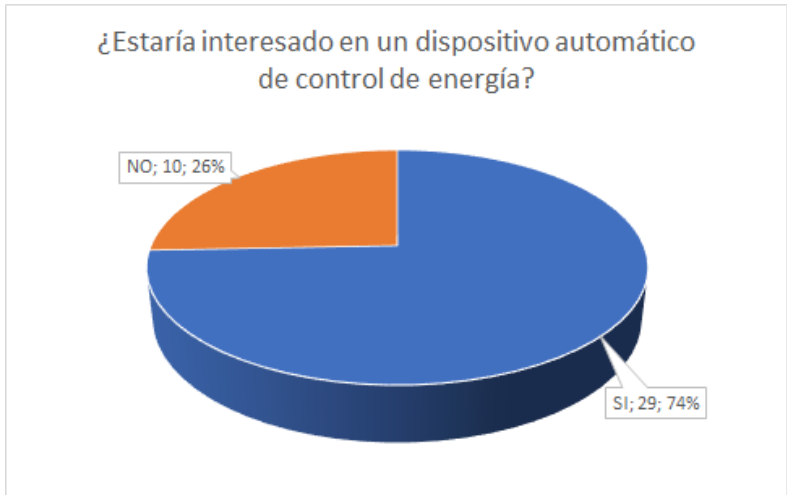
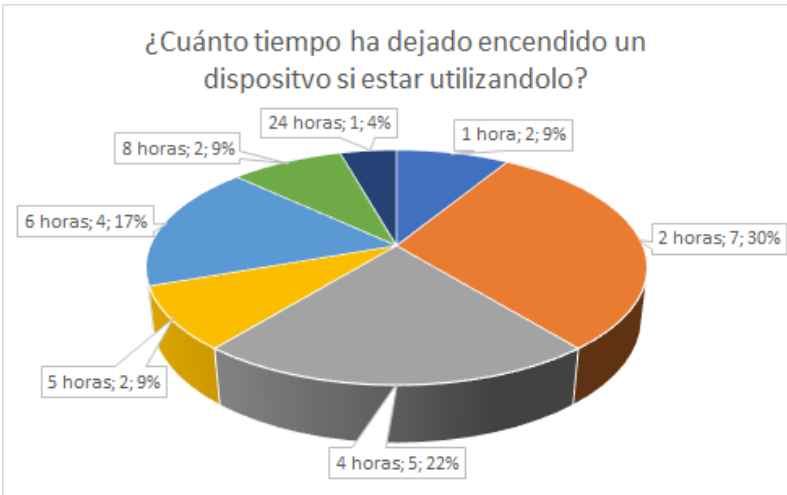
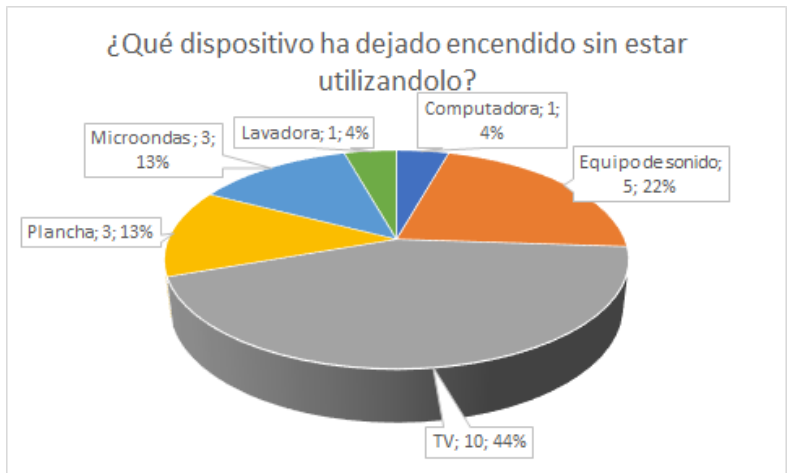


¿Realiza mantenimiento a sus dispositivos?



¿Ha dejado encendido un dispositivo cuando no lo está utilizando?





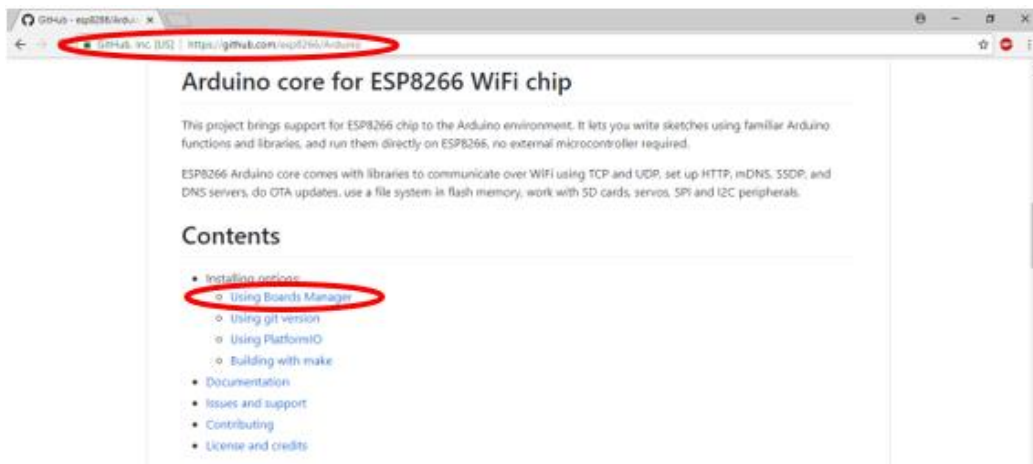
Anexo B. Instalación de librerías para Arduino y ESP-01S

Instalación de Librería WiFiESP8266

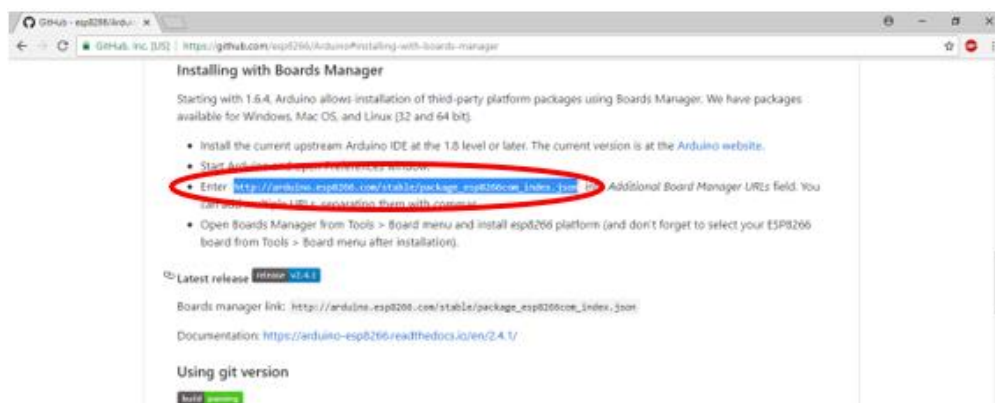
Esta librería es necesaria para controlar las operaciones relacionadas con la conexión wifi y transmisión de datos del módulo ESP-01S. Para instalar esta librería se debe agregar la dirección del repositorio al IDE de Arduino.

Procedimiento:

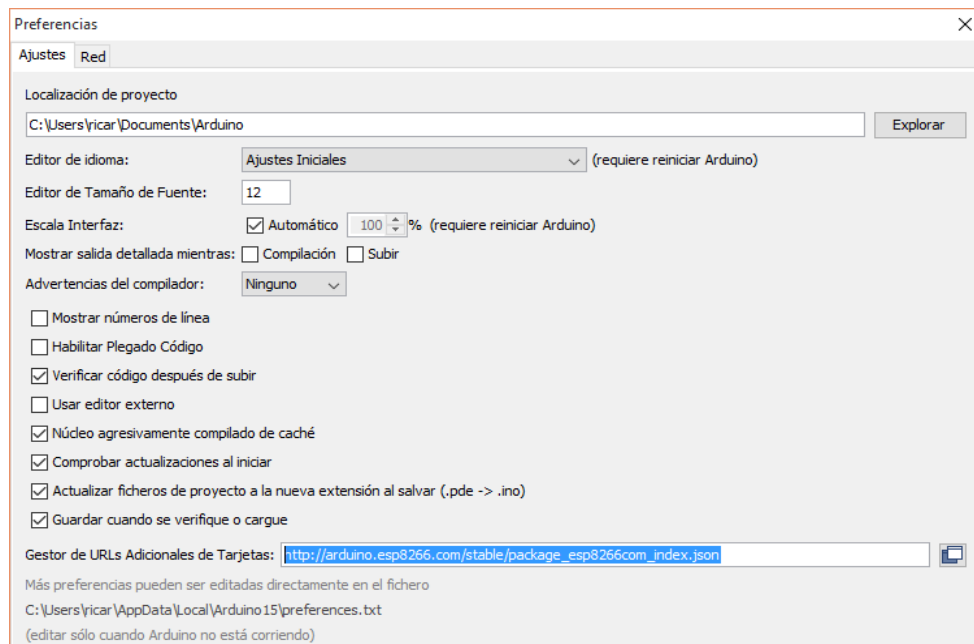
1. Entrar al repositorio de GitHub <https://github.com/esp8266/Arduino> y dirigirse a la opción de instalación Using Boards Manager.



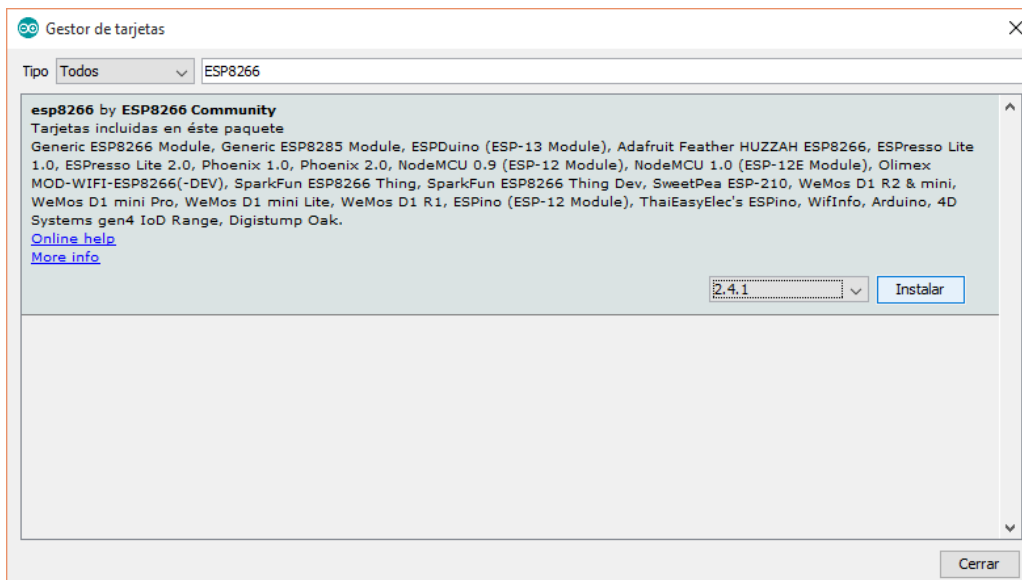
2. De la sección Installing with Boards Manager copiar la dirección http://arduino.esp8266.com/stable/package_esp8266com_index.json



3. En la sección de Preferencias del IDE de Arduino, copiar la línea anterior al Gestor de URLs Adicionales de Tarjetas,



4. En el apartado Gestor de tarjetas, buscar la librería ESP8266 y luego dar click en Instalar.

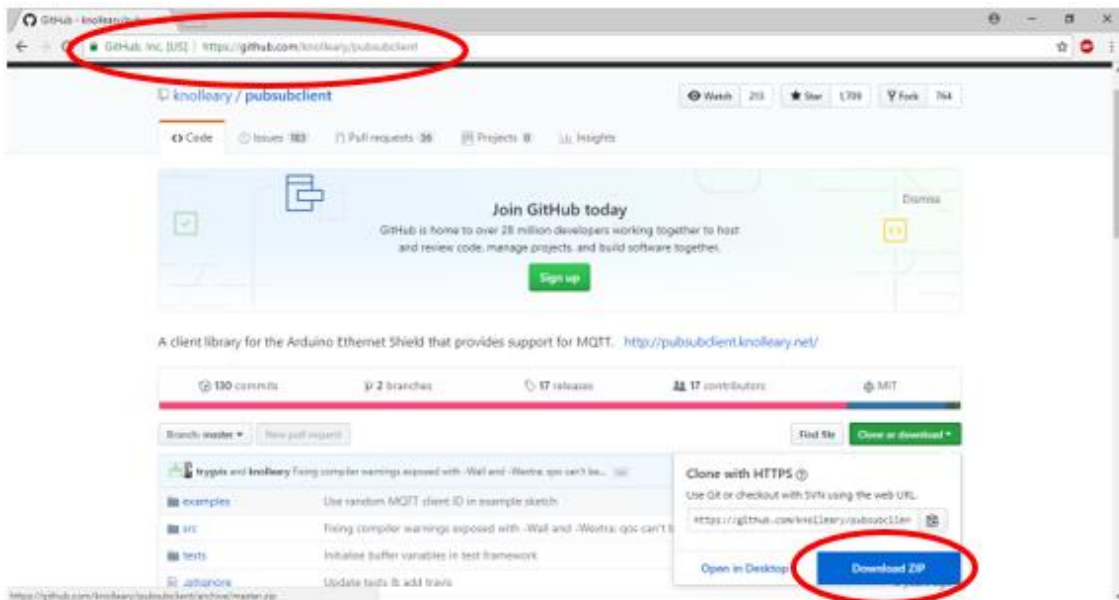


Instalación de Librería PubSubClient.h

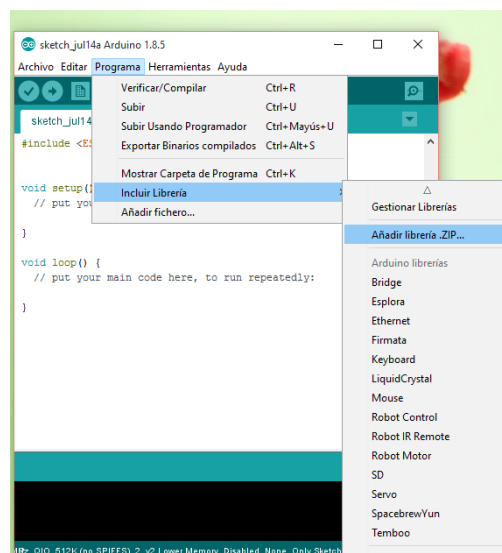
Esta librería permite implementar el protocolo MQTT en Arduino.

Procedimiento

1. Ingresar a la dirección <https://github.com/knolleary/pubsubclient> y descargar la librería como ZIP.



2. Dirigirse a la opción Programa > Incluir Librería > Añadir librería ZIP y buscar la dirección del archivo descargado.

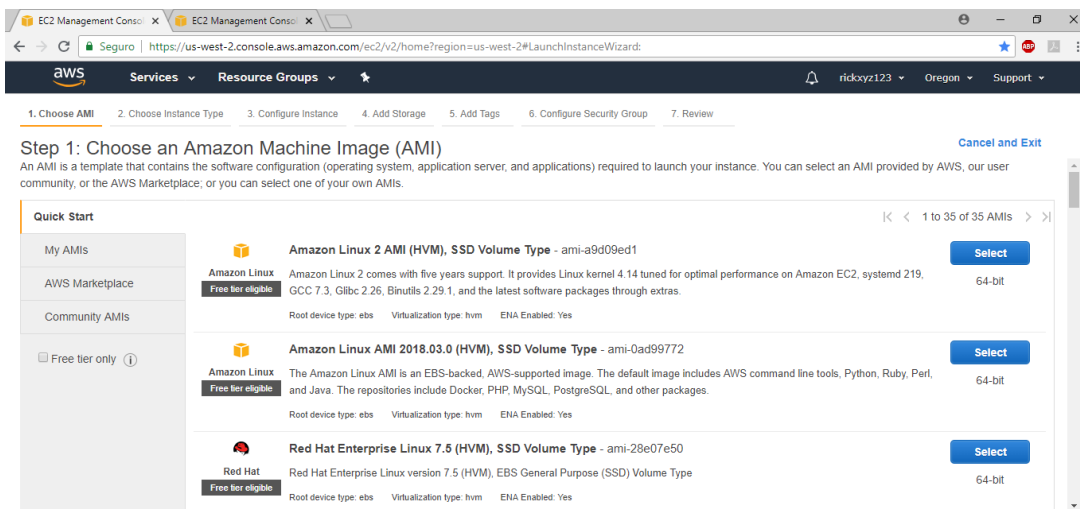


Anexo C. Desarrollo de la Plataforma de Administración y Gestión en AWS.

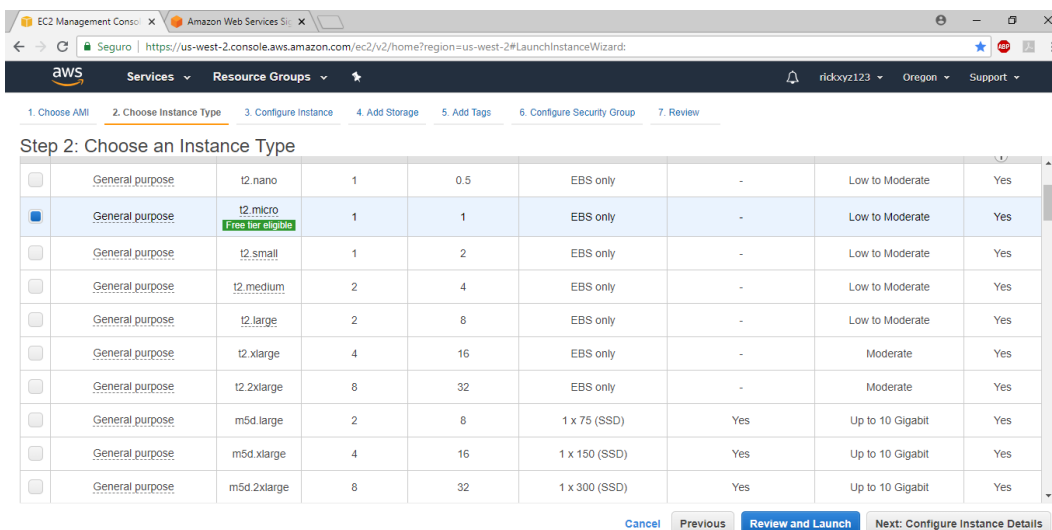
Es necesario crear una cuenta en AWS para correr instancias, en la página <https://aws.amazon.com/es/>

El proceso de levantamiento de una instancia consiste en 7 pasos:

1. Seleccionar la imagen de la máquina, en este caso se utilizó Red Hat Enterprise Linux 7.5



2. Seleccionar el tipo de instancia, se escogió una memoria de 2GB, que será suficiente para levantar el servidor.



3. Configurar detalles de la instancia, los valores se mantienen iguales.

Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot Instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances: 1 [Launch into Auto Scaling Group](#)

Purchasing option: Request Spot instances

Network: vpc-2793705f (default) [Create new VPC](#)

Subnet: No preference (default subnet in any Availability Zone) [Create new subnet](#)

Auto-assign Public IP: Use subnet setting (Enable)

Placement group: Add instance to placement group.

IAM role: None [Create new IAM role](#)

Shutdown behavior: Stop

Enable termination protection: Protect against accidental termination

Monitorin: Enable CloudWatch detailed monitoring

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Storage](#)

4. Agregar almacenamiento, se utiliza el valor de 10GiB, ya que las operaciones se realizan mediante línea de comandos.

Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

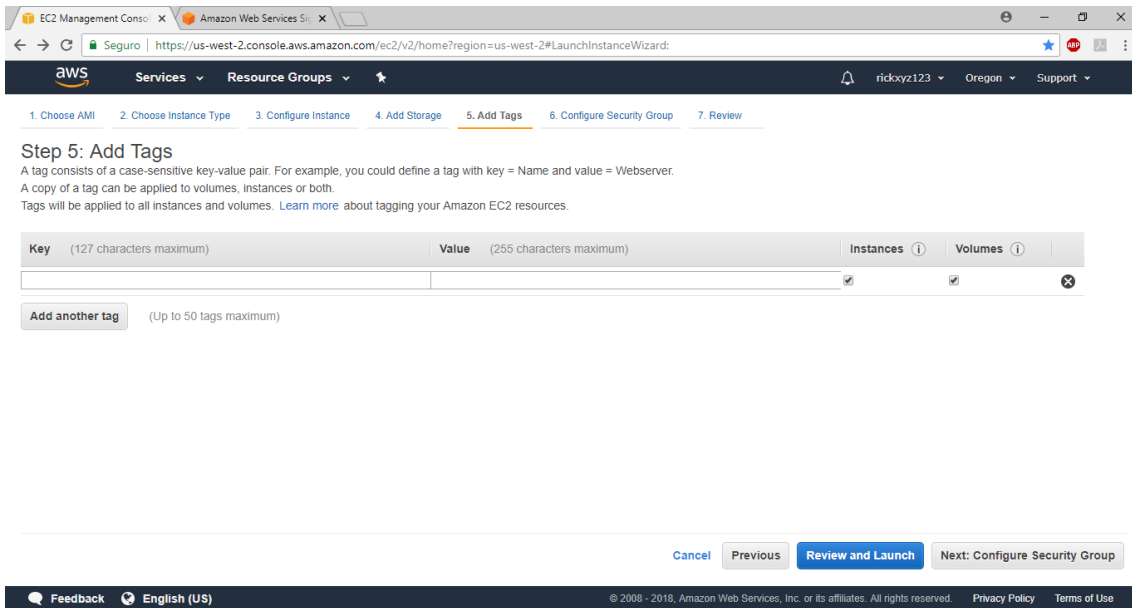
Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/sda1	snap-0a0eab8eb3b56277b	10	General Purpose SSD (GP2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

[Add New Volume](#)

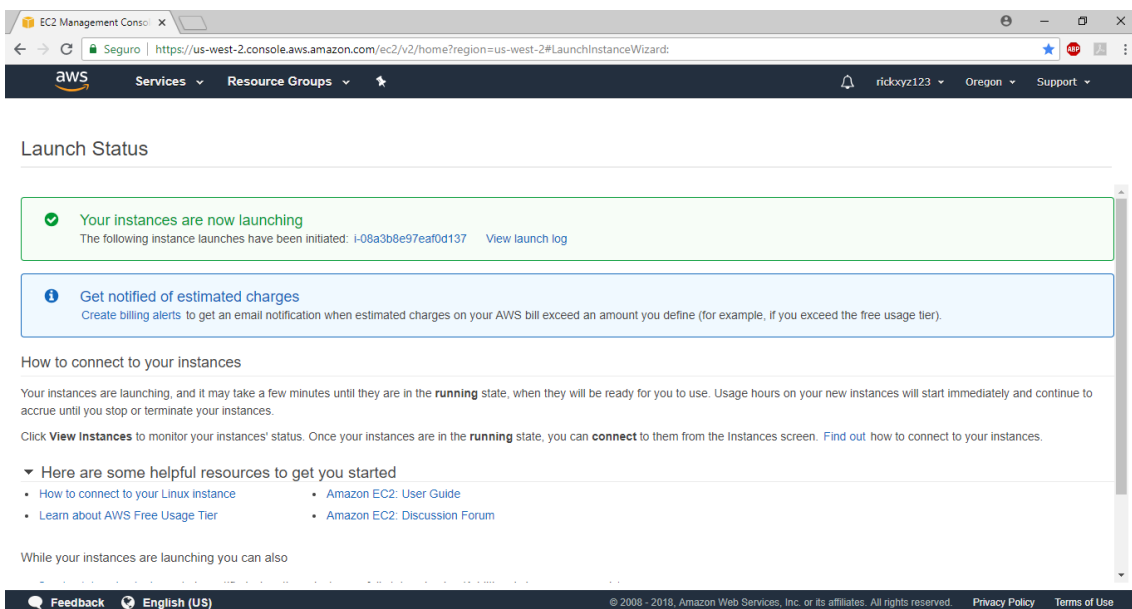
Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Tags](#)

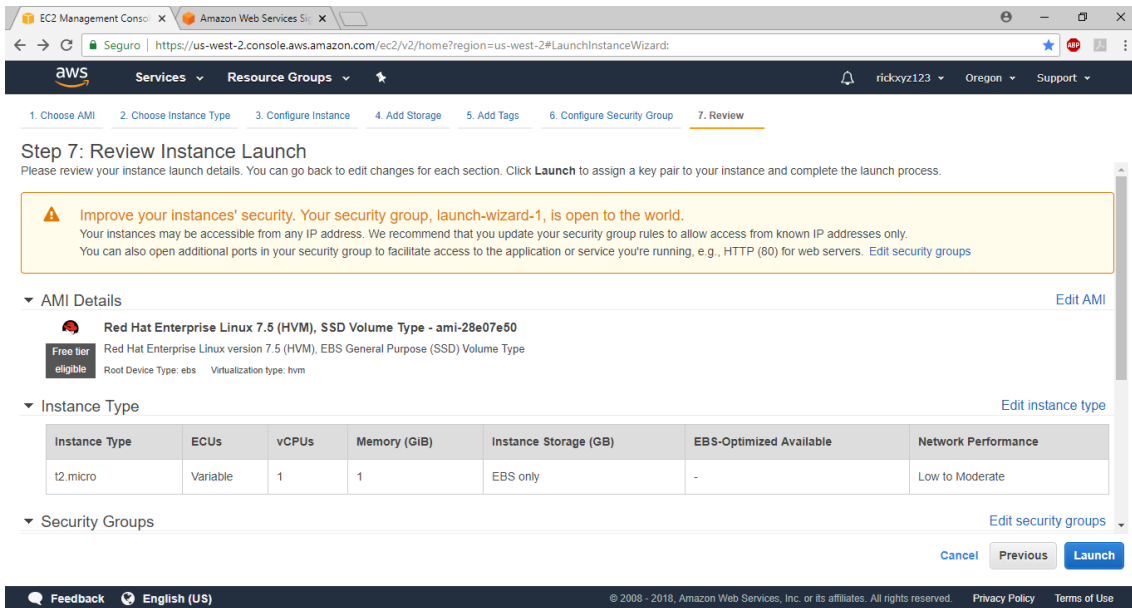
5. Agregar Tags, consiste en generar una llave y un valor para acceder posteriormente a la instancia mediante comunicación SSH.



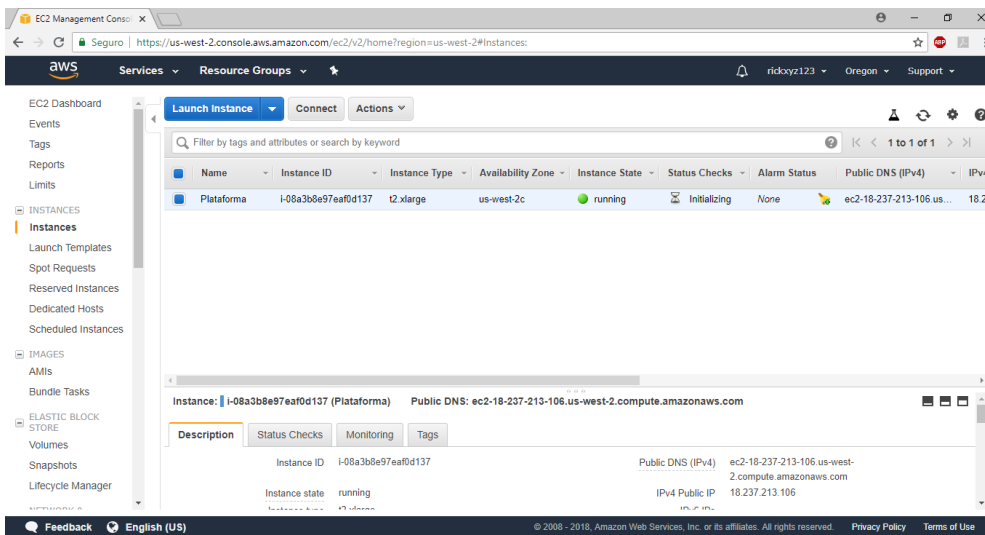
6. Configurar un grupo de seguridad, se agregan reglas de Firewall, específicamente se abre el puerto 22 para la comunicación SSH.



7. En el último paso se hace una revisión de todos los parámetros anteriores, y se ejecuta la instancia.

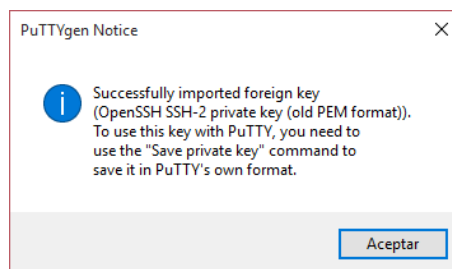
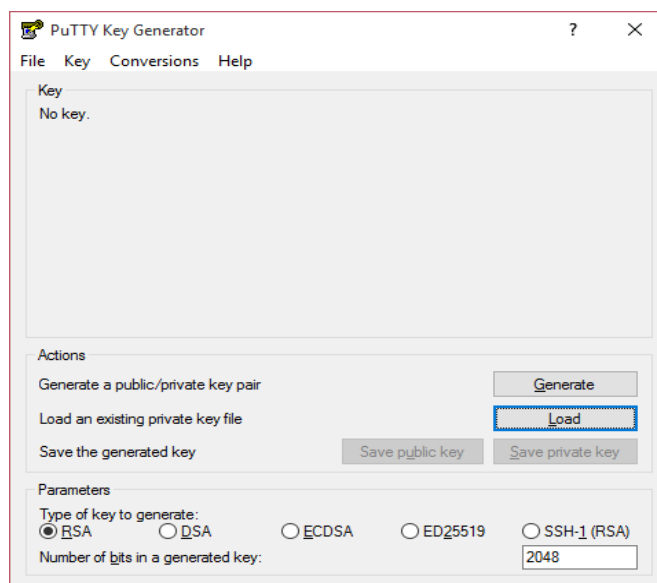


8. Se asigna una IP pública y un DNS público.

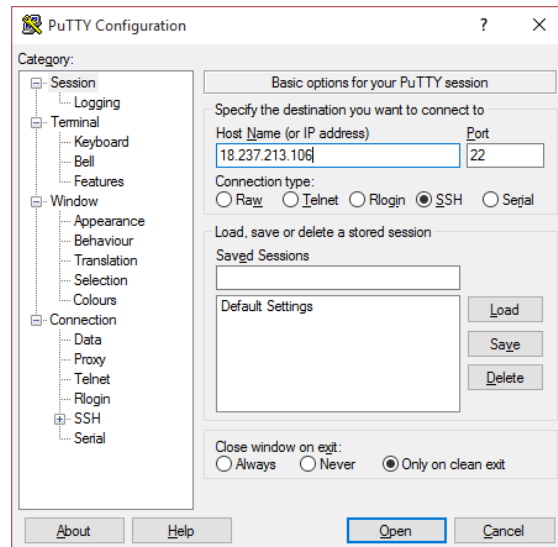


Anexo D. Acceso por SSH a la instancia creada

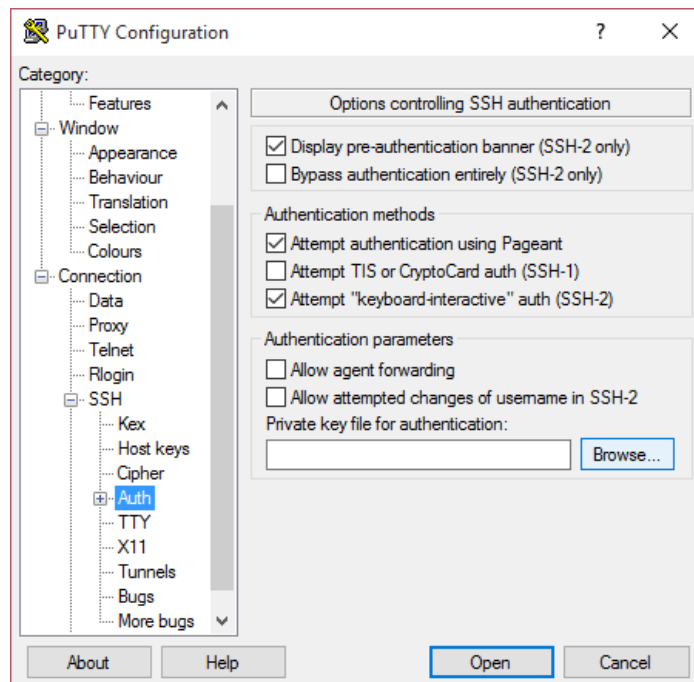
1. Para acceder a la terminal se necesita el programa PuTTYgen, el cual puede descargarse de la siguiente dirección <https://www.ssh.com/ssh/putty/windows/puttygen>
2. En este programa se carga el tag generado en el paso 5 del Anexo B, y se genera un archivo en formato. ppk, que es legible para el programa PuTTY



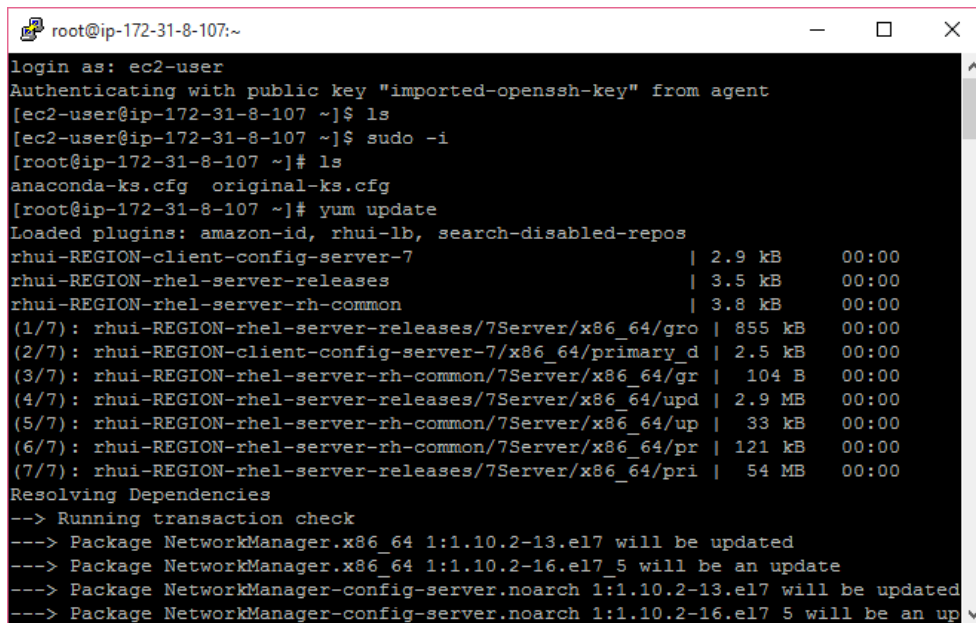
3. En el programa PuTTY se ingresa la IP pública de la instancia para establecer la comunicación por SSH.



4. En la opción Auth del menú SSH, se debe cargar el archivo generado .ppk



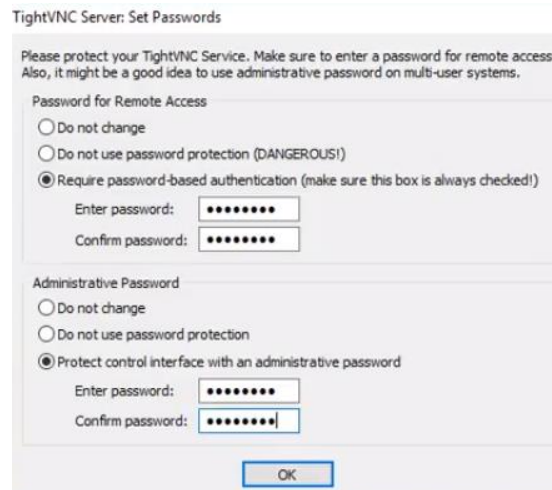
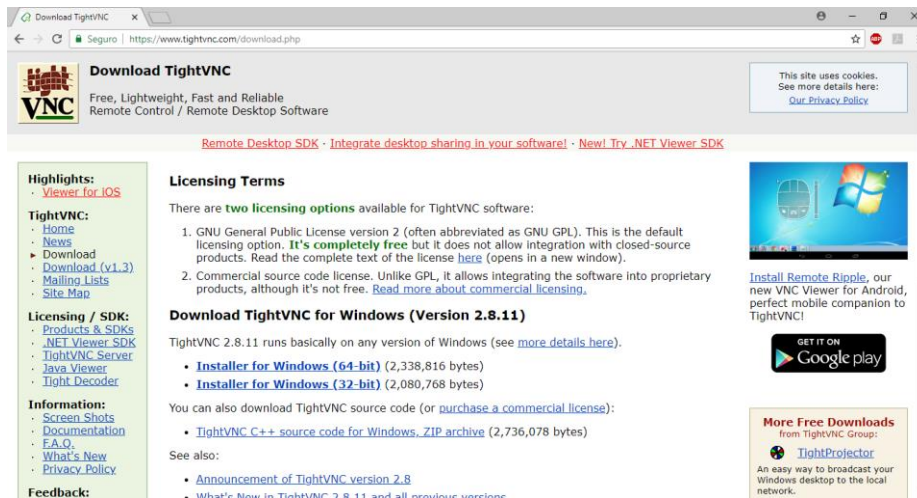
5. Una vez realizada la autenticación, se tendrá acceso a la máquina virtual, el usuario es ec2-user. Es recomendable hacer una actualización del sistema.

A terminal window titled 'root@ip-172-31-8-107:~' showing a user logging in as 'ec2-user'. The user runs 'ls', then 'sudo -i' to become root. The root user runs 'ls' showing 'anaconda-ks.cfg' and 'original-ks.cfg'. Then 'yum update' is run, displaying a list of packages to be updated with their sizes and estimated completion times. The packages include 'rhui-REGION-client-config-server-7', 'rhui-REGION-rhel-server-releases', and 'rhui-REGION-rhel-server-rh-common'. The terminal shows the progress of the update, including resolving dependencies and running a transaction check. The update process is shown as a series of progress bars for each package.

```
root@ip-172-31-8-107:~
login as: ec2-user
Authenticating with public key "imported-openssh-key" from agent
[ec2-user@ip-172-31-8-107 ~]$ ls
[ec2-user@ip-172-31-8-107 ~]$ sudo -i
[root@ip-172-31-8-107 ~]# ls
anaconda-ks.cfg  original-ks.cfg
[root@ip-172-31-8-107 ~]# yum update
Loaded plugins: amazon-id, rhui-lb, search-disabled-repos
rhui-REGION-client-config-server-7 | 2.9 kB  00:00
rhui-REGION-rhel-server-releases | 3.5 kB  00:00
rhui-REGION-rhel-server-rh-common | 3.8 kB  00:00
(1/7): rhui-REGION-rhel-server-releases/7Server/x86_64/gro | 855 kB  00:00
(2/7): rhui-REGION-client-config-server-7/x86_64/primary_d | 2.5 kB  00:00
(3/7): rhui-REGION-rhel-server-rh-common/7Server/x86_64/gr | 104 B  00:00
(4/7): rhui-REGION-rhel-server-releases/7Server/x86_64/upd | 2.9 MB  00:00
(5/7): rhui-REGION-rhel-server-rh-common/7Server/x86_64/up | 33 kB  00:00
(6/7): rhui-REGION-rhel-server-rh-common/7Server/x86_64/pr | 121 kB  00:00
(7/7): rhui-REGION-rhel-server-releases/7Server/x86_64/pri | 54 MB  00:00
Resolving Dependencies
--> Running transaction check
---> Package NetworkManager.x86_64 1:1.10.2-13.el7 will be updated
---> Package NetworkManager.x86_64 1:1.10.2-16.el7_5 will be an update
---> Package NetworkManager-config-server.noarch 1:1.10.2-13.el7 will be updated
---> Package NetworkManager-config-server.noarch 1:1.10.2-16.el7_5 will be an up
```

Anexo E. Acceso mediante escritorio remoto a la instancia.

1. Para tener un escritorio se necesita el software TightVNC, que puede descargarse de <https://www.tightvnc.com/> . En la instalación se solicitarán claves de acceso para el visor.



2. Instalar varios paquetes necesarios para la conexión con el escritorio remoto.

```
root@ip-172-31-8-107:~  
[root@ip-172-31-8-107 ~]# yum groupinstall " X Window System" Desktop  
Loaded plugins: amazon-id, rhui-lb, search-disabled-repos  
There is no installed groups file.  
Maybe run: yum groups mark convert (see man yum)  
Warning: group Desktop does not exist.  
Resolving Dependencies  
--> Running transaction check  
---> Package glx-utils.x86_64 0:8.2.0-3.e17 will be installed  
--> Processing Dependency: libGL.so.1()(64bit) for package: glx-utils-8.2.0-3.e17.x86_64
```

```
root@ip-172-31-8-107:~  
login as: ec2-user  
Authenticating with public key "imported-openssh-key" from agent  
Last login: Mon Jul 16 09:20:35 2018 from 190.152.83.95  
[ec2-user@ip-172-31-8-107 ~]$ sudo -i  
[root@ip-172-31-8-107 ~]# yum -y group install gnome
```

```
root@ip-172-31-8-107:~  
[root@ip-172-31-8-107 ~]# yum install tigervnc-server  
Loaded plugins: amazon-id, rhui-lb, search-disabled-repos  
Resolving Dependencies  
There are unfinished transactions remaining. You might consider running yum-complete-transaction, or "yum-complete-transaction --cleanup-only" and "yum history redo last", first to finish them. If those don't work you'll have to try removing/installing packages by hand (maybe package-cleanup can help).  
--> Running transaction check  
--> Package tigervnc-server.x86_64 0:1.8.0-5.e17 will be installed  
--> Finished Dependency Resolution
```

3. Se instala el navegador firefox

```
root@ip-172-31-8-107:~  
Loaded plugins: amazon-id, rhui-lb, search-disabled-repos  
Error: Need a group or list of groups  
Mini usage:  
  
install PACKAGE...  
  
Install a package or packages on your system  
  
aliases: install-n, install-na, install-nevra  
[root@ip-172-31-8-107 ~]# yum -y group install gnome  
Loaded plugins: amazon-id, rhui-lb, search-disabled-repos  
Maybe run: yum groups mark install (see man yum)  
No packages in any requested group available to install or update  
[root@ip-172-31-8-107 ~]# yum -y groupinstall  
Loaded plugins: amazon-id, rhui-lb, search-disabled-repos  
Error: Need a group or list of groups  
Mini usage:  
  
install PACKAGE...  
  
Install a package or packages on your system  
  
aliases: install-n, install-na, install-nevra  
[root@ip-172-31-8-107 ~]# yum install gnome-core xfce4 firefox
```

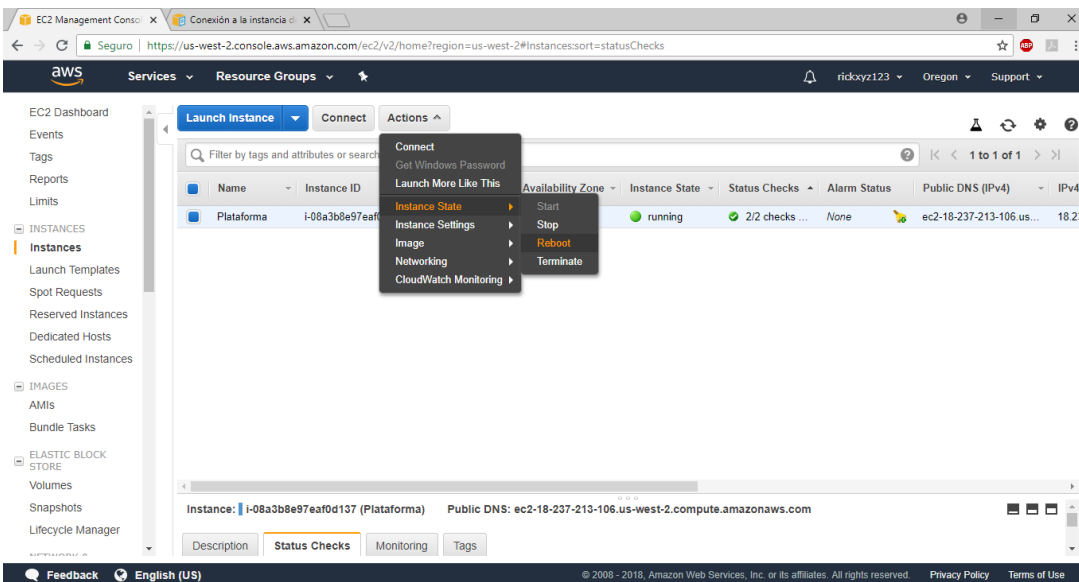
4. La instalación de vncserver necesita de una contraseña, la misma del programa TightVNC Viewer.

```
root@ip-172-31-8-107:~  
Installed:  
tigervnc-server.x86_64 0:1.8.0-5.e17  
  
Complete!  
[root@ip-172-31-8-107 ~]# vncserver  
  
You will require a password to access your desktops.  
  
Password:  
Verify:  
Would you like to enter a view-only password (y/n)? y  
Password:  
Verify:  
xauth: file /root/.Xauthority does not exist  
  
New 'ip-172-31-8-107.us-west-2.compute.internal:1 (root)' desktop is ip-172-31-8-107.us-west-2.compute.internal:1  
  
Creating default startup script /root/.vnc/xstartup  
Creating default config /root/.vnc/config  
Starting applications specified in /root/.vnc/xstartup  
Log file is /root/.vnc/ip-172-31-8-107.us-west-2.compute.internal:1.log  
[root@ip-172-31-8-107 ~]#
```

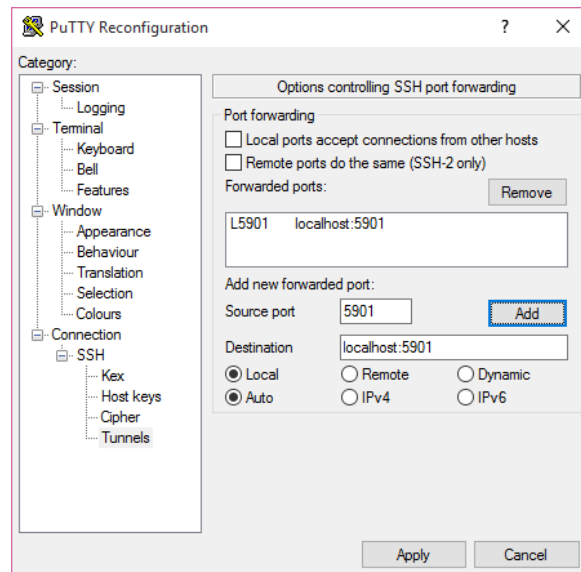
5. Para inicializar el servidor VNC, se ejecuta el comando vncserver.

```
root@ip-172-31-8-107:~/vnc  
[root@ip-172-31-8-107 .vnc]# vncserver  
  
New 'ip-172-31-8-107.us-west-2.compute.internal:2 (root)' desktop is ip-172-31-8-107.us-west-2.compute.internal:2  
  
Starting applications specified in /root/.vnc/xstartup  
Log file is /root/.vnc/ip-172-31-8-107.us-west-2.compute.internal:2.log  
[root@ip-172-31-8-107 .vnc]#
```

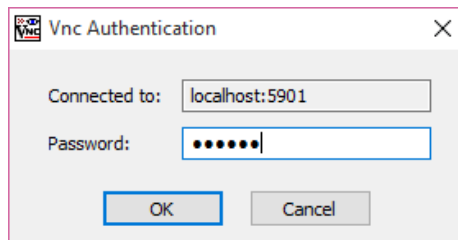
6. Luego hacer un reinicio de la instancia.



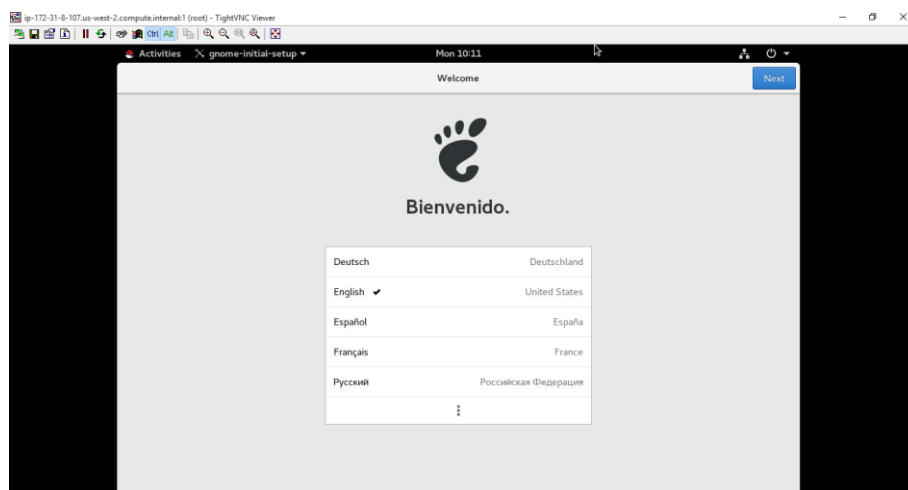
7. En PuTTY, en la opción Tunnels de menú SSH, se debe ingresar el puerto y host por el cual se comunica TightVNC Viewer.



8. El software solicitará la contraseña configurada



9. Una vez realizada la conexión, se tendrá acceso al escritorio remoto, donde se deberá hacer ciertas configuraciones básicas (idioma, teclado, entre otras.).



Anexo F. Instalación, inicio y pruebas con el Broker Mosquitto

1. Instalación de Broker Mosquitto agregando repositorios

```
[root@ip-172-31-8-107 ~]# rpm -ivh epel-release-latest-7.noarch.rpm
warning: epel-release-latest-7.noarch.rpm: Header V3 RSA/SHA256 Signature, key ID
352c64e5: NOKEY
Preparing... ##### [100%]
Updating / installing...
 1:epel-release-7-11 ##### [100%]
[root@ip-172-31-8-107 ~]# yum -y install mosquitto
```

2. Inicio y habilitación de bróker Mosquitto

```
root@ip-172-31-8-107:~
Complete!
[root@ip-172-31-8-107 ~]# systemctl start mosquitto
[root@ip-172-31-8-107 ~]# systemctl enable mosquitto
Created symlink from /etc/systemd/system/multi-user.target.wants/mosquitto.servi
ce to /usr/lib/systemd/system/mosquitto.service.
[root@ip-172-31-8-107 ~]#
```

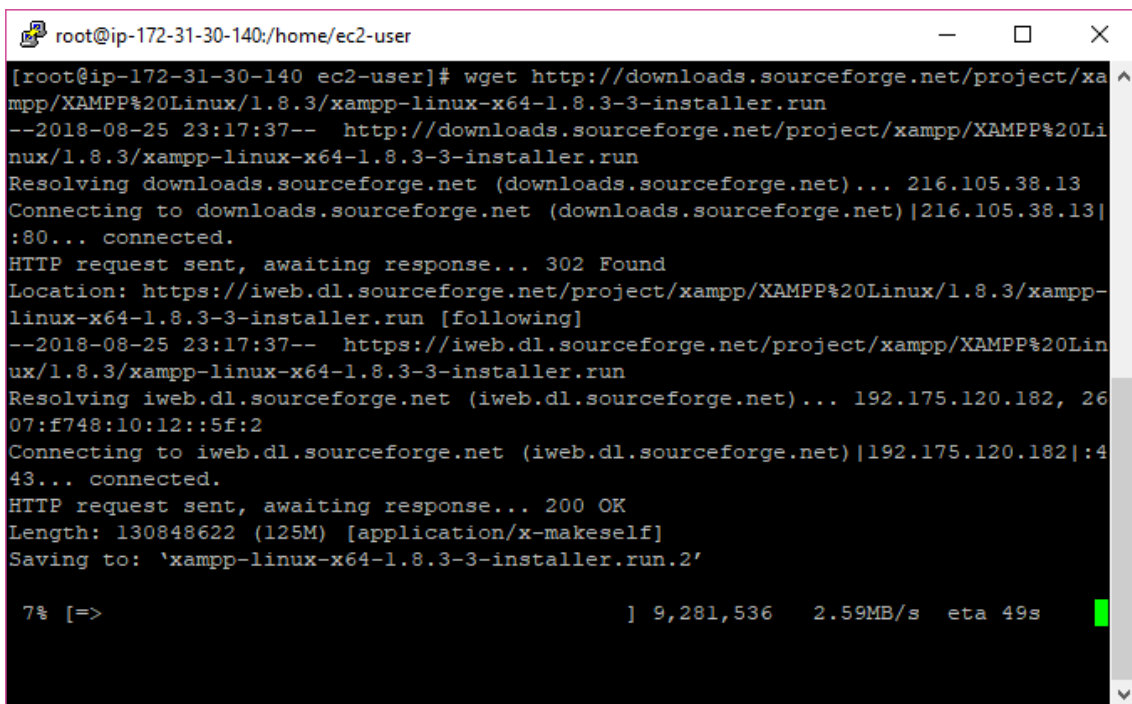
3. Publicación de mensajes a un tópico suscrito.

```
root@ip-172-31-8-107:~
[root@ip-172-31-8-107 ~]# mosquitto_pub -t "inTopic" -m 1
[root@ip-172-31-8-107 ~]# mosquitto_pub -t "inTopic" -m 0
[root@ip-172-31-8-107 ~]# mosquitto_pub -t "inTopic" -m 1
[root@ip-172-31-8-107 ~]# mosquitto_pub -t "inTopic" -m 0
[root@ip-172-31-8-107 ~]#
```

Anexo G. Instalación de Servidor XAMPP

1. La instalación de este servidor puede hacerse mediante línea de comandos, o mediante el escritorio remoto descargando el instalador XAMPP para Linux, de la página <https://www.apachefriends.org/es/download.html>. En este caso se realiza mediante CLI con los comandos que se indican a continuación.

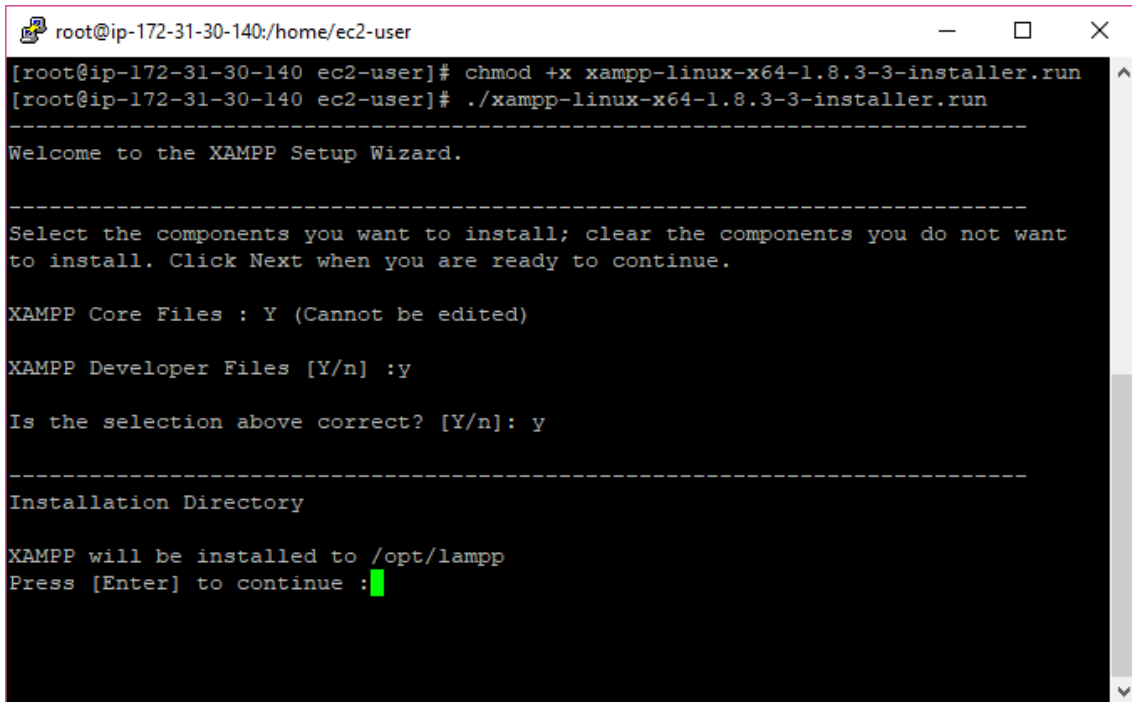
- `yum install wget`
- `wget`
`http://downloads.sourceforge.net/project/xampp/XAMPP%20Linux/1.8.3/xampp-linux-x64-1.8.3-3-installer.run`
- `chmod +x xampp-linux-x64-1.8.3-3-installer.run`
- `./xampp-linux-x64-1.8.3-3-installer.run`



```
root@ip-172-31-30-140:/home/ec2-user
[root@ip-172-31-30-140 ec2-user]# wget http://downloads.sourceforge.net/project/xampp/XAMPP%20Linux/1.8.3/xampp-linux-x64-1.8.3-3-installer.run
--2018-08-25 23:17:37-- http://downloads.sourceforge.net/project/xampp/XAMPP%20Linux/1.8.3/xampp-linux-x64-1.8.3-3-installer.run
Resolving downloads.sourceforge.net (downloads.sourceforge.net)... 216.105.38.13
Connecting to downloads.sourceforge.net (downloads.sourceforge.net)|216.105.38.13|:80... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://iweb.dl.sourceforge.net/project/xampp/XAMPP%20Linux/1.8.3/xampp-linux-x64-1.8.3-3-installer.run [following]
--2018-08-25 23:17:37-- https://iweb.dl.sourceforge.net/project/xampp/XAMPP%20Linux/1.8.3/xampp-linux-x64-1.8.3-3-installer.run
Resolving iweb.dl.sourceforge.net (iweb.dl.sourceforge.net)... 192.175.120.182, 2607:f748:10:12::5f:2
Connecting to iweb.dl.sourceforge.net (iweb.dl.sourceforge.net)|192.175.120.182|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 130848622 (125M) [application/x-makeself]
Saving to: 'xampp-linux-x64-1.8.3-3-installer.run.2'

7% [=> ] 9,281,536 2.59MB/s eta 49s
```

2. Se indica los componentes que van a instalarse y el directorio de instalación del servidor.

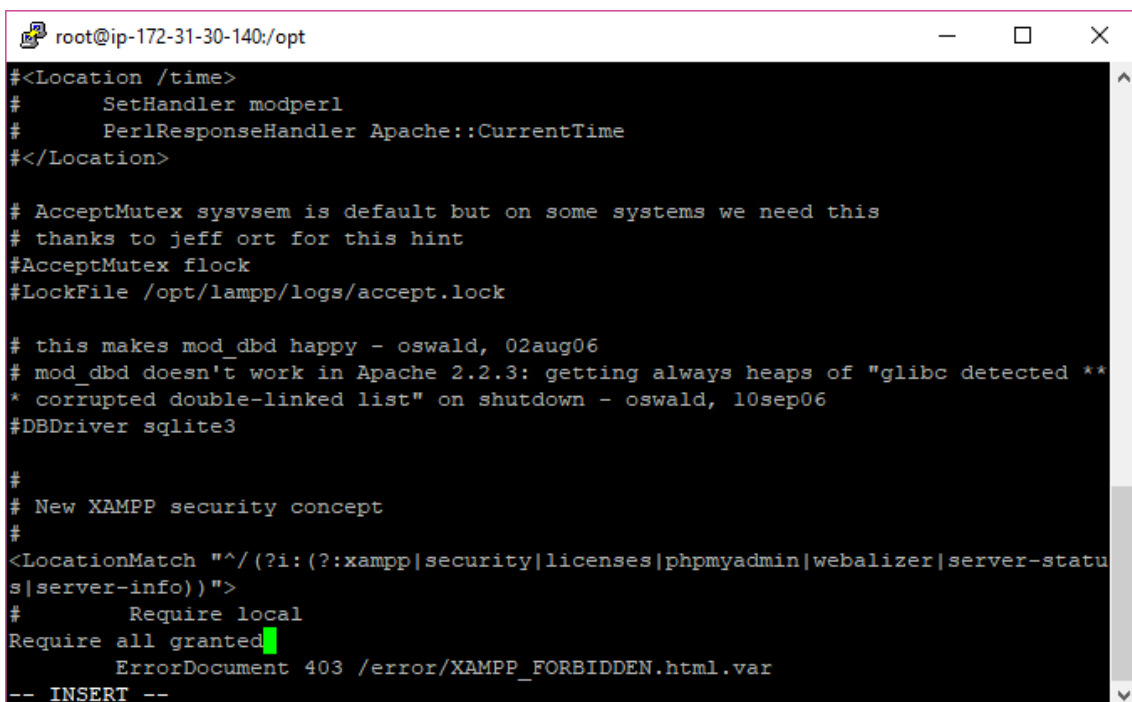


```
root@ip-172-31-30-140:/home/ec2-user
[root@ip-172-31-30-140 ec2-user]# chmod +x xampp-linux-x64-1.8.3-3-installer.run
[root@ip-172-31-30-140 ec2-user]# ./xampp-linux-x64-1.8.3-3-installer.run
-----
Welcome to the XAMPP Setup Wizard.
-----
Select the components you want to install; clear the components you do not want
to install. Click Next when you are ready to continue.

XAMPP Core Files : Y (Cannot be edited)
XAMPP Developer Files [Y/n] :y
Is the selection above correct? [Y/n]: y
-----
Installation Directory

XAMPP will be installed to /opt/lampp
Press [Enter] to continue : █
```

3. Editar el archivo `httpd-xampp.conf` para permitir el acceso hacia el servidor web por vía remota, se debe escribir *Require all granted* en lugar de *Require local*.



```
root@ip-172-31-30-140:/opt
#<Location /time>
#   SetHandler modperl
#   PerlResponseHandler Apache::CurrentTime
#</Location>

# AcceptMutex sysvsem is default but on some systems we need this
# thanks to jeff ort for this hint
#AcceptMutex flock
#LockFile /opt/lampp/logs/accept.lock

# this makes mod_dbd happy - oswald, 02aug06
# mod_dbd doesn't work in Apache 2.2.3: getting always heaps of "glibc detected **
* corrupted double-linked list" on shutdown - oswald, 10sep06
#DBDriver sqlite3

#
# New XAMPP security concept
#
<LocationMatch "^/(?:xampp|security|licenses|phpmyadmin|webalizer|server-status|server-info)">
#   Require local
Require all granted█
   ErrorDocument 403 /error/XAMPP_FORBIDDEN.html.var
-- INSERT --
```

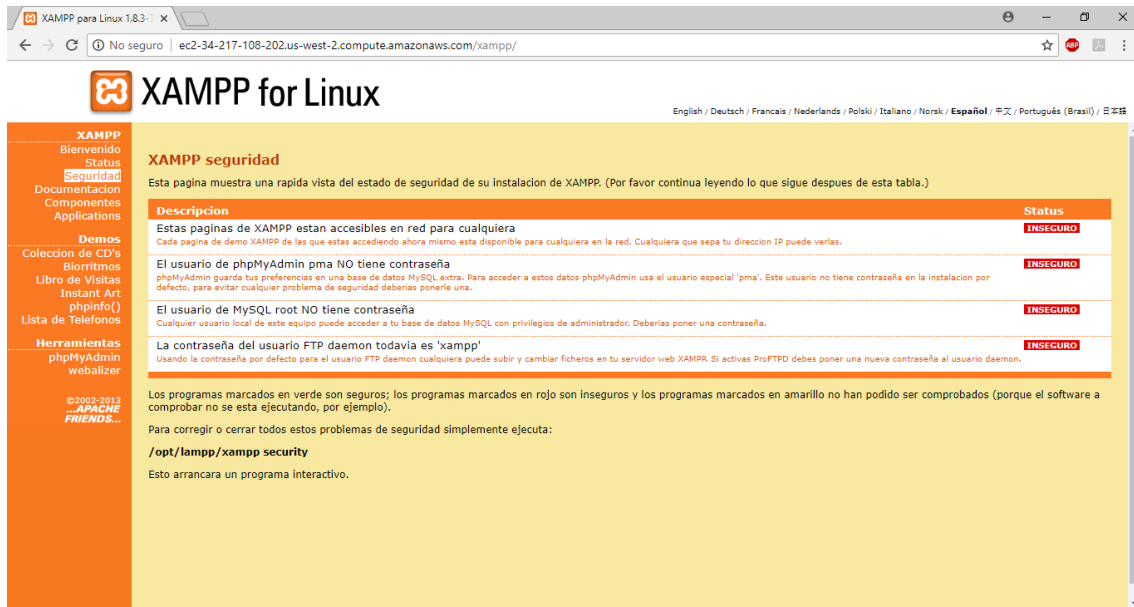

4. Reiniciar el servidor

```
root@ip-172-31-30-140:/opt/lampp
[root@ip-172-31-30-140 lampp]# lampp restart
bash: lampp: command not found..
[root@ip-172-31-30-140 lampp]# ./lampp restart
Restarting XAMPP for Linux 1.8.3-3...
XAMPP: Stopping Apache...ok.
XAMPP: Stopping MySQL...not running.
XAMPP: Stopping ProFTPD...not running.
XAMPP: Starting Apache...ok.
XAMPP: Starting MySQL...ok.
XAMPP: Starting ProFTPD...ok.
[root@ip-172-31-30-140 lampp]# █
```

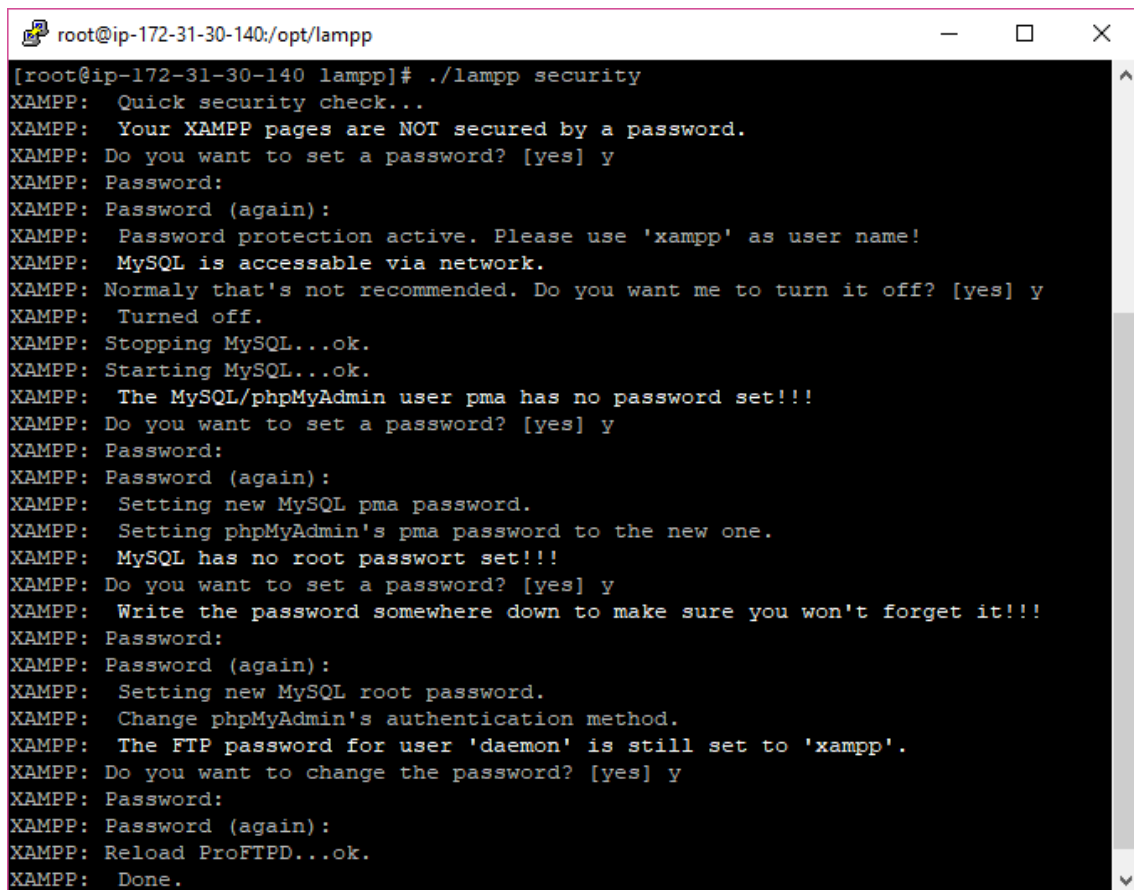
5. Al escribir la dirección *dnspublico/xampp/index.php* en el navegador, se muestran las características del servidor



6. En la opción *seguridad* se muestran ciertas características como inseguras y se marcan en color rojo.



7. Para cambiar el estatus a *seguro* se debe escribir `./lampp security` en el cmd, y seguir los pasos indicados.



8. Al realizar nuevamente los pasos 5 y 6, ya se observa un servidor con características seguras.

XAMPP para Linux 1.8.3- x

No seguro | ec2-34-217-108-202.us-west-2.compute.amazonaws.com/xampp/

XAMPP for Linux

English / Deutsch / Français / Nederlands / Polski / Italiano / Norsk / Español / 中文 / Português (Brasil) / 日本語

XAMPP
Bienvenido
Status
Seguridad
Documentación
Componentes
Applications

Demos
Colección de CD's
Biorritmos
Libro de Visitas
Instant Art
phpinfo()
Lista de Telefonos

Herramientas
phpMyAdmin
webalizer

©2002-2013
APACHE
FRIENDS...

XAMPP seguridad

Esta pagina muestra una rapida vista del estado de seguridad de su instalacion de XAMPP. (Por favor continua leyendo lo que sigue despues de esta tabla.)

Descripción	Status
Estas paginas de AMPP ya NO estan accesibles en red para cualquiera	SEGURO
El usuario de phpMyAdmin pma tiene contraseña	SEGURO
El usuario de MySQL root tiene contraseña	SEGURO
La contraseña del usuario FTP daemon ya no es 'xampp'	SEGURO

Los programas marcados en verde son seguros; los programas marcados en rojo son inseguros y los programas marcados en amarillo no han podido ser comprobados (porque el software a comprobar no se esta ejecutando, por ejemplo).

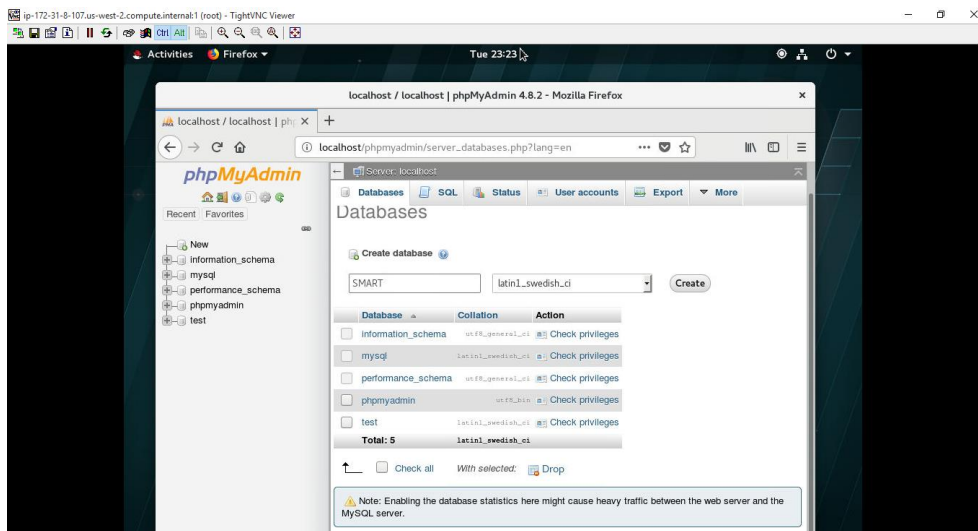
Para corregir o cerrar todos estos problemas de seguridad simplemente ejecuta:

```
/opt/lampp/xampp security
```

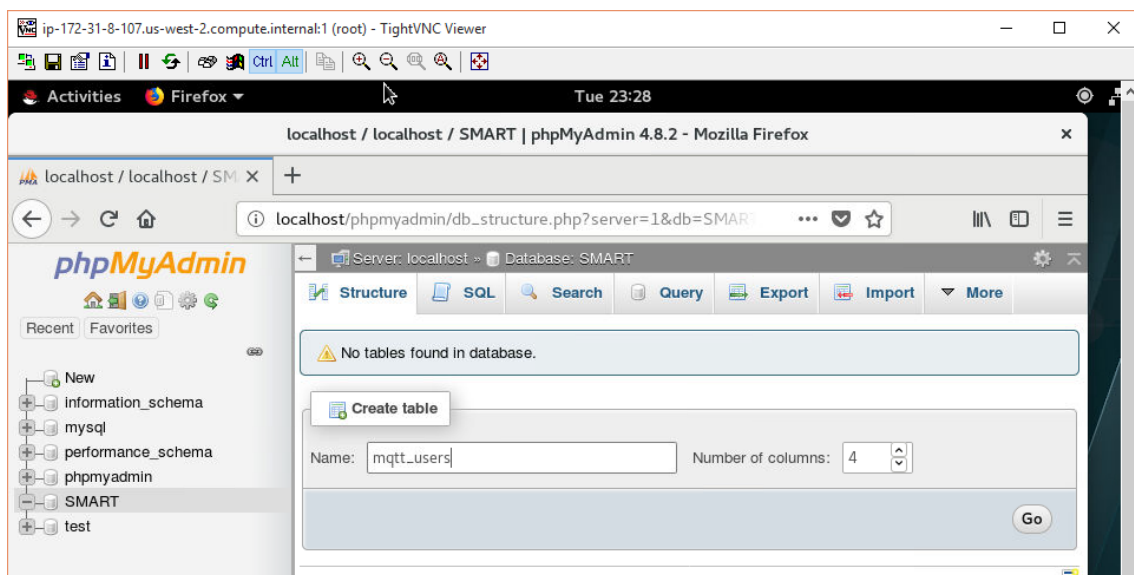
Esto arrancara un programa interactivo.

Anexo H. Creación de base de datos con el gestor phpMyAdmin

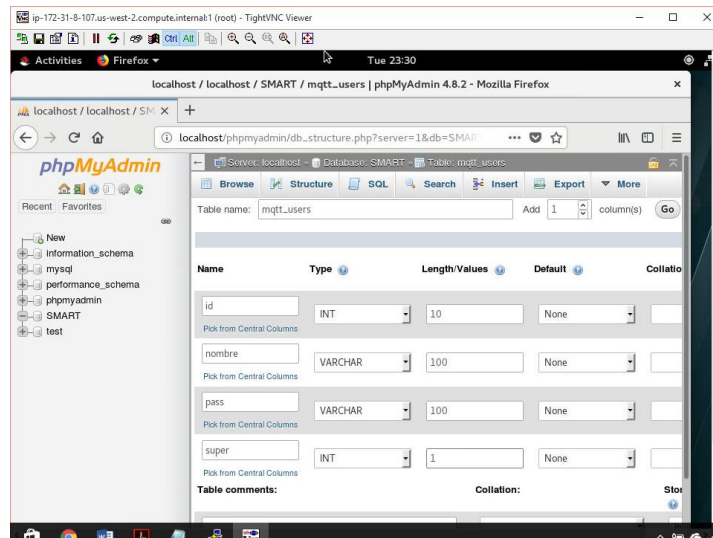
1. Para acceder al gestor phpMyAdmin, se escribe en la dirección de navegador *dnspublico/phymyadmin*. Aquí se crearán las bases de datos y tablas, en la opción Databases.



2. Al ingresar a la base de datos, se debe crear tablas de acuerdo al número de parámetros que se va a manejar (sensor1, sensor2, sensor,3 usuarios, etc). A cada tabla se le asignará un número de columnas, según los datos que se vayan a ingresar (ID, valor, hora, nombre, clave, etc).

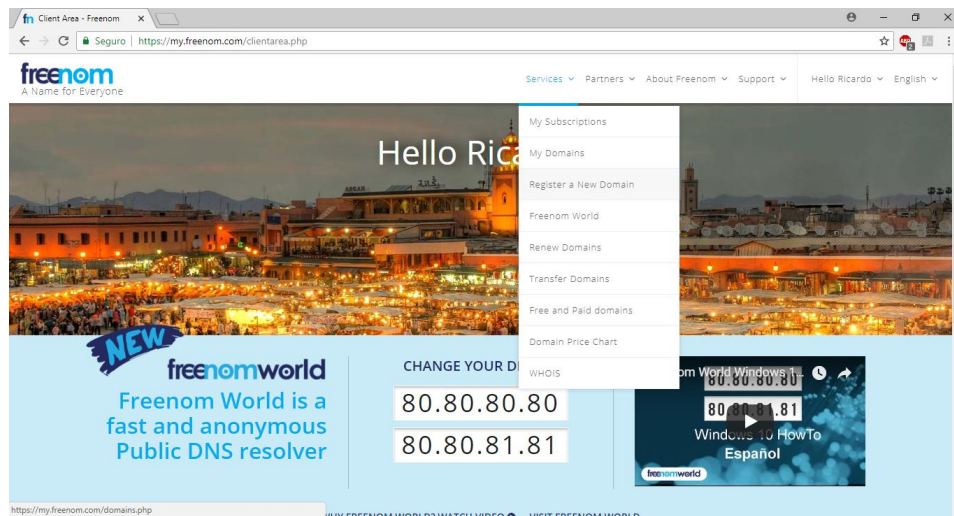


3. Según el dato que se va a almacenar en la base de datos se escoge un tipo, para valores enteros se elige INT, para nombres y contraseñas se elige VARCHAR, y para la hora y fecha se elige TIMESTAMP.

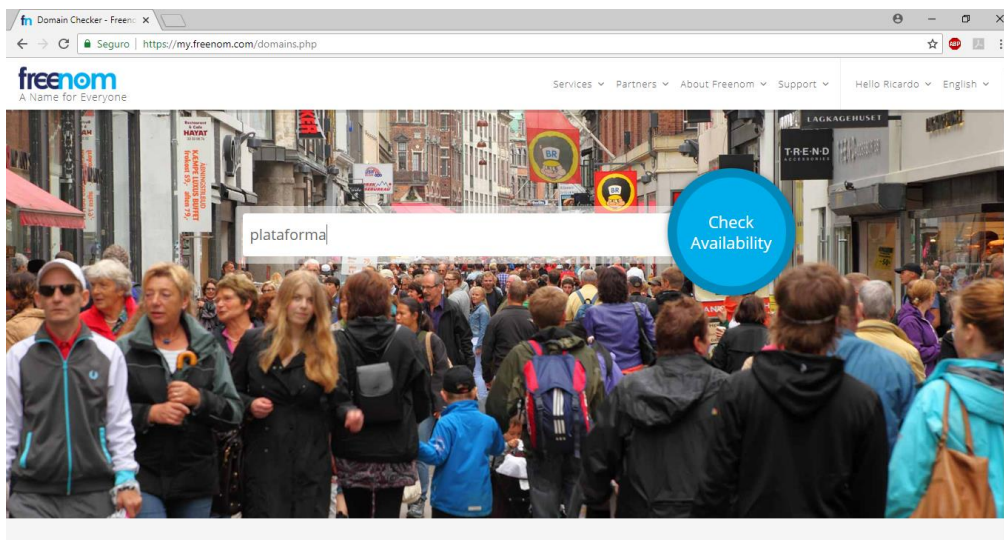


Anexo I. Asignación de DNS

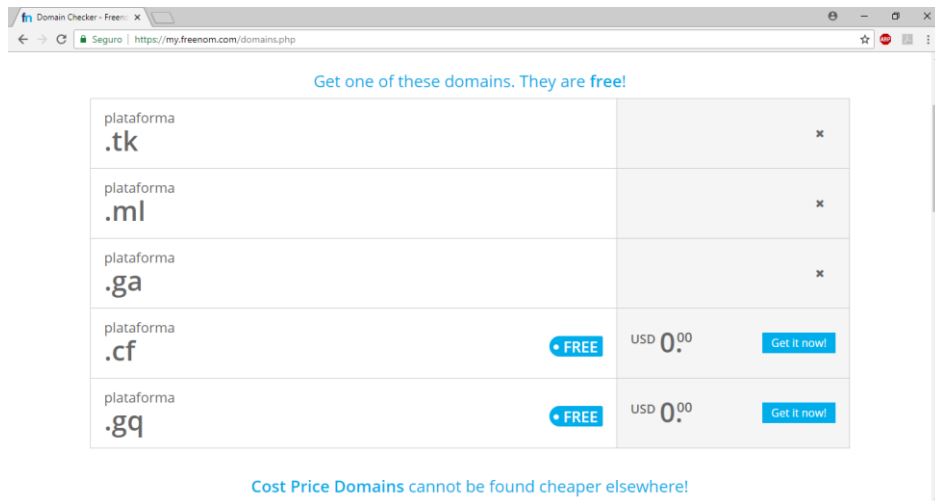
1. Ingresar al sitio web myfreenom.com y crear una cuenta, a continuación, hacer click en la opción *Register a New Domain* en el menú *Services*



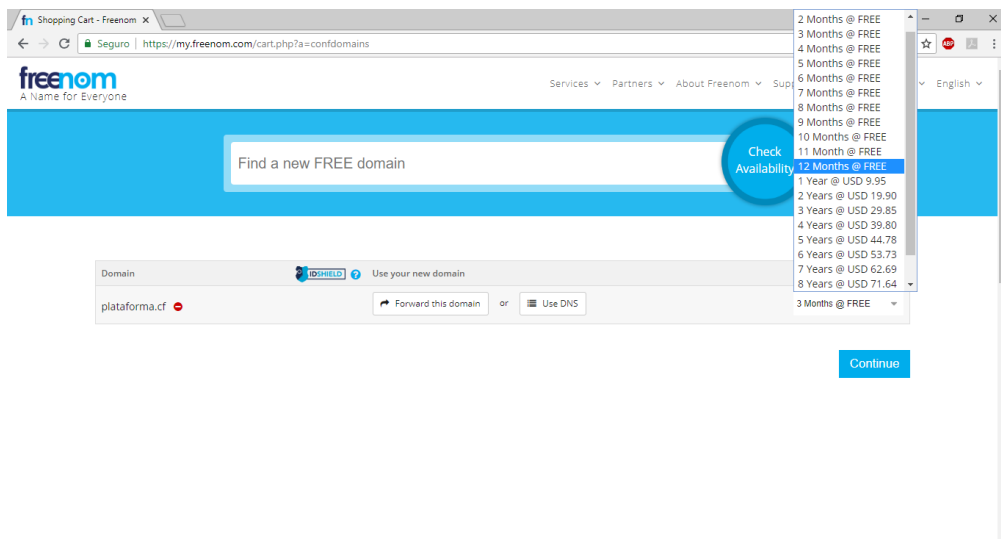
2. Escribir el dominio deseado para la página web y verificar su disponibilidad.



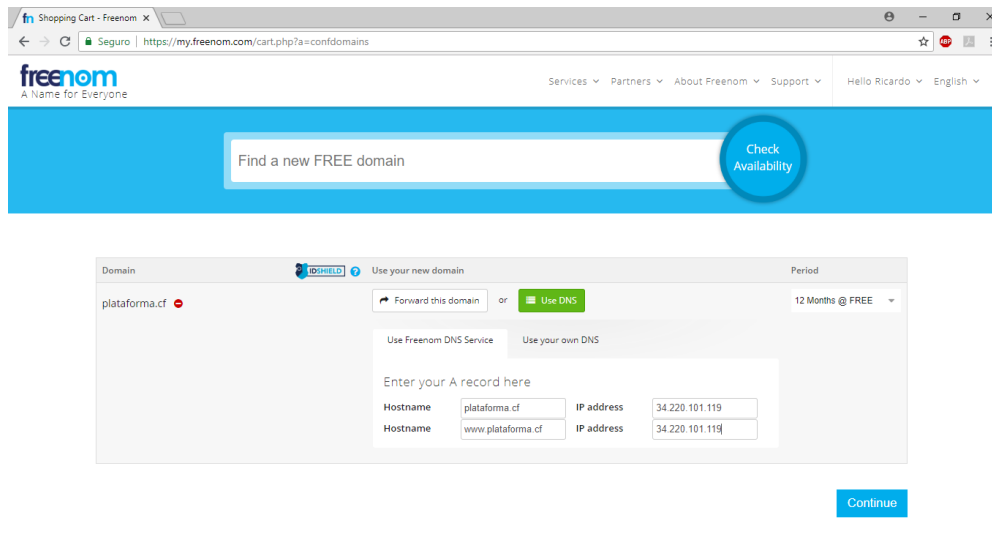
3. Escoger una extensión para el dominio creado, existen varias gratuitas.



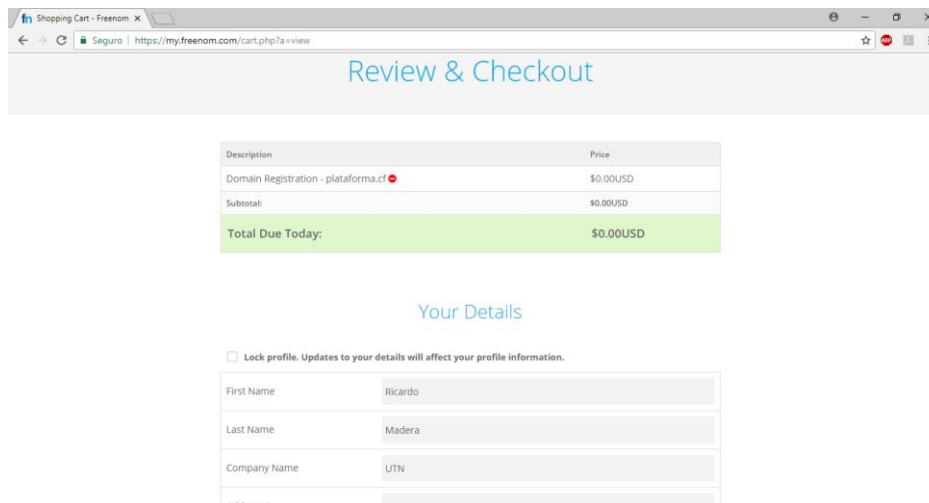
4. Escoger el tiempo deseado para el servicio de DNS, el máximo tiempo gratuito es de 12 meses, que luego puede renovarse.



5. Escribir la dirección IP a la que hacer referencia el dominio creado.



6. Verificar la creación del nuevo dominio.



Anexo J. Mediciones con y sin implementación de prototipo

Sin prototipo



Medición en la primera semana sin prototipo



Medición en la segunda semana sin prototipo



Medición en la tercera semana sin prototipo



Medición en la cuarta semana sin prototipo

Con prototipo



Medición en la primera semana con prototipo



Medición en la segunda semana con prototipo



Medición en la tercera semana con prototipo



Medición en la cuarta semana con prototipo

Anexo K. Programación

Programación de Arduino Nano

1. Se incluye librerías y se inicializa variables.

```
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

```
#include <Wire.h>
#include "RTCLib.h"
#include "LowPower.h"

RTC_DS3231 RTC;
int hora = 0;
int minutos = 0;
int segundos = 0;
const int salida = 10;
const int interrupt = 2;
const int led = 11;
const int timeThreshold = 5000;
int counter = 0;
long timeCounter = 0;
volatile int state = LOW;
const int relay1 = 4;
const int relay2 = 5;
int sensor1 = 0;
int sensor2 = 0;
int control = 0;
int control2 = 0;
int vac = 0;
int s1 = 0;
int s2 = 0;
int h, m, s = 0;

int u = 0;
int v = 0;

int hora_off;
int min_off;
int hora_on;
int min_on;

int frecuencia=1;
int frecuencia2=1;

double voltajeInst;
double voltajeMax;
double voltajeMin;
int potencia;
```


4. Recibir los datos enviados por el módulo WiFi, y evaluar de acuerdo a un conjunto de casos para apagar o encender los relevadores.

```
////////////////////////////////////
```

```
if (Serial.available())
{
  char valor = Serial.read();
  switch (valor)
  {
    case '0':
      digitalWrite(relay1, HIGH); //apagado
      if (digitalRead(relay2)==LOW)
        digitalWrite(relay2, LOW);
      control = 1;
      break;
    case '1':
      digitalWrite(relay1, LOW); //encendido
      if (digitalRead(relay2)== HIGH)
        digitalWrite(relay2, HIGH);
      control = 0;
      break;
    case '2':
      digitalWrite(relay2, HIGH); //apagado
      if (digitalRead(relay1)== LOW)
        digitalWrite(relay1, LOW);
      control2 = 1;
      break;
    case '3':
      digitalWrite(relay2, LOW); //encendido
      if (digitalRead(relay1)== HIGH)
        digitalWrite(relay1, HIGH);
      control2 = 0;
      break;

```

```
////////////////////////////////////
```

5. Hora y minuto de apagado del módulo electrónico

```
////////////////////////////////////
```

```
case 'Q': hora_off = 0; break;
case 'W': hora_off = 1; break;
case 'E': hora_off = 2; break;
case 'R': hora_off = 3; break;
case 'T': hora_off = 4; break;
case 'Y': hora_off = 5; break;
case 'U': hora_off = 6; break;
case 'I': hora_off = 7; break;
```


////////////////////////////////////
7. Frecuencia de medición
////////////////////////////////////

```
case '1': frecuencia = 1; break;
case '2': frecuencia = 2; break;
case '3': frecuencia = 3; break;
case '4': frecuencia = 4; break;
case '5': frecuencia = 5; break;

case '!': frecuencia2 = 1; break;
case '"': frecuencia2 = 2; break;
case '#': frecuencia2 = 3; break;
case '$': frecuencia2 = 4; break;
case '%': frecuencia2 = 5; break;

}
}
```

8. Comparar las horas de apagado y encendido con la hora actual, y realizar el apagado o encendido respectivo.

```
if ((now.hour() >= hora_off && now.hour() <= hora_on && now.minute() >= min_off && now.minute() < min_on))

{
  digitalWrite(relay1, HIGH); //apagado
  digitalWrite(relay2, HIGH); //apagado
  delay(100);
  digitalWrite(relay1, HIGH); //apagado
  digitalWrite(relay2, HIGH); //apagado
}

else
{
  if (control == 0)
    digitalWrite(relay1, LOW);
  if (control2 == 0)
    digitalWrite(relay2, LOW);
}
}
```

////////////////////////////////////

9. Interrupción para encender el módulo electrónico

```
////////////////////////////////////

void blink()
{
  if (millis() > timeCounter + timeThreshold)
```


11. Lectura de consumo energético del sensor2 ACS-712 y envío por comunicación serial.

```
////////////////////////////////////////////////////////////////////  
void lectura_sensor2()  
{  
  //Para hallar la potencia se suma con el voltaje rms  
  voltajeMax2 = 0.00;  
  voltajeMin2 = 5.00;  
  for (int b = 0; b < nMuestras2; b++) {  
    voltajeInst2 = 5.00 / 1023 * analogRead(A1); //Se normaliza el valor obtenido desde  
    el sensor  
  
    if (voltajeInst2 >= voltajeMax2) { //Si la corriente instantánea es mayor a la  
corriente máxima...  
      voltajeMax2 = voltajeInst2; //Se asigna ese valor a la corriente máxima  
    }  
    if (voltajeInst2 <= voltajeMin2) { //Si la corriente instantánea es mayor a la  
corriente mínima...  
      voltajeMin2 = voltajeInst2; //Se asigna ese valor a la corriente mínima  
    }  
  }  
  float Vpp2 = voltajeMax2 - voltajeMin2; //Se halla la corriente pico a pico;  
  Irms2 = (5.3568 * Vpp2) - 0.15;  
  if (Irms2 <= 0) { //Si la corriente rms es menor o igual a 0...  
    Irms2 = 0; //Se le asigna 0  
  }  
  potencia2 = Irms2 * 120;  
  
  Serial.print(vac);  
  Serial.print(',');  
  Serial.println(potencia2);  
  delay(3000);  
  Serial.flush();  
}  
  
////////////////////////////////////////////////////////////////////
```

Programación de WiFi ESP-01S

```
#include <ESP8266WiFi.h> //Librería para el módulo WiFi
#include <PubSubClient.h> //Librería para protocolo MQTT

const char* ssid = "wi-fi"; //SSID de la red WiFi
const char* password = "*****"; //Contraseña de la red WiFi
const char* mqtt_server = "plataforma.cf"; //Dirección del servidor MQTT
const char* mqttUser = "placa1"; //Usuario para conexión
const char* mqttPassword = "placa1"; //Contraseña para conexión

WiFiClient espClient; //Inicialización de cliente
PubSubClient client(espClient);

void setup_wifi() {

  delay(10);
  WiFi.begin(ssid, password); //Inicialización de conexión

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
  }
}

void callback(char* topic, byte* payload, unsigned int length) { //Recepción de datos
                                                                    enviados desde el
                                                                    servidor remoto

  if ((char)payload[0] > 0) //Envío de datos recibidos
                                                                    hacia Arduino Nano
    Serial.write((char)payload[0]);
}

void reconnect() { //Reconexión al servidor cada 5
                  segundos en caso de fallar

  while (!client.connected()) {

    if (client.connect("ESP8266", mqttUser, mqttPassword)) {
      client.subscribe("inTopic");
    } else {
      delay(5000);
    }
  }
}
```

Llamado a funciones de conexión con el servidor, comunicación serial y puerto de comunicación

```
//////////////////////////////////////////////////////////////////  
void setup() {  
  Serial.begin(9600);  
  delay(10);  
  setup_wifi();  
  client.setServer(mqtt_server, 1883);  
  client.setCallback(callback);  
}
```

```
//////////////////////////////////////////////////////////////////
```

```
void loop() {  
  if (!client.connected()) {  
    reconnect();  
  }  
  client.loop();  
}
```

Cambiar valores enviados por Arduino Nano a tipo entero

```
//////////////////////////////////////////////////////////////////
```

```
WiFiClient client;  
if (Serial.available()) {  
  String first = Serial.readStringUntil(',');  
  String sec = Serial.readStringUntil('\0');  
  int data1 = first.toInt();  
  int data2 = sec.toInt();  
}
```

```
//////////////////////////////////////////////////////////////////
```

Si existe conexión enviar valores a las bases de datos de la plataforma de administración y gestión

```
//////////////////////////////////////////////////////////////////
```

```
if (client.connect(mqtt_server, 80) > 0 && data1 > 5 )  
{  
  client.print("POST /plataforma/sensor1.php?valor=");  
  client.print(data1);  
  client.println(" HTTP/1.0");  
  client.println("User-Agent: Arduino 1.0");  
  client.println();  
  client.stop();  
  client.flush();  
  Serial.flush();  
  data1=0;  
  delay(200);  
}
```


Scripts de RHEL

Todos los scripts y carpetas relacionadas al funcionamiento del servidor se encuentran en el directorio `/opt/lampp/htdocs`

Index.html Es la primera página en aparecer al escribir *plataforma.cf* en el navegador, en ella se debe escribir un usuario y contraseña autorizados, para acceder al sistema.

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <title>Plataforma</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, user-scalable=no, initial-
scale=1, maximum-scale=1, minimum-scale=1">
    <link rel="stylesheet" href="estilos.css">
  </head>
  <body>
    <form action="validar.php" method="post">
      <h2>Ingrese sus Datos</h2>
      <input type="text" placeholder="Usuario" name="usuario">
      <input type="password" placeholder="Contraseña" name="clave">
      <input type="submit" value="Aceptar">
    </form>
  </body>
</html>
```

Validar.php Toma los datos ingresados en el script anterior y realiza una comparación con la base de datos de los usuarios registrados, si son correctos se autoriza el acceso a la página web principal, caso contrario se despliega un mensaje de error y se rechaza la conexión

```
<?php
$usuario=$_POST['usuario'];
$clave=md5 ($_POST['clave']);

$conexion=mysqli_connect("localhost", "root", "Utn123", "login");
$consulta="SELECT * FROM usuarios WHERE usuario='$usuario' and clave= '$clave'";
$resultado=mysqli_query($conexion, $consulta);
$filas=mysqli_num_rows($resultado);

if ($filas>0){
  session_start();
  $_SESSION["autentica"]="SIP";
  header("location:page_test.php");
}
```

```
else {
    echo "<font size=10>Error en datos ingresados</font>";
}
mysqli_free_result($resultado);
mysqli_close($conexion);
```

Seguridad.php Evita que la página web principal sea accedida escribiendo directamente la dirección en el navegador, por lo tanto, se debe pasar obligatoriamente por el control de seguridad de la primera página web.

```
<?php
@session_start();
if ($_SESSION["autentica"]!="SIP")
{
header("location:index.html");
exit();
}
?>
```

Ar.php Es la página web principal, donde se encuentra toda la información concerniente a las bases de datos, así como la interfaz para interactuar entre los subsistemas (módulo electrónico y plataforma de administración y gestión). El funcionamiento del código se describe por bloques.

1. Se verifica el control de seguridad y se establece el formato de la página web.

```
////////////////////////////////////
<?php
include ("seguridad.php");
require_once ("/opt/lampp/htdocs/plataforma/config.php");
?>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <title>Plataforma</title>
        <link href="ar.css"
            rel="stylesheet" type="text/css">
</head>
<body>
////////////////////////////////////
```


2. Contiene el botón salir para cerrar la sesión, y accede a la base de datos que contiene los nombres de equipos y las potencias nominales.

```

////////////////////////////////////
<section class="container">

    <p class="text-center"><a href="salir.php" class="btn btn-secondary btn-
lg">Salir</a></p>
    </section>

<?php
require ("/opt/lampp/htdocs/plataforma/con_parametros.php");
$sql_name = "SELECT * FROM equipos";
    $result = mysqli_query($con,$sql_name);
    $registros = mysqli_fetch_array($result);
require ("/opt/lampp/htdocs/plataforma/con_parametros.php");
$sql_name2 = "SELECT * FROM potencia";
    $result2 = mysqli_query($con,$sql_name2);
    $registros2 = mysqli_fetch_array($result2);
?>

////////////////////////////////////

```

3. Los campos de texto muestran los valores guardados y permiten ingresar nuevos valores de equipos y potencias.

```

////////////////////////////////////

<button onclick="equiposFunction()">Ingresar nuevos equipos</button>

<form id="formulario" method="post" action="datos_test.php">
<input type="text" name="equipo1" value="<?php
echo $registros['nombre'];
?>" placeholder="Equipo 1" id="equipo1"/>

<input type="number" min="0" name="pot1" value="<?php
echo $registros2['potencia_nominal'];
?>" placeholder="Potencia 1" id="pot1"/>

<input type="text" name="equipo2" value="<?php
echo $registros['nombre2'];
?>" placeholder="Equipo 2" id="equipo2"/>

<input type="number" min="0" name="pot2" value="<?php
echo $registros2['potencia_nominal2'];
?>" placeholder="Potencia 2" id="pot2"/>


```



```

        <span class="onoffswitch-inner"></span>
        <span class="onoffswitch-switch"></span>
    </label>
</div>
<p><b>
<?php

    require ("/opt/lampp/htdocs/plataforma/con_parametros_test.php");
    $sql_name = "SELECT * FROM equipos";
    $result = mysqli_query($con,$sql_name);
    $registros = mysqli_fetch_array($result);
    echo $registros['nombre'];
?>
</b></p>

```

```

<script type="text/javascript" src="https://code.jquery.com/jquery-3.1.1.min.js"></script>
<script type="text/javascript">
function funcion1(){
var checkBox1 = document.getElementById("myonoffswitch");
if(checkBox1.checked == true){
$.ajax({
type: "POST",
url: '/button/encendido.php',
data:{action:'call_this'},
success:function(html){;
}
});
} else {
$.ajax({
type: "POST",
url: '/button/apagado.php',
data:{action:'call_this'},
success:function(html){;
}
});
}
}
</script>

```

```

<div class="onoffswitch">
    <input type="checkbox" onclick="funcion2()" name="onoffswitch2"
class="onoffswitch-checkbox" id="myonoffswitch2" checked>
    <label class="onoffswitch-label" for="myonoffswitch2">
        <span class="onoffswitch-inner"></span>

```

```

        <span class="onoffswitch-switch"></span>
    </label>
</div>
<p><b>
<?php

    echo $registros["nombre2"];
?>
</b></p>

<script type="text/javascript" src="https://code.jquery.com/jquery-
3.1.1.min.js"></script>
<script type="text/javascript">
function funcion2(){
var checkBox2 = document.getElementById("myonoffswitch2");
if(checkBox2.checked == true){
$.ajax({
type: "POST",
url: '/button/encendido2.php',
data:{action:'call_this'},
success:function(html){
}
});
}else{
$.ajax({
type: "POST",
url: '/button/apagado2.php',
data:{action:'call_this'},
success:function(html){
}
});
}
}
}
</script>

<div class="onoffswitch">
    <input type="checkbox" onclick="funcion3()" name="onoffswitch3"
class="onoffswitch-checkbox" id="myonoffswitch3" checked>
    <label class="onoffswitch-label" for="myonoffswitch3">
        <span class="onoffswitch-inner"></span>
        <span class="onoffswitch-switch"></span>
    </label>
</div>
<p><b>
<?php

    echo $registros["nombre3"];
?>
</b></p>

```



```
Sensor1 //Recibe los valores del sensor1 y los guarda en la base de datos
<?php
require ("config.php");
$valor=mysqli_real_escape_string($con,$_GET['valor']);
$query="INSERT INTO sensor1(valor) VALUES('".$valor."')";
mysqli_query($con, $query);
mysqli_close($con);
?>
```

```
Sensor2 //Recibe los valores del sensor2 y los guarda en la base de datos
<?php
require ("config.php");
$valor=mysqli_real_escape_string($con,$_GET['valor']);
$query="INSERT INTO sensor2(valor) VALUES('".$valor."')";
mysqli_query($con, $query);
mysqli_close($con);
?>
```

```
Sensor3 //Recibe los valores del sensor3 y los guarda en la base de datos
<?php
require ("config.php");
$valor=mysqli_real_escape_string($con,$_GET['valor']);
$query="INSERT INTO sensor3(valor) VALUES('".$valor."')";
mysqli_query($con, $query);
mysqli_close($con);
?>
```

Anexo L. Manual de Usuario

1. El usuario que haga uso de este Sistema de Monitoreo de Consumo Eléctrico debe ingresar a la página web plataforma.cf e ingresar el nombre y contraseña proporcionados por el administrador del Sistema.



The image shows a web browser window with the address bar displaying 'plataforma.cf'. The main content area has a solid blue background. In the lower right quadrant, there is a white rectangular form with a grey header that reads 'Ingrese sus Datos'. Below the header, there are two input fields. The first field contains the text 'plataforma'. The second field contains a series of dots '.....'. At the bottom of the form is a blue button with the text 'Aceptar' in white.

2. A continuación deberá establecer los parámetros iniciales del sistema. Al hacer click en *Ingresar nuevos equipos* se despliegan varios campos de texto, allí se debe ingresar el nombre de cada equipo, la potencia nominal dada por el fabricante, y la frecuencia establecida con la que se desea tomar los datos de consumo eléctrico para cada uno de los equipos. Seguidamente debe ingresar el *Periodo de apagado*, es decir, el horario durante el cual se apaga el módulo electrónico para ahorrar energía, en caso de no desear activar esta funcionalidad, deberá dejarse los valores en cero.

[Salir](#)

SISTEMA DE MONITORIZACIÓN DE CONSUMO ELÉCTRICO

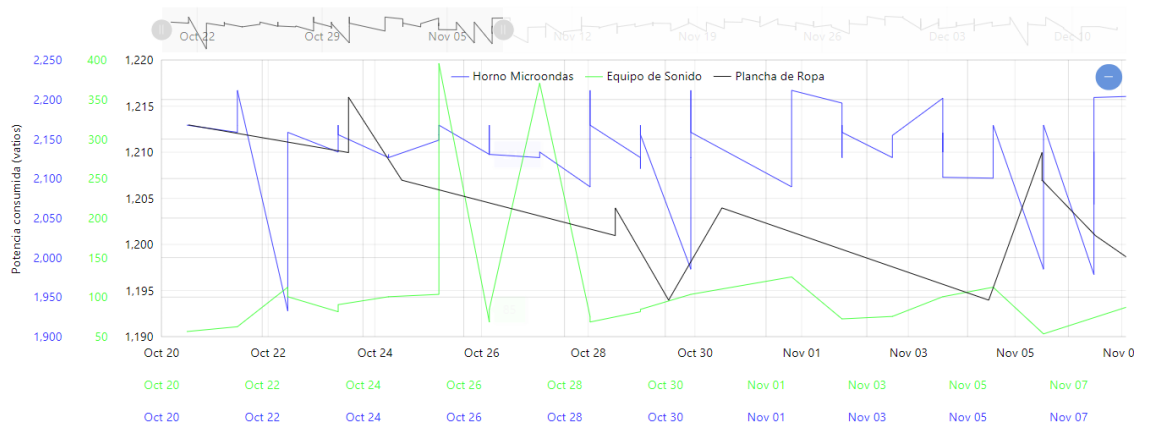
<input type="button" value="Ingresar nuevos equipos"/>			
Foco	<input type="text" value="220"/>	<input type="text" value="vatios"/>	Frecuencia de medición (minutos) <input type="text"/>
Laptop	<input type="text" value="260"/>	<input type="text" value="vatios"/>	Frecuencia de medición (minutos) <input type="text"/>
Equipo 3	<input type="text" value="1200"/>	<input type="text" value="vatios"/>	Frecuencia de medición (minutos) <input type="text"/>
<input type="button" value="Aceptar"/>			
Periodo de apagado	<input type="text" value="Desde 15:50 hasta 16:30"/>		

3. En la parte superior derecha de la página web se muestra la interfaz de encendido/apagado de los equipos, mediante la cual se controla el paso de energía hacia cada uno de ellos.

SISTEMA DE MONITORIZACIÓN DE CONSUMO ELÉCTRICO



4. En la parte central de la página web se muestra la gráfica del consumo eléctrico a través del tiempo, de los equipos registrados. Existen escalas para cada equipo, en el lado izquierdo se encuentran las escalas de la potencia consumida en vatios; y en la parte inferior se tiene las escalas de tiempo. Además se puede filtrar resultados según las fechas de medición, con la línea de tiempo que se ubica en la parte superior de la gráfica.



- Al detectarse un consumo eléctrico excesivo por parte de cualquier equipo ingresado se mostrará una alerta en la página web, y se enviará un correo electrónico a su usuario para informarle de la situación.



- Los equipos ingresados en la página web deben conectarse en el lugar adecuado del módulo electrónico, siguiendo la ubicación de los nombres.

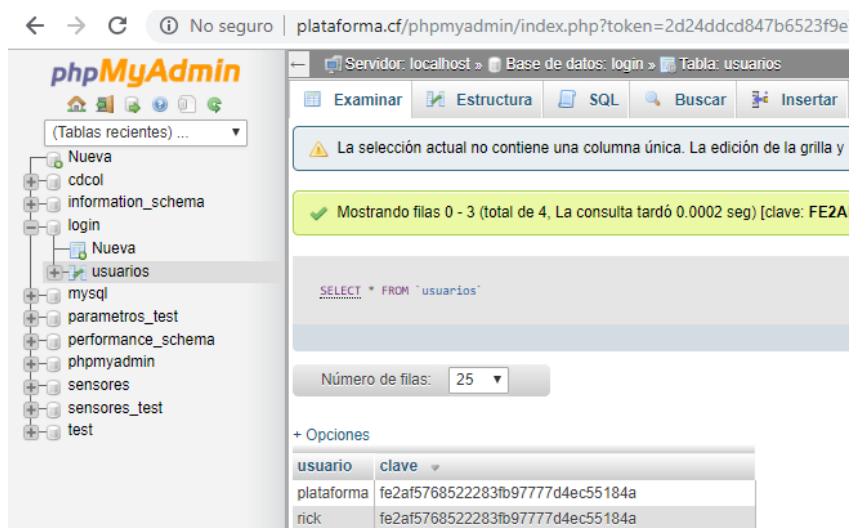


7. En caso de requerir la activación del módulo electrónico cuando se encuentre desactivado, se tiene un botón al costado izquierdo del prototipo.

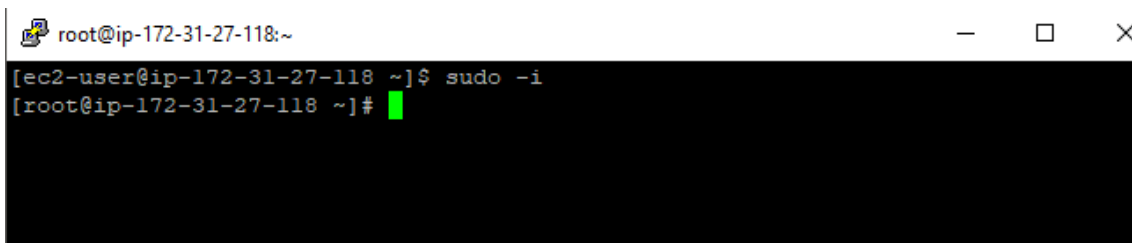


Anexo M. Manual de Administrador

1. El administrador deberá crear nuevos usuarios y generar contraseñas para cada uno de ellos, para eso debe ingresar la dirección plataforma.cf/phpmyadmin en el navegador y colocar su usuario y contraseña. A continuación, deberá crear los nuevos usuarios en la base de datos *usuarios*.



2. El acceso a la instancia se realiza mediante el software PuTTY con la dirección IP pública y la llave privada, los scripts de programación se encuentran en la dirección `/opt/lampp/htdocs`.



3. Para agregar más equipos se debe editar los archivos *amchart.php* y *ar.php*. Desde la línea 15 hasta la 22 de *amchart.php*, se encuentra la estructura de código para obtener los valores para la gráfica de un equipo, se deberá cambiar por una nueva tabla de la base de datos.

```
root@ip-172-31-27-118:/opt/lampp/htdocs
12
13
14
15 <?php
16 $sql = "SELECT *, DATE_FORMAT(tiempo, '%Y, %m-%l, %d, %H, %i') AS fecha FROM sensor1";
17 $result = mysqli_query($con,$sql);
18 while($registros = mysqli_fetch_array($result))
19 {
20     echo "data.push({ date1: new Date({$registros['fecha']}), price1: {$registros['valor']}
21 });";
22 ?>
```

En el mismo archivo se encuentra el código que dibuja la gráfica de cada equipo, en la imagen se muestra el código para dos equipos, para agregar más se debe copiar la misma estructura, cambiando las variables.

```
root@ip-172-31-27-118:/opt/lampp/htdocs
65 var dateAxis = chart.xAxes.push(new am4charts.DateAxis());
66 dateAxis.renderer.grid.template.location = 0;
67 dateAxis.renderer.labels.template.fill = am4core.color("#3333FF");
68
69 var dateAxis2 = chart.xAxes.push(new am4charts.DateAxis());
70 dateAxis2.renderer.grid.template.location = 0;
71 dateAxis2.renderer.labels.template.fill = am4core.color("#3cfff33");
72
73
74 var valueAxis = chart.yAxes.push(new am4charts.ValueAxis());
75 valueAxis.tooltip.disabled = true;
76 valueAxis.renderer.labels.template.fill = am4core.color("#3333FF");
77 valueAxis.renderer.minWidth = 60;
78
79 var valueAxis2 = chart.yAxes.push(new am4charts.ValueAxis());
80 valueAxis2.tooltip.disabled = true;
81 valueAxis2.renderer.labels.template.fill = am4core.color("#3cfff33");
82 valueAxis2.renderer.minWidth = 60;
```

En el archivo ar.php se muestra la estructura de los campos de parametrización de nuevos equipos.

```
55 <input type="text" name="equipol" value="<?php
56 echo $registros['nombre'];
57 ?>" placeholder="Equipo 1" id="equipol" >
58
59 <input type="number" style="width:7em;" min="0" name="potl" value="<?php
60 echo $registros2['potencia nominal'];
61 ?>" placeholder="Potencia 1" id="potl" />
62 <p></p>
```

También se indica la comparación de los valores prefijados con los valores recibidos por el módulo electrónico; y el llamado a las alertas.

```
172 $sql="SELECT valor FROM sensor1 where id=(SELECT MAX(id) FROM sensor1)";
173 $result=mysqli_query($con,$sql);
174 while($registros=mysqli_fetch_array($result))
175 {
176
177 if($registros["valor"]>$registros2 'potencia_nominal' +50)
178
179 {
180 require_once "/opt/lampp/htdocs/alertas_test/alerta.html";
181 require_once "/opt/lampp/htdocs/correo_equipo.php";
182 require_once ("/opt/lampp/htdocs/plataforma/con_parametros_test.php");
183
184 $sql="UPDATE eventos SET equipol=equipol+1"
185 $s=mysqli_query($con, $sql);
186
187
188 }
189 }
190 mysqli_free_result($result);
191 mysqli_close($con);
```

Se indica la estructura de los botones

```
282 <script type="text/javascript" src="https://code.jquery.com/jquery-3.1.1
.min.js"></script>
283 <script type="text/javascript">
284
285 var checked = JSON.parse(localStorage.getItem("myonoffswitch"));
286 document.getElementById("myonoffswitch").checked = checked;
287 function funcion1 () {
288 var checkBox1 = document.getElementById("myonoffswitch");
289 localStorage.setItem("myonoffswitch", checkBox1.checked);
290 if (checkBox1.checked == true) {
291 $.ajax({
292 type: "POST",
293 url: '/button/encendido.php',
294 data: {action:'call_this'},
295 success: function (html) {;
296 }
297 });
```

Y el código para el registro de eventos

```
436 <?php
437 require_once ("/opt/lampp/htdocs/plataforma/con_parametros_test.
php");
438 $sql_name = "SELECT * FROM equipos";
439 $result = mysqli_query($con,$sql_name);
440 $registros = mysqli_fetch_array($result);
441 echo $registros['nombre3']; ?> ;
442 <?php
443
444 require_once ("/opt/lampp/htdocs/plataforma/con_parametros_test.
php");
445 $sql_name = "SELECT * FROM eventos";
446 $result = mysqli_query($con,$sql_name);
447 $registros = mysqli_fetch_array($result);
448 echo $registros['equipo3'];
449 ?> alertas registradas.
450
451 </div>
```


4. El correo electrónico para envío de alerta se encuentra en la dirección *correo_equipo.php*

```
root@ip-172-31-27-118:/opt/lampp/htdocs
<?php
echo shell_exec('echo "Alerta: Consumo eléctrico excedido en uno de sus equipos"
| mail -v -s "Plataforma de Consumo Eléctrico" rickjoseph69@gmail.com' );
?>
```

5. Para almacenar los datos ingresados en la parametrización se debe editar el archivo *datos.php* siguiendo la misma estructura de los demás equipos.

```
root@ip-172-31-27-118:/opt/lampp/htdocs
?php
$equipol=$_POST["equipol"];
$equipo2=$_POST["equipo2"];
$equipo3=$_POST["equipo3"];
$pot1=$_POST["pot1"];
$pot2=$_POST["pot2"];
$pot3=$_POST["pot3"];

require_once("/opt/lampp/htdocs/plataforma/con_parametros.php");

if($con === false){
    die("ERROR: Could not connect. " . mysqli_connect_error());
}
$sql_update = "UPDATE equipos SET nombre='$equipol',nombre2='$equipo2',nombre3='$equipo3'";
$sql_update2 = "UPDATE potencia SET potencia_nominal='$pot1',potencia_nominal2='$pot2', potencia_nominal3='$pot3'";

if(mysqli_query($con, $sql_update) && mysqli_query($con, $sql_update2)){
    header('Location:ar.php');
} else {
    echo "ERROR: Could not able to execute $sql_update. " . mysqli_error($con);
}

mysqli_close($con);
?>
```

6. Para usar el prototipo en una nueva red se debe cambiar los valores de *ssid* y *password* en el programa del WiFi ESP-01S

```
#include <ESP8266WiFi.h>
#include <PubSubClient.h>
const char* ssid = "Carmita Rosero";
const char* password = "*****";
```

