



INSTITUTO TECNOLÓGICO DE HERMOSILLO
DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN
MAESTRÍA EN INGENIERÍA ELECTRÓNICA

TESIS PARA OBTENER EL GRADO DE MAESTRO EN
INGENIERÍA ELECTRÓNICA

“Desarrollo de un sistema de telemetría para monitoreo de servicios básicos en una instalación residencial con sistema fotovoltaico interconectado a la red eléctrica”

PRESENTADA POR:

ING. JORGE ALBERTO YANEZ PRECIADO

Director de Tesis:

M.C. Fredy Alberto Hernández Aguirre

Codirector de Tesis:

M.C. José Manuel Chávez

Revisores:

M.C. Rafael Armando Galaz Bustamante

Dr. Carlos Alberto Pereyda Pierre

Hermosillo, Sonora, México; junio de 2021





SECCIÓN: DIV. EST. POS. E INV.
No. OFICIO: DEPI/182/21.
ASUNTO: AUTORIZACIÓN DE IMPRESIÓN
DE TESIS.

15 Julio de 2021

**C. JORGE ALBERTO YANEZ PRECIADO,
P R E S E N T E.**

Por este conducto, y en virtud de haber concluido la revisión del trabajo de tesis que lleva por nombre **“DESARROLLO DE UN SISTEMA DE TELEMETRÍA PARA MONITOREO DE SERVICIOS BÁSICOS EN UNA INSTALACIÓN RESIDENCIAL CON SISTEMA FOTOVOLTAICO INTERCONECTADO A LA RED ELÉCTRICA”**, que presenta para el examen de grado de la MAESTRÍA EN INGENIERÍA ELECTRÓNICA, y habiéndola encontrado satisfactoria, nos permitimos comunicarle que se autoriza la impresión del mismo a efecto de que proceda el trámite de obtención de grado.

Deseándole éxito en su vida profesional, quedo de usted.

ATENTAMENTE

M.C. FREDY ALBERTO HERNÁNDEZ AGUIRRE
DIRECTOR

M.C. JOSÉ MANUEL CHÁVEZ
CO- DIRECTOR

M.C. RAFAEL ARMANDO GALAZ BUSTAMANTE
SECRETARIO



DR. CARLOS ALBERTO PEREYDA PIERRE
VOCAL

INSTITUTO TECNOLÓGICO
DE HERMOSILLO
DIVISIÓN DE ESTUDIOS
DE POSGRADO

M.C.O. ROSA IRENE SANCHEZ FERMÍN
JEFA DE LA DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

RISF/momv*





CARTA CESIÓN DE DERECHOS

En la ciudad de Hermosillo Sonora a el día 16 de agosto del año 2021 el que suscribe C. Jorge Alberto Yanez Preciado, alumno de la maestría en Electrónica adscrito a la División de Estudios de Posgrado e Investigación, manifiesta que es autor intelectual del presente trabajo de Tesis titulado "Desarrollo de un sistema de telemetría para monitoreo de servicios básicos en una instalación residencial con sistema fotovoltaico interconectado a la red eléctrica" bajo la dirección de Fredy Alberto Hernández Aguirre y ceden los derechos del mismo al Tecnológico Nacional de México/Instituto Tecnológico de Hermosillo, para su difusión, con fines académicos y de investigación.

Los usuarios de la información no deben de reproducir el contenido textual, graficas, tablas o datos contenidos sin el permiso expreso del autor y del director del trabajo. Este puede ser obtenido a la dirección de correo electrónico siguiente: lct_jorge@hotmail.com. Una vez otorgado el permiso se deberá expresar el agradecimiento correspondiente y citar la fuente del mismo.

A T E N T A M E N T E

Jorge Alberto Yanez Preciado



DEDICATORIA

Para mi familia más cercana, mis amigos, mi comité de revisión y mi gato Gerald.

AGRADECIMIENTOS

Quiero agradecer al Consejo Nacional de Ciencia y Tecnología (CONACYT) por hacer esto posible, al Tecnológico Nacional de México y al Instituto Tecnológico de Hermosillo por abrirme las puertas para realizar mis estudios y hacer uso de sus instalaciones.

Agradezco mucho la ayuda de mi director de tesis, el maestro Fredy Alberto Hernández Aguirre, de mi codirector de tesis José Manuel Chávez, a mis revisores Carlos Pereyda Pierre y Rafael Armando Galaz Bustamante, a mis compañeros de estudio y al personal del Instituto Tecnológico de Hermosillo.

Quiero agradecer especialmente a mi novia y compañera de vida Claudia Iveth Rodríguez, a su madre Ma. Del Refugio, y a sus hermanos, Yaneth, Rubén y Karla, pues su constante apoyo incondicional, fueron las bases para que esto empezara y culminara satisfactoriamente; de igual forma le agradezco a mi madre Martha, a mi Padre Jorge y a mi abuela Socorro, quienes siempre estuvieron al pendiente de mi travesía. Le agradezco a mis hermanos Ana y Sebastián, a mis familiares más cercanos, y a los Ingenieros Ramsés Bustamante, Luis Cortinas, Moisés Rentería, Tadeo Bustamante, Oswaldo Verdugo, por apoyarme en el desarrollo de algunas de las ingenierías requeridas en el proyecto.

RESUMEN

En el presente trabajo de tesis, se exhibe el desarrollo y los resultados obtenidos en el diseño de un sistema de telemetría para la medición de distintos parámetros en el hogar, para tener la posibilidad de conocer y monitorear los datos de manera remota, a través de una aplicación móvil, en tiempo real. Haciendo uso de instrumentación se realizó la medición del consumo de energía eléctrica en el hogar (kWh), el volumen de agua consumido, la presencia de gas LP, iluminación interior, temperatura y la energía generada a través de una arquitectura de panel fotovoltaico.

La arquitectura propuesta para el desarrollo de este proyecto de tesis está basada en el modelo de tres capas del internet de las cosas, haciendo uso de un microcontrolador basado en el chip ESP8266 llamado NodeMCU®, se realiza la lectura de los parámetros propuestos anteriormente, estos datos son transmitidos por medio de un protocolo de comunicación Wi-Fi (*Wireless Fidelity*) llamado Websockets®, los datos son enviados a una base de datos en internet utilizando el servicio de Firebase®, posteriormente esta base de datos se comunica con una aplicación móvil en la cual se visualizan los datos que el microcontrolador carga a la base de datos, en una interfaz gráfica programada con Android Studio®, el usuario podrá monitorear los distintos parámetros en el hogar en tiempo real y en base a su criterio, ejercer acciones preventivas y correctivas para el aprovechamiento de sus recursos energéticos.

ABSTRACT

This thesis work shows the development and results obtained in the design of a telemetry system for the measurement of different parameters at home, in order to have the possibility of knowing and monitoring the data remotely, through a mobile application and in real time. Using instrumentation, the home's electrical energy consumption (kWh), the volume of water consumed, the presence of LP gas, interior lighting, temperature and the energy generated through a photovoltaic panel architecture were measured.

The proposed architecture for the development of this thesis project is based on the three-layer model of the internet of things, using a microcontroller based on the ESP8266 chip called NodeMCU®, the reading of the parameters proposed above is performed, these data are transmitted via a Wi-Fi communication protocol (Wireless Fidelity) called Websockets®, the data are sent to a database on the internet using the Firebase® service, Later this database communicates with a mobile application in which the data that the microcontroller uploads to the database is displayed in a graphical interface programmed with Android Studio®, the user can monitor the various parameters in the home in real time and based on their criteria, exercise preventive and corrective actions for the use of their energy resources.

ÍNDICE

CAPÍTULO 1. INTRODUCCIÓN	1
1.1 ANTECEDENTES	1
PLANTEAMIENTO DEL PROBLEMA	2
1.2 OBJETIVOS GENERALES Y PARTICULARES	2
1.2.1 Objetivo general	2
1.2.2 Objetivos particulares	2
1.3 JUSTIFICACIÓN	3
1.4 METAS Y ALCANCES	4
CAPÍTULO 2. MARCO CONCEPTUAL Y REVISIÓN DEL ESTADO DEL ARTE	5
2.1 MARCO CONCEPTUAL	11
2.1.1 SENSORES ANALÓGICOS	12
2.1.2 SENSORES INTELIGENTES	12
2.1.3 APLICACIONES MÓVILES	10
2.1.4 ENTORNO DE DESARROLLO INTEGRADO	11
2.1.5 ANDROID STUDIO	11
2.1.6 MICROCONTROLADORES	11
2.1.7 CHIP INTEGRADO ESP8266	12
2.1.8 NODEMCU	13
2.1.9 <i>INTERNET OF THINGS</i>	14
2.1.10 ARQUITECTURA DE TRES CAPAS DE UNA RED IOT	14
2.1.11 PROTOCOLO DE COMUNICACIÓN MQTT	16
2.1.12 BACKEND COMO UN SERVICIO	16
2.1.13 FIREBASE	17
2.1.14 BASE DE DATOS EN TIEMPO REAL DE FIREBASE	17
2.1.15 MEDIDORES INTELIGENTES	18
2.2 ESTADO DEL ARTE	19
2.2.1 ALTERNATIVAS REPORTADAS	24
CAPÍTULO 3. DESARROLLO	25
3.1 SOLUCIÓN PROPUESTA	25
3.2 METODOLOGÍA A DESARROLLAR	27
3.3 MODELADO, SIMULACIÓN Y EXPERIMENTACIÓN	29
3.4 RESULTADOS OBTENIDOS	32
3.4.1 PRUEBA DE PROTOCOLO MQTT CON INTERFAZ NODE-RED	32

3.4.2 DESARROLLO E INTEGRACIÓN DE PROTOTIPO CON PROTOCOLO WEBSOCKET Y APLICACIÓN MÓVIL	40
3.5 VALIDACIÓN DE SENSORES SELECCIONADOS	40
a) SENSOR DE ENERGÍA	40
b) SENSOR DE FLUJO DE AGUA	45
c) SENSOR DE PRESENCIA DE GAS LP	48
d) SENSOR DE TEMPERATURA	49
e) SENSOR DE ILUMINACIÓN	50
3.6 MEDICIÓN DE PANEL FOTOVOLTAICO	51
3.7 INTEGRACIÓN DE PROTOTIPO	51
3.8 PROGRAMACIÓN DE SKETCH PARA LECTURA DE SENSORES	52
3.9 COMUNICACIÓN DE MICROCONTROLADOR CON BASE DE DATOS	54
3.10 DESARROLLO DE INTERFAZ GRÁFICA EN APLICACIÓN MÓVIL ANDROID	56
CAPÍTULO 4. ANÁLISIS DE RESULTADOS	61
4.1 ANÁLISIS DE RESULTADOS	61
4.1.1 COSTOS DE ADQUISICIÓN	61
4.1.2 COSTOS INDIRECTOS	62
4.1.3 ANÁLISIS DE COSTO	62
4.1.4 ANÁLISIS DE IMPLEMENTACIÓN DE PROTOTIPO DE SISTEMA ELECTRONICO	63
4.2 COMPARACIÓN DEL SISTEMA VS PRODUCTOS COMERCIALES	65
CAPÍTULO 5. CONCLUSIONES	67
5.1 CONCLUSIONES GENERALES	67
5.2 CONCLUSIÓN DE LA EXPERIMENTACIÓN	68
5.3 TRABAJOS FUTUROS	69
REFERENCIAS BIBLIOGRÁFICAS	70
APÉNDICES	73
APÉNDICE 1	73
APÉNDICE 2	77
APÉNDICE 3	83

LISTA DE FIGURAS

Figura 2.1. Estructura de un sensor inteligente. -----	5
Figura 2.2. Chip integrado ESP8266.-----	6
Figura 2.3. NodeMCU, placa de desarrollo basada en el IC ESP8266.-----	8
Figura 2.4. Arquitectura de tres capas de una red IoT. -----	8
Figura 2.5. Arquitectura de protocolo MQTT.-----	9
Figura 2.6. Sensor de temperatura digital DS18B20 con protocolo de comunicación 1-wire.-----	9
Figura 2.7. Sensor de iluminación digital BH1750. -----	10
Figura 2.8. Circuito integrado ESP8266.-----	13
Figura 2.9. NodeMCU, placa de desarrollo basada en el IC ESP8266.-----	14
Figura 2.10. Arquitectura de tres capas de una red IoT.-----	15
Figura 2.11. Ejemplo de una instalación eléctrica tradicional. -----	18
Figura 3.1. Arquitectura propuesta para proyecto. -----	26
Figura 3.2. Metodología. -----	27
Figura 3.3. Diagrama unifilar de conexiones requeridas para el proyecto.-----	31
Figura 3.4. Diagrama de conexión de sensor a NodeMCU. -----	33
Figura 3.5. Sensor de temperatura conectado a NodeMCU. -----	33
Figura 3.6. Línea de comandos de Windows, comando Node-RED. -----	34
Figura 3.7. Entorno de desarrollo Node-RED.-----	34
Figura 3.8. Componente MQTT in. -----	35
Figura 3.9. Configuración de objeto en Node-RED.-----	36
Figura 3.10. Parámetros de configuración de servidor CloudMQTT. -----	36
Figura 3.11. Configuración para conexión de componente con servidor CloudMQTT. -----	37
Figura 3.12. Componente configurado y conectado con servidor CloudMQTT. -----	37
Figura 3.13. Diagrama de interfaz gráfica de Node.RED. -----	38
Figura 3.14. Configuración de indicador de temperatura. -----	38
Figura 3.15. Interfaz gráfica de Node-RED.-----	39
Figura 3.16. Interfaz gráfica de Node-RED visualización de temperatura. -----	39
Figura 3.17. Diagrama eléctrico de instalación monofásica en una residencia. -----	41
Figura 3.18. Diagrama unifilar de conexiones para medición de gasto eléctrico en residencia. -----	42
Figura 3.19. Conexión física de sensor en instalación eléctrica residencial. -----	43
Figura 3.20. Facturación de periodo bimestral emitida por Comisión Federal de Electricidad. -----	44
Figura 3.21. Diagrama unifilar de conexión de sensor con el microcontrolador.-----	45
Figura 3.22. Planta para pruebas de sensor de agua. -----	46
Figura 3.23. Monitor serial registrando los datos del sensor de agua. -----	47
Figura 3.24. Monitor serial imprimiendo los datos del sensor de flujo de agua. -----	47

Figura 3.31. Diagrama unifilar de conexión de sensor MQ-2 con NodeMCU.-----	49
Figura 3.32. Monitor serial de microcontrolador imprimiendo el estado del sensor digital de presencia de gas LP.-----	49
Figura 3.33. Monitor serial de microcontrolador imprimiendo el valor de temperatura. -----	49
Figura 3.34. Diagrama unifilar de conexión de sensor de iluminación BH1750 con microcontrolador NodeMCU a través de protocolo de comunicación I^2C .-----	50
Figura 3.35. Monitor serial de microcontrolador imprimiendo el valor del sensor de iluminación. -----	50
Figura 3.36. Diagrama unifilar de conexiones de arquitectura fotovoltaica con microcontrolador. -----	51
Figura 3.37. Prototipo integrado.-----	52
Figura 3.38. Diagrama de flujo de Sketch para lectura de todos los sensores y publicación de los datos en la base de datos en tiempo real de Firebase. -----	53
Figura 3.39. Creación de proyecto en Firebase. -----	55
Figura 3.40. Pantalla principal de Firebase en el explorador web. -----	55
Figura 3.41. Valores visualizados en la base de datos.-----	56
Figura 3.42. (a) Pantalla principal de aplicación. (b) Pantalla de registro de usuario. -----	58
Figura 3.43. Pantalla de inicio de sesión. -----	59
Figura 3.44. Pantalla de visualización de datos. -----	60
Figura 4.1. Análisis de los resultados de costos. -----	63
Figura 4.2. Sketch compilado y cargado al microcontrolador sin errores.-----	64

Lista de tablas

Tabla 1. Selección de sensores para proyecto de tesis. -----	30
Tabla 2. Datos recolectados por el sensor de energía en la instalación monofásica del hogar. -----	43
Tabla 3. Conteo de pulsos recibidos por el sensor de flujo de agua para medición de un litro. -----	46
Tabla 4. Costos de adquisición de proyecto. -----	61
Tabla 5. Costos indirectos de proyecto. -----	62
Tabla 6. Comparación entre sistemas. -----	66

CAPÍTULO 1

INTRODUCCIÓN

1.1. ANTECEDENTES

En referencia a la contribución para la mejora del medio ambiente, y el desarrollo de sistemas electrónicos basados en el internet de las cosas para facilitar tareas de la vida cotidiana, se identificó que no existe un dispositivo implementado en las viviendas de México, que sea capaz de analizar, recaudar, procesar y registrar la lectura del consumo de energía eléctrica, consumo de agua y otras variables de interés para el usuario, con la finalidad de informar estos parámetros al dueño del hogar.

El sector residencial es, en el consumo de energía en México, el tercer sector de mayor consumo, significando casi el 16% del consumo energético total y poco más del 25% de la demanda de electricidad del país. [1] Además, este sector manifiesta un consumo siempre creciente a través del tiempo y la tendencia al alza de sus consumos de energía, tales como, el crecimiento de la población, el incremento del número de hogares, el tipo de equipos utilizados y el aumento del equipamiento.

En países como Estados Unidos, Alemania, y España existen empresas que se dedican a brindar este servicio, realizan la instalación del sistema y dependiendo de la empresa, se pueden visualizar las mediciones en una interfaz gráfica ya sea a través de un portal web o una aplicación.

PLANTEAMIENTO DEL PROBLEMA

Cada día se observan nuevos sistemas electrónicos en el mercado que implican el uso de una arquitectura basada en el internet de las cosas, En los hogares de México el uso de los servicios energéticos como la energía eléctrica, el agua y el gas es indispensable. Estos son utilizados diariamente para realizar los diversos quehaceres domésticos, creando la necesidad de optimizar los recursos naturales disponibles, ya que la demanda de los recursos naturales crece año tras año. Debido a la importancia de los recursos no renovables para el ser humano, el cliente debe actuar de una

manera responsable, racionalizando de la manera más eficiente posible el uso de los recursos energéticos.

En los hogares es común que se desconozca el consumo de energía eléctrica, consumo de agua y/o gas, actualmente las viviendas cuentan con un medidor electromecánico o digital para registrar el consumo de electricidad y agua, muchas veces se hace caso omiso de este dispositivo, derivado de la falta de conocimiento para interpretar los datos que se están midiendo y es hasta el final del ciclo de facturación donde se informa al contratista del servicio, el consumo de los recursos que se utilizaron en el mes. Conocer el consumo de servicios de manera periódica y en tiempo real, ayuda a detectar factores críticos que suelen encarecer las tarifas del servicio.

1.2 OBJETIVOS GENERALES Y PARTICULARES

1.2.1. Objetivo general

Diseñar un sistema electrónico, con protocolos de comunicación basados en IoT (*Internet of things*), para la medición de los servicios básicos (kWh, agua, presencia de gas, iluminación y temperatura) en una instalación residencial con sistema fotovoltaico interconectado a la red.

1.2.2. Objetivos particulares

- Investigar el estado del arte relacionado con el tema de tesis, con la finalidad de conocer los trabajos y patentes publicadas con respecto a medidores y plataformas desarrolladas para la medición de los parámetros propuestos: kWh (kilowatt-hora), agua, presencia de gas, iluminación y temperatura.
- Definir la arquitectura propuesta para la solución del problema, relacionada con el sistema de telemetría y aplicación *IoT* para la medición de parámetros propuestos, considerando una instalación fotovoltaica interconectada a la red.
- Analizar y seleccionar una plataforma digital para el procesamiento de datos y comunicación inalámbrica.
- Analizar y seleccionar los sensores para la medición de los parámetros propuestos.

- Integrar los sistemas digitales y desarrollo de algoritmos para el monitoreo local de las mediciones realizadas en una instalación residencial con sistema fotovoltaico interconectado a la red.
- Desarrollar una plataforma para la comunicación de los datos y diseño de una interfaz gráfica con aplicación móvil para la visualización de las mediciones realizadas en tiempo real.
- Integrar el sistema de telemetría completo (hardware y software) y la comunicación inalámbrica para realizar las pruebas eléctricas, corrección de errores y validación de la plataforma completa.

1.3. JUSTIFICACIÓN

Las tecnologías en sistemas embebidos han evolucionado favorablemente con el pasar de los años, actualmente se tienen sistemas y plataformas embebidas cada vez más robustas, capaces de crear un vínculo entre internet para almacenamiento de datos y la realidad que se vive día con día, esto crea una ventana de oportunidad para los proyectos que involucran el internet de las cosas, pues las plataformas son de bajo costo y brindan capacidades de programación muy vastas.

Los medidores inteligentes tienen ofertas comerciales en países mayormente desarrollados, las empresas realizan la venta de sus diferentes productos para medir distintos parámetros, y generalmente venden la licencia de su software en el cual se consultan los datos recabados por los sensores, estas alternativas tienen costos elevados debido a la ingeniería y la calidad de los materiales utilizados, los servicios de programación y/o el uso de sus instalaciones para almacenar información.

Con el desarrollo de tecnologías y protocolos basados en *IoT* se propone implementar un sistema de telemetría el cual sea capaz de tomar lecturas periódicamente de distintos parámetros propuestos.

El sistema de telemetría además de medir el consumo de energía eléctrica, se interconectará a un panel fotovoltaico con sus respectivos circuitos de acoplamiento para lograr medir la energía generada por el panel.

1.4. METAS Y ALCANCES

Para el tema de tesis propuesto se definen las siguientes metas:

- Selección y compra de la instrumentación propuesta para la lectura de las variables físicas de energía eléctrica, flujo de agua, presencia de gas, iluminación, temperatura y energía eléctrica generada por un panel fotovoltaico.
- Desarrollo de algoritmos de programación para sistema de medición para los parámetros deseados.
- Desarrollo de aplicación móvil para interfaz gráfica.
- Publicación de un artículo en congreso.

Con referencia a los alcances del proyecto se hará la selección y compra de la instrumentación requerida para el proyecto, seguido de la implementación de los sensores para interconectar mediante cableado, estos sensores a una plataforma digital que procese los datos y los envíe hacia una base de datos en internet, una vez obtenidos estos datos se desplegarán en una interfaz gráfica para proporcionar información útil al usuario final a través de una aplicación móvil para el sistema operativo Android.

Las limitaciones para el desarrollo del trabajo de tesis están relacionadas con las dificultades para la adquisición de los sensores requeridos para una medición de las variables.

CAPÍTULO 2

MARCO CONCEPTUAL Y REVISIÓN DEL ESTADO DEL ARTE

2.1. MARCO CONCEPTUAL

2.1.1. Sensores analógicos

Los sensores analógicos son aquellos que producen una señal continua en el tiempo y proporcional a la magnitud física que se está midiendo [2].

El tipo de salida producido por un sensor analógico varía dependiendo del fabricante y puede ser una señal de voltaje variable o una señal de corriente variable, es común utilizar una señal de 0 – 10 Volts o una señal de 4 – 20 mA de corriente.

Para la propuesta de sensores analógicos se analizó el sensor de corriente con número de parte SCT-013-030, el cual es un sensor de corriente no invasivo que permite medir la intensidad de corriente que atraviesa un conductor sin necesidad de cortar o modificar el cable. Este dispositivo de instrumentación brinda una medición proporcional a la intensidad de corriente eléctrica que atraviesa un circuito. La medición se realiza por inducción electromagnética. Esto permite medir la corriente instantánea en cualquier tipo de instalación eléctrica a la cual se tenga acceso al cable de fase. En la figura 2.1 se observa el sensor de corriente analógico.



Figura 2.1. Sensor de corriente analógico SCT-013-030. [3]

El sensor tiene una resistencia burden internamente de 62 Ohms, la cual proporciona un voltaje en la salida. El error de este sensor es de 1% del valor medido, y tiene una relación 1800 a 1 en el devanado secundario. Tiene un grado de protección

B en resistencia eléctrica, lo que lo hace un sensor robusto para aplicaciones industriales y en el hogar.

La salida de este sensor es de tipo analógica, y se requiere de un circuito de acondicionamiento de señal, para obtener una señal de 0 – 5 VCD que se pueda acoplar al sistema digital de medición.

A través de un algoritmo de programación en el microcontrolador se puede calcular la corriente eficaz o *RMS* empleando un sensor de corriente analógico y utilizando la siguiente fórmula:

$$I_{RMS} = \sqrt{\frac{1}{T} \int_0^T i^2 dt} = \sqrt{\frac{1}{N} \sum_{n=0}^N i^2} \quad \text{Calculo de valor efectivo de corriente [4]}$$

Además de un sensor de corriente, se requiere también un sensor de voltaje siguiendo la arquitectura de sensores analógicos para la medición de gasto eléctrico, se analizó el sensor Zmpt101b, este sensor de bajo costo de igual manera actúa como un transformador de CA – CA, con la ventaja que otorga el acondicionamiento de señal para ser conectado directamente al sistema digital de medición. En la figura 2.2 se observa el sensor analógico de voltaje.



Figura 2.2. Sensor analógico de voltaje. [5]

Soporta voltajes de entrada de hasta 250 VCA (Volts de Corriente Alterna) y entrega una señal sinusoidal de amplitud regulable por un potenciómetro en la placa. Esta señal de salida está desplazada positivamente para que la onda no tenga voltajes negativos y así poderla medir completamente con el *ADC*. El desplazamiento depende del voltaje con el que alimentemos el módulo: si el voltaje de alimentación es de 5 Volts el desplazamiento será de 2.5 Volts y si alimentamos el módulo con 3.3 Volts el

desplazamiento será de 1.65 Volts. El circuito de acondicionamiento de señal permite que el voltaje de salida del módulo pueda ser leído por cualquier microcontrolador con entrada analógica (*ADC*), de esta forma es posible medir el voltaje instantáneo y realizar cálculos de energía, como: voltaje pico a pico (*Vpp*) y voltaje eficaz (*Vrms*).

2.1.2. Sensores inteligentes

Los sensores inteligentes presentan un amplio rango de características adicionales a los sensores analógicos convencionales, se diferencian de estos porque integran las funciones de medición, acondicionamiento de señal, procesamiento digital y comunicaciones en redes permitiendo su programación según las exigencias específicas de una aplicación.

Sin lugar a duda los sensores inteligentes forman parte de una tecnología nueva y un nuevo concepto al plantear el acondicionamiento y adquisición de señales analógicas. Es importante resaltar la ventaja que presenta este tipo de sensores sobre la oferta tradicional analógica, ya que integran además de un sensor con su respectivo principio de funcionamiento, el propio acondicionamiento de señal que este requiere, que puede ser por ejemplo alguna red eléctrica de conversión de la señal analógica del sensor, en una señal de tiempo, frecuencia, digital y también incluir un elemento de procesamiento como un contador o un microcontrolador, de esta manera el sensor inteligente puede acoplarse a un bus digital, permitiendo una mejor y más rápida transferencia de información. [6]

Además de los sensores analógicos, se analizó el sensor con número de parte PZEM-004T el cual es un sensor multifunción que permite medir el voltaje *RMS* (*Root Mean Square*), corriente *RMS*, potencia activa instantánea y energía que toma una carga conectada a una línea monofásica. Cuenta además con salidas optoacopladas y almacenamiento de valores cuando se presenta una desconexión de la energía eléctrica. La ventaja de este sensor es que su salida cuenta con protocolo de comunicación serial, lo que facilita su conexión con el microcontrolador. En la figura 2.3 se muestra una imagen del sensor digital analizador de energía.



Figura 2.3. Sensor con salida a protocolo de comunicación digital PZEM-004T. [7]

Para la medición de flujo de agua se analizó el sensor YF-S201, el cual es un caudalímetro cuya salida es una señal de pulsos con un voltaje de 5 VCD (Volts de Corriente Directa), la frecuencia de esta señal varía en función del flujo. Para realizar la medición es necesario únicamente una entrada digital del sistema de medición, y haciendo un conteo del estado alto de esa entrada se puede calcular el flujo. En la figura 2.4 se observa el caudalímetro que se utilizará para el proyecto de tesis.



Figura 2.4. Sensor de flujo de agua YF-S201. [8]

Este sensor didáctico demuestra que es posible realizar la medición del volumen de un flujo de agua y poder comunicar este dato a la aplicación móvil sin la necesidad de utilizar un sensor de grado industrial.

Para la medición de presencia de gas LP, se utiliza un módulo con sensor electroquímico MQ-2, que cuenta con una resistencia variable cuando se expone a determinados gases, y con esto el sensor reacciona con los gases provocando un cambio en el valor de la resistencia.

Estos módulos tienen una salida analógica y una salida digital, la cual internamente trabaja con un comparador y con la ayuda de un potenciómetro se puede

calibrar el umbral para poder interpretar la salida digital como presencia o ausencia del gas.

La diferencia entre los distintos tipos de sensores MQ es la sensibilidad a cierta gama de gases, más sensibles a algunos gases que a otros, pero siempre detectan a más de un gas, por lo que es necesario revisar las hojas de datos para escoger el sensor adecuado para la aplicación. En este caso se plantea utilizar el sensor MQ2 pues es adecuado para detectar GLP, propano, metano, alcohol, hidrógeno, humo. Siendo más sensible al GLP y propano.

En la figura 2.5 se observa el sensor de presencia de gas, que será utilizado para detectar fugas de gas LP.



Figura 2.5. Sensor de presencia de gas MQ-2. [9]

Para la medición de la temperatura se analizó el sensor DS18B20, un termómetro digital que provee medición de temperatura en grados Celsius con una resolución programable de 9 a 12 bits. Este sensor se comunica por el protocolo 1-Wire, gracias a esto se requiere de únicamente un pin de entrada digital en el microcontrolador para poder tomar las lecturas. Su rango de medición es de -55°C hasta los 125°C con un error de $\pm 0.5^{\circ}\text{C}$. En la figura 2.6 se muestra el sensor de temperatura digital.



Figura 2.6. Sensor de temperatura digital DS18B20 con protocolo de comunicación 1-wire. [10]

Para el sensor de iluminación se considera el sensor digital BH1750, se trata de un sensor de luz, que a diferencia del *LDR (Light Dependent Resistor)* convencional,

este es digital y entrega valores de medición en Lux (lumen /m²) que es una unidad de medida estándar para el nivel de iluminación. Tiene alta precisión y un rango configurable entre 1 – 65535 lx. La interfaz de comunicación es I²C (*Inter-Integrated Circuit*), pudiéndolo implementar en la mayoría de microcontroladores, el módulo aparte de los pines de alimentación y pines I²C tiene un pin para establecer la dirección. En la figura 2.7 se muestra el sensor de iluminación digital BH1750, con protocolo I²C.



Figura 2.7. Sensor de iluminación digital BH1750. [11]

2.1.3. Aplicaciones móviles

Las aplicaciones móviles han tenido un gran incremento en popularidad en los últimos años, y el interés sigue creciendo entre los usuarios. Los sistemas operativos móviles están disponibles tanto en teléfonos inteligentes como en Tablet, incrementando el mercado existente para aplicaciones.

Android posee características valiosas para los desarrolladores, siendo esta una plataforma de código abierto con una gran comunidad de desarrolladores. Android siempre está compitiendo con iOS® el sistema operativo móvil de la empresa Apple.

Google es el creador de Android Studio, un entorno de desarrollo integrado gratuito, el cual puede ser utilizado por cualquier persona para el desarrollo de aplicaciones móviles, el único requisito de este software es tener instalado Java ya que el lenguaje de programación Android está basado en el. [12]

2.1.4. Entorno de desarrollo integrado

Los lenguajes de programación de alto nivel usualmente se escriben como texto ASCII y dependiendo del lenguaje en el que se esté realizando la programación será la interpretación que la computadora o el hardware le dé al código.

Existen varios pasos para llevar un programa desde su etapa de código fuente hasta la ejecución del mismo. Históricamente, se utilizaban distintos programas de software tales como un editor de texto, un enlazador y hasta comandos del sistema operativo para hacer una conversión del código y lograr ejecutar el programa. Sin embargo, en la actualidad, todas estas acciones han sido integradas en un único programa llamado compilador, el cual es un conjunto de diferentes programas de software que crean un entorno utilizado por programadores para desarrollar software. De ahí viene el nombre Entorno de Desarrollo Integrado. [13]

2.1.5. Android Studio

Android Studio fue creado en 2013, y a pesar de que se encontraba en fase de prueba, estaba más que claro que los desarrolladores buscaban crear el entorno de desarrollo integrado oficial para desarrollo de aplicaciones Android. El software de Android Studio está basado en el entorno de desarrollo integrado de JetBrains' IntelliJ, el cual es un entorno de desarrollo integrado basado en el lenguaje Java, que además es gratuito. [14]

2.1.6. Microcontroladores

Los sistemas o plataformas digitales, están diseñados para almacenar, procesar y comunicar información de una forma digital. Se encuentran en un amplio rango de aplicaciones incluyendo control de procesos, sistemas de comunicación, instrumentos digitales entre otros productos comerciales. Una computadora manipula información de manera digital, precisamente en forma binaria. Cada valor discreto representa un estado de encendido o apagado en un interruptor electrónico llamado transistor. Si observáramos el diagrama de bloques de una computadora, se podría observar que cuenta con una unidad central de procesamiento, *CPU (Central Processing Unit)* por

sus siglas en inglés, una memoria y un bloque de entradas y salidas. Gracias al avance en tecnología de semiconductores es posible fabricar *CPU* en un solo chip, lo que resulta en un microprocesador.

Los microcontroladores evolucionaron de chips de microprocesadores, son normalmente utilizados en sistemas automotrices, electrodomésticos y sistemas de entretenimiento en el hogar. Los microcontroladores incluyen un *CPU*, unidades de memoria, bloques de entrada y salida, además incluye ciertas funciones específicas de cada microcontrolador como temporizadores, convertidores analógicos a digital, todo esto en un mismo chip. [15]

2.1.7. Chip integrado ESP8266

El ESP8266 es un chip integrado diseñado por Espressif Systems® debido a una creciente demanda en el mercado de soluciones *IoT* de bajo costo. Este chip ofrece una solución de red *Wi-Fi* completa y programable, que permite desarrollar aplicaciones o también ayudar a otra plataforma y compartir sus funciones de red.

Este chip fue originalmente vendido como un complemento de las placas Arduino, mismo que tenía una necesidad de ser conectado al mundo exterior, cuando la gente empezó a trabajar con el chip ESP8266 y Arduino, descubrieron que la placa Arduino puede ser eliminada de la arquitectura y puede ser utilizado como un dispositivo independiente.

Gracias a distintos proveedores se han desarrollado placas que permiten trabajar el chip de una manera sencilla, creando distintas placas de desarrollo para facilitar la conexión y programación del chip.

La principal plataforma digital a analizar para la propuesta de solución parte de una placa de desarrollo basada en el chip integrado ESP8266 que lleva por nombre NodeMCU. [16] En la figura 2.8 se puede observar el chip integrado ESP8266, mismo chip en el que está basado el microcontrolador NodeMCU.



Figura 1.8. Circuito integrado ESP8266. [17]

2.1.8. NodeMCU

Actualmente esta placa de desarrollo se basa en el chip ESP8266, es la más famosa en el mercado debido a la forma en la que se encuentran expuestos sus pines *GPIO* (*General Purpose Input Output*) es muy fácil montarla y adaptarla a la arquitectura de un proyecto *IoT*. Esta placa de desarrollo ahorra la necesidad de implementar un conversor *USB* (*Universal Serial Bus*) serial para conectarnos a él y programarlo con una computadora.

NodeMCU es capaz de conectarse vía Wi-Fi con un dispositivo, como por ejemplo una computadora, una Tablet o un teléfono móvil, para recibir o transmitir información a través de una variedad de protocolos de comunicación. Al mismo tiempo es capaz de interactuar con motores, leds, sensores de temperatura, detectores de humo y relevadores gracias a sus pines de uso general.

Para conocer un poco mejor este dispositivo vamos a describir algunas de las características físicas y de software que posee:

- Procesador ESP8266, de 32 bits a 80 MHz o incluso 160 MHz.
- Voltaje de operación de 3.3 V.
- Intensidad de consumo 50 mA aproximadamente, aunque tiene picos de más de 300 mA.
- Protocolo 802.11 b/g/n.
- Interface WIFI a 2.4 Ghz.

En la figura 2.9 se adjunta una ilustración del microcontrolador NodeMCU, la plataforma digital seleccionada para este proyecto de tesis.

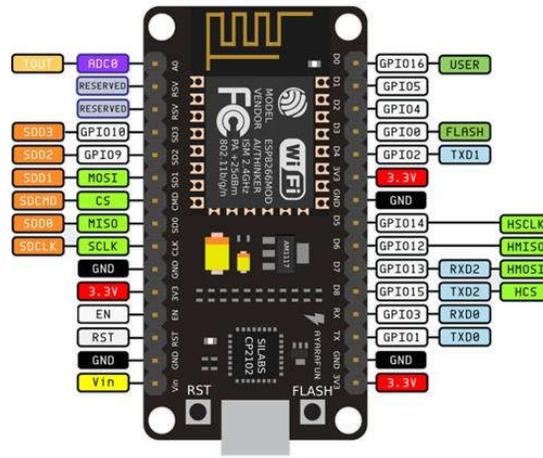


Figura 2.9. NodeMCU, placa de desarrollo basada en el IC ESP8266. [18]

2.1.9. Internet de las cosas

Una red de “cosas” potenciadas por sensores, identificadores, inteligencia de software y conectividad a Internet se llama Internet de las cosas (*IoT*). Por lo tanto, se basa en distintas tecnologías previas, de las cuales se encuentran sistemas de información generalizados, redes de sensores y computación embebida.

El internet de las cosas puede ser considerado como una intersección entre internet, objetos y datos. Estos objetos puede ser cualquier cosa como, por ejemplo, carros, personas, animales o plantas.

El principal objetivo del internet de las cosas es monitorear cosas u objetos desde cualquier parte del mundo. Es importante recalcar, que el monitoreo puede ser realizado por cualquier persona o por cualquier máquina, por ejemplo, el dueño de un hogar puede monitorear su casa a través de su teléfono móvil. [19]

2.1.10. Arquitectura de tres capas de una red *IoT*

La arquitectura del internet de las cosas comprende una colección de objetos físicos, sensores, servicios en la nube, desarrolladores, actuadores, capas de comunicación,

usuarios, capas de negocio y protocolos *IoT*. Diferentes arquitecturas fueron propuestas por diferentes investigadores, llegando a un acuerdo en que la arquitectura de esta red se divide en 3 capas, las cuales son:

- Capa de percepción: en esta capa se encuadran los diferentes sensores encargados de recoger información del entorno. Aquí se «sienten» parámetros físicos, o bien se identifican otros dispositivos inteligentes del entorno.
- Capa de red: se encarga de conectar el dispositivo a otras «cosas inteligentes», o bien a dispositivos de red o servidores. Dispone de las herramientas necesarias para transmitir datos entre dispositivos (o servidores y dispositivos de red), y también para realizar cierto grado de procesamiento de los mismos.
- Capa de aplicación: es en la que se enmarcan las aplicaciones del usuario. Desde las aplicaciones domésticas sobre uso de recursos (agua, gas, electricidad), hasta las aplicaciones logísticas para las empresas, que optimicen los recursos y el tiempo de procesado.

En la figura 2.10 se observa la arquitectura del proyecto basada en *IoT*.



Figura 2.10. Arquitectura de tres capas de una red *IoT*.

2.1.11. Protocolo de comunicación MQTT

Message Query Telemetry Transport por sus siglas en inglés *MQTT*, es un protocolo de mensajería asíncrona que disocia al emisor y al receptor en el tiempo, utiliza un modelo de publicación y suscripción. Este protocolo es flexible y compacto, lo que hace que pueda implementarse en dispositivos con recursos limitados y se adapte a situaciones con diferente demanda de recursos. La comunicación de los datos funciona gracias a un intermediario de mensajes entre los distintos clientes, que puede ser cualquier dispositivo con capacidades *Wi-Fi*. El cliente publica un mensaje en un tema que se envía al intermediario, el cual se encarga de hacer llegar el mensaje al cliente o clientes que estén suscritos a ese tema. Puesto que los mensajes están organizados por temas proporciona una jerarquía a la estructura del sistema. Este protocolo encaja perfectamente en una arquitectura *IoT*, puesto que la mayoría de los objetos que se dotan de conexión a internet poseen recursos limitados. [20]. Se observa en la figura 2.11 la arquitectura del protocolo de comunicación *MQTT*.

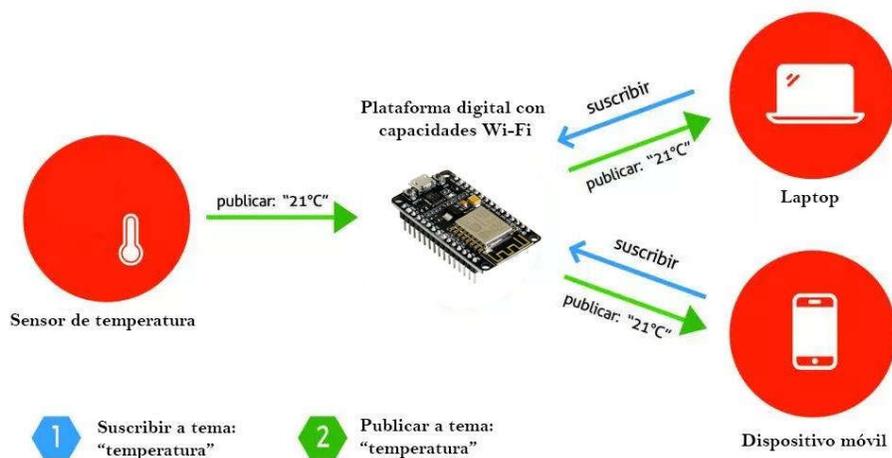


Figura 2.11. Ejemplo de arquitectura de protocolo MQTT. [21]

2.1.12. Backend como un servicio

Muchas empresas se dedican a desarrollar este tipo de servicios, el cual busca proveer herramientas a los desarrolladores de aplicaciones web y móviles con el fin de hacer más fácil el desarrollo de entornos gráficos. Entre los servicios que ofrecen estas

plataformas esta la conexión de aplicaciones a bases de datos en servidores, a la vez que proporciona características comunes como la administración de usuarios, notificaciones, integración de redes sociales entre otras características que los usuarios de aplicaciones móviles exigen en estos días.

Dentro de las ventajas que encontramos en los servicios *BaaS (Backend as a Service)* se destacan las siguientes:

- Facilita el desarrollo de aplicaciones. Gracias a los diferentes servicios que pueden ofrecer tales como análisis de datos, base de datos en tiempo real, entre otros.
- Infraestructura segura y escalable. Los proveedores de estos servicios proporcionan infraestructura escalable, por lo que en caso de necesitarlo se puede ampliar el espacio de almacenamiento para la base de datos y ampliar distintos parámetros para mejorar el servicio y mantener la calidad de las aplicaciones. [22]

2.1.13. Firebase

Firebase es uno de los *BaaS* más populares que existe desde el año 2012. Gracias a su capacidad y tecnología, tal como su base de datos en tiempo real, llamo la atención y el interés de Google por lo que fue adquirida por Google en el año 2014. Desde entonces se transformó en una poderosa herramienta, que contiene soporte para la mayoría de los casos que se presentan en el desarrollo de software en el día a día.

Firebase es el único capaz de proveer con una base de datos capaz de sincronizarse automáticamente lo que permite hacer crecer grandes aplicaciones. Cada uno de los distintos servicios que Firebase ofrece puede trabajar por separado, pero trabajan mejor juntos.

2.1.14. Base de datos en tiempo real de Firebase

La base de datos en tiempo real de Firebase permite crear aplicaciones que permitan una conexión segura desde el lado del cliente hasta el transporte de los datos. Los datos continuaran almacenados en caché localmente en el servidor, en caso de que

un dispositivo se desconecte de la base de datos, los cambios en los datos se sincronizaran una vez que se haya conectado a internet.

Además, esta base de datos ofrece un lenguaje de reglas adaptable basado en expresiones, llamado *Firestore Realtime Database Security Rules*, para establecer cómo deben permanecer organizados y cuando se pueden sobrescribir los datos.

Esta base de datos en tiempo real de Firestore soporta dispositivos Android, iOS, aplicaciones web y más. Todos los datos manejados en esta base de datos están en formato *JSON (JavaScript Object Notation)*, y cualquier cambio es reflejado inmediatamente pues se realiza una sincronización en todas las plataformas conectadas. Esto ayuda a desarrollar aplicaciones que requieran sincronizar datos en tiempo real sin mayor esfuerzo, aun si el usuario final esta desconectado de internet los datos son almacenados localmente en el servidor en un cache, y una vez que se detecta que el dispositivo se ha conectado a la base de datos y se le transfiere los datos almacenados hacia el usuario. [23]

2.1.15. Medidores inteligentes

Los medidores inteligentes según la Comisión Federal Reguladora de Energía de los Estados Unidos se definen como sistemas medidores de distintos parámetros capaces de registrar periódicamente el consumo del usuario final y proveer esta información a una red externa para el almacenamiento colectivo de los datos. [24] En la figura 2.11 se observa un diagrama de cómo se transmite la energía eléctrica a la residencia.

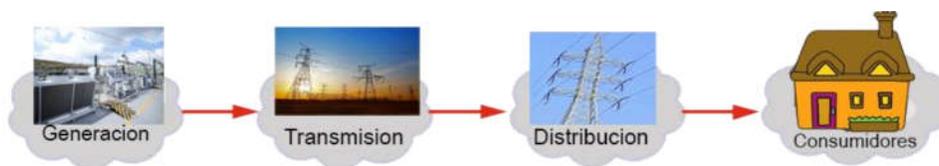


Figura 2.11. Ejemplo de una instalación eléctrica tradicional.

Con respecto a la distribución de esta energía, estas empresas centraron su atención e inversión en infraestructura con mejor tecnología, es así como surgieron los *Automated Meter Reading* o *AMR* por sus siglas en inglés, lo que les permitía leer el consumo y el estado de la instalación de sus clientes. Los medidores *AMR* tenían

comunicación limitada pues era únicamente de una vía, solamente se podían leer los datos del medidor, sin embargo, por la falta de una vía de comunicación se vuelve imposible crear aplicaciones adicionales, es aquí donde nace la necesidad de seguir invirtiendo y mejorando la infraestructura de comunicación lo que pronto se conocería como *Smart Metering* o *Advanced Metering Infrastructure*. La arquitectura utilizada en los medidores inteligentes provee una comunicación bidireccional con el medidor, y lo vuelve capaz de evaluar el estado actual en el que se encuentra la instalación eléctrica. Modelos recientes de medidores inteligentes, equipados con mejor infraestructura, trabajando además con sensores inteligentes permite llevar un paso más a las instalaciones eléctricas, permitiendo la gestión y el control remoto de las mismas. [25]

Los medidores inteligentes presentan un amplio rango de características, y pese a que no hay una norma o directiva que los defina en términos de cantidad o funcionalidad, se ha establecido que un medidor inteligente posee estas tres características:

- Lectura remota de parámetros.
- Comunicación bidireccional.
- Soporte de sistema avanzado de tarifas y facturación.

2.2. ESTADO DEL ARTE

[26] presenta el artículo "*Electronic meter reader & data base management system*", el objetivo de este documento es presentar el desarrollo de un sistema para transmitir las mediciones de un dispositivo capaz de capturar el consumo eléctrico en kWh a una estación remota para facturación y control, el medidor de energía era analógico y se digitalizó a través de circuitos electrónicos. Los datos eran recabados utilizando un microcontrolador AT89C51, posteriormente estos eran transmitidos a un servidor y a una pantalla LCD donde el usuario pueda visualizar.

[27] presenta su tesis con el tema "*Diseño de un medidor inteligente e implementación de sistema de comunicación bidireccional*", el objetivo principal en este trabajo fue desarrollar un sistema de medición que pudiera ser supervisado y controlado a distancia, para ello se realizaron diferentes actividades orientadas al

desarrollo del mismo, tales como: la simulación de algoritmos de medición, la programación de estos algoritmos en lenguaje C, pruebas a diferentes circuitos de adecuación de voltaje y corriente; así como dos métodos distintos para digitalización de señales analógicas, se analizaron diferentes opciones y fabricantes para la selección del sistema de comunicación, se aplicaron métodos estadísticos para el ajuste y escalamiento de mediciones eléctricas.

[28] presenta el artículo *“Monitoreo de variables analógicas usando Raspberry Pi® en el cual se utiliza una computadora de circuito único Raspberry Pi”*, haciendo uso de un convertidor analógico digital se obtienen datos de distintas señales analógicas, para después ser procesadas utilizando un lenguaje de programación llamado Python, el cual permite procesar los datos recabados, almacenarlos en base de datos y por último, proyectarlos en una interfaz gráfica.

[29] presenta el artículo *“Automated electric meter reading and monitoring system using ZigBee-integrated raspberry Pi single board computer via Modbus”*, en esta lectura se resalta la creciente necesidad de la electricidad, además se da a conocer el concepto de "medidores automáticos inteligentes", los cuales son dispositivos que se encargan de realizar lecturas de los parámetros eléctricos y procesan la información que recolectan los medidores automáticos inteligentes, de igual forma aplica para el consumo de agua. El sistema utiliza tecnología ZigBee® y está integrado con una computadora de circuito único Raspberry Pi.

[30] presenta el artículo *“An agricultural telemetry system implemented using an Arduino-Android interface”*, en el cual se propuso la arquitectura Arduino-Android para un sistema de telemetría inalámbrico donde el microcontrolador Arduino® se encarga de recolectar datos de diferentes sensores para parámetros tales como la temperatura, humedad, humedad del suelo y estos datos eran enviados a través de wifi a un servidor.

[31] presenta su trabajo de tesis con el título *“Diseño de un medidor de consumo de energía eléctrica en redes industriales domésticas, basado en la adquisición y digitalización de señales para el monitoreo y análisis de la calidad de energía”*, donde se realiza el modelado para un sistema de medición de consumo de potencia en una

red eléctrica monofásica, capaz de analizar y alertar las principales fallas eléctricas y registrar el consumo de energía a partir de un sistema de censado de señal de corriente y voltaje.

[32] presenta su *trabajo “Diseño y construcción de un equipo de medida de energía eléctrica de bajo coste”*, este proyecto consta de un medidor de parámetros eléctricos hecho con un microcontrolador de Arduino, con sensores de voltaje y corriente y algoritmos de programación en lenguaje lua.

[33] presenta el artículo *“Diseño y construcción de un analizador de redes eléctricas 1 para sistemas trifásicos de bajo voltaje”*, el cual habla de la importancia de la calidad de la red eléctrica en las industrias para evitar sobrecalentamientos en conductores entre otros factores que afectan a los dispositivos eléctricos / electrónicos, en este trabajo se utilizó hardware para digitalizar y medir señales de corriente y voltaje de un sistema trifásico, se utilizaron convertidores *ADC (Analog-to-Digital Convertor)* para la digitalización de estas señales.

[34] presenta el artículo *“IoT based data processing for automated industrial meter reader using Raspberry Pi”*, en el cual se utiliza un computador de circuito único Raspberry Pi encargado de digitalizar imágenes, utilizando una cámara, se fotografía el sensor y obtiene la lectura de la medida, después estos datos son almacenados en la nube.

[35] presenta su trabajo *“Diseño e implementación de un sistema de medición de consumo de energía eléctrica y agua potable remoto con interacción al usuario basado en el concepto internet de las cosas”*, este trabajo desarrollado ha logrado el diseño e implementación de un sistema de medición de consumo de energía eléctrica y agua potable remoto con interacción al usuario basado en el concepto "Internet de las cosas". Se han establecido los métodos de medición de energía eléctrica y agua potable para la lectura de los datos que son capturados y enviados por el dispositivo de comunicación ESP8266 que ha sido seleccionado. Igualmente se ha diseñado e implementado una plataforma de software de comunicación y almacenamiento de datos usando tecnologías afines a las necesidades del Internet de las cosas como Websocket y Node.js®, que aprovechan una comunicación bidireccional, full dúplex.

[36] presenta el artículo "*Medidor electrónico interactivo de consumo de energía eléctrica para uso residencial*", en el cual se presenta el diseño y la implementación de un medidor electrónico interactivo para uso residencial que permite monitorear el consumo de energía eléctrica en tiempo real, posibilitando la visualización del consumo actual o del histórico en una pantalla de cristal líquido, o a través de una interfaz gráfica en un computador; de tal forma que el usuario final cuente con información suficiente para tomar decisiones que propicien el uso racional de la energía eléctrica. El uso del circuito integrado ADE7753 como dispositivo central para la medición de energía eléctrica ofrece economía y versatilidad en el equipo desarrollado, ya que este dispone de todo el hardware necesario para el tratamiento de las señales de corriente y de voltaje previamente acondicionadas, con lo que se consigue obtener datos de manera rápida, precisa y en forma digital, lo que facilita de manera significativa la gestión de la información, además de tratarse de un dispositivo de tamaño reducido, y además, ofrece alta confiabilidad en las medidas tomadas al permitir la calibración mediante el uso de medidores de referencia.

[37] presenta el artículo "*Energy management including photovoltaic panel and energy storage for Smart Grids through mobile application*" en el cual se propone el uso de una aplicación móvil para Android con el fin de visualizar los datos obtenidos de la energía generada de forma solar y el consumo eléctrico de una instalación, para así hacer al cliente más consciente del uso que le da a la energía. Se concluyó que es viable usar una aplicación móvil para el proyecto, esta aplicación se encargaba de brindar al usuario final varios métodos para reducir el consumo eléctrico, ya sea por el uso consciente de la electricidad y también mediante el control del sistema de almacenamiento de energía en horas pico.

[38] presenta el artículo "*Control strategy for small-scale photovoltaic systems connected to single-phase grids*", en esta lectura, se estudia el control de un inversor monofásico en su profundidad. La estrategia de control utilizada se basa en el control orientado a voltaje para controlar la potencia activa y pasiva. Siguiendo lo anterior da como resultado el control de un inversor conectado a la red monofásico.

[39] presenta el artículo *“Raspberry Pi: Data logging IoT device”*, en el cual se analiza un motor, gracias a una red de sensores interconectados a un computador Raspberry Pi, estos datos son almacenados en la nube donde el usuario final puede ingresar con sus credenciales para mayor seguridad y consultar las lecturas de los sensores como información ya procesada.

[40] presenta el artículo *“Diseño de un medidor de energía eléctrica bidireccional y monitoreado mediante aplicación móvil”*, el cual habla del diseño e implementación de un medidor de energía eléctrica con comunicación bidireccional, enlazado a una aplicación móvil. Gracias a este medidor inteligente se obtuvieron mediciones y se generaron informes por parte del usuario final, además de poder administrar vía servidor remoto cada uno de los medidores. Se espera que con la implementación del medidor bidireccional propuesto y su monitoreo mediante la aplicación móvil desarrollada, los usuarios finales puedan llevar un mayor control sobre sus consumos de energía en tiempo real desde cualquier lugar en donde se encuentren.

[41] *presentan el artículo “IoT Enabled Home With Smart Security”*, en esta lectura se implementó un sistema basado en *IoT* encargado de leer distintos parámetros tales como temperatura con un sensor LM35, se utilizó un foto resistor para detección de luz, sensor infrarrojo pasivo, interruptores magnéticos y sensores de humo, se utilizó un servidor web *IoT* llamado *Thingspeak* utilizando el protocolo MQTT®, donde se podían visualizar los datos obtenidos.

[42] presenta el artículo *“Sistema de control y telemetría de datos mediante una aplicación móvil en Android basado en IoT para el monitoreo de datos”*, en este proyecto se muestra el desarrollo de una aplicación móvil implementada en *App Inventor* para el monitoreo y control de sistemas eléctricos de mediana y baja potencia mediante el principio del internet de las cosas *IoT*, el hardware se basa en una tarjeta WeMos D1 mini, la cual posee un microcontrolador ESP8266 y tiene la tarea de medir, procesar, controlar y enviar las variables de medición a la nube mediante la plataforma *IoT* thinger.io, el resultado obtenido es una herramienta de hardware y software de bajo costo (menor a 20 dólares) para controlar equipos electrónicos a través de internet, brindando oportunidad de mejora y de bajo costo en la transferencia

tecnológica a pequeñas y medianas empresas interesadas en aplicar este tipo de tecnología a sus procesos.

2.2.1. Alternativas reportadas

Durante la investigación en la literatura se encontró que los prototipos de medidores inteligentes no involucran mediciones relacionadas con el consumo de gas LP ya que representa un importante costo a considerar al cotizar los sistemas, Raspberry Pi y Arduino son las plataformas más utilizadas por su bajo costo y su basta capacidad de programación. Los medidores basados en FPGA® no se adaptan a la arquitectura *IoT* de este proyecto por lo que se descartó utilizar ese tipo de microcontroladores

En la oferta comercial, existen alternativas para este propósito y se denominan medidores inteligentes, la empresa EKM Metering® ofrece un medidor inteligente capaz de medir parámetros eléctricos para consumo de energía eléctrica, consumo de agua y consumo de gas, el costo de los dispositivos actualmente es superior a los \$600 USD, sin incluir el precio del software requerido para visualizar los datos.

En países como Estados Unidos y Reino Unido existen empresas tales como: NPower, British Gas, Scottish Power, Shell energy, las cuales se dedican a la venta, instalación y soporte para medidores inteligentes, estas empresas se encargan de revisar la instalación eléctrica en el hogar, así como instalaciones de gas o agua para adaptar su sistema a las necesidades del usuario final y poder seleccionar los mejores componentes requeridos.

CAPÍTULO 3

DESARROLLO

3.1. SOLUCIÓN PROPUESTA

El Internet de las cosas ha marcado la pauta de una revolución tecnológica sin precedentes, desde su inicio se ha buscado favorecer al ser humano con la inmensa variedad de aplicaciones que se pueden realizar con dispositivos actuales. Cada vez se pueden conectar más dispositivos a Internet por una buena razón, los datos son valiosos, un dispositivo puede proporcionar datos útiles sobre su estado o procesos en el entorno. También pueden recibir y transmitir información a Internet para que puedan ser monitoreados en tiempo real.

Y es aquí donde se crea una oportunidad para una aplicación de Internet de las cosas en el hogar, ya que el usuario promedio que paga servicios básicos tales como energía eléctrica y agua potable, desconoce el consumo que realiza a lo largo del periodo de facturación, si bien se tienen medidores análogos electromecánicos que indican el consumo del periodo, la interpretación de estos datos puede escapar al entendimiento del usuario al no estar familiarizado con la interpretación de las unidades de medición. Atendiendo la problemática, se implementó un sistema electrónico capaz de tomar mediciones de estos servicios y como complemento, tomar lectura de distintos parámetros de interés en el hogar tales como la iluminación, temperatura y presencia de gas LP, contando con una interfaz gráfica programada en Android Studio para la posterior visualización de los datos, donde el usuario final pueda consultar desde su Smartphone los datos procesados para una fácil interpretación de los mismos, todo esto en tiempo real y de manera bidireccional, gracias a la rapidez de los protocolos de comunicación del Internet de las cosas.

Partiendo de la arquitectura de tres capas de una red *IoT*, para la primera capa se define una red de sensores, esto implica realizar una búsqueda para la selección de los sensores necesarios para la instrumentación requerida de acuerdo con las mediciones de los distintos parámetros, tomando en consideración las características

de estos sensores, se seleccionan los que cumplan con las especificaciones requeridas para que estos puedan comunicarse con la plataforma digital, encargada de recabar los distintos datos y digitalizarlos en un servidor de Internet, donde a su vez estos datos son mostrados en una aplicación móvil para el sistema operativo Android.

Esta red de sensores representa la capa de percepción en la arquitectura de red mencionada en el capítulo anterior, gracias a ella se obtienen las lecturas de los parámetros de interés a medir.

Se requiere una plataforma digital capaz de interpretar estos datos para posteriormente centralizarlos en un servidor en Internet, estos dos elementos forman parte de la capa de red para la arquitectura propuesta.

Por último, pero no menos importante, el propósito de estos datos es que se visualicen en una interfaz gráfica programada en una aplicación móvil para el sistema operativo Android, formando así la capa de aplicación. De esta manera se implementa una arquitectura sólida basada en el Internet de las cosas. En la figura 3.1 se muestra la arquitectura del proyecto basada en las tres capas del internet de las cosas.



Figura 3.1. Arquitectura propuesta para proyecto.

3.2 METODOLOGÍA A DESARROLLAR

La metodología a seguir para este proyecto de tesis se muestra en la figura 3.2.

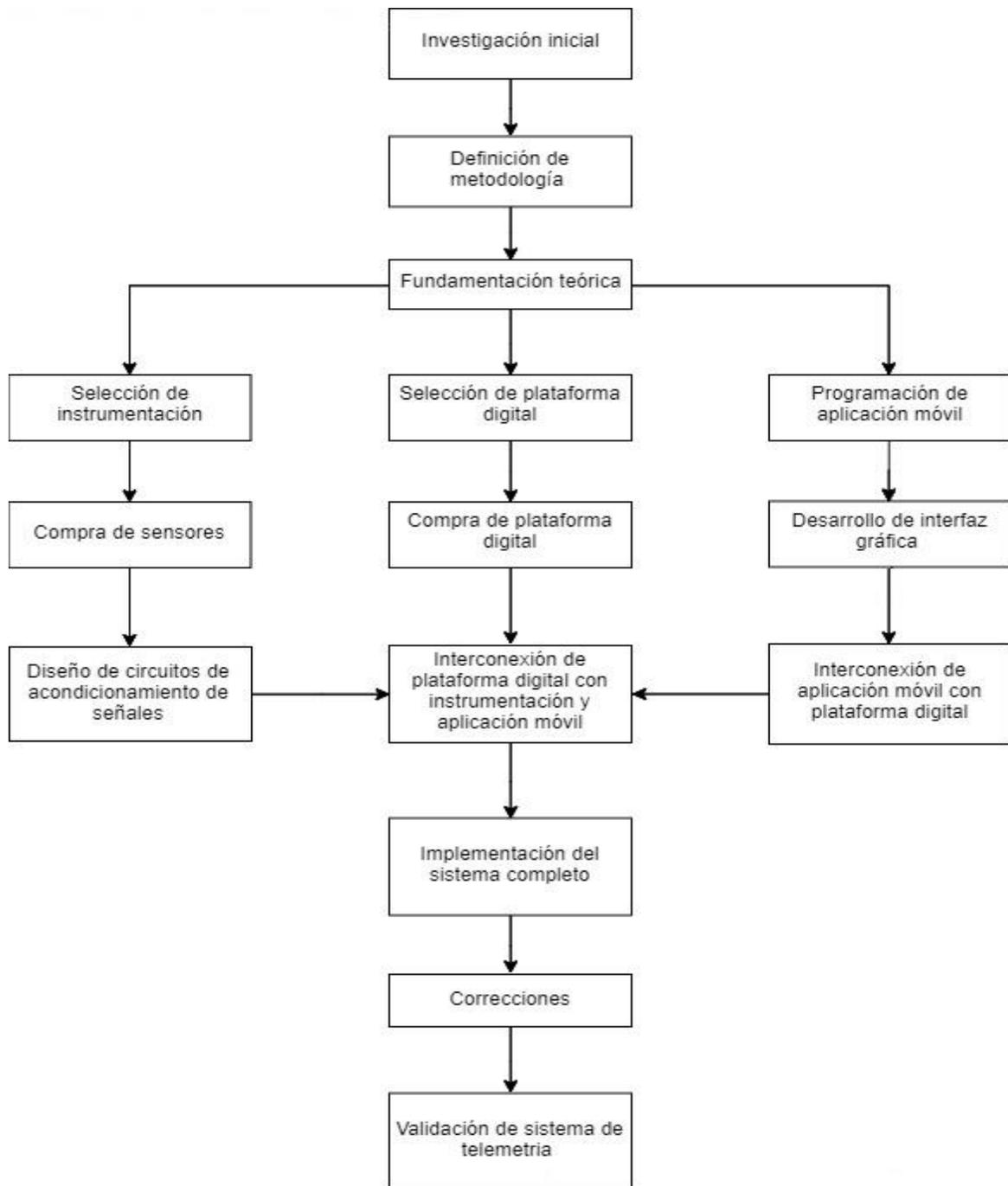


Figura 3.2. Metodología.

La metodología de la figura 3.2 fueron los cimientos para el desarrollo de este trabajo de tesis, el proyecto se dividió en tres etapas principales que son:

- Instrumentación: Análisis, la selección y compra de los sensores requeridos para el proyecto.
- Interconexión: Desarrollo de algoritmos de comunicación para la interconexión de la instrumentación, selección de plataforma digital para procesamiento de datos, renta de un servidor de Internet para el almacenamiento de datos y una aplicación móvil.
- Interfaz gráfica: Desarrollo de aplicación móvil para Android donde el usuario final consulte los datos recabados de los distintos sensores en tiempo real.

También es importante el estado del arte y marco teórico del tema de tesis, pues el fundamento teórico fue la base que sustente el prototipo que se desarrollará en este proyecto. Partiendo desde la selección de instrumentación, fue necesario seleccionar los sensores que se adaptasen mejor a la plataforma digital, se tiene que revisar diferentes principios de funcionamiento en los que se basan los sensores, seleccionar niveles de voltaje adecuados y el tipo de salida, con esa información se buscó una marca de sensores y posteriormente se realizó la compra. La selección de la plataforma digital se ligó directamente a la instrumentación, pues el número de entradas digitales que se necesitan impacta directamente en el costo y las características requeridas.

Para el desarrollo de la aplicación móvil se hizo uso del software Android Studio, ya que es el entorno de desarrollo de aplicaciones móviles para Android por defecto, esta aplicación móvil cuenta con todos los protocolos requeridos para lograr conectarse a una base de datos, en este caso se utilizó la base de datos en tiempo real de Firebase.

3.3 MODELADO, SIMULACIÓN Y EXPERIMENTACIÓN

Como parte de las actividades se realizaron distintas investigaciones para conocer más acerca de la instrumentación requerida, posteriormente se realizó una selección de los sensores necesarios para llevar a cabo este proyecto de tesis. Se requieren medir distintos parámetros, los cuales son:

- Corriente I_{RMS}
- Voltaje V_{RMS}
- Flujo de agua.
- Flujo de gas LP.
- Temperatura.
- Iluminación.

Para conocer el gasto de energía eléctrica es necesario tomar lecturas de la corriente instantánea y el voltaje instantáneo en un circuito eléctrico, ya que una vez se obtienen los dos parámetros se realizan cálculos para obtener los voltajes y las corrientes eficaces [43]. Para el prototipo se propone una instalación eléctrica residencial monofásica de corriente alterna.

La instrumentación requerida para conocer el gasto de energía eléctrica se puede clasificar en 2 tipos de sensores:

- Sensores analógicos cuya salida es una variación de voltaje, esto hace más fácil el desarrollo de un algoritmo de programación con un microcontrolador para obtener los datos de las lecturas
- Sensores basados en circuitos integrados dedicados a la medición de consumo eléctrico, con salida a protocolo de comunicación digital, el cual sería ideal para la arquitectura pues este tipo de sensores requieren únicamente entradas digitales para la transmisión de la información, además estos sensores no requieren circuito de acondicionamiento de señal, pues la información es procesada antes de ser enviada, lo que reduce el ruido por intervenciones externas en una lectura analógica.

Una vez presentados los sensores que se analizaron, se realizó una comparación entre características, precio, tiempo de entrega y otros factores para la selección de la instrumentación que se compró. En la tabla 1 se destacan los sensores seleccionados para la arquitectura de la red de sensores utilizada en este proyecto de tesis.

Tabla 1. Sensores seleccionados para el proyecto de tesis.

Descripción	Número de parte	Tiempo de entrega	Tipo de comunicación	Precio
Sensor analizador de energía digital	PZEM-004T	2 semanas	<i>UART</i>	\$1000.00
Sensor para flujo de agua con salida a pulsos	YF-S201	1 semana	Frecuencia digital	\$200.00
Sensor digital de presencia de gas	MQ-2	1 semana	Salida digital de 2 estados	\$50.00
Sensor de temperatura digital	DS18B20	2 semanas	Protocolo 1–wire	\$200.00
Sensor de iluminación con salida digital	BH1750	1 semana	<i>I²C</i>	\$100.00

Con esta selección de sensores, se requiere que la plataforma digital tenga por lo menos 9 pines de entrada digital, se propone que el microcontrolador para este proyecto de tesis sea la placa de desarrollo basada en el circuito integrado ESP8266, llamada NodeMCU, ya que la capacidad de procesamiento que provee esta placa es suficiente para lograr leer los datos de los sensores propuestos y además tiene un precio muy económico lo que ayudaría a reducir el impacto del costo del proyecto.

Hasta ahora se han definido dos de tres capas de la arquitectura *IoT*, se definió la capa de percepción, la cual será la red de sensores establecida, y la capa de red la cual será la placa de desarrollo NodeMCU. Ambas capas pueden interconectarse a través de cableado electrónico, lo que deja pendiente una capa por definir, llamada también capa de aplicación. Para esta capa se programa una interfaz gráfica en una aplicación móvil para sistema operativo Android, esta aplicación tiene los protocolos de comunicación requeridos para enlazar la plataforma digital con la aplicación, y a

través de estos mismos protocolos enviar los datos para ser proyectados en la aplicación.

Ahora que se han definido en su totalidad las capas de la arquitectura, se crea un detalle de la arquitectura del proyecto, pues ya se han seleccionado anteriormente los sensores utilizados, la plataforma digital, los protocolos de comunicación y la interfaz gráfica. En la figura 3.3 se adjunta un diagrama unifilar de las conexiones de los sensores con la plataforma digital.

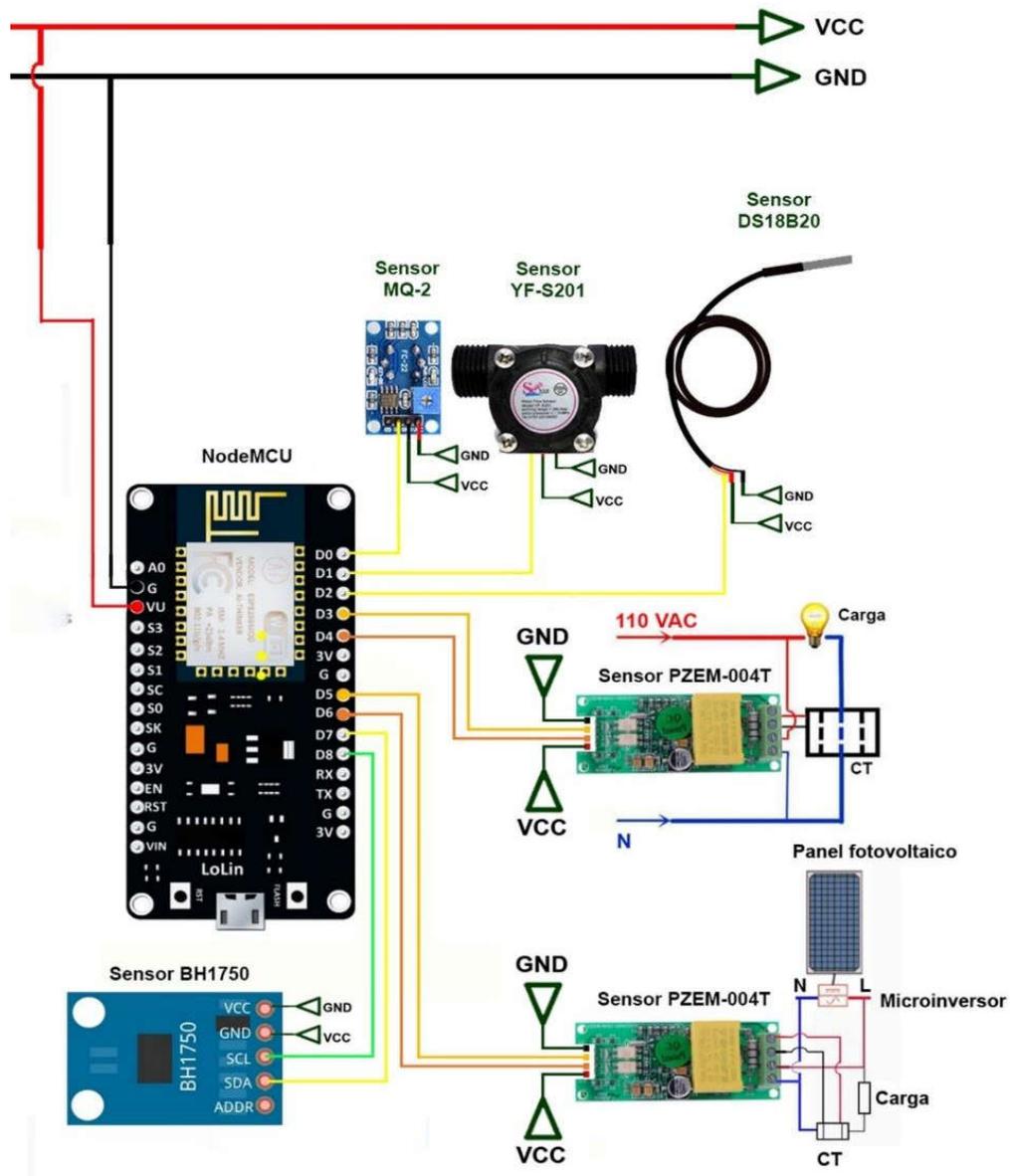


Figura 3.3. Diagrama unifilar de conexiones requeridas para el proyecto.

3.4 RESULTADOS OBTENIDOS

3.4.1. Prueba de protocolo MQTT con interfaz Node-RED®

Para el desarrollo de este prototipo se analizaron distintas tecnologías y protocolos de comunicación para seleccionar los óptimos, en total se probaron las siguientes tecnologías de comunicación:

- *Wi-Fi*
- Bluetooth
- *SPI (Serial Peripheral Interface)*
- *1-Wire*
- *I²C*
- *UART (Universal Asynchronous Receiver-Transmitter)*
- Websockets
- *MQTT*

Para la comunicación *Wi-Fi* se trabajó en el desarrollo de dos prototipos, el primero de ellos con el fin de probar el protocolo de comunicación *MQTT*, y el segundo para utilizar el protocolo *Websockets*.

Con el primer prototipo fue probado el sensor de temperatura DS18B20, utilizando el microcontrolador NodeMCU, se programó el microcontrolador para recibir el dato de la temperatura a través de un protocolo digital. Posteriormente esta información se comunicaba a una interfaz gráfica desarrollada en el entorno grafico *Node-RED*.

Para la conexión del circuito del prototipo con protocolo de comunicación *MQTT*, se requiere cablear el sensor de temperatura a la alimentación que brinda el microcontrolador NodeMCU. El cable de la señal se conectó al pin de entrada digital 2, empleando una resistencia de 4.7 kΩ En la figura 3.4 se observa el diagrama para la conexión del sensor de temperatura con el microcontrolador

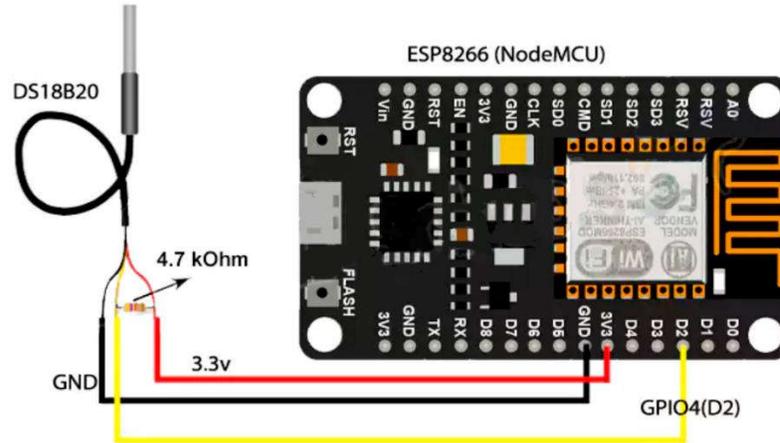


Figura 3.4. Diagrama de conexión de sensor a NodeMCU

La conexión del sensor de temperatura se muestra en la figura 3.12, este circuito fue conectado en un protoboard con la finalidad de probar el funcionamiento del sensor.

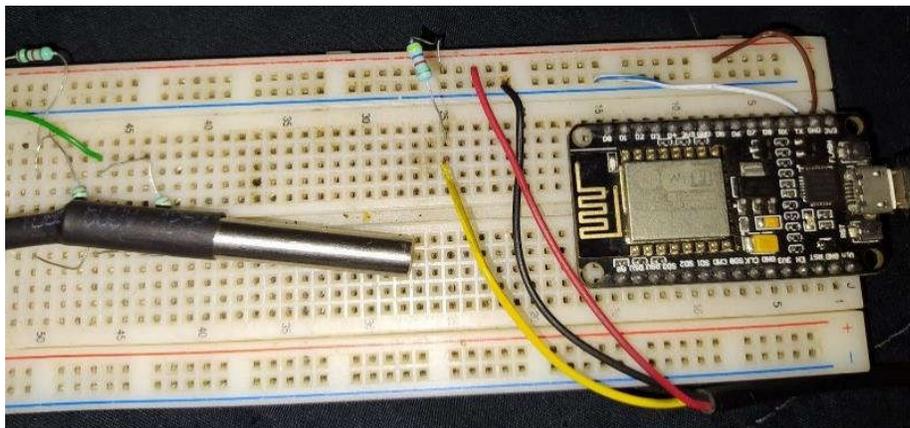
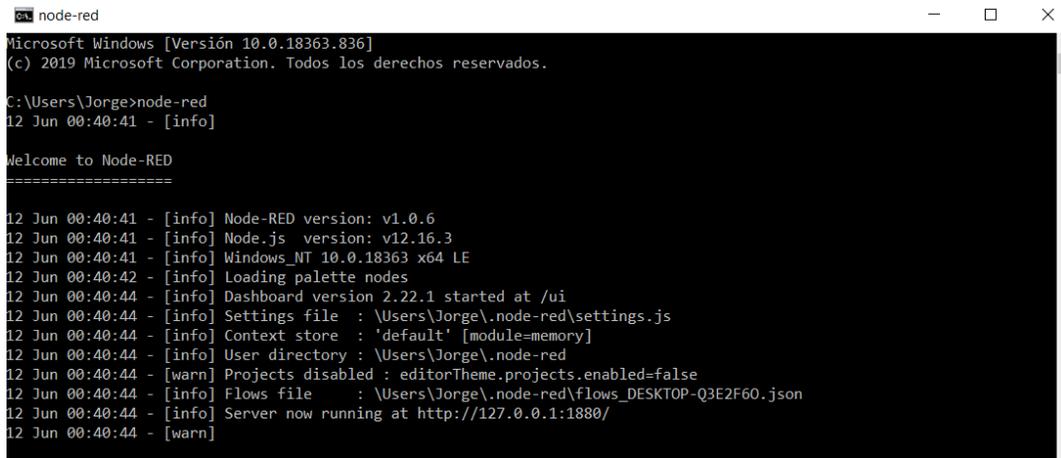


Figura 3.5. Sensor de temperatura conectado a NodeMCU.

Para esta prueba se conectó la plataforma NodeMCU con un servidor *MQTT* en línea llamado CloudMQTT. Este servidor en línea permite registrarse con una cuenta y de manera gratuita, utilizar sus servicios para tener un intermediario capaz de transmitir la información que se publique en los temas o tópicos a los dispositivos conectados. El microcontrolador NodeMCU publica en un tópico de este servidor con el nombre de “temp”, el valor de la lectura del sensor de temperatura en ese instante, el cliente en esta prueba es una interfaz gráfica desarrollada en *Node-RED*, un motor de flujos con enfoque *IoT* que permite definir gráficamente flujos de servicios a través

de protocolos estándares como *MQTT* y *Websocket*. Para este prototipo se programó una interfaz que cuenta con un indicador visual que permite al usuario ver el valor de la temperatura que esté midiendo el sensor. Con el programa de *Node-RED* instalado en el sistema operativo de Windows, se puede iniciar esta herramienta a través de la línea de comandos, introduciendo el comando *Node-RED*. En la figura 3.12 se observa la terminal de Windows, donde inicia el software *Node-RED*.



```
node-red
Microsoft Windows [Versión 10.0.18363.836]
(c) 2019 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Jorge>node-red
12 Jun 00:40:41 - [info]

Welcome to Node-RED
=====
12 Jun 00:40:41 - [info] Node-RED version: v1.0.6
12 Jun 00:40:41 - [info] Node.js version: v12.16.3
12 Jun 00:40:41 - [info] Windows_NT 10.0.18363 x64 LE
12 Jun 00:40:42 - [info] Loading palette nodes
12 Jun 00:40:44 - [info] Dashboard version 2.22.1 started at /ui
12 Jun 00:40:44 - [info] Settings file : \Users\Jorge\.node-red\settings.js
12 Jun 00:40:44 - [info] Context store : 'default' [module=memory]
12 Jun 00:40:44 - [info] User directory : \Users\Jorge\.node-red
12 Jun 00:40:44 - [warn] Projects disabled : editorTheme.projects.enabled=false
12 Jun 00:40:44 - [info] Flows file : \Users\Jorge\.node-red\flows_DESKTOP-Q3E2F60.json
12 Jun 00:40:44 - [info] Server now running at http://127.0.0.1:1880/
12 Jun 00:40:44 - [warn]
```

Figura 3.6. Línea de comandos de Windows, comando Node-RED.

Una vez iniciado el servicio en la computadora, se accede en el explorador la dirección en la que está corriendo el servidor de *Node-RED*: <http://127.0.0.1:1880/>. En la figura 3.13 se muestra la interfaz del software *Node-RED*.

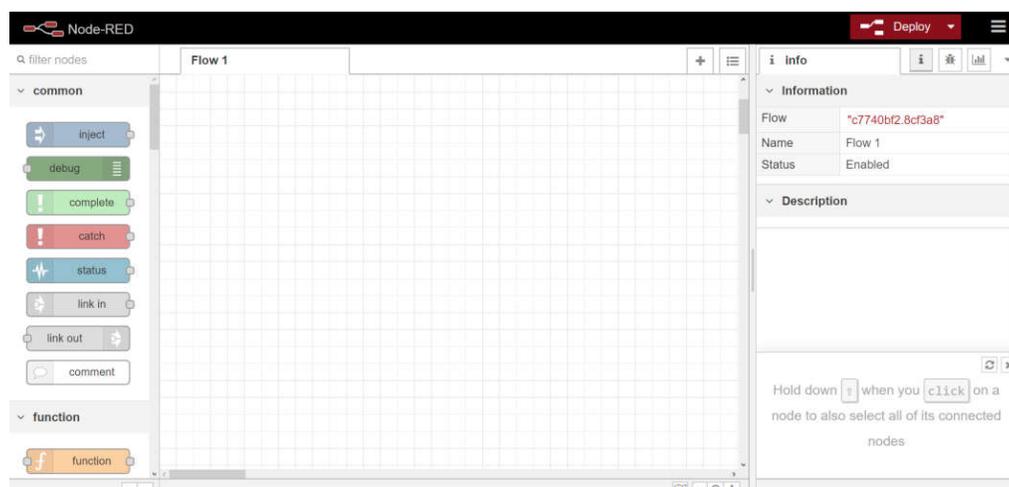


Figura 3.7. Entorno de desarrollo Node-RED.

Con el software ya instalado y ejecutado, se inicia con la programación de la interfaz gráfica. Primero se agrega el objeto *MQTT* in para suscribir un tópico en el servidor, por ejemplo, el tópico “temp”, configurado previamente en la plataforma NodeMCU, esto permite publicar cada dos segundos la temperatura medida, creando una conexión entre el microcontrolador y la interfaz gráfica con *Node-RED*. En la figura 3.8 se muestra el componente que permite utilizar el protocolo *MQTT* en el software *Node-RED*.

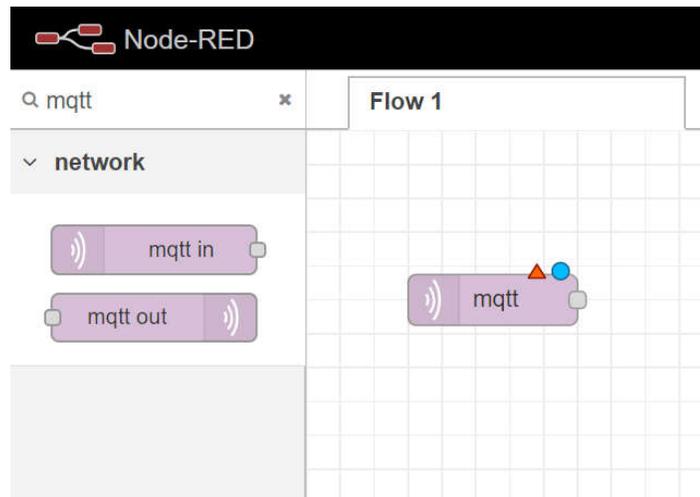


Figura 3.8. Componente MQTT in.

Una vez añadido el componente realizamos la configuración, haciendo doble clic en él para introducir los parámetros que se requieren para conectar la interfaz gráfica con el intermediario, primero se introducen las credenciales y la dirección del servidor CloudMQTT®. En la figura 3.9 se observa la pantalla de configuración de un objeto en el software Node-RED.

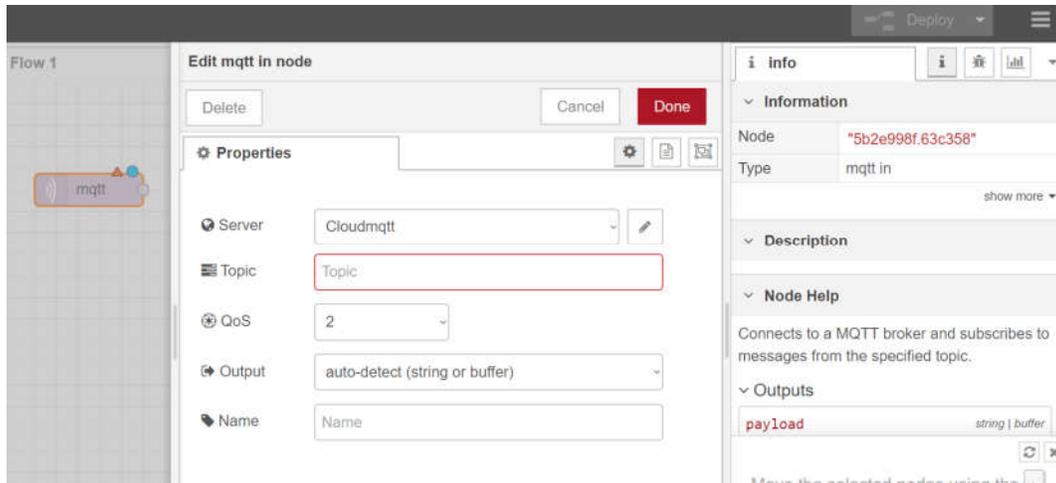


Figura 3.9. Configuración de objeto en Node-RED.

Al crear la cuenta en la página oficial de CloudMQTT se obtienen los datos que se deben introducir para que el componente de *Node-RED* pueda conectarse con el servidor. En la figura 3.10 se muestran los parámetros del servidor CloudMQTT.

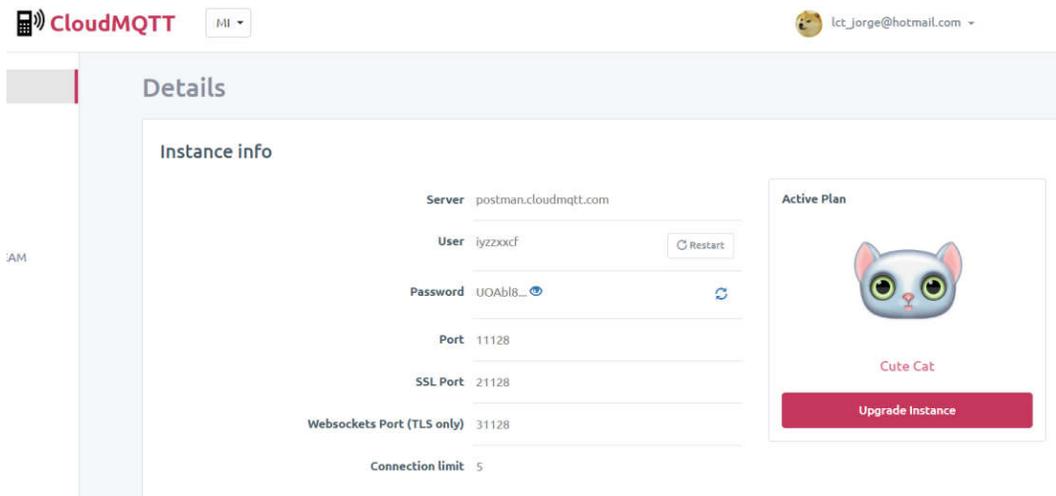


Figura 3.10. Parámetros de configuración de servidor CloudMQTT.

Para configurar el servidor es necesario llenar los campos en blanco de servidor y puerto, estos datos se obtienen en la página web de CloudMQTT y se observan en la figura 3.16.

En la figura 3.11 se muestra la pantalla de configuración para conectar un servidor con el software *Node-RED*.

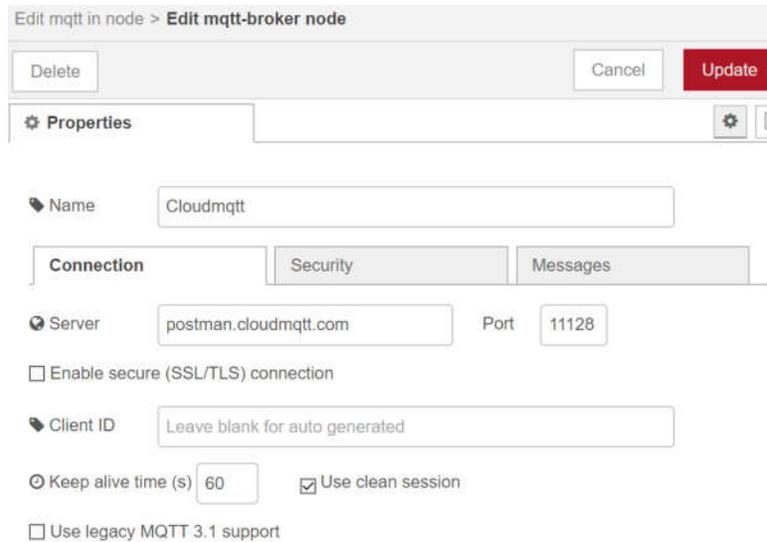


Figura 3.11. Configuración para conexión de componente con servidor CloudMQTT.

Ya que se han introducido los datos, es necesario hacer clic en el botón actualizar, el componente muestra el estatus indicando si pudo realizar la conexión al servidor de manera exitosa, para esta prueba la conexión fue exitosa por lo que se aprecia el estado conectado. En la figura 3.12 se observa que el objeto configurado ya está conectado al servidor.

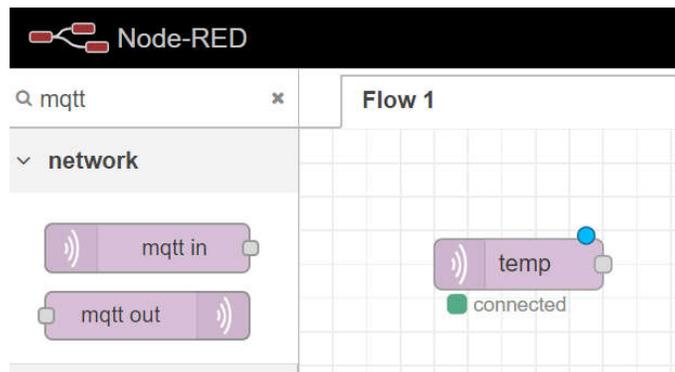


Figura 3.12. Componente configurado y conectado con servidor CloudMQTT.

Una vez se tiene el valor de la temperatura en la plataforma, es necesario añadir un elemento que permita visualizarlo, en este caso se utilizó un indicador. En la figura 3.13 se muestra el diagrama de conexiones del Sketch de *Node-RED*.

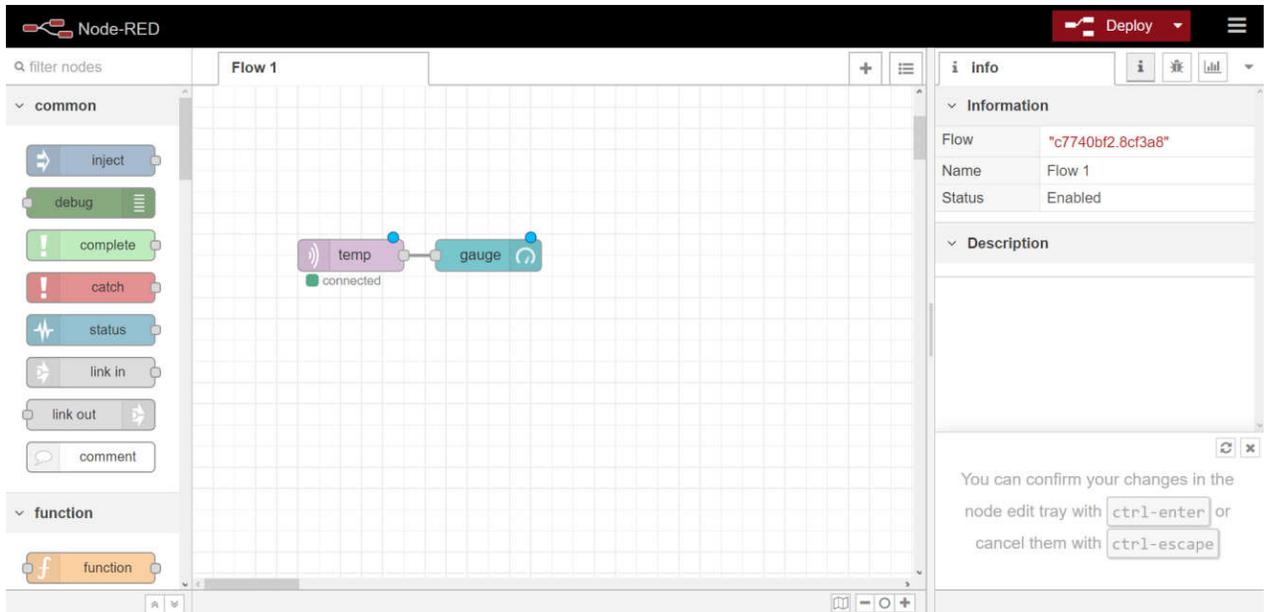


Figura 3.13. Diagrama de interfaz gráfica de Node-RED.

En los parámetros de configuración del indicador, se requiere ajustar los rangos de valores mínimo y máximo de la temperatura. En la figura 3.14 se observa la configuración de un objeto visual, mostrando la temperatura al igual que un termómetro analógico.

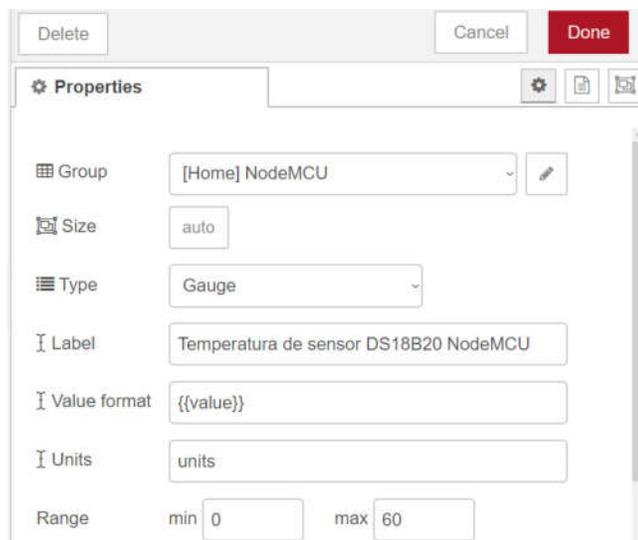


Figura 3.14. Configuración de indicador de temperatura.

Finalmente, se accedió a la dirección <http://127.0.0.1:1880/ui>, ya que es ahí donde se muestran los objetos visuales que fueron programados en el diagrama de flujo de *Node-RED*, como se puede observar en la figura 3.15.

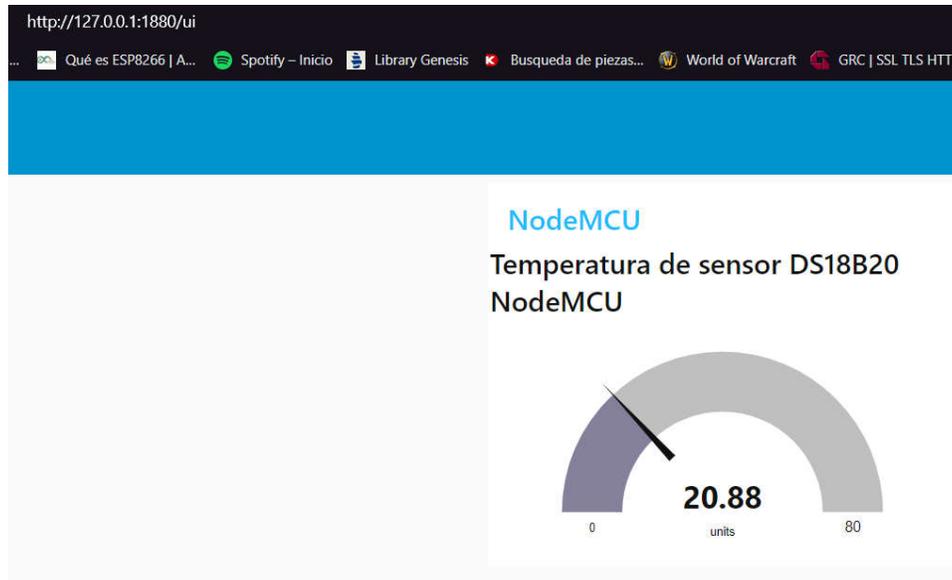


Figura 3.15. Interfaz gráfica de Node-RED.

Incrementamos la temperatura del sensor para verificar la comunicación entre la interfaz gráfica y el microcontrolador, se aprecia que asciende con éxito como se observa en la figura 3.16.

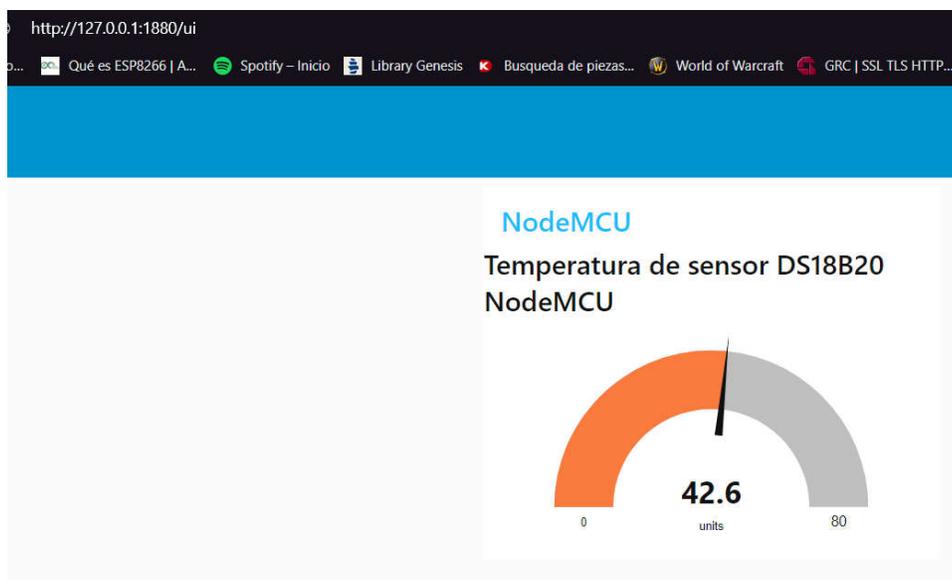


Figura 3.16. Interfaz gráfica de Node-RED visualización de temperatura.

3.4.2. Desarrollo e integración de prototipo con protocolo Websockets y aplicación móvil

Para el desarrollo del prototipo se decidió utilizar el protocolo de comunicación *Websockets* pues elimina la necesidad de añadir componentes de comunicación a la arquitectura, al ser un protocolo de última generación, se tienen ventajas sobre protocolos convencionales y es que además de un protocolo de comunicación, se obtienen diferentes servicios importantes para la aplicación, tales como una base de datos en tiempo real, analizadores de datos y notificaciones. Para implementar este tipo de servicios utilizando otro protocolo de comunicación se tendría que programar en un determinado lenguaje de programación el *backend* necesario para realizar todo esto desde cero, lo cual representa una carga excesiva para el desarrollo de este prototipo.

Para la visualización de los datos capturados por el microcontrolador, se desarrolló una aplicación móvil con el software Android Studio, por las ventajas que ofrece sobre una plataforma web. El entorno de desarrollo Android Studio facilita la adaptación de diferentes tipos de dispositivos móviles, permitiendo que se pueda enfocar la programación en los algoritmos de comunicación de datos y la visualización de los mismos de manera general y funcional.

3.5. VALIDACIÓN DE SENSORES SELECCIONADOS

a) Sensor de energía.

La primer validación de los sensores será la del sensor de energía eléctrica, la prueba que se realizó para la validación del sensor se describe a continuación: Se instaló el sensor de energía eléctrica en la acometida de la red eléctrica monofásica en la residencia, con el objetivo de tomar lecturas diarias del consumo eléctrico registrado por el sensor, estos valores se introdujeron en una hoja de cálculo en Excel para poder realizar la sumatoria de los kWh que se consumieron en el periodo bimestral, para posteriormente poderlo cotejar con la factura de Comisión Federal de Electricidad.

Para esta prueba se requirió manipular la instalación eléctrica residencial, al carecer de una formación eléctrica sólida, se solicitó ayuda a un experto en el tema para poder instalar el sensor de manera segura, siempre manejando los circuitos en frío, en la figura 3.17 se adjunta un diagrama de la instalación eléctrica en la residencia.

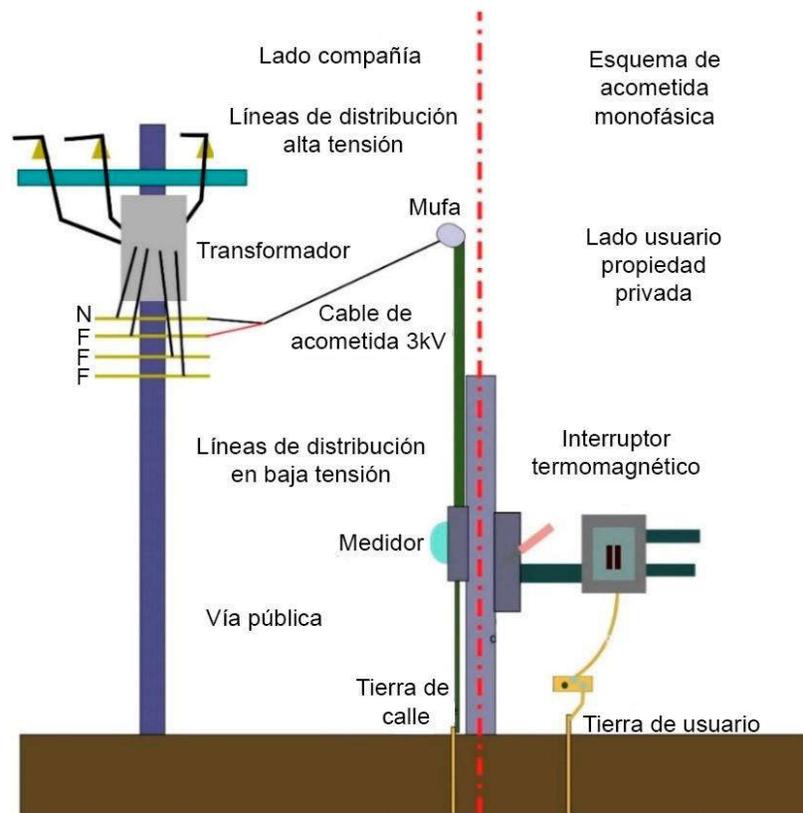


Figura 3.17. Diagrama eléctrico de instalación monofásica en una residencia.

Para conocer el gasto energético total de la residencia fue necesario conectar el sensor en el centro de carga. Se adjunta un diagrama unifilar de conexiones que fueron requeridas para lograr empezar a medir el gasto eléctrico en la residencia. En la figura 3.18 se observa el diagrama unifilar de conexiones del sensor de energía eléctrica con el microcontrolador.

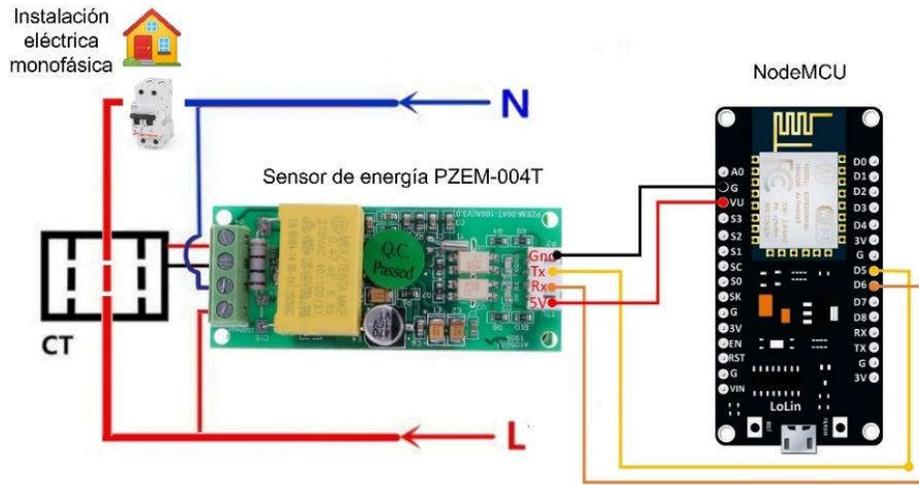


Figura 3.18. Diagrama unifilar de conexiones para medición de gasto eléctrico en residencia.

Con el sensor conectado a la instalación eléctrica se dejó operando de manera continua a lo largo de un periodo de facturación bimestral, donde con ayuda de un programa hecho en Arduino se obtiene la medición diaria de los kWh leídos por el sensor, estos datos posteriormente, eran manualmente introducidos a un archivo de Excel ya que fue la forma más rápida de organizarlos, se presentan los datos recabados en la tabla 2. En la figura 3.19 se muestra la conexión física del sensor en la instalación eléctrica.

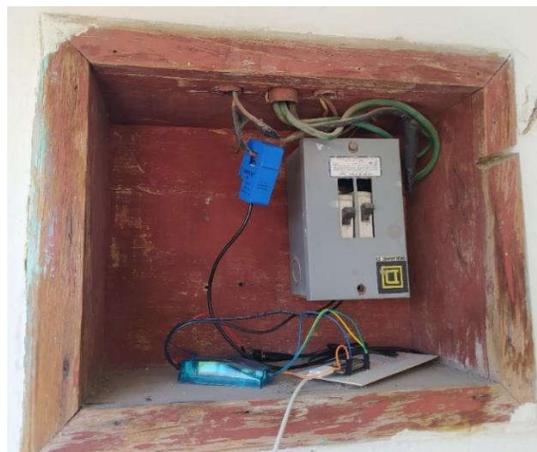


Figura 3.19. Conexión física de sensor en instalación eléctrica residencial.

Tabla 2. Datos del sensor de energía en la instalación monofásica del hogar.

Energía consumida en instalación monofásica		
Fecha	Medición de sensor en kWh	kWh acumulados
6/1/2021	2.5047	2.5047
7/1/2021	2.2719	4.7766
8/1/2021	2.3976	7.1742
9/1/2021	1.8921	9.0663
10/1/2021	2.6056	11.6719
11/1/2021	2.4208	14.0927
12/1/2021	3.1647	17.2574
13/1/2021	2.8643	20.1217
14/1/2021	1.6641	21.7858
15/1/2021	3.3014	25.0872
16/1/2021	2.9764	28.0636
17/1/2021	2.7194	30.783
18/1/2021	3.1778	33.9608
19/1/2021	2.5104	36.4712
20/1/2021	3.0417	39.5129
21/1/2021	3.4463	42.9592
22/1/2021	2.8331	45.7923
23/1/2021	3.0734	48.8657
24/1/2021	2.1977	51.0634
25/1/2021	4.4731	55.5365
26/1/2021	3.3494	58.8859
27/1/2021	3.1799	62.0658
28/1/2021	2.4935	64.5593
29/1/2021	1.8029	66.3622
30/1/2021	2.6411	69.0033
31/1/2021	2.3041	71.3074
1/2/2021	3.0138	74.3212
2/2/2021	1.7738	76.095
3/2/2021	2.2717	78.3667
4/2/2021	2.6133	80.98
5/2/2021	2.0193	82.9993

6/2/2021	1.9309	84.9302
7/2/2021	2.3374	87.2676
8/2/2021	2.1341	89.4017
9/2/2021	4.7611	94.1628
10/2/2021	3.8037	97.9665
11/2/2021	4.3595	102.326
12/2/2021	4.1747	106.5007
13/2/2021	1.9734	108.4741
14/2/2021	3.6432	112.1173
15/2/2021	4.2477	116.365
16/2/2021	4.7481	121.1131
17/2/2021	4.8897	126.0028
18/2/2021	3.7301	129.7329
19/2/2021	5.2041	134.937
20/2/2021	4.0473	138.9843
21/2/2021	5.4097	144.394
22/2/2021	5.1905	149.5845
23/2/2021	5.3781	154.9626
24/2/2021	4.7074	159.67
25/2/2021	3.8307	163.5007
26/2/2021	4.5044	168.0051
27/2/2021	5.3738	173.3789
28/2/2021	5.0547	178.4336
1/3/2021	3.0471	181.4807
2/3/2021	3.7841	185.2648
3/3/2021	3.2602	188.525
4/3/2021	4.0719	192.5969
5/3/2021	2.6061	195.203
6/3/2021	3.5594	198.7624
7/3/2021	2.6719	201.4343
8/3/2021	3.0172	204.4515

En la figura 3.20 se adjunta una gráfica ojiva con los datos mostrados en la tabla 1.

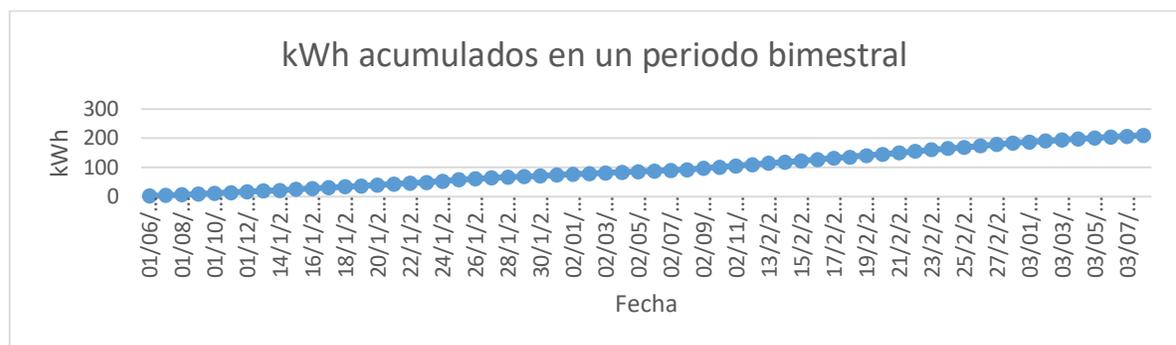


Figura 3.20. Gráfica de consumo de energía eléctrica en kWh.

Con el gasto eléctrico registrado durante un periodo de facturación bimestral, se cotejó la medición total de kWh realizada por el sensor de energía eléctrica PZEM-004T y la factura emitida por el proveedor de servicio de energía eléctrica Comisión Federal de Electricidad. En la figura 3.21 se adjunta la factura emitida por el proveedor de servicio, el cual marca una lectura de 206 kWh por el periodo, y el sensor registró una lectura de 204 kWh obteniendo menos de 0.5% de error en la medición.



Figura 3.21. Facturación de periodo bimestral emitida por Comisión Federal de Electricidad.

b) Sensor de flujo de agua.

En la validación del sensor de flujo de agua se diseñó una planta con la finalidad de tener un flujo constante y medible de agua, para calibrar el sensor y obtener resultados más precisos, la conexión de este sensor con el microcontrolador se muestra en la figura 3.22.

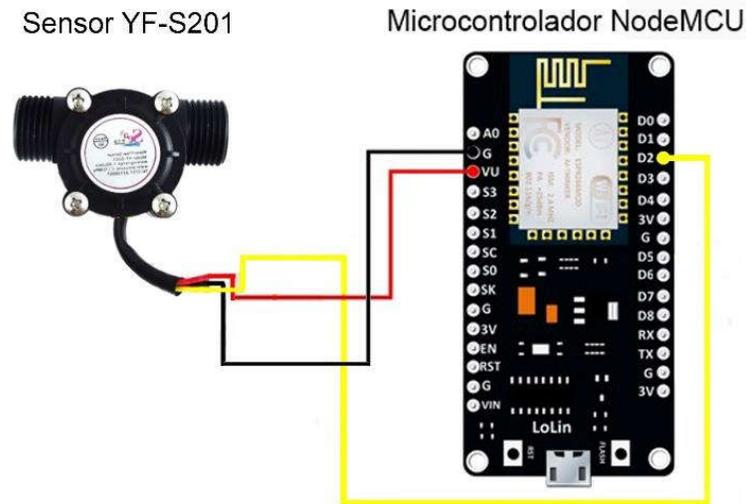


Figura 3.22. Diagrama unifilar de conexión de sensor con el microcontrolador.

Se cuenta con un tanque de agua de 18 litros, conectado a través de una manguera a una bomba de agua encargada de aumentar la presión en el circuito para que haya un flujo constante. La bomba de agua se acopla al sensor utilizando un acople reductor de cobre, el flujo pasa por el caudalímetro de manera vertical, ya que el fabricante indica que es necesario que el flujo de agua pase el sensor de manera perpendicular al suelo. Posterior al sensor de flujo de agua hay un contenedor de distintas capacidades, con el fin de comprobar que las mediciones del volumen de agua sean correctas. En la figura 3.23 se muestra una fotografía de la planta que se describió anteriormente.

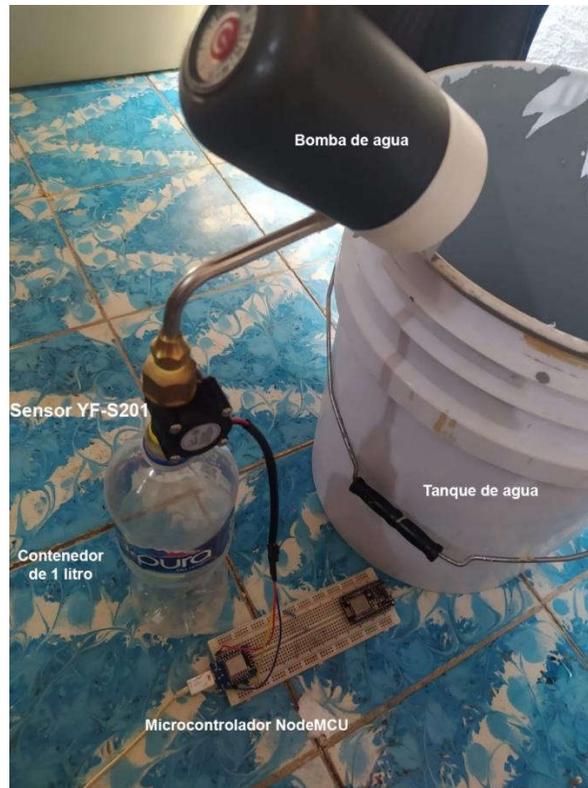


Figura 3.23. Planta para pruebas de sensor de agua.

Para obtener mayor precisión en las lecturas del sensor se realizó una calibración, para esto se programó un Sketch encargado de contar los pulsos que recibía el microcontrolador del sensor al llenar un contenedor de un litro de agua, se realizó esta prueba 10 veces, para obtener el promedio de los distintos valores y calcular el factor de precisión, en la tabla 3 se muestran los valores obtenidos.

Tabla 3. Conteo de pulsos recibidos por el sensor de flujo de agua para medición de un litro.

Calibración de sensor de flujo de agua		
No. de prueba	Tamaño de contenedor	No. de pulsos
Prueba 1	1 litro	457
Prueba 2	1 litro	435
Prueba 3	1 litro	425
Prueba 4	1 litro	414
Prueba 5	1 litro	415
Prueba 6	1 litro	430
Prueba 7	1 litro	420
Prueba 8	1 litro	418
Prueba 9	1 litro	418
Prueba 10	1 litro	437

Con los 10 cálculos obtenidos se puede obtener el promedio de pulsos por litro, y posteriormente crear un Sketch que sea capaz de obtener el valor de consumo de agua en litros. Para probar la precisión de este código se hicieron dos pruebas, utilizando la misma planta y cambiando el contenedor final para saber la cantidad exacta de agua que atraviesa el sensor, para la primera prueba se utilizó un contenedor de 600 ml. Al llenarse el contenedor de 0.6 litros, se obtuvieron los datos mostrados en la figura 3.24.

```

COM3
10:28:09.300 -> Caudal: 1.688L/min Volumen: 0.441 L
10:28:10.575 -> Caudal: 1.688L/min Volumen: 0.469 L
10:28:11.562 -> Caudal: 1.688L/min Volumen: 0.497 L
10:28:12.589 -> Caudal: 1.688L/min Volumen: 0.525 L
10:28:13.574 -> Caudal: 1.688L/min Volumen: 0.553 L
10:28:14.564 -> Caudal: 1.688L/min Volumen: 0.581 L
10:28:15.583 -> Caudal: 1.547L/min Volumen: 0.607 L
10:28:16.569 -> Caudal: 0.141L/min Volumen: 0.609 L
10:28:17.559 -> Caudal: 0.000L/min Volumen: 0.609 L
10:28:18.576 -> Caudal: 0.000L/min Volumen: 0.609 L
10:28:19.557 -> Caudal: 0.141L/min Volumen: 0.612 L
10:28:20.579 -> Caudal: 0.000L/min Volumen: 0.612 L
10:28:21.563 -> Caudal: 0.000L/min Volumen: 0.612 L
10:28:22.585 -> Caudal: 0.000L/min Volumen: 0.612 L
10:28:23.573 -> Caudal: 0.000L/min Volumen: 0.612 L
10:28:24.562 -> Caudal: 0.000L/min Volumen: 0.612 L
10:28:25.576 -> Caudal: 0.000L/min Volumen: 0.612 L
Autoscroll [x] Mostrar marca temporal Ambos NL & CR 9600 baudio Limpiar salida
  
```

Figura 3.24. Monitor serial registrando los datos del sensor de flujo de agua.

Para la segunda prueba se cambió el contenedor de agua por un contenedor de agua con capacidad de 1.5 Litros, el microcontrolador arrojó el resultado que se observa en la figura 3.25.

```

COM3
10:32:32.403 -> Caudal: 1.688L/min Volumen: 1.118 L
10:32:33.419 -> Caudal: 1.688L/min Volumen: 1.146 L
10:32:34.402 -> Caudal: 1.688L/min Volumen: 1.174 L
10:32:35.418 -> Caudal: 1.547L/min Volumen: 1.200 L
10:32:36.400 -> Caudal: 1.688L/min Volumen: 1.228 L
10:32:37.392 -> Caudal: 1.688L/min Volumen: 1.256 L
10:32:38.412 -> Caudal: 1.028L/min Volumen: 1.287 L
10:32:39.398 -> Caudal: 1.688L/min Volumen: 1.315 L
10:32:40.415 -> Caudal: 1.688L/min Volumen: 1.343 L
10:32:41.396 -> Caudal: 1.547L/min Volumen: 1.369 L
10:32:42.412 -> Caudal: 1.547L/min Volumen: 1.395 L
10:32:43.400 -> Caudal: 1.688L/min Volumen: 1.423 L
10:32:44.390 -> Caudal: 1.547L/min Volumen: 1.449 L
10:32:45.410 -> Caudal: 1.547L/min Volumen: 1.474 L
10:32:46.398 -> Caudal: 1.547L/min Volumen: 1.500 L
10:32:47.418 -> Caudal: 1.547L/min Volumen: 1.526 L
Autoscroll [x] Mostrar marca temporal Ambos NL & CR 9600 baudio Limpiar salida
  
```

Figura 3.25. Monitor serial imprimiendo los datos del sensor de flujo de agua.

c) Sensor de presencia de gas LP.

En el caso del sensor de presencia de gas, para realizar la conexión con el microcontrolador, se requiere utilizar un pin de entrada digital, alimentación de voltaje y tierra, con esas tres conexiones el sensor está listo para trabajar con el microcontrolador, se adjunta un diagrama de conexión en la figura 3.26.

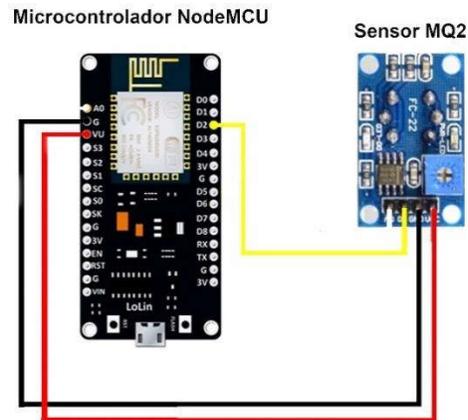


Figura 3.26. Diagrama unifilar de conexión de sensor MQ-2 con NodeMCU.

Para la validación del sensor de presencia de gas, se programó un Sketch el cual lee periódicamente el valor de la entrada digital D2, el sensor mantiene el pin de la señal de salida en estado alto, lo que indica al microcontrolador que el sensor no está detectando presencia de gas LP en ese momento, caso contrario al estar la entrada digital D2 en estado bajo, esto representa una detección de gas LP. El sensor tiene un potenciómetro, el cual se ajustó para que el umbral de detección del gas LP esté calibrado, y hacerlo capaz de detectar una fuga de gas LP. Se realizó la prueba para probar la funcionalidad del sensor, acercando el sensor a una fuente de gas LP por periodos muy cortos de tiempo, evitando a toda costa el peligro de explosión que pudiese ocurrir. En la figura 3.27 se muestra una captura del monitor serial donde se notifica la presencia del gas LP.

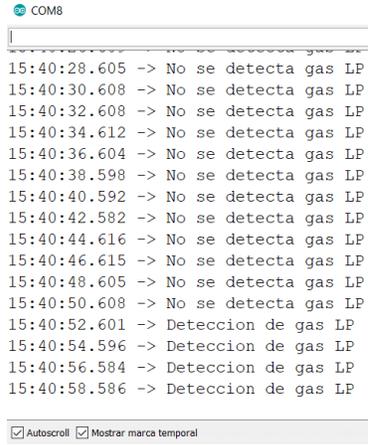


Figura 3.27. Monitor serial de microcontrolador imprimiendo el estado del sensor digital de presencia de gas LP.

d) Sensor de temperatura.

El DS18B20 es un sensor digital de temperatura que utiliza el protocolo de comunicación *1-Wire*, este protocolo necesita solo un pin de datos para comunicarse y permite conectar más de un sensor en el mismo bus. Para la conexión de este sensor con el microcontrolador, se alimenta con 3.3 Volts y 0 Volts, el cable de señal se conecta directamente a cualquier entrada digital.

Con el sensor conectado se programó un Sketch capaz de hacer lecturas de temperatura, en la figura 3.28 se adjunta la captura de pantalla del monitor serial del programa de Arduino.

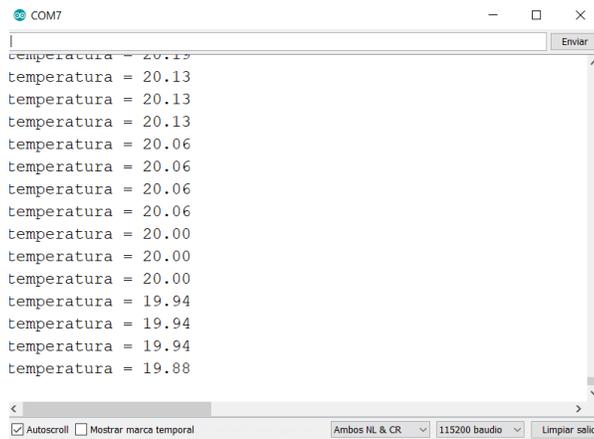


Figura 3.28. Monitor serial de microcontrolador imprimiendo el valor de temperatura.

e) Sensor de iluminación.

Para la prueba del sensor de iluminación BH1750 se programó un sencillo Sketch con el que se probó la comunicación del sensor con el microcontrolador y obtener las lecturas, ya que el protocolo que maneja el sensor digital es I^2C . En la figura 3.29 se muestra el diagrama unifilar de conexiones entre el microcontrolador y el sensor digital.

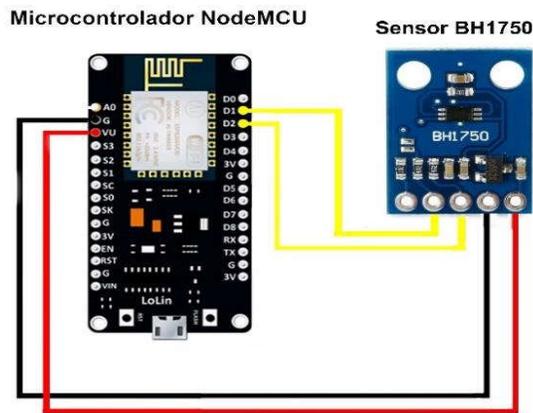


Figura 3.29. Diagrama unifilar de conexión de sensor de iluminación BH1750 con microcontrolador NodeMCU a través de protocolo de comunicación I^2C .

En la figura 3.30 se observa una captura de pantalla donde se muestra el monitor serial del microcontrolador imprimiendo el valor de la iluminación en Lux.

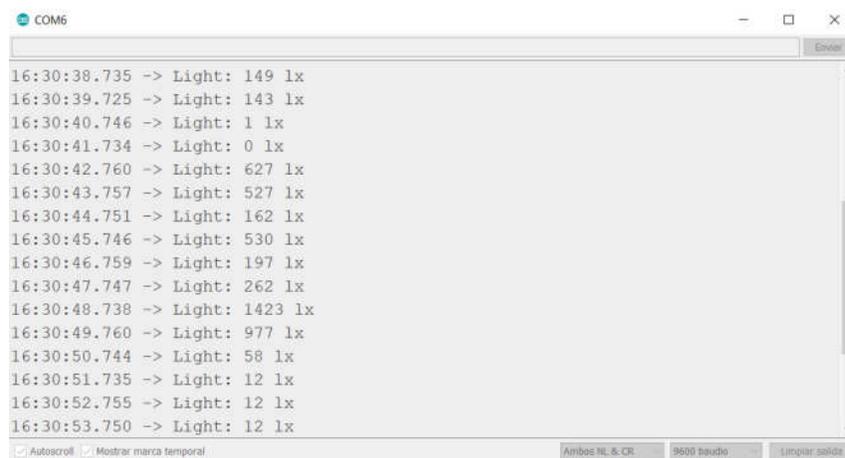


Figura 3.30. Monitor serial de microcontrolador imprimiendo el valor del sensor de iluminación.

3.6. MEDICIÓN DE PANEL FOTOVOLTAICO

Cuando una placa o panel fotovoltaico recibe luz solar, los electrones comienzan a moverse dentro de los paneles de silicio, lo que produce electricidad de corriente continua. Las células fotovoltaicas utilizan esta energía como la entrada de voltaje en un dispositivo microinversor, este se encarga de convertir la energía eléctrica generada en corriente continua, a energía eléctrica de corriente alterna. La instalación eléctrica en las viviendas funciona con corriente alterna, por lo que la energía producida por las placas solares no es útil por sí misma.

Si se produce más electricidad de la que se necesita, también se puede almacenar en baterías o ser vendida a través de la red eléctrica del proveedor del servicio, siempre y cuando se tengan los circuitos de acoplamiento requeridos para recibir una compensación económica.

El sensor seleccionado para analizar la energía es capaz de tomar la lectura de los parámetros eléctricos generados por el inversor, lo que permite conocer la energía generada por el panel fotovoltaico. En la figura 3.31 se muestra la conexión entre el sensor y el microinversor conectado al panel fotovoltaico.

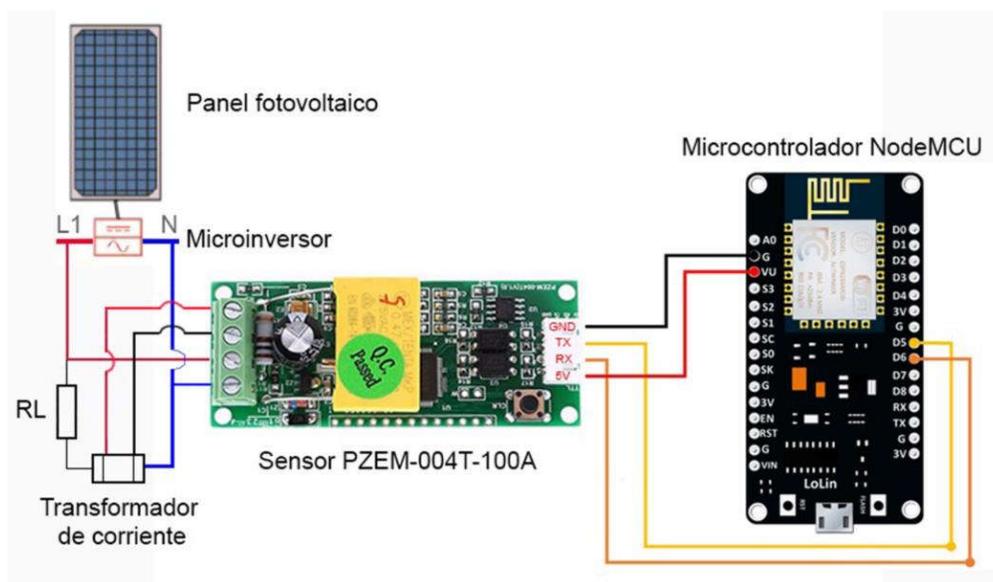


Figura 3.31. Diagrama unifilar de conexiones de arquitectura fotovoltaica con microcontrolador.

3.7. INTEGRACIÓN DE PROTOTIPO

Ya presentados y validados los sensores que se utilizaron en la arquitectura del sistema de telemetría, se procedió con la integración del microcontrolador con los sensores y una fuente de voltaje para alimentar los sensores, conformando el prototipo del proyecto. Se utilizó una placa fenólica de cobre perforada, que permitió realizar las conexiones de los sensores con el microcontrolador, soldando estas conexiones con estaño.

En la figura 3.32 se muestra el microcontrolador con los sensores y la fuente de voltaje, soldados a la placa fenólica de cobre.

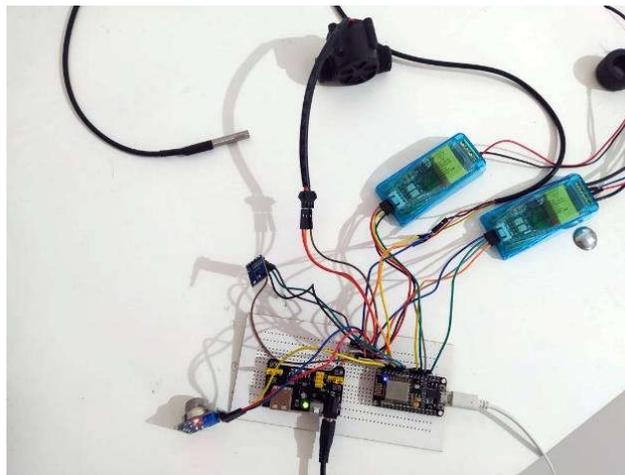


Figura 3.32. Prototipo integrado.

3.8. PROGRAMACIÓN DE SKETCH PARA LECTURA DE SENSORES

Con las pruebas y validaciones que se realizaron en los sensores, se integró el prototipo, haciendo uso de los códigos de los Sketchs creados anteriormente para unificarlos en un solo programa, utilizando funciones para que las mediciones que toma el microcontrolador se hagan de manera ordenada y periódica. En la figura 3.33 se observa el diagrama de flujo del código.

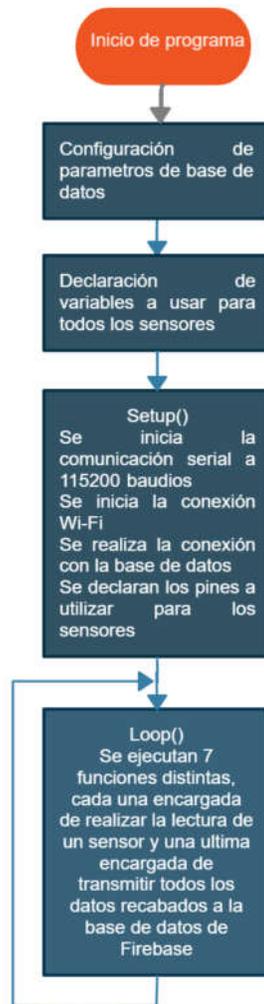


Figura 3.33. Diagrama de flujo de Sketch para lectura de todos los sensores y publicación de los datos en la base de datos en tiempo real de Firebase.

El Sketch que se programó para este prototipo es capaz de comunicar el microcontrolador con la base de datos de Firebase en tiempo real, a través de *Wi-Fi* para cargar los datos de los sensores.

El algoritmo funciona de manera secuencial utilizando funciones, comienza por la etapa de declaración de elementos, se especifican las librerías requeridas por el Sketch, y se crean las variables utilizadas en el código, estas últimas incluyen la medición de consumo de energía eléctrica en kWh, consumo de agua en litros, temperatura en grados centígrados, presencia de gas, e iluminación en lux. Por último se declararon los pines digitales que utiliza el puerto serial para comunicarse con los

sensores digitales de energía que maneja el protocolo *UART* a 9600 baudios.

Después de la primera etapa del algoritmo, el Sketch entra en el ciclo "main", mismo que se ejecuta una única vez, dentro de este ciclo se configuran algunos parámetros requeridos para que el microcontrolador pueda conectarse a la red *Wi-Fi* de la residencia, también configura los distintos pines para ser utilizados por los sensores, con los distintos protocolos que estos requieren.

La última etapa del algoritmo es el ciclo "loop" el cual se repite indefinidamente, solo se detiene cuando el microcontrolador se queda sin energía de alimentación, en este ciclo se programaron seis funciones distintas, de las cuales cinco de ellas se encargan de tomar la lectura de un sensor para cada parámetro, una función para los dos sensores de energía, otra para el sensor de flujo de agua, otra para el sensor de presencia de gas, otra para el sensor de temperatura, y una más para el sensor de iluminación, con la finalidad de darle una estructura uniforme al algoritmo. La sexta función es la encargada de transmitir la información hacia la base de datos de Firebase, donde son almacenados los valores, y a su vez, son desplegados en tiempo real en la interfaz gráfica de la aplicación móvil.

3.9. COMUNICACIÓN DE MICROCONTROLADOR CON BASE DE DATOS

Para que el microcontrolador pueda leer y escribir en la base de datos de Firebase, fue necesario incluir en el algoritmo, parámetros de configuración con el fin de que el Sketch sea capaz de saber la dirección del servidor donde se aloja la base de datos y poder establecer una conexión, se requiere crear una cuenta de usuario en la página oficial de Firebase, para poder obtener las credenciales de la base de datos.

Una vez se accede al sitio oficial de Firebase con una cuenta de Google, es posible crear un proyecto, únicamente se le debe asignar un nombre al proyecto para identificarlo como se muestra en la figura 3.34.



Figura 3.34. Creación de proyecto en Firebase.

Una vez se ha creado el proyecto de manera exitosa, accederemos al panel de control del proyecto, en donde hay distintas pestañas con información del mismo. En la figura 3.35 se muestra la pestaña de autenticación, en ella se encuentran dos parámetros requeridos para la configuración del algoritmo de NodeMCU para que le sea posible conectarse a la base de datos y comenzar a cargar las lecturas de los sensores.



Figura 3.35. Pantalla principal de Firebase en el explorador web.

Al obtener los parámetros de configuración de la base de datos el microcontrolador está listo para enviar o recibir información de Firebase, al estar los sensores ya integrados al sistema de telemetría, se pueden enviar datos para comprobar que se puedan visualizar en la página web.

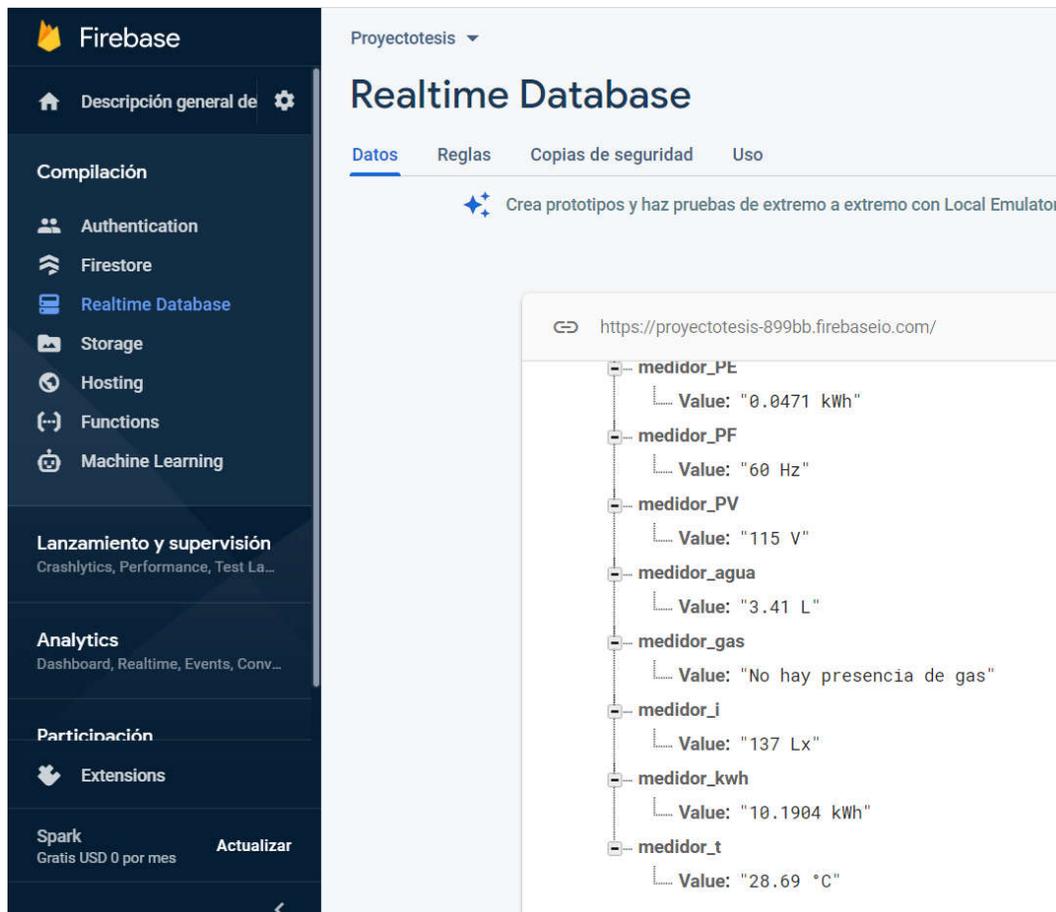


Figura 3.36. Valores visualizados en la base de datos.

3.10. DESARROLLO DE INTERFAZ GRÁFICA EN APLICACIÓN MÓVIL ANDROID

Para el desarrollo de la aplicación móvil, se utilizó el software Android Studio, ya que es el entorno de desarrollo integrado para aplicaciones móviles Android por excelencia, fue desarrollado por Google, mismo desarrollador del sistema operativo móvil Android.

Ya con el software instalado, fue necesario actualizar las *API* (Application Programming Interface) con la finalidad de tener su versión más reciente y poder crear un dispositivo virtual, este contiene un perfil de hardware, una imagen del sistema, un área de almacenamiento, entre otras propiedades. El software proporciona perfiles de hardware predefinidos para dispositivos comunes, aunque también puede definirse un dispositivo diferente creando un perfil de hardware nuevo.

Para el caso de Android Studio, es diferente la programación realizada en el microcontrolador, ya que este trabaja de manera secuencial, la programación en el lenguaje Android es orientada a objetos, para el desarrollo de la aplicación móvil fue necesario diseñar la interfaz gráfica de la aplicación, la cual conlleva todos los elementos que interactúan con el usuario. Para este prototipo se desarrollaron 4 pantallas en total, las cuales se describen a continuación:

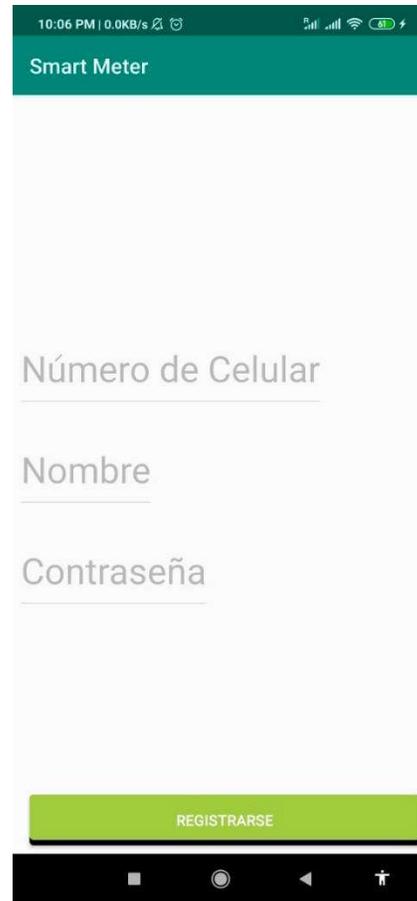
- Pantalla principal. En esta pantalla el usuario puede decidir entre registrar un usuario o iniciar sesión con una cuenta ya existente.
- Pantalla de registro de usuario. En esta pantalla el usuario final es capaz de registrarse con un número telefónico, un nombre de usuario y una contraseña para crear una cuenta.
- Pantalla de inicio de sesión. En esta pantalla el usuario puede iniciar sesión para pasar a la pantalla personalizada donde se despliegan los datos.
- Pantalla de interfaz gráfica. En esta pantalla se muestran los datos en tiempo real de las mediciones tomadas por los sensores.

En la figura 3.37 (a) se observa la pantalla principal, cuenta con 2 botones, cada uno dirige al usuario a su respectiva pantalla.

En la figura 3.37 (b) se muestra una captura de la pantalla de registro de usuario, los datos introducidos aquí son almacenados en la base de datos.



(a)



(b)

Figura 3.37. (a) Pantalla principal de aplicación. (b) Pantalla de registro de usuario

La pantalla de registro procesa los valores introducidos, la aplicación hace una solicitud a la base de datos para validar si previamente se ha registrado un usuario con el mismo número de celular, nombre o contraseña, en caso de que se haya introducido anteriormente el mismo dato, la aplicación notifica a través de una ventana emergente que ya hay un usuario existente, y en caso contrario, si se valida que son valores nuevos son cargados a la base de datos para que posteriormente el usuario pueda entrar a su cuenta.

En la figura 3.38 se muestra una captura de la pantalla de inicio de sesión, el usuario debe introducir su número telefónico y la contraseña creada previamente para poder iniciar sesión en la aplicación. La aplicación realizará la validación de las credenciales, cotejando los valores introducidos con los valores registrados en la base

de datos de usuarios, en caso de que ambos valores coincidan con los que se encuentran almacenados, se redirige al usuario a la pantalla de visualización de datos, cabe mencionar que los datos mostrados varían ya que cada cuenta de usuario es capaz de almacenar valores diferentes entre sí, nunca serán compartidos con otras cuentas respetando la privacidad.

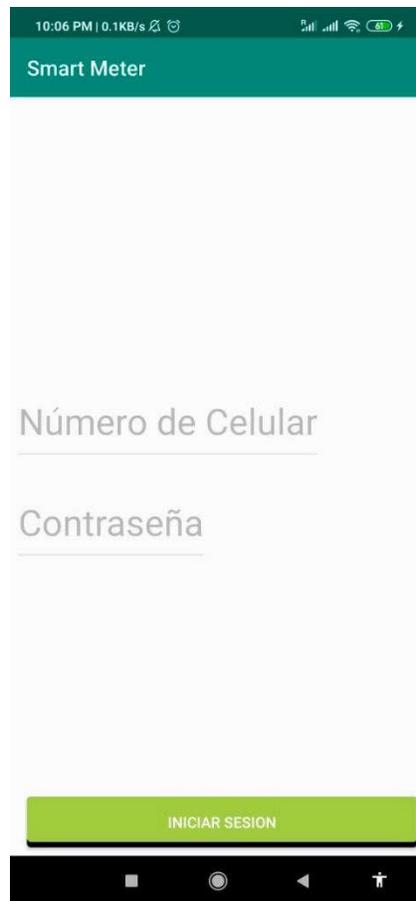


Figura 3.38. Pantalla de inicio de sesión.

En la figura 3.39 se observa la pantalla más importante de todas, es en esta pantalla donde se muestra la interfaz gráfica programada, la cual despliega los datos recabados que se obtienen de la medición de los distintos sensores. Esta pantalla está constantemente actualizando los valores en tiempo real conforme el microcontrolador los carga a la base de datos, mientras la plataforma digital tenga alimentación de

voltaje, se realizará la lectura de los sensores y estos valores serán cargados periódicamente al banco de datos, en un ciclo programado para repetirse indefinidamente por el microcontrolador.

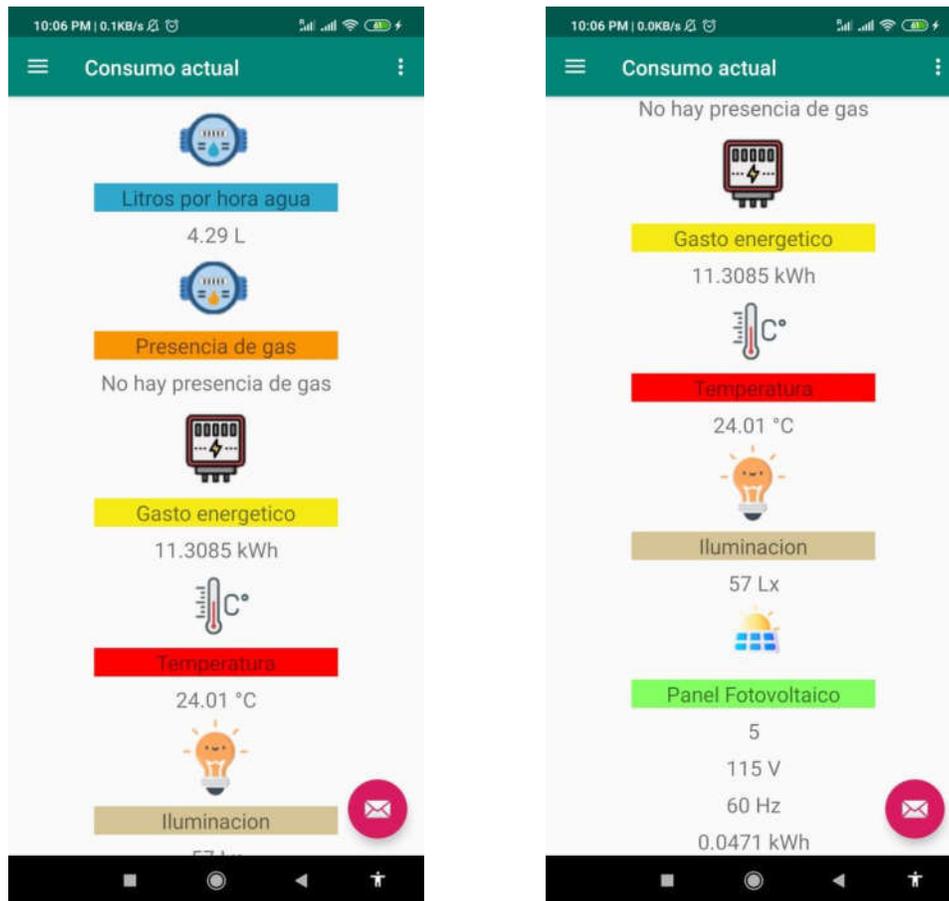


Figura 3.39. Pantalla de visualización de datos.

CAPÍTULO 4

ANÁLISIS DE RESULTADOS

4.1. ANÁLISIS DE RESULTADOS

En este capítulo se analizará el costo-beneficio del proyecto, tomando en cuenta el tiempo total invertido, en el diseño, el desarrollo y la implementación, el costo de la adquisición de los sensores requeridos, entre otros costos a tomar en cuenta.

4.1.1. Costos de adquisición

Para obtener el costo de estos componentes se presenta en la tabla 4, un desglose de costos para conocer el costo total de adquisición en el desarrollo del proyecto.

Tabla 4. Costos de adquisición de proyecto

Tabla de costos			
Componente	Cantidad	Precio	Total
Microcontrolador NodeMCU	1	\$ 206.00	\$ 206.00
Sensor analizador de energía PZEM-004T	2	\$ 502.00	\$ 1004.00
Sensor digital de iluminación BH1750	1	\$ 100.00	\$ 100.00
Sensor de flujo de agua YF-S201	1	\$ 200.00	\$ 200.00
Sensor de presencia de gas MQ-2	1	\$ 50.00	\$ 50.00
Sensor de temperatura DS18B20	1	\$ 200.00	\$ 200.00
Placa fenólica perforada	1	\$ 60.00	\$ 60.00
Fuente para protoboard MB102 3.3v / 5v	1	\$ 30.00	\$ 30.00
3 piezas header, tira de postes hembra 40 pines 2 mm	1	\$ 33.00	\$ 33.00
Tubo De Soldadura Estaño Plomo 60/40 2,75m	1	\$ 30.00	\$ 30.00
Despachador De Agua Eléctrico Recargable	1	\$ 270.00	\$ 270.00
Cautín tipo lápiz para soldadura de 30 W	1	\$ 100.00	\$ 100.00
Kit de destornillador con puntas varias	1	\$ 250.00	\$ 250.00
		Costo total:	\$ 2,533.00

El costo total de adquisición para desarrollar el proyecto equivale a \$2,533.00 pesos mexicanos.

4.1.2. Costos indirectos

En el cálculo de los costos indirectos para el desarrollo del sistema electrónico, se consideran diferentes gastos monetarios, entre ellas la contratación de servicios de profesionistas para apoyarse en las pruebas que se realizaron para validar los sensores. Se desglosan estos conceptos con su respectivo costo en tabla 5.

Tabla 5. Costos indirectos de proyecto.

Actividad	Costo
Contratación de electricista	\$ 250.00
Asesoría de ferretería para compra de acoples y reductores	\$ 150.00
Servicios de base de datos de Firebase	\$ 200.00
Asesoría para desarrollo de aplicación móvil	\$ 1,200.00
Costo total:	\$ 1,800.00

El costo total del desarrollo del sistema electrónico, se obtiene al sumar los costos de diseño, de adquisición e indirectos, dando como resultado la valuación total del proyecto en: \$4,333.00 MXN, costo que incluye la creación, facturación, instalación y puesta en marcha del sistema.

4.1.3. Análisis de costo

Al inicio del proyecto se realizó una estimación de los costos que se preveían en el proyecto, en la figura 4.1 se observa una gráfica de barras donde se compara la estimación realizada con los costos reales del proyecto.

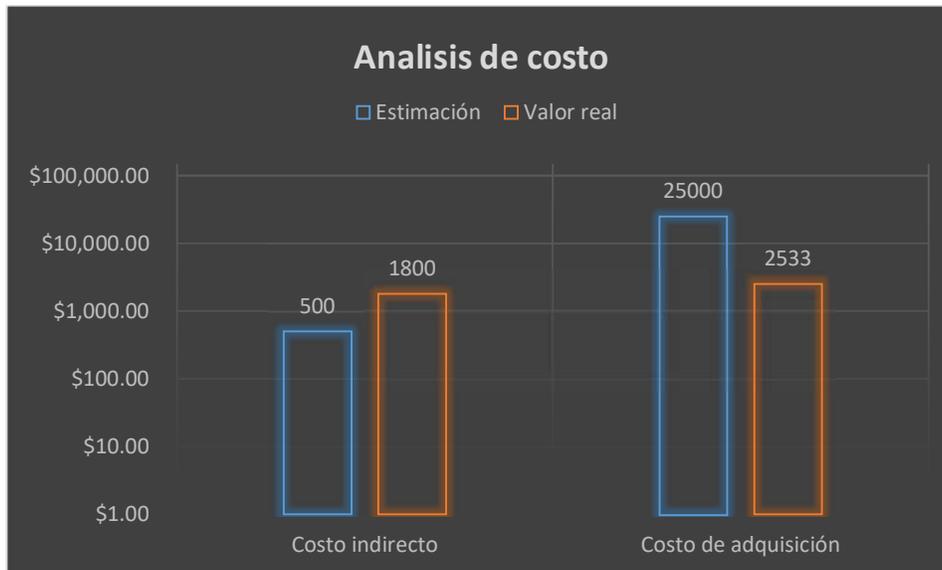


Figura 4.1. Análisis de los resultados de costos

El costo indirecto se incrementó en un 80%, debido a que en un principio no se contempló las asesorías de un Ingeniero en Sistemas para el desarrollo de la aplicación móvil, sin embargo, fue un requerimiento esencial para el proyecto, ya que se buscaba utilizar tecnologías innovadoras en el ámbito y no las tradicionales que se encontraron en la literatura científica.

El costo de adquisición decreció a un 9.86% del presupuesto contemplado inicialmente, debido a que se hicieron cambios en el proyecto, se descartó el uso de un sensor de grado industrial para la medición de flujo de gas LP con el fin de conocer el consumo de gas LP en una residencia, el costo del sensor de aproximadamente \$23,928.00 MXN lo hizo inasequible, por lo que se reemplazó por un sensor de presencia de gas con la finalidad de alertar al usuario final si hay una fuga en su vivienda.

4.1.4. Análisis de implementación de prototipo de sistema electrónico

Para la integración del prototipo fue necesario hacer distintas pruebas que validarán la ingeniería desarrollada tanto de la parte electrónica del proyecto como de la parte de comunicaciones y sistemas. Cada algoritmo programado para interactuar con cada uno de los 5 sensores utilizados en el proyecto de tesis se comprobó que funcionara

de manera continua, y sin errores, validando en todo momento que los datos sean enviados a la base de datos de manera correcta.

Con la integración de la instrumentación con el microcontrolador hecha, se probó que el algoritmo no tuviera errores, ni problemas para leer simultáneamente los seis sensores de la arquitectura, gracias a que el tiempo requerido para tomar la lectura de los sensores es demasiado corto en orden de los milisegundos, aproximadamente 4 ms, el microcontrolador es capaz de obtener la lectura de los 6 sensores en menos de 30 ms.

En la figura 4.2 se muestra la compilación del Sketch programado encargado de leer todos los sensores y reportar los datos hacia la base de datos.



```
// Librerías requeridas para utilizar Firebase
#include <common.h>
#include <FirebaseFS.h>
#include <Firebase_ESP_Client.h>
#include <Utils.h>

#include <ESP8266WiFi.h>
#include <Arduino.h>
#include <PZEM004Tv30.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <Wire.h>
#include <BH1750.h>
```

Guardado.

```
Executable segment sizes:
IRAM : 495156      - code in flash      (default or ICACHE_FLASH_ATTR)
IRAM : 30080 / 32768 - code in IRAM      (ICACHE_RAM_ATTR, ISRs...)
DATA : 1352 )      - initialized variables (global, static) in RAM/HEAP
RODATA : 1880 ) / 81920 - constants          (global, static) in RAM/HEAP
BSS : 29416 )      - zeroed variables      (global, static) in RAM/HEAP
El Sketch usa 528468 bytes (50%) del espacio de almacenamiento de programa. El máximo es 1044464 bytes.
Las variables Globales usan 32648 bytes (39%) de la memoria dinámica, dejando 49272 bytes para las variables
```

1 NodeMCU 1.0 (ESP-12E Module), 80 MHz, Flash, Legacy (new can return nullptr), All SSL ciphers (most compatible), 4MB (FS:2MB OTA:~1019KB)

Figura 4.2. Sketch compilado y cargado al microcontrolador sin errores.

El dispositivo demostró ser capaz de mantener su funcionamiento, gracias al algoritmo programado para encargarse de que esté operando independientemente de si hay conexión *Wi-Fi* o no, aun con el *Wi-Fi* desconectado, el código es capaz de seguir haciendo lecturas en los sensores, y una vez el microcontrolador reestablece la conexión, se cargan los valores a la base de datos, y de igual manera, se actualizan los datos en la interfaz gráfica, la única excepción es cuando se corta el suministro de energía del microcontrolador, en este caso el microcontrolador es incapaz de seguir tomando lecturas de los sensores, idealmente no se debería interrumpir el suministro eléctrico en el hogar, sin embargo suele ocurrir, variando la frecuencia de estas interrupciones de servicio según la zona residencial en la que se habite, cuando esto ocurre, hay tramas de datos que no son tomadas en cuenta, creando discrepancias entre los valores reales y los medidos.

Las pruebas para la medición de energía eléctrica generada por el sistema de panel fotovoltaico y microinversor fueron programadas para el tercer y cuarto semestre del programa de Maestría en Ingeniería electrónica ya que el Instituto Tecnológico de Hermosillo cuenta con la infraestructura necesaria para experimentar con el prototipo. Derivado de la cuarentena presentada en el año 2020 por el virus SARS-CoV-2 y las recomendaciones emitidas por la Secretaría de Educación Pública, se limitó el acceso a los laboratorios y las instalaciones, misma razón por la que no hubo la oportunidad de coincidir con personal capacitado capaz de mostrar la instalación del panel fotovoltaico y su ubicación, con el fin de realizar las pruebas necesarias para la documentación.

4.2. COMPARACIÓN DEL SISTEMA Y PRODUCTOS COMERCIALES

Con el fin de validar la innovación del sistema electrónico desarrollado en el proyecto de tesis, se presenta la tabla 6 donde se realiza una comparación de las características de dos sistemas comerciales existentes y el prototipo desarrollado, destacando en algunos aspectos entre las ofertas actualmente.

Tabla 6. Comparación entre sistemas.

Sistema	Sistema electrónico de telemetría Smart Meter	Analizador de redes eléctricas marca CIRCUTOR	Medición remota de agua, gas y electricidad (solo disponible en estados unidos)
Registra más de un parámetro	✓	X	✓
Arquitectura	Basada en microcontrolador	-	Basada en microcontrolador
Visualización de datos	Aplicación Móvil	Portal web	Plataforma en línea
Vida útil del dispositivo	1 año aproximadamente	-	-
Protocolo de comunicación	Websockets	RS-485 MODBUS/RTU	-
Requiere licencia de software	X	X	✓
Protocolo de conexión	Wi-Fi	Wi-Fi / Ethernet	Wi-Fi / Ethernet
Costo aproximado	\$ 2,090.00	\$ 13,270.00	\$ 13,402.88

La oferta comercial de sistemas de medición de distintos parámetros en México es inexistente, lo más apegado a este tipo de sistemas son analizadores de instalaciones eléctricas que se encargan de tomar lecturas periódicas de la corriente, el voltaje y calculan la energía que se está consumiendo en la instalación, en otros países si existen ofertas comerciales de medidores inteligentes, capaces de registrar distintos parámetros tales como consumo de agua o consumo de gas, lo que eleva los precios de estos sistemas ya que la medición de gas resulta ser costosa, la medición de consumo de gas posteriormente puede ser implementada al sistema desarrollado en este prototipo, siempre y cuando se cuente con el presupuesto requerido para ello.

Además de la ventaja del costo y de la lectura de distintos parámetros que ofrece el sistema desarrollado en este proyecto de tesis, se destaca que no se requieren licencias de software para hacer uso de la interfaz gráfica del sistema ya que fue desarrollada en un entorno gratuito lo que también ofrece una ventaja competitiva sobre los demás sistemas.

CAPÍTULO 5

CONCLUSIONES

5.1. CONCLUSIONES GENERALES

En el presente trabajo de tesis, se realizaron los desarrollos de ingeniería y un prototipo para validar la arquitectura diseñada en base a instrumentación innovadora, protocolos de comunicación de última generación enfocados para aplicaciones en el internet de las cosas, y una interfaz programada desde cero con un software libre. De esta forma se cumplieron los objetivos declaradas al inicio del proyecto, con la diferencia que el prototipo contemplaba la conexión de un panel fotovoltaico con un dispositivo microinversor para medir la energía generada por el panel, en este caso se consideró en la arquitectura el sensor de energía, mas no se llevó a cabo la conexión del panel fotovoltaico con el sistema de telemetría.

Es importante mencionar que en la búsqueda de la literatura científica no se encontró ningún dispositivo para residencias que sea capaz de registrar la lectura de consumo de energía eléctrica, consumo de agua y distintos parámetros en el hogar, procesando estos datos con un microcontrolador y utilizando la base de datos de Firebase para mostrarlos en una aplicación móvil para Android.

Con base en las investigaciones y los trabajos realizados se concluye que es posible desarrollar un sistema de telemetría para los servicios básicos en instalaciones residenciales, así como para medición de parámetros adicionales, haciendo un análisis cauteloso de la instrumentación que se requiere para lograr medir los parámetros deseados, y seleccionar correctamente los sensores. Las pruebas y los experimentos realizados fueron suficientes para validar la plataforma digital desarrollada con el módulo NodeMCU. Este módulo proporciona una amplia gama de herramientas para el desarrollo de interfaces gráficas, y el sistema operativo Android es una plataforma totalmente gratuita con una variedad de funciones para desarrollar diversas aplicaciones

La implementación de este sistema de telemetría impacta principalmente en la forma en la que el usuario utiliza los recursos energéticos en su hogar, ya que la visualización en tiempo real de estos parámetros ayuda a que el usuario este consiente del consumo que realiza en su residencia, de esta manera puede tomar decisiones congruentes sobre la forma en la que utiliza los recursos energéticos disponibles, lo que se traduce en un ahorro tanto de recursos como de dinero.

La principal aportación de este trabajo de tesis a la disciplina de estudio es una arquitectura funcional para el desarrollo de un sistema de telemetría residencial utilizando tecnología reciente, ya que en la búsqueda de la literatura, se encontró que si bien ya se han diseñado y creado sistemas de medición de distintos parámetros, estos no utilizan los protocolos de comunicación que se implementaron en este proyecto de tesis, tal es el caso de Firebase, que si bien tiene una infinidad de aplicaciones y de usuarios utilizando el servicio, en este ramo se carecen de proyectos que implementen la medición de variables en el hogar con la visualización a través de una aplicación móvil utilizando la base de datos de Firebase.

5.2. CONCLUSIÓN DE LA EXPERIMENTACIÓN

Los resultados de la experimentación sobre el prototipo indican que es posible realizar la medición distintos parámetros utilizando un solo microcontrolador, para este caso se miden 6 sensores en tiempo real, gracias a la capacidad de la plataforma digital para leer sensor por sensor, tomar el dato, procesarlo y posteriormente cargarlo a la base de datos.

El sistema si bien cuenta con un nodo centralizado encargado de tomar las lecturas de los distintos sensores, es posible expandir el sistema, para añadir nodos extras que sean capaz de tomar lectura de mas parámetros, y de igual forma, desplegarlos en la interfaz gráfica programada.

El principal inconveniente con el prototipo es la capacidad para mantener energizado el dispositivo, para este proyecto no se consideró utilizar una batería, desde un inicio, partiendo desde la base en el que existe un suministro ideal de energía

eléctrica para alimentar el dispositivo, cosa que difiere de la realidad, al haber apagones de luz.

El uso de una arquitectura websockets para este prototipo logró reducir la robustez de la arquitectura, ya que en protocolos convencionales utilizados para este tipo de arquitecturas como el protocolo MQTT, obliga al microcontrolador a interactuar con un intermediario, lo cual hace que la programación de los algoritmos entre el microcontrolador, la base de datos y la interfaz gráfica sean más tediosos de programar.

5.3. TRABAJOS FUTUROS

Como recomendaciones para trabajos futuros es importante realizar las pruebas con la arquitectura fotovoltaica, para obtener las mediciones de energía eléctrica generada con el sensor PZEM-004T conectado a la salida del microinversor.

Además, se puede trabajar en la implementación de algoritmos con inteligencia artificial para el análisis de los datos recabados, para que pueda aprender sobre los patrones regulares de consumo del usuario y pueda alertar a este usuario cuando algo este fuera de los rangos establecidos como regulares, esto representa una carga de programación muy extensa para el tiempo asignado en este proyecto de tesis.

También se puede reducir costos en el material necesario para la canalización de los sensores instalados en el hogar, utilizando varios módulos de microcontrolador NodeMCU, se puede transmitir los datos de manera inalámbrica sin tener que centralizar todos los datos en un mismo microcontrolador, lo que facilitaría la misma instalación de los sensores.

REFERENCIAS BIBLIOGRÁFICAS

- [1] INEGI, “Encuesta Nacional sobre Consumo de Energéticos en Viviendas Particulares, ENCEVI 2018. Instituto Nacional de Estadística y Geografía,” 2018.
- [2] C. Marcelo Cheminet, “Maestría en Energía Energía Informacional – Smart Metering,” 2018.
- [3] “Sensor de Corriente AC 30A no invasivo - SCT-013-030.” <https://naylorlampmechatronics.com/sensores-corriente-voltaje/154-sensor-de-corriente-ac-30a-no-invasivo-sct-013-030.html> (accessed Jun. 30, 2021).
- [4] V. Miron-alexe, “Comparative Study Regarding Measurements of,” *Int. Symp. Fundam. Electr. Eng.*, 2016.
- [5] “Sensor De Voltaje Alterna Zmpt101b.” <https://adichip.net/tienda/producto/sensor-de-voltaje-alterna-ac250v-2ma-zmpt101b-para-arduino/> (accessed Jun. 30, 2021).
- [6] J. Higuera and J. Polo, “Hacia la convergencia de las redes inalámbricas de sensores con IEEE 1451,” *Mundo Electrónico*, no. February, 2008, [Online]. Available: <http://isi.upc.es/en/node/679>.
- [7] “PZEM-004T V3 Module | Arduino & NodeMCU Code, Circuit, Pinout And Library.” [https://innovatorsguru.com/pzem-004t-v3/#iLightbox\[2d4adf73f2493e23a06\]/0](https://innovatorsguru.com/pzem-004t-v3/#iLightbox[2d4adf73f2493e23a06]/0) (accessed Dec. 14, 2020).
- [8] “Sensor de flujo de agua (Hall) 1/2" YF-S201.” <https://www.naylorlampmechatronics.com/sensores-liquido/108-sensor-de-flujo-de-agua-12-yf-s201.html> (accessed Dec. 14, 2020).
- [9] “Tutorial sensores de gas MQ2, MQ3, MQ7 y MQ135.” https://www.naylorlampmechatronics.com/blog/42_Tutorial-sensores-de-gas-MQ2-MQ3-MQ7-y-MQ13.html (accessed Dec. 14, 2020).
- [10] “Sensor de temperatura DS18B20 con protector metálico – Sumador.” <https://sumador.com/en/products/sensor-de-temperatura-ds18b20-con-protector> (accessed Dec. 14, 2020).
- [11] “Tutorial módulo sensor de luz BH1750.” https://www.naylorlampmechatronics.com/blog/44_Tutorial-módulo-sensor-de-luz-BH1750.html (accessed Dec. 14, 2020).
- [12] B. C. Zapata, *Android Studio Essentials*. 2015.
- [13] K. L. Busbee, “Programming Fundamentals - A Modular Structured Approach using C++,” *Solutions*, 2011.
- [14] T. Hagos, *Android Studio IDE Quick Reference*. Berkeley, CA: Apress, 2019.
- [15] B. Vogelvang, *Theory and Applications with EViews*. 2005.
- [16] O. Unigwe, D. Okekunle, and A. Kiprakis, “Economical distributed voltage control in low-voltage grids with high penetration of photovoltaic,” *CIREN - Open Access Proc. J.*, vol. 2017, no. 1, pp. 1722–1725, Oct. 2017, doi: 10.1049/oap-cired.2017.1227.
- [17] “Modulo Wifi ESP8266 ESP-12F - UNIT Electronics.” <https://uelectronics.com/producto/modulo-wifi-esp8266-esp-12f/> (accessed Jun. 30, 2021).
- [18] “NodeMCU ESP8266 Pinout, Specifications, Features & Datasheet.” <https://components101.com/development-boards/nodemcu-esp8266-pinout-features-and->

datasheet (accessed Jun. 30, 2021).

- [19] M. Jabraeil Jamali, B. Bahrami, A. Heidari, P. Allahverdizadeh, and F. Norouzi, *Towards the Internet of Things: Architectures, Security, and Applications*. 2019.
- [20] J. de la Cal Calleja, "Control Domótico basado en el protocolo MQTT," 2019.
- [21] "▷ MQTT: Protocolo de comunicación para pequeños dispositivos móviles »." <https://ccnadesdecero.es/mqtt-protocolo-comunicacion-pequenos-dispositivos-moviles/> (accessed Jun. 30, 2021).
- [22] K. Lane, "Descripción general de Backend as a Service There," *API Evang.*, vol. 2013, no. May, 2013, doi: 10.1016/S0022-3913(12)00047-9.
- [23] A. Kumar, *Mastering Firebase for Android*. 2018.
- [24] US Department of Energy, "Assessment of Demand Response and Advanced Metering," *Fed. Energy Regul. Comm.*, 2008.
- [25] N. Uribe-Pérez, L. Hernández, D. de la Vega, and I. Angulo, "State of the Art and Trends Review of Smart Metering in Electricity Grids," *Appl. Sci.*, vol. 6, no. 3, pp. 1–24, 2016, doi: 10.3390/app6030068.
- [26] N. Mehmood, Z. A. Ali, A. A. Siddiqui, M. Asif, and S. Wasi, "Electronic meter reader & data base management system," *Proc. - 2011 6th IEEE Jt. Int. Inf. Technol. Artif. Intell. Conf. ITAIC 2011*, vol. 1, no. August 2011, pp. 68–71, 2011, doi: 10.1109/ITAIC.2011.6030152.
- [27] A. Valdiosera, "Diseño de medidor inteligente e implementación de sistema de comunicación bidireccional," p. 365, 2013.
- [28] S. Velásquez and C. Gómez, "Monitoreo de variables analogicas usando raspberry PI," *Universidad, Cienc. y Tecnol.*, vol. 17, no. 69, pp. 170–175, 2013.
- [29] A. C. D. Bonganay, J. C. Magno, A. G. Marcellana, J. M. E. Morante, and N. G. Perez, "Automated electric meter reading and monitoring system using ZigBee-integrated raspberry Pi single board computer via Modbus," *2014 IEEE Students' Conf. Electr. Electron. Comput. Sci. SCEECS 2014*, 2014, doi: 10.1109/SCEECS.2014.6804531.
- [30] W. Lim, H. K. Torres, and C. M. Oppus, "An agricultural telemetry system implemented using an Arduino-Android interface," *2014 Int. Conf. Humanoid, Nanotechnology, Inf. Technol. Commun. Control. Environ. Manag. HNICEM 2014 - 7th HNICEM 2014 Jt. with 6th Int. Symp. Comput. Intell. Intell.*, no. November, 2014, doi: 10.1109/HNICEM.2014.7016199.
- [31] L. Muñoz, "Diseño de un medidor de consumo de energia electrica en redes industriales domesticas, basado en la adquisicion y digitalizacion de señales para el monitoreo y analisis de la calidad de energia," p. 104, 2014.
- [32] D. E. Nieto, "Diseño y construcción de un equipo de medida de energía eléctrica de bajo coste," 2015.
- [33] I. F. Rincón Roa, W. G. Cadena Téllez, and M. F. Flórez Franco, "Diseño y construcción de un analizador de redes eléctricas para sistemas trifásicos de bajo voltaje," *I+D Rev. Investig.*, vol. 6, no. 2, pp. 57–75, 2015, doi: 10.33304/revinv.v06n2-2015005.
- [34] P. H. Kulkarni, P. D. Kute, and V. N. More, "IoT based data processing for automated industrial meter reader using Raspberry Pi," *2016 Int. Conf. Internet Things Appl. IOTA 2016*, pp. 107–111, 2016, doi: 10.1109/IOTA.2016.7562704.

- [35] G. G. Valbuena, D. Alexis, and P. Agudelo, "GuacanemeValbuenaGerardo2016," 2016.
- [36] E. A. Quintero Salazar, "Medidor electrónico interactivo de consumo de energía eléctrica para uso residencial," *Prospectiva*, vol. 14, no. 1, p. 60, 2016, doi: 10.15665/rp.v14i1.639.
- [37] R. F. Eggea, M. Ferreira, A. R. Aoki, and R. J. Riella, "Energy management including photovoltaic panel and energy storage for Smart Grids through mobile application," *2015 IEEE PES Innov. Smart Grid Technol. Lat. Am. ISGT LATAM 2015*, pp. 177–181, 2016, doi: 10.1109/ISGT-LA.2015.7381149.
- [38] I. Yahyaoui, F. Tadeo, and M. V. Segatto, "Control strategy for small-scale photovoltaic systems connected to single-phase grids," *IREC 2016 - 7th Int. Renew. Energy Congr.*, 2016, doi: 10.1109/IREC.2016.7478889.
- [39] T. Tavade and P. Nasikkar, "Raspberry Pi: Data logging IOT device," *Int. Conf. Power Embed. Drive Control. ICPEDC 2017*, pp. 275–279, 2017, doi: 10.1109/ICPEDC.2017.8081100.
- [40] J. Flores Figueroa, M. A. Romero Ochoa, M. Soto Rodriguez, and M. G. Vazquez Palma, "Diseño de un medidor de energía eléctrica bidireccional y monitoreado mediante aplicación móvil / Design of a two-way electric power meter monitoring with a mobile application," *RECI Rev. Iberoam. las Ciencias Comput. e Informática*, vol. 7, no. 14, pp. 1–30, 2018, doi: 10.23913/reci.v7i14.84.
- [41] M. L. R. Chandra, B. V. Kumar, and B. Sureshbabu, "IoT enabled home with smart security," *2017 Int. Conf. Energy, Commun. Data Anal. Soft Comput. ICECDS 2017*, pp. 1193–1197, 2018, doi: 10.1109/ICECDS.2017.8389630.
- [42] A. A. Rodriguez Aya, J. A. Figueredo Luna, and J. A. Chica Garcia, "Sistema de control y telemetría de datos mediante una aplicación móvil en Android basado en IoT para el monitoreo de datos," *Espacios*, vol. 39, no. 22, p. 30, 2018, [Online]. Available: <http://www.revistaespacios.com/a18v39n22/18392230.html>.
- [43] M. Hernandez, "Corriente Y Potencia Eléctrica," *Subsecr. Educ. Media Super.*, vol. Primera ed, pp. 106–108, 2016, [Online]. Available: <http://www.sems.gob.mx/work/models/sems/Resource/12235/1/images/voltaje.pdf>.

APÉNDICES

Apéndice 1. Algoritmo de microcontrolador para lectura y carga de datos.

```
Proyectotesissistemaintegrado $
// Librerías requeridas para utilizar Firebase
#include <common.h>
#include <FirebaseFS.h>
#include <Firebase_ESP_Client.h>
#include <Utils.h>

#include <ESP8266WiFi.h>
#include <Arduino.h>
#include <PZEM004Tv30.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <Wire.h>
#include <BH1750.h>

/* 1. definir credenciales de Wi-Fi */
#define WIFI_SSID "INFINITUM1109_2.4"
#define WIFI_PASSWORD "S873wLIZKe"

/* 2. Definir el host y la clave api de firebase */
#define FIREBASE_HOST "proyectotesis-899bb.firebaseio.com"
#define API_KEY "AIzaSyA4Yvta_3dB1oElUU8D3xW2dZgXMMLIUNQ"

/* 3. Definir el email y la contraseña del usuario */
#define USER_EMAIL "lctjorgel@gmail.com"
#define USER_PASSWORD "Taric123"

//Definir un objeto de datos
FirebaseData fbdo;

FirebaseAuth auth;
FirebaseConfig config;

String path = "/statusmedidor";

// Declaración de variables para uso de sensores
|
// Variables de sensores de energia
float voltage, current, power, frequency, pf, VoltagePF, CurrentPF, PowerPF, EnergyPF, FrequencyPF, PFPF;
float kWh = 0.0;
char kWhs[6];
char kWhsPF[6];
const char * pgas;

// Variable de sensor de temperatura
float temperatureC;
const int oneWireBus = D2;
OneWire oneWire(oneWireBus);
DallasTemperature sensors(&oneWire);

// Variable de sensor de presencia de gas LP
const int MQ_PIN = 16;

// Variable de sensor de caudal de agua
#define SENSOR D1
long currentMillis = 0;
long previousMillis = 0;
int interval = 1000;
boolean ledState = LOW;
```

```

float calibrationFactor = 3.7420;
volatile byte pulseCount;
byte pulse1Sec = 0;
float flowRate;
unsigned long flowMilliLitres;
unsigned int totalMilliLitres;
float flowLitres;
float totalLitres;

//Variable de sensor de iluminación
BH1750 lightMeter;
uint16_t lux;

// Declaración de puertos seriales para lectura de sensores de energía
PZEM004Tv30 pzem(D3, D4); // (RX,TX) connect to TX,RX of PZEM
PZEM004Tv30 pzem2(D5, D6); // (RX,TX) connect to TX,RX of PZEM

void IRAM_ATTR pulseCounter()
{
    pulseCount++;
}

void setup() {

    Serial.begin(115200); // Iniciamos serial a una velocidad de 9600 Baudios
    WiFi.mode(WIFI_OFF);
    delay(1000);
    WiFi.begin(WIFI_SSID, WIFI_PASSWORD); // Conexion de microcontrolador con WiFi
    Serial.print("connecting");
    while (WiFi.status() != WL_CONNECTED) {
        Serial.print(".");
        delay(500);
    }
    Serial.println();
    Serial.print("connected: "); // Mensaje de conexion exitoso
    Serial.println(WiFi.localIP()); // Si la conexion WiFi se realizo nos imprime la direccion IP

    /* asignar el host del proyecto */
    config.host = FIREBASE_HOST;
    config.api_key = API_KEY;

    /* Assign the user sign in credentials */
    auth.user.email = USER_EMAIL;
    auth.user.password = USER_PASSWORD;

    /* Iniciamos la conexion con la base de datos*/
    Firebase.begin(&config, &auth);
    Firebase.reconnectWiFi(true);

    // configuracion de sensor de temperatura
    pinMode(D2, INPUT_PULLUP);
    // configuracion de sensor de agua
    pinMode(SENSOR, INPUT_PULLUP);
    pulseCount = 0;
    flowRate = 0.0;
    flowMilliLitres = 0;
    previousMillis = 0;

```

```

    attachInterrupt(digitalPinToInterrupt(SENSOR), pulseCounter, FALLING);

    // configuracion de sensor de iluminacion
    Wire.begin(D7,D8);
    lightMeter.begin();
    //pzem.resetEnergy();
}

void loop() {
    LecturaAgua();
    LecturaVoltaje1();
    LecturaTemp();
    LecturaPGAS();
    dtostrf(kWh, 6, 4, kWhs);
    dtostrf(EnergyPF, 6, 4, kWhsPF);
    lux = lightMeter.readLightLevel();
    /* LecturaPreGas();*/
    Tdatos();
    delay(100);
}

void LecturaAgua(){
    currentMillis = millis();
    if (currentMillis - previousMillis > interval)
    {

        pulse1Sec = pulseCounter;
        pulseCount = 0;
        flowRate = ((1000.0 / (millis() - previousMillis)) * pulse1Sec) / calibrationFactor;
        previousMillis = millis();
        flowMilliLitres = (flowRate / 60) * 1000;
        flowLitres = (flowRate / 60);
        totalMilliLitres += flowMilliLitres;
        totalLitres += flowLitres;
        Serial.print("Flow rate: ");
        Serial.print(float(flowRate));
        Serial.print("L/min");
        Serial.print("\t");
        Serial.print("Output Liquid Quantity: ");
        Serial.print(totalMilliLitres);
        Serial.print("mL / ");
        Serial.print(totalLitres);
        Serial.println("L");
    }
}

void LecturaPGAS() {
    bool state= digitalRead(MQ_PIN);
    if (!state)
    {
        pgas = "Cuidado! gas LP detectado";
    }
    else
    {
        pgas = "No se detecta gas LP";
    }
}

```

```

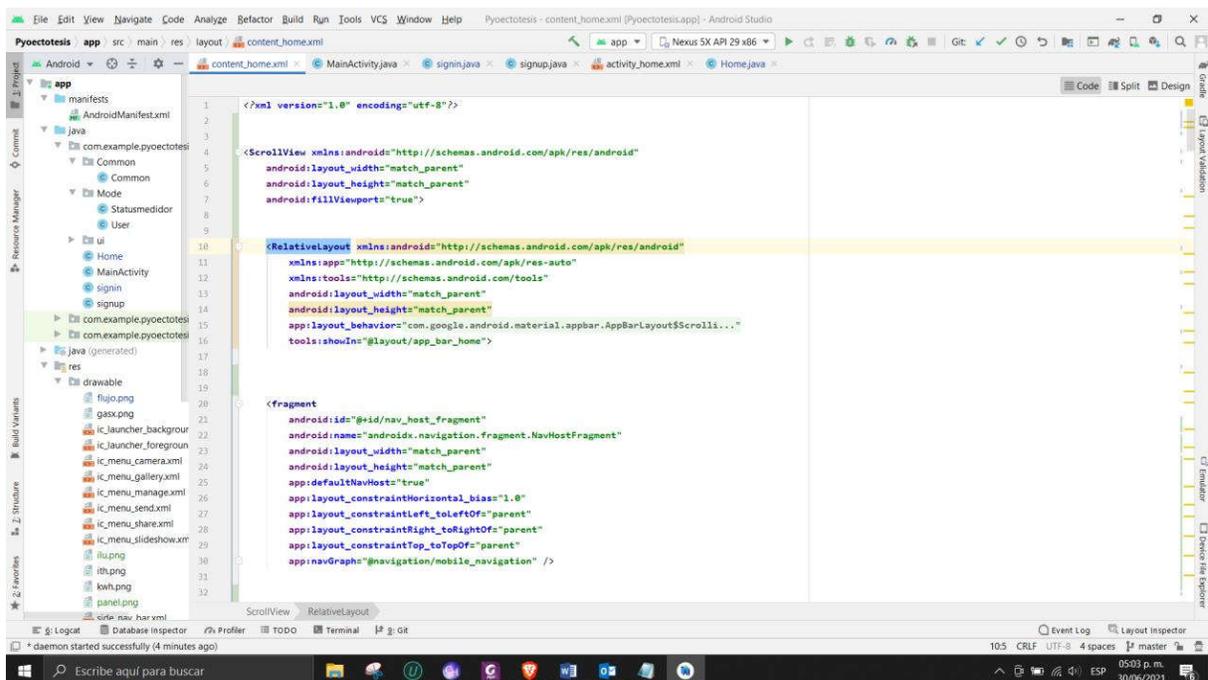
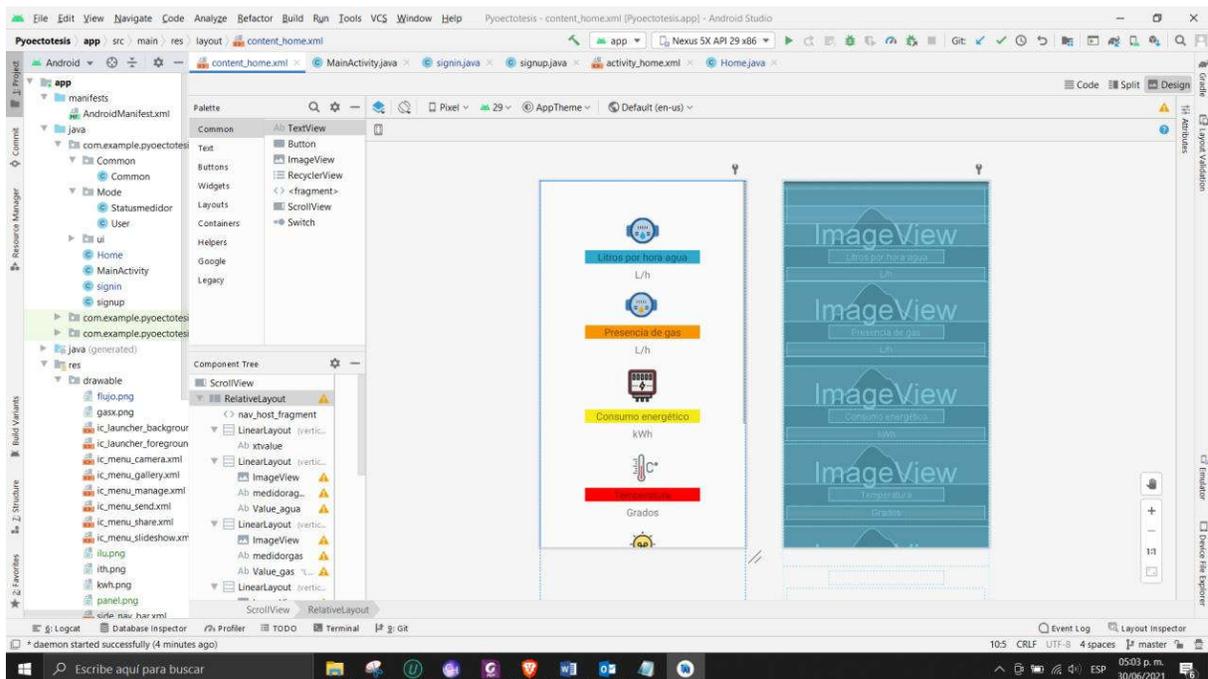
void Lecturatemper() {
    sensors.requestTemperatures();
    temperatureC = sensors.getTempCByIndex(0);
}
void LecturaVoltaje1() {
    // Lectura de sensor de energía
    voltage = pzem.voltage();
    current = pzem.current();
    power = pzem.power();
    kWh = pzem.energy();
    frequency = pzem.frequency();
    pf = pzem.pf();

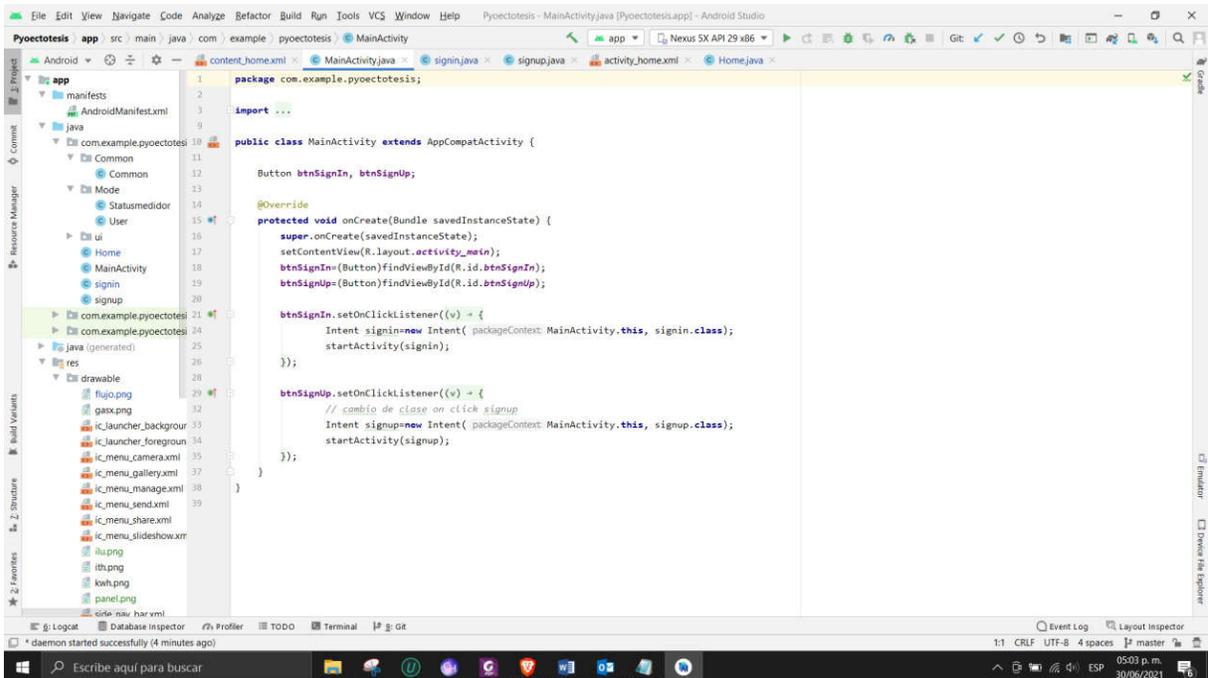
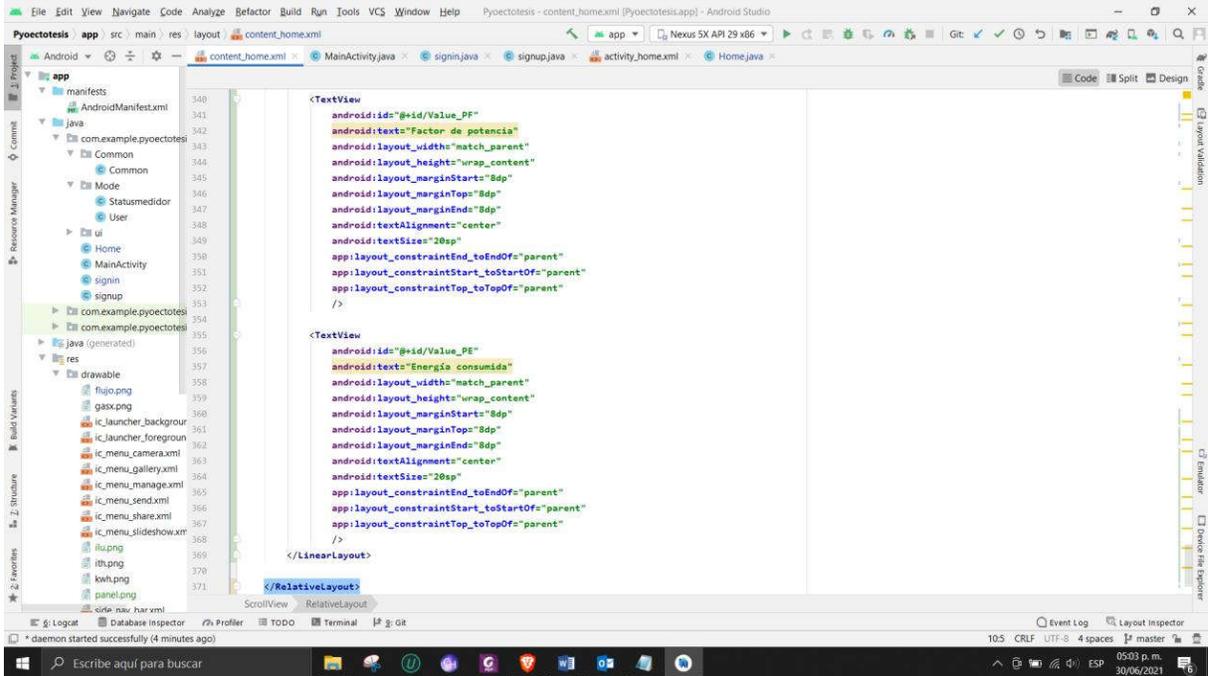
    // Lectura de sensor de energía para panel fotovoltaico
    VoltagePF = pzem2.voltage();
    CurrentPF = pzem2.current();
    PowerPF = pzem2.power();
    EnergyPF = pzem2.energy();
    FrequencyPF = pzem2.frequency();
    PFPF = pzem2.pf();
}

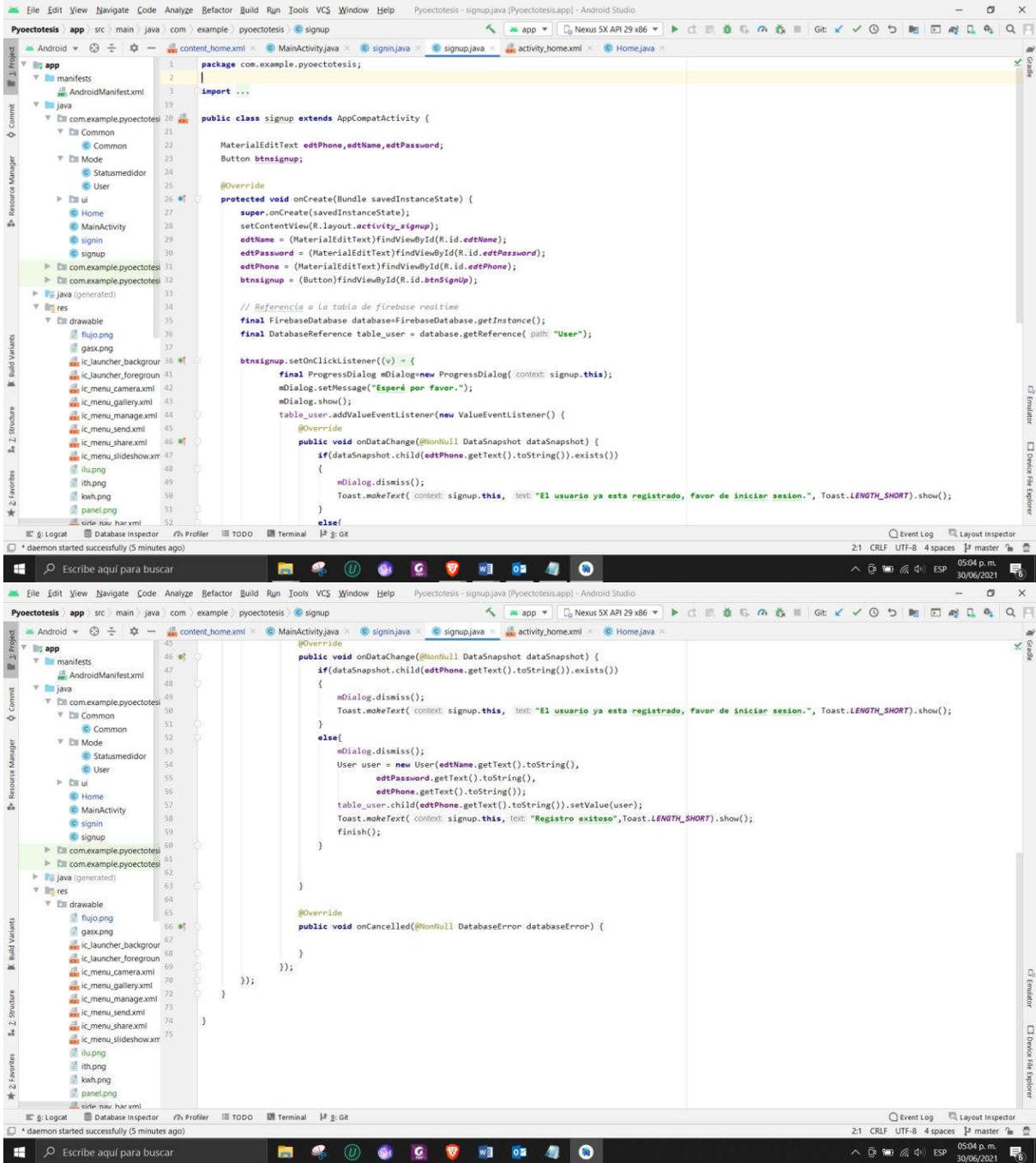
void Tdatos() {
    String Path = path + "/medidor_kwh/Value";
    Firebase.RTDB.setString(&fbdo, Path.c_str(), String(kWhs)+" kWh");
    delay(10);
    Path = path + "/medidor_t/Value";
    Firebase.RTDB.setString(&fbdo, Path.c_str(), String(temperatureC)+" °C");
    delay(10);
    Path = path + "/medidor_PC/Value";
    Firebase.RTDB.setFloat(&fbdo, Path.c_str(), CurrentPF);
    delay(10);
    Path = path + "/medidor_PF/Value";
    Firebase.RTDB.setFloat(&fbdo, Path.c_str(), FrequencyPF);
    delay(10);
    Path = path + "/medidor_PV/Value";
    Firebase.RTDB.setFloat(&fbdo, Path.c_str(), VoltagePF);
    delay(10);
    Path = path + "/medidor_PE/Value";
    Firebase.RTDB.setString(&fbdo, Path.c_str(), kWhsPF);
    delay(10);
    Path = path + "/medidor_gas/Value";
    Firebase.RTDB.setString(&fbdo, Path.c_str(), pgas);
    delay(10);
    Path = path + "/medidor_agua/Value";
    Firebase.RTDB.setFloat(&fbdo, Path.c_str(), totalLitres);
    delay(10);
    Path = path + "/medidor_i/Value";
    Firebase.RTDB.setString(&fbdo, Path.c_str(), String(lux)+" Lx");
}
}

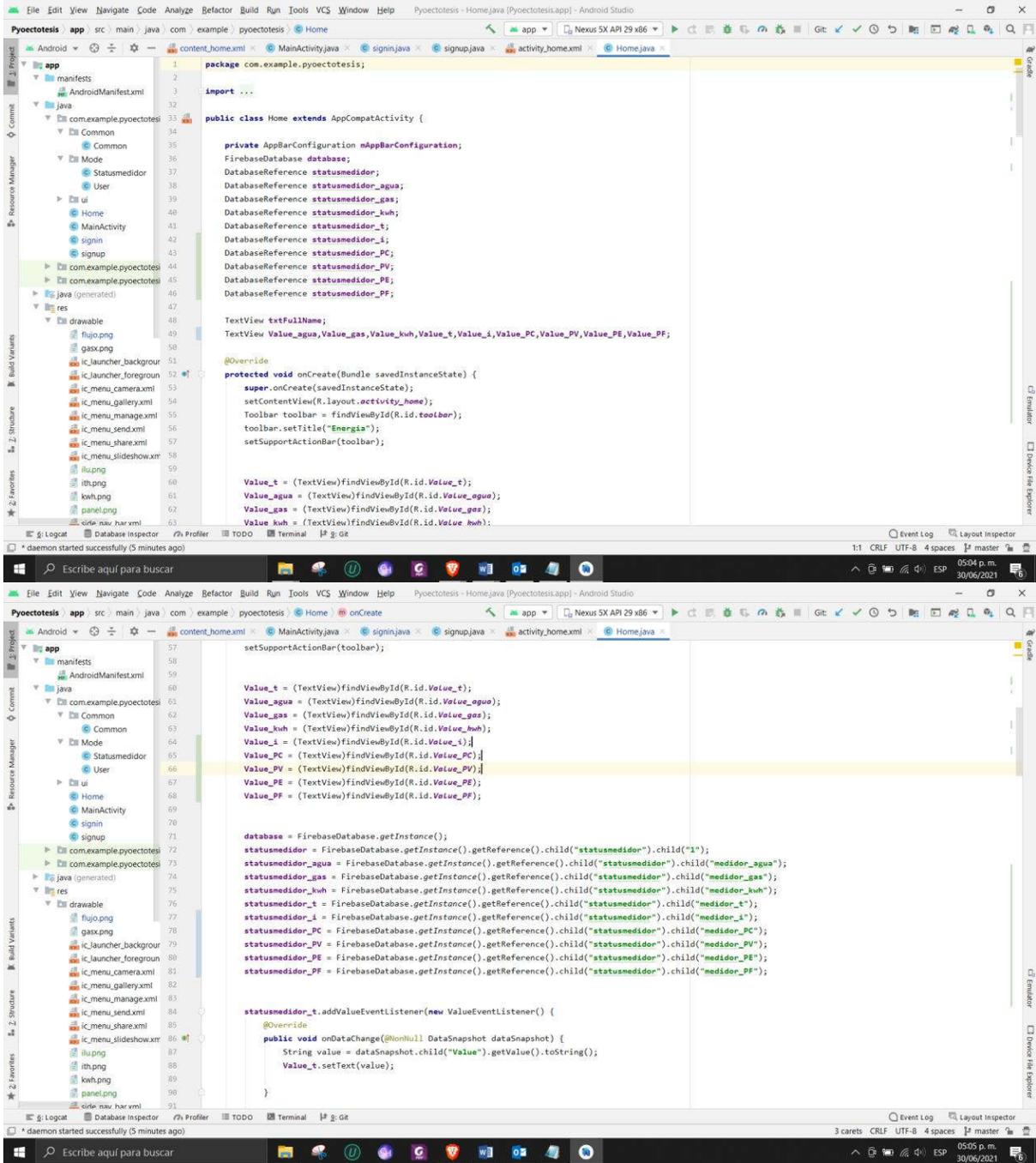
```

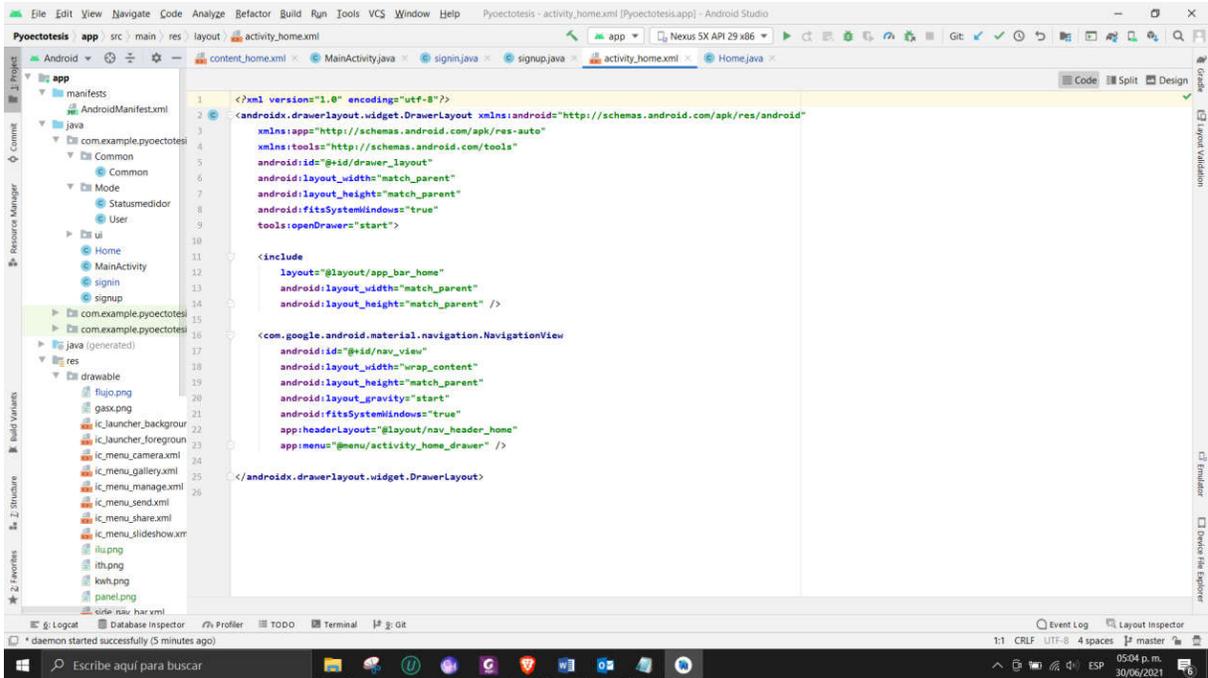
Apéndice 2. Código de aplicación móvil.











Apéndice 3. Manual de operaciones.

**Manual de operación de prototipo para sistema de
telemetría.**

Instituto Tecnológico de Hermosillo

Departamento de estudios de posgrado e investigación.

Maestría en ingeniería electrónica.

28/03/2021

Hermosillo, Sonora, México



Descripción de funcionamiento.

El sistema de telemetría permite a un usuario enlazar una aplicación móvil para el sistema operativo Android, con un microcontrolador, programado para realizar lectura de 6 distintos sensores, con los que podremos conocer el gasto de energía eléctrica en kWh, el gasto de agua en Litros, la presencia de gas LP, la iluminación, la temperatura y la energía generada por un panel fotovoltaico.

El prototipo está programado para trabajar de manera continua, siempre y cuando se cuente con alimentación de energía ideal es decir que no sufra de interrupciones totales o parciales y sea de voltaje constante, por esto se consideró una fuente de 5V que mantendrá energizado el dispositivo.

Requerimientos y configuración de sistema.

Como requerimientos para realizar la instalación del sistema de manera autónoma, se requieren conocimientos de instalaciones eléctricas para poder manipular el cableado de forma segura, ya que la manipulación de elementos eléctricos siempre es peligrosa, se requiere conocer la medida de la rosca de la llave del suministro de agua para que el sensor quede implementado sin pérdidas de líquido y presión, lo que mejorara la precisión de la medición una vez calibrado. En cuanto al sensor de energía requerido para las mediciones del panel fotovoltaico, será necesario instalarlo en paralelo con la salida del microinversor o el dispositivo de la arquitectura que convierta la corriente continua en corriente alterna.

Para la configuración del sistema únicamente es requerido conocer los parámetros de acceso para la red Wi-Fi del hogar, en este caso son dos parámetros, el nombre de la

red también llamado SSID y la contraseña de la red. Es necesario para el prototipo contar con conexión a internet, para poder cargar los datos, de igual forma si no cuenta con internet disponible en ese momento el microcontrolador seguirá tomando lecturas de los sensores para actualizar la base de datos cuando haya disponibilidad de internet.

Instalación de sistema.

Los sensores que componen el sistema son 6, dos para medición de parámetros eléctricos, uno para medición de flujo de agua, uno para medición de presencia de gas, uno para iluminación y el último para medición de temperatura.

Cada sensor debe ser instalado en un punto específico de la residencia, instalándose uno en la acometida de la instalación eléctrica para registrar los valores de corriente y voltaje, para lograr obtener la energía consumida total en la residencia, el segundo sensor de energía eléctrica se instalaría en la salida de la infraestructura fotovoltaica, el sensor de flujo de agua se instala en el circuito hidráulico donde se desee conocer el consumo de agua, o directamente en la llave de paso ya que esto registraría el total de litros de agua que pasen a la residencia, el sensor de presencia de gas se instala en la instalación de gas LP, para registrar fugas será necesario instalar el sensor en un punto medio donde acapare la mayoría de la canalización así como el propio tanque de gas y los elementos que consuman este recurso, el sensor de temperatura e iluminación se instalaran en donde el usuario desee mantener controlado estos parámetros, puede ser en el área designada para trabajar en el hogar como una

especie de taller para el caso de la iluminación, y en la recámara para el caso del sensor de temperatura.

La canalización se dificulta en la instalación de estos sensores ya que se cuenta únicamente con un microcontrolador, por esta razón para el proyecto de tesis se decidió probar estos sensores con plantas, para comprobar el funcionamiento de los sensores.

Puesta en marcha.

Una vez que el microcontrolador haya sido configurado, e instalado en la residencia con los sensores a utilizar se alimentara la fuente de voltaje externa de 5v, misma que alimentara a los sensores para reducir la carga energética del microcontrolador, al momento de ser energizado la fuente el microcontrolador encenderá su chip interno, lo que empezara a correr el algoritmo programado en la memoria del dispositivo, y configurara todos los parámetros necesarios para poderse conectar a la red Wi-Fi y empezar a realizar lecturas de los distintos sensores, no es necesario realizar pasos extra para que estas mediciones comiencen a subirse a la base de datos, y posteriormente poder visualizarlas en la aplicación móvil.

Mantenimiento.

En la búsqueda en la literatura de las hojas de datos de los sensores utilizados en este proyecto de tesis no se encontró nada referente a la vida útil de los mismos, se recomienda cotejar las mediciones realizadas con los recibos de facturación de las empresas que brindan el servicio para las mediciones de consumo eléctrico y consumo

hídrico, con el fin de poder asegurarse que las mediciones sean fiables, el microcontrolador puede ser fácilmente reemplazado ya que el algoritmo de programación toma menos de 2 minutos en ser programado en la memoria interna del microcontrolador, y al tener un bajo costo lo hace perfecto para refacciones.