



**UNIVERSIDAD TECNOLÓGICA CENTROAMERICANA**

**FACULTAD DE INGENIERÍA**

**PROYECTO FASE I**

**SISTEMA DE MONITOREO DE CONSUMO DE ENERGÍA ELÉCTRICA EN  
TIEMPO REAL**

**PREVIO A LA OBTENCIÓN DEL TÍTULO "INGENIERO EN  
TELECOMUNICACIONES"  
PRESENTADO POR:**

**21221069 CRISTIAN ANTHONY COUSIN HANDAL**

**ASESOR: ING. ANA CRISTINA REYES LEIVA**

**CAMPUS SAN PEDRO SULA; ABRIL, 2020**

## **RESUMEN EJECUTIVO**

Este informe define como objetivo conceptualizar un sistema de monitoreo de energía eléctrico que almacene información y provea métricas sobre la energía consumida en la carga en una interfaz accesible desde un ordenador con acceso al navegador. Estas métricas tienen el objetivo de informar al usuario sobre el costo incurrido en tiempo real de la energía eléctrica consumida.

La funcionalidad del anterior se da mediante un sensor de corriente que es instalado en la placa programable Arduino, el cual se encarga de procesar y enviar los datos al servidor que corre una aplicación desarrollada con php y es la encargada de almacenar la información y mostrar la interfaz con las métricas, sobre la carga, al usuario.

Los resultados muestran que desarrollar este sistema de monitoreo puede ser una solución casera para medir cargas de potencia moderada de un manera económica. También se muestra la utilidad que brindan las plataformas de desarrollo de código abierto como Arduino y Laravel, con las que se puede trabajar sin ningun costo o con uno bastante accesible en el caso del Arduino.

## **ABSTRACT**

The main purpose of this study is to conceptualize an electrical energy monitoring system that stores information and provides metrics on the energy consumed by a load in an interface easily accessible from a computer with access to the browser. These metrics are intended to inform the user about the real time cost of the electrical energy consumed by the load.

The functionality of the described system is mainly given by a current sensor that is installed on the Arduino programmable board, which is responsible for processing and sending the data to the server that runs an application developed with php and is responsible for storing the information and displaying the interface with metrics for the user to see.

The results show that developing this monitoring system can be a homemade solution to measure moderate power loads in an economical way. Also shown is the utility provided by open source development platforms such as Arduino and Laravel, with which you can work without any cost or with one that is quite accessible in the case of the Arduino.

# ÍNDICE DE CONTENIDO

I. INTRODUCCIÓN .....	1
II. PLANTEAMIENTO DEL PROBLEMA .....	2
2.1 PRECEDENTES DEL PROBLEMA.....	2
2.2 DEFINICIÓN DEL PROBLEMA .....	2
2.3 JUSTIFICACIÓN .....	2
2.4 PREGUNTAS DE INVESTIGACIÓN .....	3
2.5 OBJETIVOS.....	3
2.5.1 OBJETIVO GENERAL .....	3
2.5.2 OBJETIVOS ESPECÍFICOS .....	3
III. MARCO TEÓRICO .....	4
3.1 ENERGÍA ELÉCTRICA.....	4
3.1.1 DEFINICIÓN DE ENERGÍA ELÉCTRICA .....	4
3.1.2. MEDICIÓN DE ENERGÍA ELÉCTRICA .....	4
3.2 EL WATTHORÍMETRO .....	4
3.3 LA DEMANDA DE ENERGÍA ELÉCTRICA .....	6
3.4 DEFINICIÓN DEL MICROCONTROLADOR .....	6
3.5 EL ARDUINO .....	7
3.5.1 BENEFICIOS DE UTILIZAR EL ARDUINO .....	7
3.5.2 LA PLACA ARDUINO UNO .....	8
3.5.3 EL ARDUINO ETHERNET SHIELD .....	8
3.5.4 PROGRAMANDO EN EL ARDUINO .....	10
3.6 LA RASPBERRY PI .....	10
3.6.1 PLACA RASPBERRY PI .....	10
3.6.3 SISTEMA OPERATIVO DEL RASPBERRY PI .....	11
3.6.4 PYTHON EN LA RASPBERRY PI.....	11
3.7 LENGUAJE PHP .....	11
3.7.1 USOS PRIMARIOS DE PHP .....	11
3.7.2 MYSQL .....	12
3.7.3 PHP Y MYSQL .....	12
3.7.4 FRAMEWORKS CON PHP .....	12
3.8 LARAVEL.....	13
3.8.1 REQUERIMIENTOS PARA LA INSTALACIÓN .....	13
3.8.2 INSTALACIÓN.....	13

3.8.3 DESARROLLANDO APLICACIONES CON LARAVEL .....	14
IV. METODOLOGÍA .....	15
4.1 ENFOQUE.....	15
4.2 VARIABLES DE INVESTIGACIÓN .....	15
4.3 TÉCNICAS E INSTRUMENTOS APLICADOS .....	15
4.4 METODOLOGÍA DE ESTUDIO .....	16
4.4.1 SOFTWARE DE PROCESAMIENTO DE DATOS.....	18
4.4.2 LAS MÉTRICAS CLAVE.....	18
4.4.3 SOFTWARE EN EL SERVIDOR Y LA INTERFAZ GRÁFICA.....	18
4.5 CRONOGRAMA DE ACTIVIDADES .....	19
V. RESULTADOS Y ANÁLISIS .....	20
5.1 MEDICIÓN DEL CONSUMO DE ENERGÍA .....	20
5.2 ESTABLECER LA COMUNICACIÓN CON EL SERVIDOR .....	21
5.3 ENUMERAR LAS MÉTRICAS CLAVE .....	22
VI. CONCLUSIONES .....	25
VII. RECOMENDACIONES .....	26
BIBLIOGRAFÍA .....	27

## ÍNDICE DE ILUSTRACIONES

Ilustración 1 - Diagrama básico de un wathhorímetro .....	5
Ilustración 2 - Vista frontal de la placa Arduino UNO .....	8
Ilustración 3 - Vista frontal de la placa Arduino Ethernet Shield .....	9
Ilustración 4 - Montaje del Arduino Ethernet Shield sobre el Arduino UNO.....	9
Ilustración 5 - Vista frontal de la placa Raspberry Pi 4 Modelo B .....	10
Ilustración 6 - ACS712 y ACS714 .....	16
Ilustración 7 - Diagrama de instalación Arduino UNO, Ethernet Shield y ACS712 .....	17
Ilustración 8 - Diagrama resumen del sistema .....	18
Ilustración 9 - Vista de la interfaz del usuario .....	24

## ÍNDICE DE ECUACIONES

Ecuación 1 .....	4
Ecuación 2 – Voltaje y corriente medido por el ACS712 .....	17

## **I. INTRODUCCIÓN**

Los sistemas de monitoreo de energía eléctrica en tiempo real tienen como función principal cuantificar el consumo de energía utilizando un arreglo de medidores electrónicos que se conectan a una red y envían los datos de sus mediciones a servidores que guardan estos en una base de datos que luego, por medio de software especializado, son analizados para proporcionar información clave que puede ser utilizada para optimizar el circuito. Estos datos también proporcionan información eléctrica sobre el circuito en tiempo real accesible desde la nube, lo cual agrega flexibilidad y facilidad de uso al sistema.

Con el aumento del consumo de energía eléctrica de las personas se abren puertas para soluciones innovadoras. En la última década, el Internet ha crecido hasta convertirse en la herramienta más importante de los seres humanos, este permite conectar sistemas con dispositivos móviles, servidores y otros ordenadores para desarrollar estructuras de comunicación complejas de alta competencia.

El propósito u objetivo principal es estudiar y lograr definir conceptos para medir la energía eléctrica en tiempo real de dispositivos electrónicos. Un sistema como este simplificará la forma en la que se monitorean los circuitos eléctricos en casas particulares u oficinas de la mediana y pequeña empresa.

Esta investigación permitirá el estudio de sistemas que podría traer grandes ahorros e información sobre el consumo de energía que tienen las personas, también se justifica este estudio para proporcionar información a aquellos que busquen indagar en el tema de sincronizar dispositivos electrónicos particulares con sistemas de red y software de análisis en tiempo real.

Se definirá el problema, justificándolo, desarrollando preguntas que se deben responder y definiendo objetivos, luego se estudiará la teoría que enmarca el tema. Esto permitirá desarrollar una metodología para el estudio del sistema en la cual se detallará el proceso. Por último se definirá la conclusión y los resultados.

## **II. PLANTEAMIENTO DEL PROBLEMA**

La creciente necesidad de energía eléctrica que tienen las personas causa problemas bastante notables, como el aumento del costo de vida, daños sustanciales al medio ambiente, entre otros. El problema es más visible en países pobres ya que en estos el costo de la energía tiende a ser mucho mayor al de los desarrollados.

### **2.1 PRECEDENTES DEL PROBLEMA**

El creciente costo de la energía eléctrica es causado originalmente por varias razones como el crecimiento de la población global y especialmente la falta de visión de los funcionarios públicos para preparar la infraestructura necesaria que reduciría los costos de la misma.

Otro aspecto a considerar es que muchas personas en países en desarrollo no cuentan con una educación suficiente para comprender que la optimización de los sistemas eléctricos en sus hogares o lugares de trabajo puede lograr generarles ahorros importantes en sus vidas.

### **2.2 DEFINICIÓN DEL PROBLEMA**

El problema objeto de este informe es el alto costo de energía eléctrica que tienen las personas por la falta de optimización en sus sistemas eléctricos.

### **2.3 JUSTIFICACIÓN**

Medir el consumo de energía eléctrica en un sistema conectado al internet puede traer beneficios a corto plazo como el monitoreo a tiempo real para la prevención de picos en el consumo, la previsibilidad de los costos mensuales de la energía, entre otros y beneficios a largo plazo como la generación de ahorros gracias a la mejor administración de la energía.



## **2.4 PREGUNTAS DE INVESTIGACIÓN**

1. ¿Cómo se puede medir el consumo de energía eléctrica en tiempo real de un circuito eléctrico?
2. ¿Cómo se puede almacenar la información de consumo de energía de un circuito eléctrico en una base de datos utilizando un Arduino o Raspberry Pi?
3. ¿Qué métricas importantes se pueden definir al analizar la información de consumo de energía de un circuito eléctrico?

## **2.5 OBJETIVOS**

En los siguientes objetivos se define puntualmente el propósito de este informe, acotando el enfoque de nuestra metodología, basándonos en las preguntas de investigación previamente definidas.

### **2.5.1 OBJETIVO GENERAL**

Conceptualizar un sistema de monitoreo de energía eléctrica en tiempo real utilizando dispositivos electrónicos o virtuales para la medición de circuitos eléctricos y otros dispositivos que permiten conectar estos al internet como el Arduino o Raspberry Pi.

### **2.5.2 OBJETIVOS ESPECÍFICOS**

1. Analizar formas o herramientas con las que se puede medir el consumo de energía eléctrica en tiempo real de un circuito eléctrico.
2. Determinar cómo una carga puede conectarse al internet y establecer una comunicación con un servidor utilizando dispositivos electrónicos como el Arduino o Raspberry Pi.
3. Enumerar métricas que provean información importante sobre el consumo de energía eléctrica de un dispositivo.

### **III. MARCO TEÓRICO**

A continuación se definen las bases teóricas de los temas necesarios para el entendimiento y sustento de este proyecto. En esta sección se resume la información acerca de la energía eléctrica, formas de medir su consumo, los problemas de la energía en Honduras, herramientas para medir la energía eléctrica y el software necesario para interpretar y mostrar la información recopilada.

#### **3.1 ENERGÍA ELÉCTRICA**

##### **3.1.1 DEFINICIÓN DE ENERGÍA ELÉCTRICA**

La energía eléctrica es aquella que se produce por la diferencia de potencial eléctrico entre dos puntos y habilita que se establezca una corriente eléctrica entre los mismos cuando hay un medio conductor. Esta energía puede ser almacenada y transformada en otras formas de energía según la aplicación que se busque (Young et al., 2009).

##### **3.1.2. MEDICIÓN DE ENERGÍA ELÉCTRICA**

La energía eléctrica entregada a una carga entre dos momentos se define como:

$$E(t_1) = \int_0^{t_1} p(t) dt = \int_0^{t_1} u(t) I(t) dt$$

Ecuación 1

(Iwansson et al., 1999)

#### **3.2 EL WATTHORÍMETRO**

El dispositivo de medición de energía más común es el wathhorímetro. Este medidor es un instrumento barato y exacto que puede trabajar correctamente durante largos períodos de tiempo con poco mantenimiento. Registra el consumo de energía de una carga mediante el conteo de las vueltas de un disco giratorio de aluminio. El giro del disco es causado por la potencia que pasa a través del medidor (Wolf & Smith, 1992).

Su funcionamiento es semejante al de un motor de inducción. La bobina de corriente y el núcleo metálico sobre el que está enrollada originan un campo

magnético. La bobina de voltaje y el núcleo metálico originan un campo magnético adicional. En el disco de aluminio se originan corrientes parásitas debido a la variación del campo magnético de la bobina de corriente (Wolf & Smith, 1992).

Esas corrientes parásitas interactúan con el campo magnético de la bobina de voltaje y se ejerce un par en el disco. Como no hay resortes de restricción, el disco continúa girando mientras se alimenta corriente a través del medidor. El par en el disco es proporcional al producto del voltaje y la corriente. Así, mientras mayor sea la corriente que pase por el medidor, más rápido girará el disco (Wolf & Smith, 1992).

El número de vueltas es una medida de la energía consumida por la carga. El eje en el que está montado el disco está unido a un mecanismo de engranes y éste con un conjunto de indicadores con carátulas semejantes a las de un reloj. Al leer los valores en esas carátulas en diferentes momentos se puede calcular cuánta energía pasó a través del medidor durante el intervalo de tiempo entre lecturas. (Wolf & Smith, 1992).

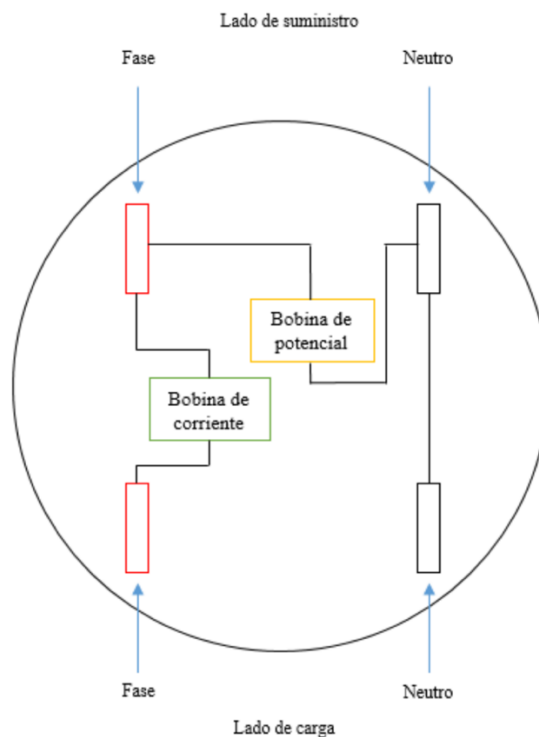


Ilustración 1 - Diagrama básico de un wathorímetro

### **3.3 LA DEMANDA DE ENERGÍA ELÉCTRICA**

Se espera que la demanda de electricidad de los usuarios aumente al 30% para 2040 (que en 2017), como se informó en (International Energy Agency [IEA], 2017). Esta creciente demanda se debe al crecimiento de la urbanización, la población y los electrodomésticos, que han sobrecargado la infraestructura eléctrica existente. Satisfacer esta demanda podría afectar el medio ambiente con la contaminación y el calentamiento global. En particular, el problema de la crisis energética y la contaminación ambiental causada por la combustión de CO<sub>2</sub> se ha citado como problemas críticos en el mercado energético.

La actual estructura de red sobrecargada también enfrenta problemas de confiabilidad debido a la insuficiente supervisión en tiempo real, identificación de fallas, automatización, transparencia y flexibilidad.

Con recursos energéticos distribuidos, el sistema energético se volverá más descentralizado y flexible. Sin embargo, con la infraestructura eléctrica existente, la energía de los recursos renovables no puede lograr la eficiencia total. Por ejemplo, China produce la mayor parte de la energía verde pero aún enfrenta crisis energéticas porque no puede entregar energía a su alta población (REN21, 2019).

### **3.4 DEFINICIÓN DEL MICROCONTROLADOR**

La caracterización de un microcontrolador se compone de tres bloques fundamentales: la CPU (**C**entral **P**rocessing **U**nit), la memoria, y la entrada y salida (Perez & Areny, 2007, pp. 11). La CPU es el "cerebro" del microcomputador y actúa bajo el control del programa almacenado en memoria. (Perez & Areny, 2007, pp. 12). La memoria de un microcontrolador es el lugar donde están almacenados el programa que se ejecuta y los datos o variables utilizados por el programa (Perez & Areny, 2007, pp. 55).. El acceso a los diversos dispositivos de entrada y salida se realiza a través de algunos registros de la memoria de datos, denominados registros de funciones especiales (Perez & Areny, 2007, pp. 29).

## 3.5 EL ARDUINO

Arduino es una plataforma electrónica de código abierto basada en hardware y software fácil de usar. Las placas Arduino pueden leer entradas (luz en un sensor, un dedo en un botón o un mensaje de Twitter) y convertirlo en una salida: activar un motor, encender un LED, publicar algo en línea. Puede decirle a su placa qué hacer enviando un conjunto de instrucciones al microcontrolador de la placa. Para hacerlo, utiliza el lenguaje de programación Arduino (basado en el cableado) y el software Arduino (IDE), basado en el procesamiento. (Arduino, 2020).

### 3.5.1 BENEFICIOS DE UTILIZAR EL ARDUINO

Con los años, Arduino ha sido el cerebro de miles de proyectos, desde objetos cotidianos hasta instrumentos científicos complejos. Una comunidad mundial de creadores (estudiantes, aficionados, artistas, programadores y profesionales) se ha reunido en torno a esta plataforma de código abierto, sus contribuciones se han sumado a una increíble cantidad de conocimiento accesible que puede ser de gran ayuda para principiantes y expertos por igual.

Todas las placas Arduino son completamente de código abierto, lo que permite a los usuarios construirlas de forma independiente y eventualmente adaptarlas a sus necesidades particulares. El software también es de código abierto y está creciendo a través de las contribuciones de los usuarios de todo el mundo.

Arduino también simplifica el proceso de trabajar con microcontroladores, pero ofrece alguna ventaja para maestros, estudiantes y aficionados interesados sobre otros sistemas:

**Barato:** las placas Arduino son relativamente baratas en comparación con otras plataformas de microcontroladores.

**Multiplataforma:** el software Arduino (IDE) se ejecuta en los sistemas operativos Windows, Macintosh OSX y Linux.

**Entorno de programación simple y claro:** el software Arduino (IDE) es fácil de usar para principiantes, pero lo suficientemente flexible como para que los usuarios avanzados también lo aprovechen.

Software de código abierto y extensible: el software Arduino se publica como herramientas de código abierto, disponibles para su extensión por programadores experimentados.

**Hardware de código abierto y extensible:** los planes de las placas Arduino se publican bajo una licencia Creative Commons, por lo que los diseñadores de circuitos experimentados pueden hacer su propia versión del módulo, extenderlo y mejorarlo. (Arduino, 2020).

### 3.5.2 LA PLACA ARDUINO UNO

La placa denominada UNO es la más reconocida, utilizada y documentada de toda la familia Arduino.

Arduino Uno es una placa de microcontrolador basada en ATmega328P (hoja de datos). Tiene 14 pines de entrada / salida digital (de los cuales 6 se pueden usar como salidas PWM), 6 entradas analógicas, un resonador cerámico de 16 MHz (CSTCE16M0V53-R0), una conexión USB, un conector de alimentación, un encabezado ICSP y un botón de reinicio . Contiene todo lo necesario para soportar el microcontrolador.

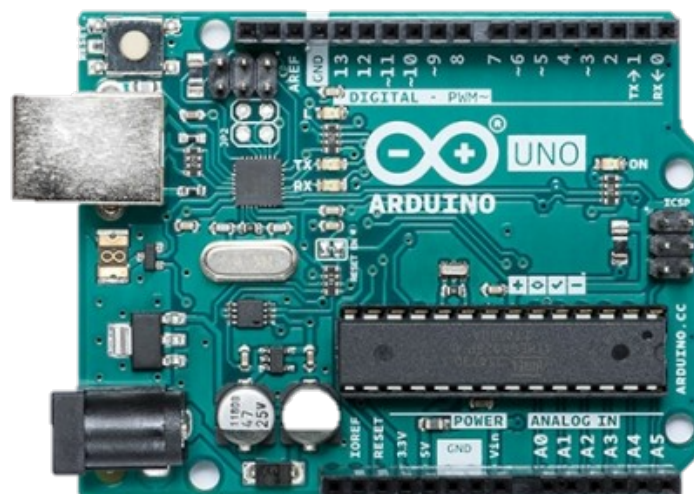


Ilustración 2 - Vista frontal de la placa Arduino UNO

### 3.5.3 EL ARDUINO ETHERNET SHIELD

El Arduino Ethernet Shield V1 permite que una placa Arduino se conecte a internet. Se basa en el chip ethernet Wiznet W5100 (hoja de datos). El Wiznet W5100 proporciona una pila de red (IP) capaz de TCP y UDP. Admite hasta cuatro conexiones de socket simultáneas. Use la biblioteca de *Ethernet* para escribir bocetos (Sketches) que se conectan a Internet utilizando el Shield. El Ethernet Shield se conecta a una placa Arduino usando encabezados largos de alambre que

se extienden a través del escudo. Esto mantiene el diseño del pin intacto y permite apilar otro escudo en la parte superior.

La revisión más reciente de la placa expone el pinout 1.0 en rev 3 de la placa Arduino UNO.

El Ethernet Shield V1 tiene una conexión estándar RJ-45, con un transformador de línea integrado y alimentación a través de Ethernet habilitada.

Hay una ranura para tarjeta micro-SD integrada, que se puede usar para almacenar archivos para servir en la red. Es compatible con todas las placas Arduino / Genuino. Se puede acceder al lector de tarjetas micro SD incorporado a través de la Biblioteca SD. Cuando se trabaja con esta biblioteca, SS está en el Pin 4. La revisión original del escudo contenía una ranura para tarjeta SD de tamaño completo; Esto no es compatible.

El escudo también incluye un controlador de reinicio, para garantizar que el módulo Ethernet W5100 se reinicie correctamente en el encendido. Las revisiones anteriores del escudo no eran compatibles con el Mega y deben reiniciarse manualmente después del encendido. (Arduino, 2020).

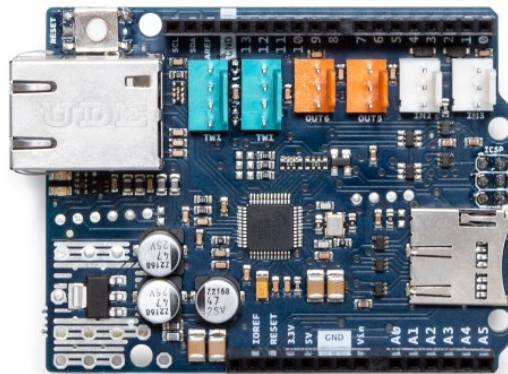


Ilustración 3 - Vista frontal de la placa Arduino Ethernet Shield

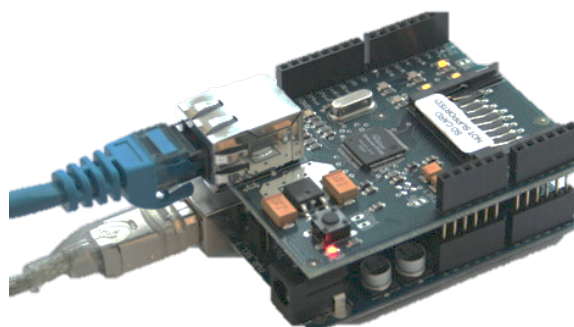


Ilustración 4 - Montaje del Arduino Ethernet Shield sobre el Arduino UNO

### 3.5.4 PROGRAMANDO EN EL ARDUINO

Las placas Arduino en su mayoría son programadas usando el IDE oficial de la organización. El Software Arduino, contiene un editor de texto para escribir código, un área de mensajes, una consola de texto, una barra de herramientas con botones para funciones comunes y una serie de menús. Se conecta al hardware Arduino para cargar programas y comunicarse con ellos. (Arduino, 2020).

## 3.6 LA RASPBERRY PI

La Raspberry Pi es una computadora de bajo costo del tamaño de una tarjeta de crédito que se conecta a un monitor de computadora o TV, y utiliza un teclado y mouse estándar. Es capaz de hacer todo lo que esperarías que hiciera una computadora de escritorio, desde navegar por Internet y reproducir videos de alta definición, hasta hacer hojas de cálculo, procesamiento de textos y jugar juegos (Raspberry Pi Foundation [RPF], 2020).

Además, Raspberry Pi tiene la capacidad de interactuar con el mundo exterior y se ha utilizado en una amplia gama de proyectos de creadores digitales, desde máquinas de música y detectores de padres hasta estaciones meteorológicas y tuiteando casas de pájaros con cámaras infrarrojas (Raspberry Pi Foundation [RPF], 2020).

### 3.6.1 PLACA RASPBERRY PI

La placa PI 4 Modelo B es la última versión del Raspberry Pi y cuenta con el procesador Broadcom BCM2711 de 64-bit a una frecuencia de 1.5GHz, hasta 4GB de LPDDR4-3200 SDRAM, tarjeta de red inalámbrica, un puerto gigabit ethernet, 2 puertos USB 3.0, puerto de 40 pines de Raspberry Pi, 2 puertos mini HDMI y una entrada de 5V DC (Upton & Halfacree, 2016).

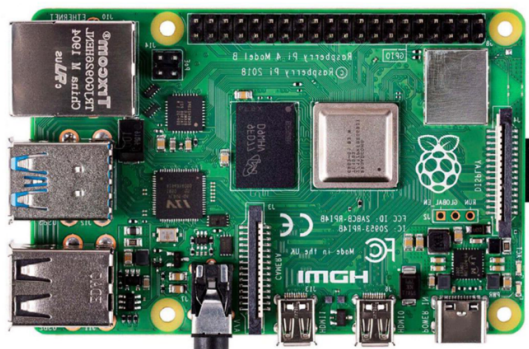


Ilustración 5 - Vista frontal de la placa Raspberry Pi 4 Modelo B



### 3.6.3 SISTEMA OPERATIVO DEL RASPBERRY PI

Raspberry Pi oficialmente soporta su propio sistema operativo creado en base a Linux y este se denomina como Raspbian. Eso no significa que la placa soporta exclusivamente este sistema pues esta puede ser utilizada con casi cualquier distribución de Linux incluyendo unas muy famosas como: Fedora, Ubuntu y Arch Linux (Robins, 2019). Hay muchos componentes que conforman una distribución moderna de Linux. Estos componentes trabajan juntos para proporcionarle todas las características modernas que espera en una computadora. Hay varios componentes clave a partir de los cuales se construye Raspbian. Estos componentes son: el gestor de arranque Raspberry Pi, el kernel de Linux, daemons, el shell, las utilidades de Shell, el servidor gráfico X.Org y el entorno de escritorio (Harrington, 2015).

### 3.6.4 PYTHON EN LA RASPBERRY PI

Python es un lenguaje interpretado que viene preinstalado en la Raspberry Pi, y es el idioma de elección para desarrollar scripts para este dispositivo (Richardson & Wallace, 2013). Es ventajoso aprender Python usando Raspberry Pi por las siguientes razones (Yamanoor & Yamanoor, 2017):

- Tiene una sintaxis muy simple y, por lo tanto, es muy fácil de entender.
- Ofrece la flexibilidad de implementar ideas como una secuencia de scripts. Esto es útil para los aficionados a implementar sus ideas.
- Hay bibliotecas de Python para el GPIO de Raspberry Pi. Esto permite una fácil interfaz de sensores / dispositivos con la Raspberry Pi.

## 3.7 LENGUAJE PHP

PHP es un simple pero potente lenguaje de programación diseñado inicialmente para crear contenido HTML (**H**yper**T**ext **M**arkup **L**anguage) (Lerdorf et al., 2006).

### 3.7.1 USOS PRIMARIOS DE PHP

PHP es predominantemente usado para tres cosas:

*Scripting en el servidor:* PHP fue diseñado originalmente para crear contenido web dinámico, y todavía es el más adecuado para esa tarea. Para generar HTML, se necesita el analizador PHP y un servidor web para enviar los documentos.

*Scripting desde la línea de comando:* PHP puede ejecutar scripts desde la línea de comandos. Puede usar los scripts de línea de comandos para tareas de administración del sistema, como copia de seguridad y análisis de registros.

*Desarrollo de interfaces gráficas para el cliente:* Usando PHP también se puede escribir aplicaciones multiplataforma con una interfaz gráfica completa (Lerdorf et al., 2006).

### 3.7.2 MySQL

MySQL es el programa de base de datos de código abierto más popular del mundo. Con MySQL ejecutándose en un servidor, puede usarse fácilmente para sistemas comerciales o sitios web basados en bases de datos. Fácil de usar y configurar, MySQL también es capaz de manejar aplicaciones industriales. Dependiendo de la computadora en la que esté instalado, MySQL puede almacenar varios terabytes de información por tabla (Jordan, 2006).

### 3.7.3 PHP Y MySQL

PHP facilita el desarrollo web, porque todo el código que se necesita está contenido en el marco del lenguaje. Si bien PHP es excelente para desarrollar la funcionalidad web, no es una base de datos. La base de datos elegida por los desarrolladores web de PHP es MySQL, que actúa como un archivador de datos procesados por PHP. MySQL automatiza las tareas más comunes relacionadas con el almacenamiento y la recuperación de información específica del usuario según los criterios proporcionados (Davis & Phillips, 2006).

### 3.7.4 FRAMEWORKS CON PHP

En los primeros días de la web dinámica, escribir una aplicación web era muy diferente de lo que es hoy. Los desarrolladores fueron responsables de escribir el código no solo para la lógica empresarial única de las aplicaciones, sino también para cada uno de los componentes que son tan comunes en los sitios, como: autenticación de usuario, validación de entrada, acceso a bases de datos, plantillas y más. Hoy los programadores tienen docenas de marcos de desarrollo (o frameworks) de aplicaciones y miles de componentes y bibliotecas de fácil acceso (Stauffer, 2017).

Los frameworks como Laravel, Symfony, Silex, Lumen y Slim, unen una colección de componentes de terceros utilizando archivos de configuración, proveedores de servicios, estructuras de directorios prescritas y programas de arranque de aplicaciones. Por lo tanto, el beneficio de usar un marco en general es que alguien ha tomado decisiones no solo sobre los componentes individuales para el desarrollador, sino también sobre cómo esos componentes deben encajar (Stauffer, 2017).

### **3.8 LARAVEL**

Laravel es un framework para desarrollar aplicaciones web con sintaxis expresiva y elegante (Laravel LLC, 2020). Laravel ayuda a que sus ideas se conviertan en realidad sin código desperdiciado, utilizando estándares de codificación modernos, rodeados de una comunidad vibrante, con un ecosistema de herramientas que les da poder (Stauffer, 2017).

#### **3.8.1 REQUERIMIENTOS PARA LA INSTALACIÓN**

Los siguientes son requisitos para instalar Laravel en una computadora:

1. Una aplicación de terminal.
2. Una conexión a internet.
3. Un entorno de desarrollo configurado con Apache, MySQL y PHP v7.0.
4. Acceso al comando *git*.
5. Acceso al comando *composer*.

(Nutile, 2006).

#### **3.8.2 INSTALACIÓN**

Si se cumplen todos los requisitos, uno puede seguir adelante y descargar Laravel con un solo comando desde la terminal:

```
composer create-project --prefer-dist laravel/laravel
```

Después de instalarse se debe correr con el siguiente comando desde la terminal:

```
php artisan serve
```

Al correr este comando, se podrá acceder en el navegador a <https://localhost:8000> para visualizar la aplicación (Laravel LLC, 2020).

### 3.8.3 DESARROLLANDO APLICACIONES CON LARAVEL

Laravel sigue el patrón de lógica MVC (Modelo, Vista, Controlador). Pero tiene muchas más funciones que han mejorado las características del patrón, elevándolas a un nuevo nivel. Laravel viene con un mecanismo de inyección de dependencia por defecto. El mecanismo DI es una forma de agregar una instancia de una clase a otra clase utilizando un constructor de clase. El flujo de trabajo del patrón MVC es simple: la acción del usuario llega al controlador desde la vista. El controlador notifica al modelo. En consecuencia, el modelo actualiza el controlador. Después de eso, el controlador vuelve a actualizar la vista (Sinha, 2019).

## IV. METODOLOGÍA

En este capítulo se aborda el marco metodológico del proyecto, en el cual se describe el proceso y el análisis determinado para intentar responder a las preguntas de investigación.

### 4.1 ENFOQUE

En este informe se desarrolla un análisis cualitativo de las partes que forman el sistema de monitoreo de energía propuesto, partiendo del uso de dispositivos electrónicos y software para el almacenamiento y procesamiento de datos, todo esto con el fin de definir métricas importantes que permitan tener un mayor grado de entendimiento del consumo de energía de una carga.

### 4.2 VARIABLES DE INVESTIGACIÓN

La variable dependiente de este proyecto es el sistema de monitoreo de energía en tiempo real. Las variables independientes son las siguientes:

1. Herramientas para medir la energía eléctrica: Estas herramientas tienen como objetivo proveer formas para medir la energía eléctrica y con el requisito que deben poder ser leídos por un microcontrolador. Pueden ser evaluadas cualitativamente para determinar y escoger las que mejor se adapten a las necesidades de la variable dependiente.
2. Herramientas para conectarse con el servidor: El microcontrolador y el software que recopila los datos leídos. Guiándose por el análisis cualitativo, se determinará el microcontrolador a utilizar y los requisitos del software para procesar los datos, almacenarlos y mostrarlos al usuario en tiempo real.
3. Métricas: Estas serán evaluadas por sus características cualitativas, con el objetivo de que provean un vistazo general del consumo de energía de la carga el cual permita monitorearla para fines definidos por el usuario.

### 4.3 TÉCNICAS E INSTRUMENTOS APLICADOS

Para poder desarrollar el sistema según lo propuesto, son necesarias las siguientes técnicas e instrumentos: Arduino UNO, Sensor de corriente, kit de electrónica básico, ordenador, conocimientos básicos de programación en C y PHP.

#### 4.4 METODOLOGÍA DE ESTUDIO

Se propone un sistema que mida, por medio de un sensor de corriente, la intensidad de la corriente que pasa por una carga; luego que lea estos datos con el Arduino junto con su accesorio el Arduino Ethernet Shield y que el software que corre en este procese los datos y los envíe al ordenador principal/base de datos, finalmente se planifica que los datos se muestren al usuario en una interfaz gráfica con métricas útiles.

De la amplia variedad de dispositivos electrónicos para medir la corriente, los más aptos para trabajar con el Arduino son el ACS712 y el ACS714. Estos dispositivos presentan cualidades bastante semejantes como la capacidad para medir la intensidad en circuitos de forma no intrusiva (pasiva), y proporcionan en su salida una tensión lineal proporcional a la corriente que atraviesa sus pines de entrada. La única diferencia encontrada para efectos del sistema es el costo de adquisición, lo cual hacen del ACS712 una mejor alternativa.

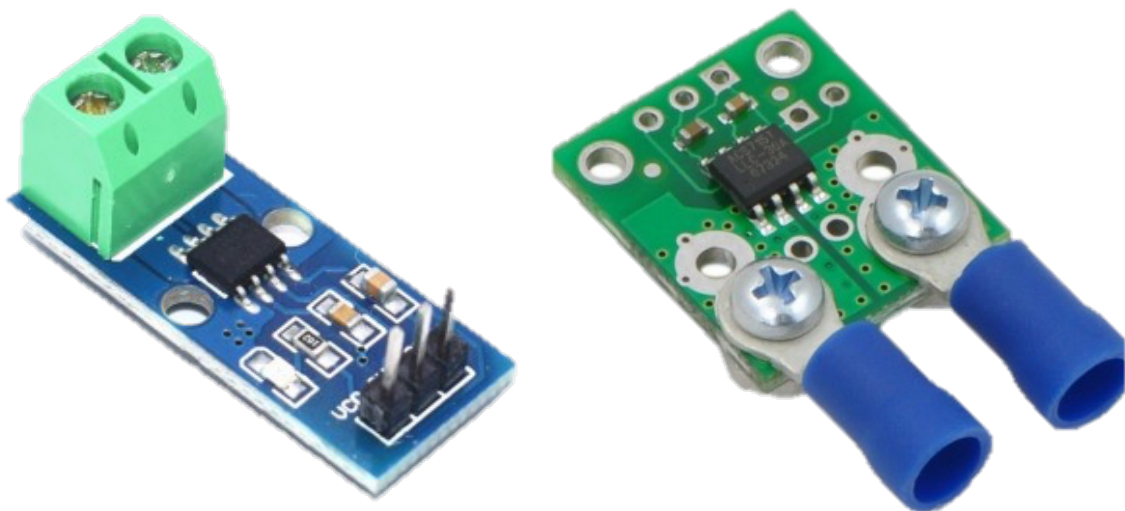


Ilustración 6 - ACS712 y ACS714

La instalación del ACS712 y el Arduino Ethernet Shield en el Arduino se detalla en la siguiente ilustración.

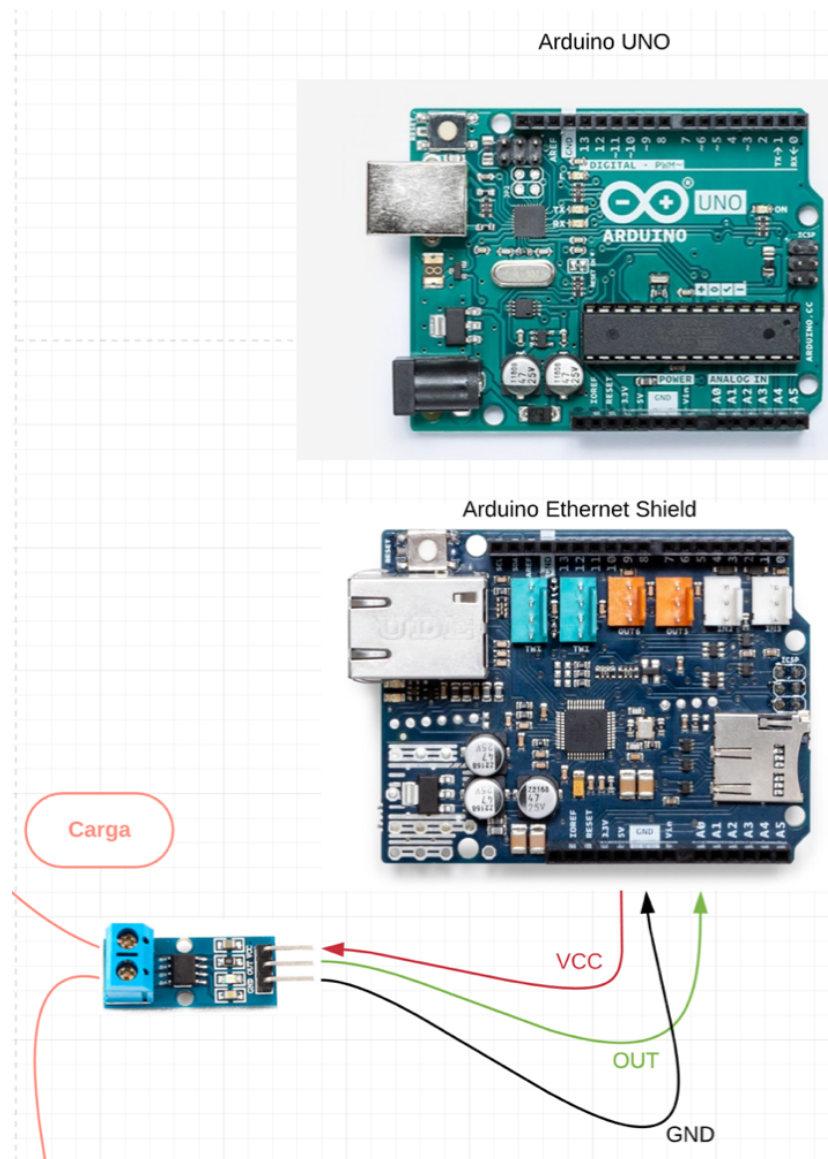


Ilustración 7 - Diagrama de instalación Arduino UNO, Ethernet Shield y ACS712

La tensión de salida (OUT) del ACS712 es proporcional a la intensidad y este incluye un desplazamiento de 2.5V que posiciona en el centro a la referencia.

$$V = 2.5 + K \cdot I \Rightarrow I = (V - 2.5)/K$$

Ecuación 2 - Voltaje y corriente medido por el ACS712

La sensibilidad del sensor (K) es de 185 mV/A según los datos del fabricante.

El Arduino interpretará la señal del sensor y la enviará de forma periódica al servidor que hospeda la base de datos el cual también hospeda la aplicación de PHP que alberga la interfaz del usuario. El diseño final del sistema se resume en la siguiente figura:

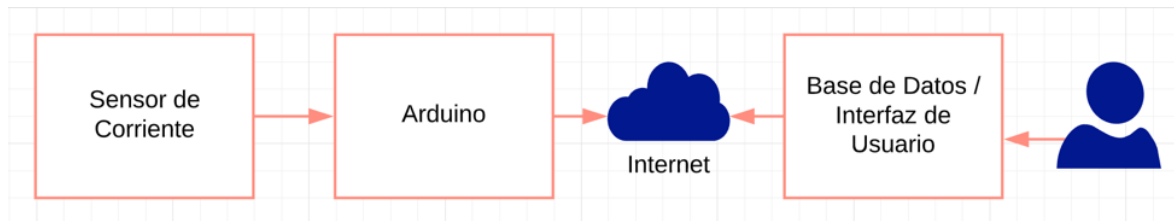


Ilustración 8 - Diagrama resumen del sistema

#### 4.4.1 SOFTWARE DE PROCESAMIENTO DE DATOS

Cuando el ACS712 hace una lectura, se obtiene el valor proporcional a la corriente en ese momento, este valor oscila a una frecuencia de 60Hz en los que se obtienen picos. El software debe ser capaz de obtener las lecturas, calcular la corriente RMS y la potencia.

#### 4.4.2 LAS MÉTRICAS CLAVE

El propósito del sistema consiste en generar datos relevantes sobre el consumo de energía eléctrica en tiempo real de una carga o circuito. Considerando lo anterior, es decisión del usuario el uso que le de a este. Es por esto que para comenzar el sistema debe mostrar la mayor cantidad de información sobre las medidas hechas, comenzando con la información sin procesar y terminando con métricas que puedan ayudar a algunos usuarios.

#### 4.4.3 SOFTWARE EN EL SERVIDOR Y LA INTERFAZ GRÁFICA

El software (servidor) debe ser capaz de exponer un punto de conexión para que otros dispositivos (clientes) puedan enviarle información, luego de recibir la información se guardará en una base de datos y se calcularán las métricas con los datos recopilados mostrándolos en una interfaz gráfica. Ya finalizado el software y la interfaz gráfica se debe programar al Arduino para que proceda a enviar los datos de sus lecturas.



## 4.5 CRONOGRAMA DE ACTIVIDADES

Para desarrollar el proyecto primero se deben adquirir los materiales definidos en la sección 4.3, estos son bien comunes y no debería de existir un problema para encontrarlos en tiendas de electrónica básicas.

Una vez adquiridos se puede comenzar inmediatamente con la instalación de los mismos. La siguiente actividad es la que se documenta en la sección 4.4.2, este software debe correr en el hardware instalado y se puede probar utilizando las herramientas de software del fabricante (Arduino).

Al tener el sistema de medición local instalado y funcional se puede continuar a implementar el código de la sección 4.4.4 para integrar este al servidor y comenzar a almacenar los datos.

Finalmente se deja la parte de optimización, pruebas y mejoras para cerciorarse que todo funcione adecuadamente.

Actividades	Día 1	Día 2	Día 3	Día 4	Día 5
Compra de materiales					
Instalación de dispositivos electrónicos					
Desarrollo e instalación del software de procesamiento de datos en dispositivos electrónicos					
Desarrollo del software y la interfaz gráfica					
Pruebas y optimizaciones					

## V. RESULTADOS Y ANÁLISIS

En este capítulo se lleva a la ejecución lo definido y diseñado en los capítulos anteriores.

### 5.1 MEDICIÓN DEL CONSUMO DE ENERGÍA

El sketch a continuación es el programa que correrá en el arduino para leer la corriente y calcular la corriente RMS y la potencia cada medio segundo.

```
float s = 0.185; // Sensibilidad en V/A
void setup()
{
  Serial.begin(9600);
}
void loop()
{
  // Calcular la corriente en base a lecturas del sensor por 10 segundos
  float i = readIntensity(500);
  // Corriente RMS
  float iRMS = i * 0.707;
  // Potencia
  float p = iRMS * 120.0;
}
// Esta función retorna la corriente promedio basado en lecturas durante
// el tiempo dado
float readIntensity(int ms)
{
  float v;
  // millis(): Devuelve el número de milisegundos transcurridos desde
  // que la placa Arduino comenzó a ejecutar el programa.
  long t = millis();
  float i = 0;
  float max = 0;
  float min = 0;
  while ((millis() - t) < ms)
  {
    // analogRead(): Devuelve un valor entre 0 y 1023
    // (correspondientes a 0V y 5V)
    v = analogRead(A0) * (5.0 / 1023.0);
    i = (v - 2.5) / s;
    if (i > max) {
      max = i;
    }
    if (i < min) {
      min = i;
    }
  }
  return (((max - min) / 2) - r);
}
```

Con lo anterior el software se puede cargar al Arduino y el siguiente paso es crear la aplicación en el servidor con Laravel.

## 5.2 ESTABLECER LA COMUNICACIÓN CON EL SERVIDOR

Para crear una nueva aplicación con Laravel:

**laravel new project**

Para correr la aplicación:

**php artisan serve**

Lo anterior permite ver la interfaz gráfica en la dirección <http://127.0.0.1:8000/> del ordenador.

Luego se debe crear una base de datos con MySQL en el sistema y configurarla en el archivo **.env**.

El siguiente paso es crear la tabla en donde se guardarán los datos que envíe el cliente (Arduino). Para esto, Laravel también proporciona un comando de ayuda:

**php artisan make:migration create\_medidas\_table**

En donde "medidas" es el nombre de la tabla en la base de datos. Esto creará un archivo dentro de la carpeta ./database/migrations que contiene la información de la tabla que se crea al instalar la aplicación. En este archivo se crearán los campos de la tabla dentro de la función denominada up():

```
public function up()
{
    Schema::create('medidas', function (Blueprint $table) {
        $table->id(); // Identificador comenzando en 1
        $table->string('i_rms'); // Corriente RMS
        $table->string('potencia'); // Potencia
        $table->timestamps();
    });
}
```

Al guardar este archivo se considera configurada la tabla y podemos correr lo siguiente para proceder a crearla en el ordenador:

**php artisan migrate**

El siguiente paso es exponer la ruta en donde se ingresan datos a la tabla de medidas.

```
// Se expone la ruta en http://127.0.0.1:8000/medidas
Route::post('/medidas', function () {
    DB::table('medidas')->insert([
        'i_rms' => request()->i_rms,
        'potencia' => request()->potencia,
        'created_at' => now()
    ]);
    return response('Datos ingresados');
});
```

El código anterior expone la ruta en <http://127.0.0.1:8000/medidas> y requiere que se le envíen dos parámetros llamados i\_rms y potencia, las cuales son la corriente rms y la potencia calculada, respectivamente.

Con esto queda completo la funcionalidad básica en el servidor.

### **5.3 ENUMERAR LAS MÉTRICAS CLAVE**

Siguiendo la premisa del problema definido, las métricas más importantes son aquellas que provean información para que el usuario tome medidas y logre un ahorro en el consumo de energía de la carga medida.

Las métricas que el sistema calculará son:

1. Kilowatt \* Hora (kWh) Consumidos: Es la unidad de venta de la energía eléctrica. Esta se calcula con el promedio de la potencia en la carga en el período de una hora en kilowatts.
2. Potencia máxima y mínima: Esta nos revela en que momento la carga llega a consumir su potencia máxima y mínima.
3. Total a pagar: Es el total a pagar por los kWh consumidos según la tarifa ingresada.

Para crear la interfaz gráfica y calcular las métricas, el primer paso para es modificar el archivo `/resources/views/welcome.blade.php` a que contenga la estructura que muestre las métricas y alguna señalización básica:

```
<h3>Sistema de Monitoreo de Energía</h3>
<ul>
  <li>kWh: {{ $kwh }}</li>
  <li>Potencia Máx: {{ $p_max }}</li>
  <li>Potencia Min: {{ $p_min }}</li>
  <li>Total a pagar: {{ $costo }}</li>
</ul>
```

Para calcular las métricas con los datos y mostrar los valores en la interfaz:

```
Route::get('/', function () {
    $precio_por_kwh = 3.7041;
    $medidas = DB::table('medidas')->get();
    // round(): función para redondeas a la cantidad de digitos especificados
    $kwh = round(getkwh(), 2);
    // min() y max() calculan el minimo y máximo de una columna
    $p_min = round($medidas->min('potencia') / 1000, 2);
    $p_max = round($medidas->max('potencia') / 1000, 2);
    $costo = round($kwh * $precio_por_kwh, 2);
    return view('welcome', [
        'kwh' => $kwh,
        'p_max' => $p_max,
        'p_min' => $p_min,
        'costo' => $costo,
    ]);
});
function getkwh()
{
    // Leer todas las medidas de la base de datos
    $medidas = DB::table('medidas')->get();
    // La función avg() calcula el promedio de una columna
    $potenciaPromedio = $medidas->avg('potencia');
    // Las horas pasadas desde el primer ingreso y el último
    $horas = Carbon::parse($medidas->first()->created_at)->diffInHours($medidas->last()->created_at);
    // Calcular los kWh
    $kwh = $potenciaPromedio * $horas / 1000;
    return $kwh;
}
```

Finalmente se agrega la librería Ethernet al sketch previamente hecho en el Arduino IDE, se crea la variable con la dirección MAC de la tarjeta de red (Arduino Ethernet Shield) y se declara el cliente HTTP:

```
#include <Ethernet.h>
byte mac[] = { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 };
EthernetClient client;
```

Luego modificamos la función setup() a que incluya lo siguiente debajo del código que ya contiene:

```
Ethernet.begin(mac)
// Se le debe dar un segundo para que la librería pueda conectarse
delay(1000);
client.connect(server, 80)
```

Lo anterior verificará que la conexión con el servidor se haga exitosamente, luego se debe enviar la información mediante HTTP con la siguiente función:

```
void enviarDatos(float iRMS, float potencia)
{
    client.println($"GET /medidas?i_rms={iRMS}&potencia={potencia} HTTP/1.1");
    client.println("Access-Control-Allow-Origin: *");
    client.println("Host: 127.0.0.1:8000");
    client.println("Connection: close");
    client.println();
}
```

La función anterior se agrega al final de la función loop() pasándole los valores calculados:

```
float i = readIntensity(1000);
float iRMS = i * 0.707;
float p = iRMS * 120.0;
enviarDatos(iRMS, p);
```

Con lo anterior se completa el sistema y se puede comenzar a observar las métricas calculadas en tiempo real.

## Sistema de Monitoreo de Energía

- kWh promedio: 6.55 kW\*h
- Potencia Máx: 5.4 kW
- Potencia Min: 1.44 kW
- Total a pagar: 24.26 Lempiras

Ilustración 9 - Vista de la interfaz del usuario

## VI. CONCLUSIONES

En este informe se desarrolló el diseño de un sistema para realizar mediciones en tiempo real del consumo de energía de una carga. Este sistema cumple con las premisas iniciales de brindar las métricas relevantes a los usuarios, y lo hace siendo bastante cómodo en cuanto a su costo monetario y de tiempo. Gracias a las plataformas de hardware y software abierto de Arduino y Laravel, el desarrollo de esta solución se hizo sin complicaciones mayores y por la forma en la que está desarrollado permite a otras personas mejorarlo o implementar sus propias ideas sobre el.

1. *Se desarrollo un sketch que al ser instalado en el arduino junto con el sensor de corriente, permite hacer lecturas periodicas sobre una carga. Pese a que hay varias formas en las que se puede medir el consumo de energía de un circuito, utilizar sensores electrónicos que actuen de forma pasiva es una de las mejores y menos intrusivas. El sensor utilizado en el diseño del sistema no solo es de un costo bien asequible, también es bien facil de instalar y utilizar con el Arduino.*
2. *Se conectó el Arduino, por medio del Ethernet Shield y software desarrollado, a un servidor; permitiendole el almacenamiento de las medidas sobre la carga al sistema. La plataforma Arduino es muy versatil en las formas de aplicación ya que proporciona una interfaz física y una interfaz virtual optimizadas para los usuarios. También cuenta con una amplia variedad de accesorios y módulos que pueden conectarse de forma eficiente y a bajo costum, como el Ethernet Shield, que le permite utilizar el protocolo http para conectarse con otros dispositivos. La interfaz gráfica que provee el Arduino IDE también ayudóen este proyecto ya que permitió desarrollar todo un algoritmo sobre ella, sin necesidad de terceros para instalarlo en la placa.*
3. *Se determinaron y enumeraron las métricas importantes sobre los datos recopilados por el sistema y se calcularon por medio de un algoritmo en el servidor, también se habilitó una interfaz gráfica para la visualización de estas. Se concluye que las métricas calculadas a partir de los datos recopilados son de una gran utilidad al usuario puesto que le permite observar el consumo de energía de la carga desde su ordenador o smartphome (si esta conectado en la misma red del servidor) cuando este lo requira.*

## **VII. RECOMENDACIONES**

Los objetivos planteados fueron cumplidos satisfactoriamente y según esta premisa se puede deducir que el diseño implementado funciona para el fin establecido en mayoría. Sin embargo, siempre hay espacio para mejoras y recomendaciones para otras implementaciones, las limitantes más importantes del desarrollo se considera que fueron que solamente se puede monitorear una carga y que el sensor de corriente utilizado tiene un límite de 30 amperios. Para ello se recomienda a otras implementaciones que se considere lo anterior para mejorar el alcance.

El sistema desarrollado presenta una posible solución al problema de monitoreo de energía en lugares como hogares y oficinas en donde un individuo con conocimientos básicos de electrónica y desarrollo de software puede implementar el sistema sin costos mayores.

Algunas recomendaciones para mejorar este sistema son las siguientes:

1. La forma en la que el Arduino envía los datos al servidor es de una naturaleza precipitada, lo que causa leve saturación siempre y cuando haya una buena conexión. Se recomienda enviar más datos en menos solicitudes ya que de esta manera prevenimos la saturación de red esperando la respuesta del servidor y garantizando que los datos lleguen con éxito.
2. En este informe se utiliza el sistema en una red local, sin embargo, este puede ser adaptado con bien pocos cambios necesarios para ser usado en la red y permitir al usuario monitorear su carga desde cualquier lugar.
3. Permitirles a los usuarios administrar la base de datos desde alguna interfaz con que contenga filtros y campo de búsqueda para darle un historial de medidas podría ser de gran utilidad.

Con el fin de mejorar en los puntos mencionadas, se incita a estudiar más dispositivos electrónicos que puedan medir pasivamente la energía consumida. Estos no necesariamente deben ser sensores de corriente.



## BIBLIOGRAFÍA

1. Davis, M. E., & Phillips, J.A. (2006). *Learning PHP and MySQL*. O'Reilly Media.
2. Harrington, W. (2015). *Learning Raspbian*. Packt Publishing.
3. International Energy Agency. (2017). *World Energy Outlook 2017*. IEA. <https://www.iea.org/reports/world-energy-outlook-2017>
4. Iwansson, K., Sinapius, G., Hoornaert, W., & Middelhoek, S. (1999). *Measuring Current, Voltage and Power*. Elsevier.
5. Jordan, K. (2006). *The visibooks guide to MySQL basics*. Visibooks.
6. Laravel LLC. (2020). *The PHP Framework for Web Artisans*. Laravel. <https://laravel.com>
7. Laravel LLC. (2020). Installation. Laravel. <https://laravel.com/docs/7.x#installing-laravel>
8. Nutile, A. (2006). *Laravel 5.x cookbook*. Packt Publishing.
9. Pérez, F. E. V., & Areny R. P. (2007). *Microcontroladores: fundamentos y aplicaciones con PIC*. MARCOMBO, S.A.
10. Raspberry Pi Foundation. (2020). *What is raspberry pi?*. Raspberry Pi. <https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/>
11. REN21. (2019) *Renewables 2018 Global Status Report*. REN21. [https://www.ren21.net/wp-content/uploads/2019/05/GSR2018\\_Full-Report\\_English.pdf](https://www.ren21.net/wp-content/uploads/2019/05/GSR2018_Full-Report_English.pdf)
12. Richardson, M., & Wallace, S. (2013). *Getting started with raspberry pi* (2nd ed.). O'Reilly Media.
13. Robins, S. O. (2019). *Programming the Raspberry Pi 4: Learning Python with Raspberry Pi 4 and Learn It Well*. Amazon Digital Services LLC.
14. Sinha, S. (2019). *Beginning Laravel: build websites with Laravel*. Apress.
15. Stauffer, M. (2017). *Laravel: up and running*. O'Reilly Media.
16. Lerdorf, R., Tatroe, K., & MacIntyre, P. (2006). *Programming Php* (2<sup>nd</sup> ed.). O'Reilly Media.

17. Upton, E., & Halfacree, G. (2016). *Raspberry pi user guide* (4th ed.). John Wiley & Sons Ltd.
18. Wolf, S., & Smith, R. F. M. (1992). *Guia para mediciones electrónicas y practicas de laboratorio* (2nd ed.). Pearson Education.
19. Yamanoor, S., & Yamanoor, S. (2017). *Python Programming with Raspberry Pi*. Packt Publishing.
20. Young, H. D., Freedman, R. A., Ford, A. L., Sears, F. W., & Zemansky, M. W. (2009). *Física universitaria con física moderna* (12th ed.). Pearson Education.
21. Arduino. (2020). *Arduino – Introduction*. Arduino.  
<https://www.arduino.cc/en/Guide/Introduction>
22. Arduino. (2020). *Arduino – Environment*. Arduino.  
<https://www.arduino.cc/en/Guide/Environment#toc14>
23. Arduino. (2020). *Arduino – ArduinoEthernetShieldV1*. Arduino.  
<https://www.arduino.cc/en/Main/ArduinoEthernetShieldV1>