

SISTEMA DE ANALÍTICA DE DATOS ESCALABLE PARA REGISTROS DE
CONSUMO ELÉCTRICO RESIDENCIAL

JUAN SEBASTIÁN BELEÑO DÍAZ

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA

2015

SISTEMA DE ANALÍTICA DE DATOS ESCALABLE PARA REGISTROS DE
CONSUMO ELÉCTRICO RESIDENCIAL.

JUAN SEBASTIÁN BELEÑO DÍAZ

Trabajo de grado para optar por el título de:
Ingeniero de Sistemas

DIRECTOR

RAÚL RAMOS POLLÁN

DOCTOR EN INGENIERÍA INFORMÁTICA

CODIRECTOR

JORGE HERNANDO RAMÓN SUÁREZ

MAGÍSTER EN INGENIERÍA DE COMPUTACIÓN, INFORMACIÓN Y CONTROL

CODIRECTOR

GABRIEL ORDÓÑEZ PLATA

DOCTOR EN INGENIERÍA INDUSTRIAL

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECAICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA

2015

TABLA DE CONTENIDO

	Pág.
INTRODUCCIÓN	15
1. JUSTIFICACIÓN Y PLANTEAMIENTO DEL PROBLEMA	17
2. OBJETIVOS	18
2.1. OBJETIVO GENERAL.....	18
2.2. OBJETIVOS ESPECÍFICOS.....	18
3. MARCO DE REFERENCIA	19
3.1. ESTADO DEL ARTE	19
3.2. MARCO TEÓRICO.....	21
3.2.1. Potencia eléctrica.	21
3.2.2. Factor de potencia.....	22
3.2.3. Curvas de carga.	22
3.2.3.1. Método de medición.	22
3.3. SISTEMAS DISTRIBUIDOS.....	23
3.4. MAPREDUCE.....	24
3.5. HADOOP MAPREDUCE	26
3.6. OCTOPY	28
3.7. MINCEMEAT.PY	28
4. METODOLOGÍA	30
4.1. SCRUM.....	30
4.1.1. Backlog product.....	31
4.1.2. Sprint planning.	32
4.1.3. Sprint backlog.	32
4.1.4. Sprint.	32
4.1.5. Daily scrum.	32
4.1.6. Sprint review & retro.....	33
4.1.7. Working software.	33
4.1.8. Burndown chart.....	34

4.1.9.	Task board.....	34
5.	DESARROLLO DEL PROYECTO	35
5.1.	DESCRIPCIÓN.....	35
5.1.1.	Módulo panel de control.....	35
5.1.2.	Módulo de resumen.....	36
5.1.3.	Módulo usuario promedio.....	37
5.1.4.	Módulo usuarios.....	38
5.1.5.	Módulo de sincronización de nuevos dato.....	40
5.2.	CICLOS DE DESARROLLO	41
5.2.1.	Roles	41
5.2.2.	Primer sprint.....	41
5.2.2.1.	Requisitos.....	41
5.2.2.2.	Prototipo #1.....	42
5.2.2.3.	Implementación y pruebas.....	44
5.2.3.	Segundo sprint.....	46
5.2.3.1.	Requisitos.....	46
5.2.3.2.	Prototipo #2.....	46
5.2.3.3.	Implementación y pruebas.....	50
5.2.4.	Tercer sprint	52
5.2.4.1.	Requisitos.....	52
5.2.4.2.	Prototipo #3.....	53
5.2.4.3.	Implementación y pruebas.....	57
5.2.5.	Cuarto sprint.....	57
5.2.5.1.	Requisitos.....	57
5.2.5.2.	Prototipo #4.....	58
5.2.5.3.	Implementación y pruebas.....	61
5.3.	PRUEBAS DE ESCALABILIDAD	62
5.3.1.	Arquitectura de pruebas.....	62
5.3.2.	Simulación de datos.....	63
5.3.3.	Implementación.....	63

5.3.4. Resultados..... 64

5. CONCLUSIONES70

6. RECOMENDACIONES72

BIBLIOGRAFÍA73

ANEXOS.....75

LISTA DE FIGURAS

	Pág.
Figura 1. Variables que afectan las características de las curvas de carga.....	20
Figura 2. Ejemplo del wordcount con MapReduce.....	25
Figura 3. Hadoop MapReduce.....	26
Figura 4. Metodología SCRUM.....	31
Figura 5. Panel de control.....	36
Figura 6. Tabla de resumen.....	37
Figura 7. Usuario promedio.....	38
Figura 8. Usuarios.....	39
Figura 9. Sincronizar nuevos datos.....	40
Figura 10. Burndown Chart del primer sprint.....	44
Figura 11. Usuarios – prototipo #1.....	45
Figura 12. Usuario – prototipo #1.....	45
Figura 13. Burndown chart del segundo sprint.....	50
Figura 14. Panel de control – prototipo #2.....	51
Figura 15. Potencia de distorsión de un usuario – prototipo #2.....	52
Figura 16. Burndown chart del tercer sprint.....	56
Figura 17. Energía por días de la semana – prototipo #3.....	57
Figura 18. Burndown chart del cuarto sprint.....	61
Figura 19. Arquitectura de pruebas para 8 clientes.....	62
Figura 20. Script en bash para ejecución de múltiples clientes.....	64
Figura 21. Curva de escalabilidad para volúmenes de datos C.E.R. menores.....	67
Figura 22. Curva de escalabilidad para volúmenes de datos C.E.R. mayores.....	68

LISTA DE TABLAS

	Pág.
Tabla 1. Sprint Backlog del primer sprint	43
Tabla 2. Sprint Backlog del segundo sprint.....	49
Tabla 3. Sprint Backlog del tercer sprint	55
Tabla 4. Sprint Backlog del cuarto sprint	61
Tabla 5. Tabla de escalabilidad del procesamiento de datos C.E.R. con MapReduce.....	67

TABLA DE ANEXOS

	Pág.
Anexo A. Manual de uso de la plataforma.....	75
Anexo B. Script MapReduce consumosxusuario.py.....	92

AGRADECIMIENTOS

El autor agradece:

A los grupos de investigación GISEL y CPS de la Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones por brindarme el apoyo necesario para la realización de este proyecto de grado.

A mi director de proyecto, el profesor Raúl Ramos Pollán por siempre estar dispuesto a solucionar las dudas que tuve y por guiarme cada vez que necesitaba ayuda.

A mis codirectores Gabriel Ordóñez Plata y Jorge Hernando Ramón Suárez por su constante acompañamiento y sabiduría para explicar los temas donde tenía poco conocimiento.

A los estudiantes de Ingeniería Eléctrica Ray Hayder Rosales Delgado y Marcelo Meza Granados que facilitaron el proceso de recolección de requisitos y el análisis de los datos que arrojó este proyecto.

A Alejandro Pinzón y Daniel Gualdrón de Guarumo, por darme la oportunidad de aprender a solucionar problemas de la vida real a través de la aplicación del conocimiento que aprendí durante tanto años de estudio en la universidad.

DEDICATORIA

Dedico este trabajo a mis padres Martín Beleño Piñeres y Martha Luz Díaz Alvarado que me han apoyado incondicionalmente en las situaciones difíciles por las que he pasado durante toda mi vida, admiro el valor que han tenido para apoyar mis decisiones aunque estas decisiones les haya afectado bastante.

A mis hermanos Martín Andrés Beleño Díaz y Sergio Luis Beleño Díaz que me acompañaron durante todo el camino y con los que he compartido momentos de calidad que siempre me han levantado el ánimo para seguir adelante.

A mi tía Enalba Beleño y al señor Gustavo Ramírez que me acogieron durante toda mi estancia en esta ciudad como un hijo más y que me brindaron todas las facilidades para que pudiese estudiar y progresar en mi vida académica, profesional y personal.

A mis amigos y compañeros de la universidad que siempre estuvieron ahí cuando yo los necesitaba.

Juan Sebastián Beleño Díaz

RESUMEN

TÍTULO: SISTEMA DE ANALÍTICA DE DATOS ESCALABLE PARA REGISTROS DE CONSUMO ELÉCTRICO RESIDENCIAL.¹

AUTOR: JUAN SEBASTIÁN BELEÑO DÍAZ²

PALABRAS CLAVES: big data, analítica, escalabilidad, consumo eléctrico residencial, curvas de carga.

DESCRIPCIÓN:

En este documento se muestra el desarrollo de una plataforma con arquitectura de procesamiento escalable de datos de consumos eléctricos residenciales (C.E.R.), este proyecto se realizó para facilitar el análisis de los comportamientos de los usuarios y poder caracterizarlos según los datos C.E.R. de cada uno.

En este proyecto se utilizó una herramienta de Big Data llamada MapReduce que permite paralelizar el procesamiento en cierto tipo de problemas dentro del que se encuentra el problema que en este proyecto se aborda, además para verificar la escalabilidad de la plataforma fue necesario simular datos C.E.R. y medir los tiempos de ejecución del procesamiento con diversos volúmenes de datos y diferente número de clientes del programa.

Los resultados en escalabilidad de la plataforma que se explican en este documento son justificados por la herramienta utilizada, para pocos datos la escalabilidad es inexistente y para volúmenes de datos de millones la escalabilidad empieza a ser visible, además se realiza una herramienta que posteriormente es utilizada para el análisis de datos C.E.R. reales de usuarios ubicados en el barrio de La Feria en el municipio de Piedecuesta, estos datos fueron proporcionados por los grupos de investigación GISEL y CPS de la Universidad Industrial de Santander.

ABSTRACT

¹ Proyecto de grado modalidad de investigación

² Facultad de Ingenierías Físico – Mecánicas. Escuela de Ingeniería de Sistemas e Informática. Director: PhD Raúl Ramos Pollán. Co-directores: Msc. Jorge Hernando Ramón Suárez, PhD. Gabriel Ordóñez Plata.

TITLE: SCALABLE DATA ANALYSIS SYSTEM FOR RECORDS OF RESIDENTIAL ELECTRICITY CONSUMPTION.³

AUTHOR: JUAN SEBASTIÁN BELEÑO DÍAZ⁴

KEYWORDS AND PHRASES: big data, analytics, scalability, residential electricity consumption, load curves.

DESCRIPTION:

In this paper is shown the development of a system with scalable processing architecture for records of residential electricity consumption (R.E.C.), this Project was done to facilitate the user behavior analysis and to characterize them according to R.E.C. data of each user.

In this project was necessary to use a big data tool called MapReduce that allows to parallelizing certain kind of problems, included the problem that is treated in this projects, also to verify the system scalability was necessary to simulate R.E.C. data records and measure the processing execution time with different data volumen and different number of program clients.

The results in the system scalability that are explained in this document are justified by the tool used in this project, for few data the scalability does not exist and for millions of records the scalability start to be visible, also it is make a tool that later was used to analyze R.E.C data records of real users located in the barrio La Feria in the city of Piedecuesta, these records were provided for the research groups GISEL and CPS in the Industrial University of Santander and it were collected using gauges developed by those research groups and are patentables.

³ Final undergraduate project.

⁴ Physics Mechanical Engineering Faculty. Informatic and System Engineering School. Director: PhD Raúl Ramos Pollán. Co-directors: Msc. Jorge Hernando Ramón Suárez, PhD. Gabriel Ordóñez Plata.

INTRODUCCIÓN

Actualmente las empresas dedicadas a la distribución de energía eléctrica se encuentran en una búsqueda constante para optimizar el uso de los recursos que manejan, esta búsqueda contribuye en gran manera a la mejora en la calidad del servicio que se le presta al usuario final y a la rentabilidad de la empresa prestadora del servicio.

Las empresas de distribución de energía eléctrica en Colombia están obligadas a seguir los lineamientos de calidad y seguridad del servicio que establece la Comisión de Regulación de Energía y Gas (CREG), pero esto debe realizarse minimizando las pérdidas de energía eléctrica en la prestación del servicio, por lo que se realizan mediciones periódicas de potencia activa, reactiva y otros parámetros en ciertos puntos de la red de distribución eléctrica para mantener control del sistema.

Los datos recolectados durante las mediciones necesitan ser estudiados para el análisis de los patrones de comportamiento en el consumo de los usuarios en toda la red de distribución eléctrica, debido a que ayuda al cálculo adecuado de las tarifas, pero también ayuda en gran medida para establecer planes de contingencia en los transformadores de distribución, mejorando así la calidad en la prestación del servicio y el aumento en la rentabilidad de la empresa como consecuencia de la disminución en las pérdidas en la prestación del servicio.

El desarrollo de esta plataforma se planteó en colaboración con los grupos GISEL y CPS de la Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones (E3T) quienes suministraron los datos para la validación de los resultados de este trabajo.

Las validaciones se realizaron con aproximadamente 200000 registros de consumo de usuarios reales ubicados en el barrio La Feria – Piedecuesta y se espera que en entornos de producción se manejen volúmenes del orden de varios millones de datos, debido a estos requisitos se desarrolló una plataforma que es escalable computacionalmente (validado con el módulo de simulación de datos) y que fue construida por ciclos Scrum para facilitar el análisis de datos teniendo en cuenta los parámetros de la etapa de medición y recolección de datos, obteniendo información detallada de los usuarios clasificada por su consumo de energía eléctrica en el período de tiempo provisto.

1. JUSTIFICACIÓN Y PLANTEAMIENTO DEL PROBLEMA

En esta sección se establecen los argumentos que permiten exponer el problema y la justificativa del desarrollo del proyecto para solucionar dicho problema.

Actualmente los grupos GISEL y CPS de la Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones (E3T), cuentan con datos recolectados periódicamente de algunos usuarios en diversos puntos del Área Metropolitana de Bucaramanga referentes a su consumo eléctrico. Algunos de los datos que registran son: La potencia activa, la potencia reactiva, los valores eficaces de la tensión y la corriente, la frecuencia, la fecha del registro, la hora del registro, el usuario, entre otros.

Al estar continuamente midiendo nuevos datos de los usuarios, los datos están aumentando a medida que pasa el tiempo, por lo que se hace necesario procesarlos y analizarlos para que puedan ofrecer información útil, luego de procesados liberar espacio y seguir recolectando nuevos datos.

A partir de esta necesidad se planteó este proyecto para la creación de una plataforma que pudiese tomar como base la información recolectada por los medidores y procesarla para detallar el comportamiento del consumo eléctrico de los usuarios y mostrarlos de una manera que pudiesen ser fácilmente estudiados y analizados por los designados de los grupos GISEL y CPS para esta labor (MEZA & ROSALES, 2015), ofreciendo información útil con la que se podrían tomar decisiones claves acerca de los usuarios y los transformadores a los que están asociados estos usuarios.

2. OBJETIVOS

2.1. OBJETIVO GENERAL

Definir e implementar un conjunto de herramientas *Big Data* que soporten ciclos de procesamiento y análisis de datos de consumo eléctrico residencial (C.E.R) sobre una arquitectura computacionalmente escalable.

2.2. OBJETIVOS ESPECÍFICOS

- Definir un caso de uso con un ciclo de procesamiento de datos C.E.R.
- Evaluar y seleccionar las herramientas Big Data apropiadas para la implementación del caso de uso anterior sobre una arquitectura escalable.
- Implementar un prototipo con el conjunto de herramientas seleccionadas y evaluar su rendimiento con los datos disponibles.
- Desarrollar un generador de datos C.E.R. simulados.
- Evaluar el rendimiento del sistema implementado con volúmenes de datos mayores obtenidos del generador desarrollado en el objetivo anterior.
- Elaborar guías de uso del sistema implementado para facilitar la incorporación de nuevos ciclos de procesamiento de datos C.E.R.

3. MARCO DE REFERENCIA

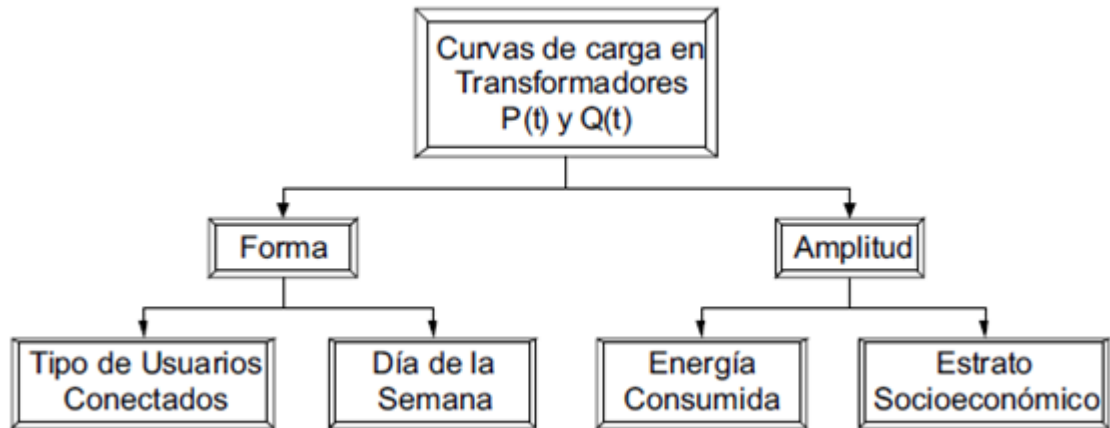
3.1. ESTADO DEL ARTE

La búsqueda por la optimización de los recursos en la industria de distribución de energía eléctrica ha facilitado un campo de explotación para la investigación en diversas partes del mundo para obtener curvas de carga y los perfiles de carga de los usuarios comerciales, industriales y residenciales.

En Colombia resaltan los estudios realizados por la Universidad del Valle donde se realiza una reseña de la mayoría de métodos usados para predecir las curvas de carga (método de medición, de agregación de curvas, de lógica difusa y de redes neuronales) y se aplica el concepto de redes neuronales para prever las curvas de carga de transformadores de distribución en el futuro a partir de datos anteriores en una serie de transformadores en Buenaventura y se entrena para entregar resultados con un 90% de precisión, además se muestra cómo afectan los diferentes parámetros a las curvas de carga (RAMÍREZ, PEREA, CAICEDO & CASTRO, 2008).

La *Figura 1* muestra las variables que se consideran para obtener las curvas de carga.

Figura 1. Variables que afectan las características de las curvas de carga.



Fuente: (RAMÍREZ, PEREA, CAICEDO & CASTRO, 2008).

En Sudamérica son notables las contribuciones que ha hecho Brasil en el estudio de disminuciones de las pérdidas operativas de los transformadores, caracterizando la carga en sectores residenciales, comerciales e industriales usando el método de agregación de curvas (GALLEGO, FRANCISQUINI, GÓMEZ & PADILHA, 2007), además de obtener los perfiles de carga diario para consumidores residenciales y basándose en datos provistos por empresas de distribución de energía eléctrica del estado de São Paulo (JARDINI, TAHAN, UN, GOUVEA & FIGUEIREDO, 2000). La USP también ha realizado avances en el uso de herramientas de *Big Data* para el procesamiento de datos de consumos de energía eléctrica en regiones densamente pobladas y con la colaboración de las distribuidores de energía eléctrica de la región obteniendo resultados altamente valorados en esa industria (MARQUES, 2006).

3.2. MARCO TEÓRICO

3.2.1. Potencia eléctrica. Es la relación del flujo energía por unidad de tiempo o la cantidad de energía entregada o absorbida por un elemento en un determinado tiempo. La unidad de medida en el Sistema Internacional de Unidades es el watt (W). La potencia eléctrica es producida por una corriente I en amperes(A) debido a la carga eléctrica Q coulomb(C) cada t segundos, producida por una tensión eléctrica en volts (V). (SERWAY & JEWETT, 2009)

$$P = \text{Trabajo realizado por unidad de tiempo} = \frac{QV}{t} = IV \quad (1)$$

- **Potencia activa:** se define como la cantidad energía eléctrica o trabajo, que se transporta o que se consume en una determinada unidad de tiempo y que efectivamente se aprovecha como potencia útil. Es el valor medio de la potencia instantánea.
- **Potencia reactiva:** se define como la cantidad de energía o trabajo, que se transporta en una determinada unidad de tiempo y que no es aprovechada como potencia útil, este tipo de potencia se presenta generalmente en dispositivos eléctricos que incluyan elementos almacenadores de energía como bobinas o condensadores. Está potencia se transforma en un campo magnético o en un campo eléctrico.
- **Potencia aparente:** Es el producto del valor eficaz de la tensión (dimensionamiento de aislamiento) y el valor eficaz de la corriente (dimensionamiento de la capacidad de transporte). Es una cantidad matemática y como tal no define una dirección del flujo de energía.

$$S = VI = \sqrt{P^2 + Q^2} \quad (2)$$

Donde P es potencia activa y Q es potencia reactiva. (ORDÓÑEZ, 2006)

- **Potencia aparente según Budeanu:** La pérdida de calidad bajo condiciones no sinusoidales ocasiona una definición diferente para la potencia aparente:

$$S = \sqrt{P^2 + Q^2 + D^2} \quad (3)$$

Donde P es potencia activa, Q es potencia reactiva y D es potencia de distorsión. (ORDÓÑEZ, 2006)

3.2.2. Factor de potencia. Es la razón entre la potencia que se transforma en trabajo útil y la potencia para la cual se dimensiona el sistema y está definida por la siguiente ecuación:

$$fp = \frac{P}{S} \quad (4)$$

Es un parámetro que permite establecer la eficiencia en el transporte de la energía eléctrica. (ORDÓÑEZ, 2006)

3.2.3. Curvas de carga. Las curvas de carga es una gráfica que muestra el comportamiento de consumo eléctrico de un usuario y se emplean para cuantificar las pérdidas en sistemas de distribución de energía, para establecer posibles tarifas de cobro que varíen según la hora del día en que ocurre el consumo y para optimizar el uso de los transformadores y demás elementos de un sistema de distribución de energía eléctrica. (RAMÍREZ et al., 2008).

3.2.3.1. Método de medición. La obtención de la curva de carga para cada transformador de distribución se realiza a través de mediciones directas, donde se censan las variables eléctricas todos los días de la semana en intervalos

normalmente de 1, 5 o 15 minutos, durante un período de tiempo. Esta información es procesada mediante herramientas estadísticas que permite obtener la curva de carga del transformador con una alta precisión (RAMÍREZ et al., 2008).

3.3. SISTEMAS DISTRIBUIDOS

Un sistema distribuido es uno en el que sus componentes están localizados en una red de computadores que comunican y coordinan sus acciones solo pasando mensajes entre ellos. Esta definición conlleva a características especialmente significativas para los sistemas distribuidos: concurrencia de componentes, no dependencia de sincronización e independencia a fallos de componentes.

Los retos que nacen de la construcción de sistemas distribuidos son la heterogeneidad de los componentes, confianza (que permite a los componentes ser agregados o reemplazados), seguridad, escalabilidad - la habilidad de trabajar bien cuando la carga o el número de usuarios se incrementa - manejo de fallos, concurrencia de componentes, transparencia y entrega de calidad de servicio.

- **Concurrencia:** En una red de computadores, la ejecución concurrente de programas es la norma, se puede hacer un trabajo de computador mientras otro lo realiza en el suyo, compartiendo recursos como páginas web o archivos cuando sea necesario. La capacidad del sistema para manejar recursos compartidos puede incrementarse agregando más recursos (por ejemplo: computadores) a la red. La coordinación de la concurrencia en la ejecución de programas que comparten recursos es también un tema importante y recurrente.
- **Sincronización:** cuando los programas necesitan cooperar para coordinar sus acciones por el intercambio de mensajes, una coordinación cercana

frecuentemente depende de una idea compartida del tiempo en la que ocurren las acciones del programa, pero hay límites para la precisión con la que un computador puede sincronizarse al reloj de la red - no hay una noción global del tiempo exacto, esto es una consecuencia directa del hecho que la comunicación es para enviar mensajes a través de una red.

- Tolerancia a fallas: Todos los sistemas de computadores pueden fallar y es responsabilidad del diseñador del sistema planificar las consecuencias de posibles fallos. Sistemas distribuidos pueden fallar de diferentes maneras, fallas en la red provocan el aislamiento de los computadores que están conectadas a ella, pero eso no significa que paren de ejecutar el programa; de hecho los programas pueden no detectar si la red ha fallado o se ha vuelto inusualmente lenta; de manera similar el fallo en un computador o el cierre inesperado en un programa del sistema, no será reportado inmediatamente a los otros componentes con los que se comunica. Cada componente del sistema puede fallar independientemente, dejando a los demás funcionando. (COULOURIS, DOLLIMORE, KINDBERG & BLAIR, 2011)

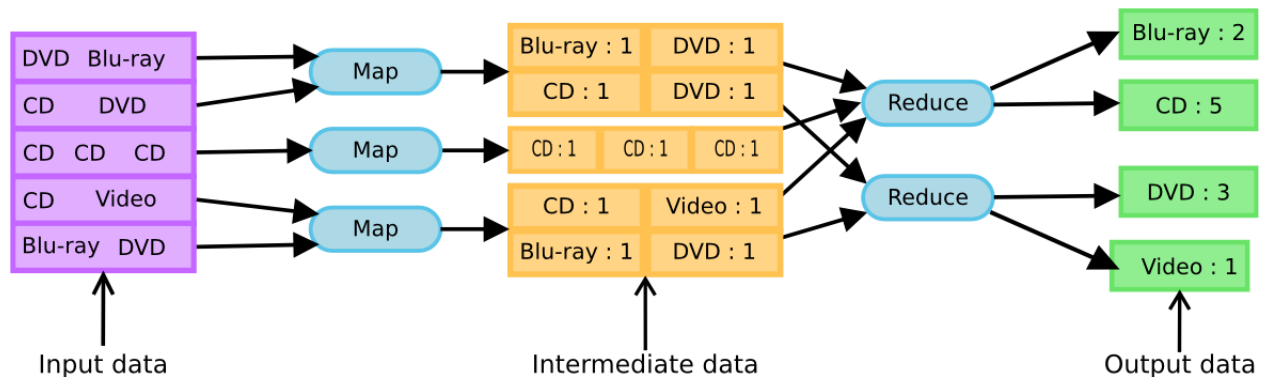
3.4. MAPREDUCE

Es un modelo de programación y una implementación asociada para procesar y generar grandes datos. Los usuarios especifican una función de *Map* que procesa un par clave/valor para generar un conjunto de pares intermedios de clave/valor, y una función *Reduce* que une todos los valores intermedios con la misma clave intermedia. Muchos problemas de la vida real son expresables por este modelo.

Programas escritos en este estilo funcional son automáticamente paralelizados y ejecutados en grandes grupos de máquinas. El sistema en tiempo de ejecución se

encarga de los detalles de la partición de los datos de entrada, agendar la ejecución del programa a través de un conjunto de máquinas, manejo de los fallos de las máquinas y la administración de la comunicación entre máquinas. Esto permite a los programadores sin experiencia alguna con sistemas paralelos y distribuidos, fácilmente hacer uso de recursos de grandes sistemas distribuidos. La implementación de MapReduce puede ejecutarse en grandes conjuntos de máquinas y es altamente escalable: un ejemplo típico de MapReduce puede procesar muchos terabytes de datos en miles de máquinas. MapReduce es fácil de usar: cientos de programas que usan MapReduce han sido implementados y ejecutados en los servidores de Google cada día. (DEAN & GHEMAWAT, 2004)

Figura 2. Ejemplo del wordcount con MapReduce



Fuente: MAPREDUCEFOUNDATION

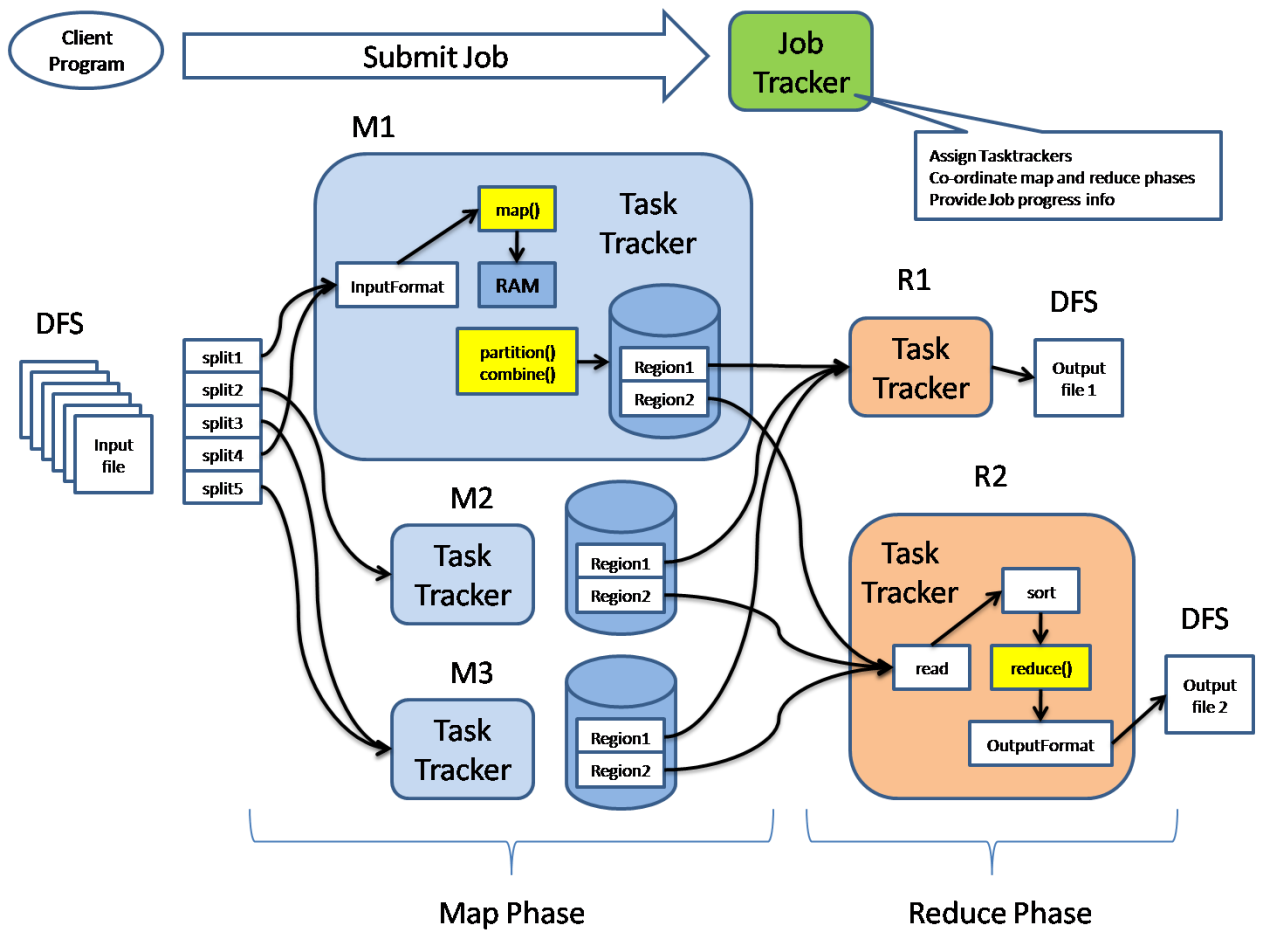
En la *Figura 2* se muestra un ejemplo común usando esta tecnología. Este ejemplo es el de contador de palabras donde se provee una lista de palabras como entradas, el *Map* se encarga de agrupar parejas (clave, valor), donde la clave corresponde a cada palabra que aparece en las entradas y el valor es 1 para cada una de ellas; para generar los datos intermedios es necesario pasar por un proceso *shuffle* donde se barajan y clasifican las parejas de datos con misma clave, finalmente el *Reduce* suma los valores de las parejas de datos que tengan la misma clave, obteniendo (clave2, valor2), donde la clave2 representa una

palabra sin repetir que aparezca en las entradas y el valor2 representa el número de veces que aparece dicha palabra en las entradas.

3.5. HADOOP MAPREDUCE

Hadoop MapReduce es un *framework* para la fácil escritura de aplicaciones que procesen vastas cantidades de datos (conjuntos de datos de varios terabytes) en paralelo en grandes grupos de máquinas (miles de nodos) de una manera confiable y con buena tolerancia a fallas.

Figura 3. Hadoop MapReduce



Fuente: DZone

Como se ve en la *Figura 3* un trabajo de MapReduce usualmente divide las entradas de los conjuntos de datos en pedazos que son procesados por la tarea de *Map* completamente paralela. El *framework* organiza los resultados de la función *Map*, que es entonces ingresada en la tarea de *Reduce*. Típicamente ambas entradas y resultados de un trabajo son almacenadas en un sistema de archivos. El *framework* se encarga de agendar las tareas, monitoreandolas y re-ejecutando las tareas fallidas.

Típicamente el nodo de cómputo y el nodo de almacenamiento son el mismo, eso es, el *framework MapReduce* y el *Hadoop Distributed File System* se ejecutan en el mismo conjunto de nodos. Esta configuración permite al *framework* agendar efectivamente las tareas en los nodos donde la información está presente, resultando en un gran ancho de banda agregado a través del *cluster*.

El *framework MapReduce* consiste de un único maestro *JobTracker* y un esclavo *TaskTracker* por cada nodo. El maestro es responsable por agendar las tareas a los esclavos, monitorearlas y re-ejecutar las tareas fallidas. Los esclavos ejecutan las tareas como lo ordena el maestro.

Mínimamente, las aplicaciones especifican las ubicaciones de las entradas/salidas y de declarar las funciones *Map* y *Reduce*, implementando las interfaces apropiadas y/o clases abstractas. Estas y otros parámetros comprenden la configuración del trabajo. El cliente de Hadoop entonces envía el trabajo (jar/ejecutable etc.) y la configuración del *JobTracker* asume la responsabilidad de distribuir el software/configuración a los esclavos, agendando tareas y monitoreandolas, proveyendo el estado e información de diagnóstico del trabajo en el cliente.

Aunque el *framework Hadoop* está implementado en Java, las aplicaciones *Mapreduce* no necesitan estar escritas en Java. (APACHE HADOOP, 2014)

3.6. OCTOPY

Octo.py es una implementación sencilla de *MapReduce* para Python, está inspirada en el *MapReduce* de Google y Starfish para Ruby.

Octo.py no está adecuada a todas las necesidades de la computación distribuida, pero es un enfoque simple adaptable a una gran proporción de tareas paralelizables. Si el código tiene un ciclo *for* existe una gran probabilidad que adaptar el código para hacerlo distribuido y si ya se tiene un código en Python para las funciones *Map* y *Reduce* sólo es necesario cambios triviales.

El alcance de este proyecto es más centrado en Starfish, ejecutando código en *clusters* de computadores, sin embargo la semántica de datos lleva mayor similitud con *MapReduce* de Google, con excepción en la parte del ordenamiento de resultados intermedios. (DIELS, 2014)

3.7. MINCEMEAT.PY

Mincemeat.py es una implementación en python del *framework* de computación distribuida *MapReduce*.

Todo el código está contenido en un solo archivo de python (actualmente pesa menos de 13 KB) que depende de las librerías estándar de python, cualquier computadora con python y mincemeat.py puede ser parte de un *cluster*. Los clientes pueden unirse y dejar el *cluster* en un tiempo variable sin afectar el proceso entero, además usa una autenticación en ambos lados de cada conexión, asegurando que sólo el código autorizado es ejecutado.

Mincemeat.py es distribuido bajo Licencia MIT y consecuentemente es libre para todos los usuarios, incluyendo su uso comercial, personal, y académico, y puede ser modificado y redistribuido sin restricciones. (FAIRLEY, 2014)

4. METODOLOGÍA

En el desarrollo de software existen una amplia variedad de metodologías que pueden ser aplicadas en proyectos, dependiendo siempre de las características propias de los mismos como la variación de requisitos, la calidad de software, número de desarrolladores del proyecto, tiempos de desarrollo, documentación del producto etc.

La plataforma desarrollada tenía definidas de manera general algunas de las funcionalidades, pero fue necesario realizar una verificación en cuanto a algunos detalles de funcionalidad, la presentación de los resultados y la interfaz con los usuarios del sistema, por lo anteriormente descrito este proyecto fue desarrollado usando la metodología Scrum que permite realizar iteraciones cortas para obtener resultados tangibles en poco tiempo.

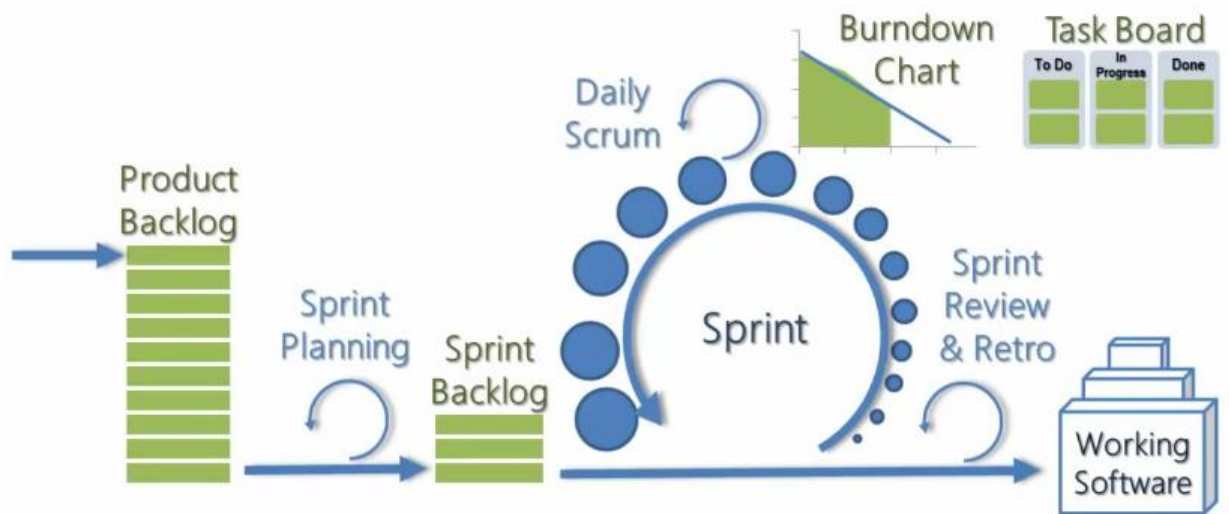
4.1. SCRUM

Scrum es una manera para que los equipos trabajen juntos en el desarrollo de un producto. El desarrollo de un producto usando Scrum ocurre en pequeñas partes, pieza a pieza se construye el producto final. Creando productos una parte a la vez estimula la creatividad y permite a los equipos responder a la realimentación y cambio, para construir lo que es necesario (SCHWABER & SUTHERLAND, 2014).

Más específicamente, Scrum es un marco de trabajo simple para equipos de colaboración efectivos en proyectos complejos. Scrum provee un pequeño conjunto de reglas que crean una estructura suficiente para que los equipos estén atentos en la innovación para resolver desafíos que de otro modo serían insuperables.

Sin embargo, Scrum es mucho más que un simple marco de trabajo, Scrum respalda las necesidades humanas en el trabajo: pertenecer, aprender, hacer, crear, crecer, mejorar, ser creativo e interactuar con otras personas. En otras palabras, Scrum aprovecha los rasgos y características innatas en las personas para que puedan hacer grandes cosas juntas. En la *Figura 4* se muestra la estructura de la metodología SCRUM.

Figura 4. Metodología SCRUM.



Fuente: Scrum.org

4.1.1. Backlog product. Es una lista ordenada de todo lo que se necesita en el producto y es la única fuente de requisitos para cualquier cambio que pueda tener el producto. Esto nunca está completo, las versiones tempranas sólo incluyen los requisitos mejor entendidos, es dinámico para identificar cambios en el producto que lo hagan más apropiado, competitivo y útil.

4.1.2. Sprint planning. Es una reunión que se realiza antes de empezar un *Sprint* donde principalmente se responde a dos preguntas: ¿Qué será entregado como resultado del *Sprint*?, ¿Cómo se trabajará para que ese resultado sea logrado?

4.1.3. Sprint backlog. Es un conjunto de requisitos seleccionados del *Product Backlog* para trabajarlos en el *Sprint* y además se incluye un plan para entregar el incremento del producto y de realización de la meta del *Sprint*.

4.1.4. Sprint. Es una iteración con duración de aproximadamente un mes o menos donde se crea un producto incremental, estas iteraciones tienen duración consistente con el esfuerzo de desarrollo, un nuevo *Sprint* comienza inmediatamente después de la conclusión del *Sprint* anterior.

Durante el *Sprint*:

- Ningún cambio que pueda afectar el objetivo del *Sprint* es realizado.
- La calidad del objetivo no disminuye.
- El equipo de desarrollo permanece constante.
- El alcance debe ser clarificado y renegociado entre el dueño del producto y el equipo de desarrollo.

4.1.5. Daily scrum. Es una reunión de 15 minutos donde el equipo de desarrollo se sincroniza y crea un plan para las próximas 24 horas. Esto es realizado por la inspección del trabajo realizado desde la última *Daily Scrum* y se pronostica el trabajo que puede ser realizado antes del próximo. En esta reunión cada miembro del equipo de desarrollo responde a estas preguntas:

- ¿Qué ha sido logrado desde la última reunión?
- ¿Qué debería ser hecho antes de la próxima reunión?
- ¿Qué obstáculos están presentes?

4.1.6. Sprint review & retro. Es una revisión que se realiza al final de un *Sprint* para inspeccionar el incremento y adaptar el *Product Backlog* si es necesario, esto incluye los siguientes elementos:

- ¿Qué fue “hecho” y qué no fue “hecho”?
- ¿Qué fue bien durante el Sprint?, ¿Qué problemas se presentaron y cómo se solucionaron?
- Se demuestra el trabajo que fue “hecho” y se responden preguntas acerca del incremento.
- El grupo colabora en lo que se debe hacer después.

4.1.7. Working software. Es el resultado del *Sprint*, lo que viene siendo el software entregable y funcional que se agrega al software general para hacer un incremento de las funcionalidades.

4.1.8. Burndown chart. Es un gráfico decreciente que se va alimentando con el resultado obtenido en cada iteración cada avance significa una caída en lo que falta por hacer en el proyecto.

4.1.9. Task board. Es un tablero donde se ponen las tareas por hacer, las que están en progreso y las que están realizadas. (SCHWABER & SUTHERLAND, 2014)

5. DESARROLLO DEL PROYECTO

En esta sección se muestran las características que fueron agregadas durante las iteraciones del proyecto teniendo en cuenta que se usó la metodología de desarrollo Scrum.

5.1. DESCRIPCIÓN

Esta plataforma web está compuesta de 6 módulos que facilitan el análisis de los comportamientos de los usuarios en cuanto a su consumo eléctrico y el análisis de los datos del transformador al que están asociados dichos usuarios.

5.1.1. Módulo panel de control. Este módulo muestra una clasificación por tipo de consumo de los medidores asociados al transformador en estudio, los medidores se dividen en medidores con consumo normal que corresponde a los usuarios que en todo momento tienen un registro de tensión, frecuencia y factor de potencia dentro de los rangos normales según NTC 5001/2008 (ICONTEC, 2008), medidores con consumo anormal que son los usuarios que no están dentro de los rangos de NTC 5001/2008 (ICONTEC, 2008) y medidores no residenciales.

Se muestra también tablas y gráfica por día de la semana de: La energía activa, la energía reactiva, la energía de distorsión, la potencia activa, la potencia reactiva, la potencia aparente, el porcentaje de uso del transformador y la potencia de distorsión correspondientes al transformador en estudio. La *Figura 5* muestra las características del módulo panel de control.

Figura 5. Panel de control.



Fuente: Autor

5.1.2. Módulo de resumen. En este módulo se evalúan y se puede exportar a Excel un reporte por día de la semana de los datos: Energía activa, energía reactiva, promedio del valor eficaz de la tensión, valor eficaz máximo de la tensión, valor eficaz mínimo de la tensión, porcentaje de permanencia en factor de potencia, frecuencia máxima, frecuencia mínima, potencia activa máxima, potencia activa mínima, potencia reactiva máxima y potencia reactiva mínima por cada usuario asociado al transformador en estudio. La *Figura 6* presenta las características del módulo resumen.

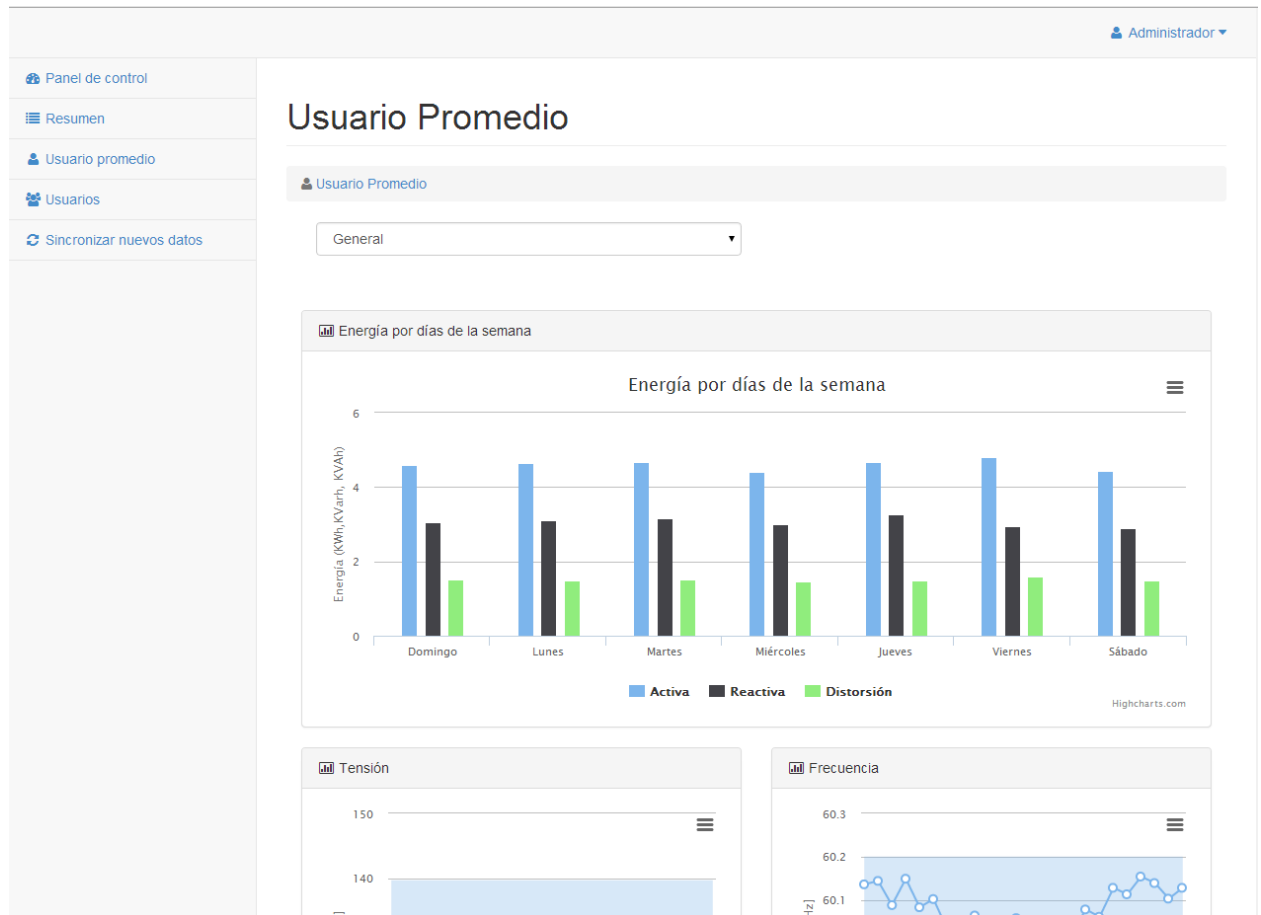
Figura 6. Tabla de resumen.

Contador		Energía P	Energía Q	Vprom	Vmax (Hora)	Vmin (Hora)	%Perm en fp	fpmáx (Hora)	fpmín (Hora)	Pmáx (Hora)	Pmín (Hora)	Qmáx (Hora)	Qmín (Hora)
1	Domingo	7.44	1.11	126.67	130.60	121.92	37.50	0.96	0.71	0.76	0.01	0.11	0.00
1	Lunes	5.63	1.15	125.50	129.91	119.77	16.67	0.95	0.65	0.84	0.03	0.12	0.02
1	Martes	8.21	0.98	125.34	128.99	122.22	37.50	0.96	0.73	0.74	0.08	0.07	0.02
1	Miércoles	5.76	1.07	125.39	130.75	119.22	29.17	0.94	0.72	0.57	0.02	0.09	0.00
1	Jueves	7.11	1.11	125.33	130.88	120.70	45.83	0.97	0.73	0.74	0.03	0.10	0.01
1	Viernes	5.98	0.90	125.24	129.20	121.62	41.67	0.96	0.74	0.69	0.02	0.07	0.00
1	Sábado	4.36	1.00	127.19	131.02	123.26	8.33	0.92	0.70	0.47	0.02	0.10	0.00
2	Domingo	3.02	1.89	126.67	130.63	121.86	8.33	0.93	0.51	0.30	0.01	0.24	0.00
2	Lunes	3.49	1.85	125.50	129.93	119.87	12.50	0.94	0.59	0.35	0.03	0.17	0.00
2	Martes	3.73	1.71	125.32	129.00	122.25	12.50	0.92	0.66	0.37	0.06	0.13	0.00
2	Miércoles	2.95	1.62	125.34	130.75	119.22	4.17	0.94	0.63	0.27	0.01	0.16	0.00

Fuente: Autor

5.1.3. Módulo usuario promedio. Este módulo clasifica por días de la semana o el total de los resultados, procesados en tablas y gráficas para los usuarios asociados al transformador de los siguientes parámetros: La energía activa, la energía reactiva, la energía de distorsión, el valor eficaz de la tensión, la frecuencia, el factor de potencia, la potencia activa, la potencia reactiva, la potencia aparente y la potencia de distorsión. La obtención de los datos de usuario promedio está basado en la norma IEC 61000-4-30 (IEC, 2004). En la *Figura 7* se muestra la vista del módulo de usuario promedio.

Figura 7. Usuario promedio.



Fuente: Autor

5.1.4. Módulo usuarios. En este módulo se presenta un listado de todos los medidores y usuarios que están asociados al transformador en estudio y se muestra la clasificación a la que pertenece, donde se puede realizar una búsqueda por usuario y ver los resultados de los datos procesados del usuario seleccionado.

Los datos de todos los usuarios son promediados por día de la semana y hora para almacenar la media y desviación estándar de cada dato, que es mostrado en forma gráfica y en tablas. Se considera que el consumo de energía eléctrica de los

usuarios se comporta como una distribución normal (JARDINI et al, 2000). La *Figura 8* muestra el módulo de usuarios.

Figura 8. Usuarios.

Usuario	# de contador	Consumo	Opciones
218090	1	Factor de potencia anormal	Ver
218091	2	Factor de potencia anormal	Ver
573662	3	Tensión anormal	Ver
218096	4	Tensión anormal	Ver
218095	5	Factor de potencia anormal	Ver
218094	6	Factor de potencia anormal	Ver
218084	7	Tensión anormal	Ver
218085	8	Tensión anormal	Ver
218086	9	Factor de potencia anormal	Ver
218087	10	Factor de potencia anormal	Ver
218088	11	Factor de potencia anormal	Ver

Fuente: Autor

5.1.5. Módulo de sincronización de nuevos dato. Este módulo muestra la información que sirve de guía para establecer los requisitos que se deben tener y los pasos a seguir para procesar correctamente nuevos datos recolectados en los medidores de los usuarios y se realiza un llamado a la acción a ejecutar para procesar los datos una vez finalizado los pasos. En la *Figura 9* se muestran las características de la vista del módulo de sincronización de datos.

Figura 9. Sincronizar nuevos datos.

Administrador ▾

Panel de control

Resumen

Usuario promedio




Usuarios

Sincronizar nuevos datos

Sincronizar nuevos datos

Sincronizar nuevos datos

En caso de necesitar actualizar los datos que alimentan este sistema, es necesario que siga con atención las siguientes instrucciones.

1. Ingrese al panel de administración del hosting de esta página, con el usuario y contraseña provistos anteriormente.
2. Haz clic en el icono  phpMyAdmin
3. Identifica el nombre de la base de datos provisto con anterioridad y clic en **conectar a esta base de datos**.
4. Para cada tabla selecciona la opción  Vaciar
5. Haz clic en la opción  **Importar**
6. Selecciona el archivo de la nueva base de datos.
7. Selecciona **utf-8** en **Conjunto de caracteres del archivo**
8. Selecciona la opción Permitir la interrupción de una importación en caso que el script detecte que se ha acercado al límite de tiempo PHP. (Esto podría ser un buen método para importar archivos grandes; sin embargo, puede dañar las transacciones.)
9. Elige el formato del archivo a importar.
10. En caso de ser necesario, selecciona la **Modalidad SQL compatible** y selecciona la casilla **No utilizar AUTO_INCREMENT con el valor 0**.
11. Clic en **continuar**.
12. En caso que el archivo de importación sea muy pesado, divide el archivo en varios subarchivos y repite los pasos del 5 al 11 hasta que la importación de todos los archivos se complete.
13. Click en el botón **Sincronizar** de abajo.

Sincronizar

Fuente: Autor

5.2. CICLOS DE DESARROLLO

Este proyecto se desarrolló usando la metodología Scrum por lo se realizaron varias iteraciones con diferentes requisitos en cada uno de ellos para alcanzar el producto que satisface las necesidades de los usuarios de la plataforma.

5.2.1. Roles

Cliente: Grupo de investigación GISEL

Scrum Master: Juan Sebastián Beleño Díaz (Autor de este proyecto)

Equipo:

Juan Sebastián Beleño Díaz (Desarrollador)

Estudiantes designados por el grupo GISEL para usar los datos de la plataforma

Raúl Ramos Pollán (Director de proyecto)

5.2.2. Primer sprint

5.2.2.1. Requisitos

- Crear un sistema de autenticación para evitar que los datos sean vistos por personas no deseadas.
- Para cada medidor se evalúan y muestran en tablas y gráficas por día de la semana y hora del día, la media y desviación estándar de: El valor eficaz de la tensión, la frecuencia, el factor de potencia, la potencia activa, la potencia reactiva y la potencia aparente.

5.2.2.2. Prototipo #1. El primer prototipo tiene como función principal garantizar la seguridad de los datos que fueron aportados por los grupos de investigación GISEL y CPS y mostrar un listado de todos los contadores asociados en el que se puedan observar gráficamente y en tablas los datos de los consumos por días de la semana y horas para observar de manera general los datos de cada contador.

La *Tabla 1* muestra las tareas asociadas a cada requisito especificado anteriormente, se muestran también las horas asociadas a las tareas y los días en lo que se realizaron.

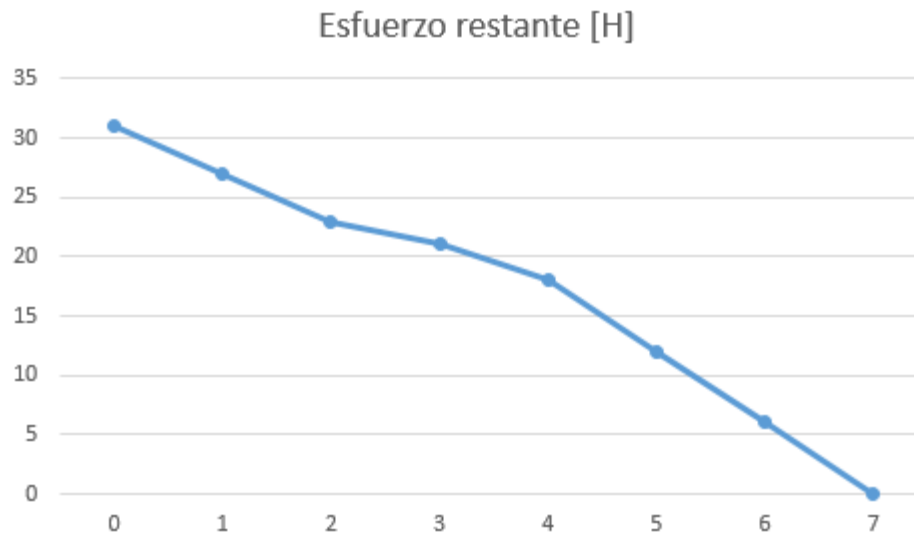
Requisito	Tarea	Intensidad en horas por día						
		Día 1	Día 2	Día 3	Día 4	Día 5	Día 6	Día 7
Crear un sistema de autenticación para evitar que los datos sean vistos por personas no deseadas.	Crear vista de inicio de sesión	2						
	Crear servicio de validación de usuario y contraseña		2					
	Crear vista de cambio de contraseña		2					
	Crear servicio de cambio de contraseña			2				
	Crear servicio para salir de				1			

	la sesión							
Para cada contador	Crear vista de usuarios	2						
calcular y mostrar en tablas y gráficas por día de la semana y hora del día	Crear servicio para listar a los medidores y los usuarios de esos medidores				2			
la media y desviación estándar de:	Crear vista de usuario					6		
el valor eficaz de la tensión, la frecuencia, el factor de potencia, la potencia activa, la potencia reactiva y la potencia aparente.	Crear servicio para sacar la media y desviación estándar de los datos y almacenarlos en BD						6	
	Crear servicio para listar la media y desviación estándar por contador							6

Tabla 1. Sprint Backlog del primer sprint

En la *Figura 10* se muestra el diagrama de progreso de la realización de las tareas asociadas a los requisitos del prototipo #1.

Figura 10. Burndown Chart del primer sprint



Fuente: Autor

5.2.2.3. Implementación y pruebas. Al intentar probar el sistema de autenticación, se encontró que el sistema ignoraba si el usuario o contraseña contenía minúsculas o mayúsculas, el problema fue solucionado realizando unas pequeñas modificaciones en la configuración que por defecto se incluye en MySQL, esto también solucionó el problema en la vista de cambio de contraseña. Los medidores son listados y permite acceder a los datos del contador seleccionado, esto se puede ver en las figuras 11 y 12. Los datos fueron verificados por los estudiantes de Ingeniería Eléctrica que realizan otro trabajo de grado simultáneo donde se analizan los resultados obtenidos (MEZA & ROSALES, 2015)

Figura 11. Usuarios – prototipo #1

Administrador ▾

Usuarios

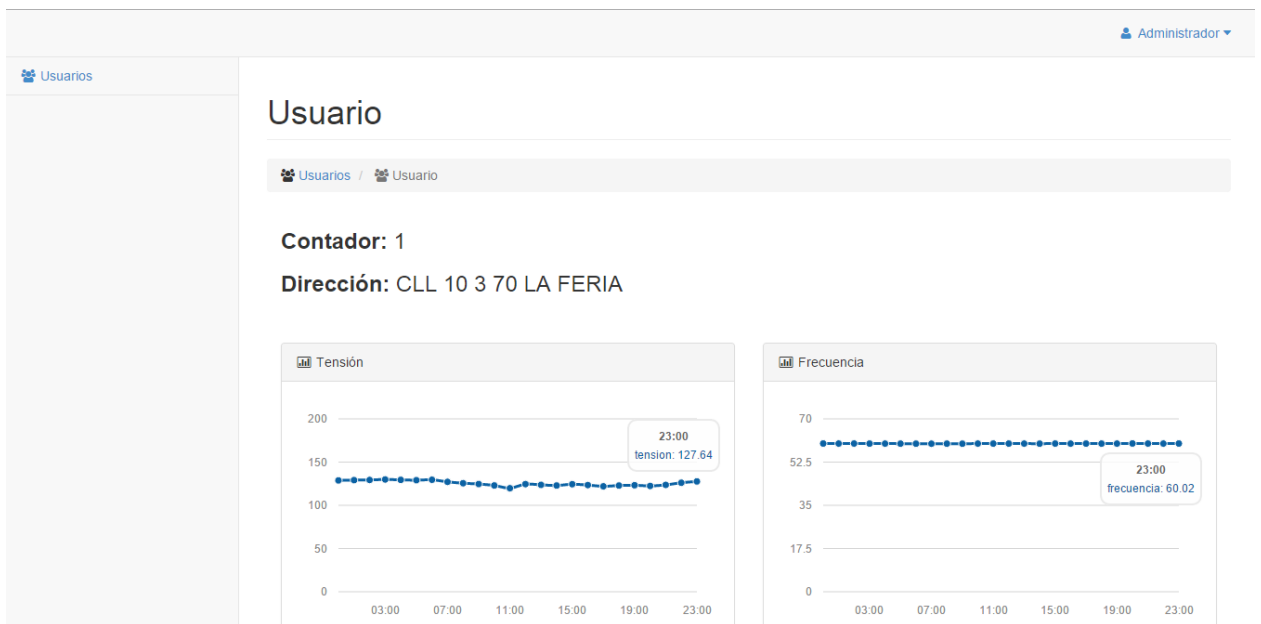
Usuarios

Usuarios

# de contador	Consumo	Opciones
1	Factor de potencia anormal	Ver
2	Factor de potencia anormal	Ver
3	Factor de potencia anormal	Ver
4	Factor de potencia anormal	Ver
5	Factor de potencia anormal	Ver
6	Factor de potencia anormal	Ver
7	Factor de potencia anormal	Ver

Fuente: Autor

Figura 12. Usuario – prototipo #1



Fuente: Autor

5.2.3. Segundo sprint

5.2.3.1. Requisitos

- Clasificar a los medidores por el comportamiento en el consumo de energía eléctrica y por sus características.
- Calcular la potencia de distorsión según el modelo propuesto por Budeanu para los medidores por día de la semana y hora del día.
- Evaluar para el transformador por día de la semana y hora del día la media y desviación estándar de: El valor eficaz de la tensión, la frecuencia, el factor de potencia, la potencia activa, la potencia reactiva, la potencia aparente y la potencia de distorsión.

5.2.3.2. Prototipo #2. Este prototipo agrega información útil a los datos que se muestran en la plataforma web, se separan los usuarios con consumo anormal (por valor eficaz de la tensión, la frecuencia y el factor de potencia), los usuarios con consumo normal y los usuarios no residenciales (como lámparas de alumbrado público).

Según la NTC 5001/2008 (ICONTEC, 2008) para que del valor eficaz de la tensión sean aceptables, es necesario que para el 95% de los datos del valor eficaz de la tensión se encuentre en el rango de $V_n \pm 10\%$ (para este caso 127 ± 12.7); en cuanto a la frecuencia en el 95% de los datos deben estar en el rango de 59,8 Hz a 60,2 Hz y para el factor de potencia el rango aceptable está entre 0.9 a 1; por lo que para cada medidor y teniendo en cuenta que los datos de los consumos de energía eléctrica residenciales se comportan como una distribución normal (JARDINI et al, 2000), se plantearon los siguientes condicionales para ubicar los datos entre los rangos mencionados anteriormente:

$$Si T_{prom} + 1.96 * \sigma < 139.7 \text{ AND } T_{prom} - 1.96 * \sigma > 114.3 \Rightarrow T \in \text{Rango adecuado}$$

$$Si F_{prom} + 1.96 * \sigma < 60.2 \text{ AND } F_{prom} - 1.96 * \sigma > 59.8 \Rightarrow F \in \text{Rango adecuado}$$

$$Si FP_{prom} - 1.96 * \sigma > 0,9 \Rightarrow FP \in \text{Rango adecuado}$$

Para evaluar los datos del transformador se suman los datos de media y desviación de los datos de todos los medidores asociados.

En la *Tabla 2* se muestra un desglosado de las tareas realizadas para satisfacer los requisitos de este sprint.

Requisito	Tarea	Intensidad en horas por día						
		Día 1	Día 2	Día 3	Día 4	Día 5	Día 6	Día 7
Clasificar a los medidores por el comportamiento en el consumo eléctrico y por sus características.	Crear vista de clasificación de medidores por consumo normal, consumo anormal y no residenciales	2	1					
	Crear servicio de clasificación de medidores normales, anormales y no residenciales		2	1				

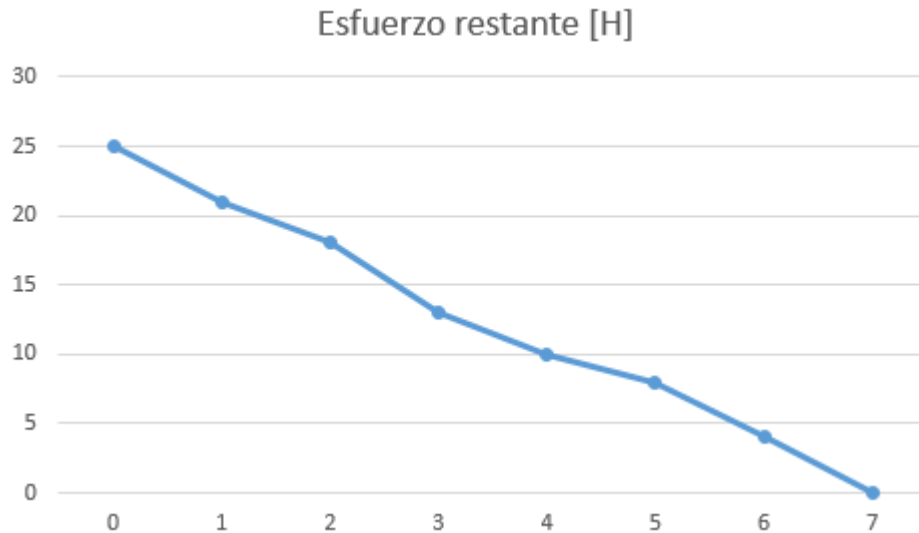
	<p>Crear vista de clasificación de medidores por consumo anormal por valor eficaz de la tensión, la frecuencia y el factor de potencia.</p>	2						
	<p>Crear servicio de clasificación de medidores de consumo anormal por valor eficaz de la tensión, la frecuencia y el factor de potencia.</p>			2				
<p>Calcular la potencia de distorsión según el modelo de potencias de Budeanu para los medidores</p>	<p>Modificar el servicio de cálculo de medias y desviaciones de datos para calcular la potencia de distorsión</p>			1	2			

por día de la semana y hora del día.	Modificar la vista de usuario para agregar la tabla y gráficas de la potencia de distorsión					2		
Calcular para el transformador por día de la semana y hora del día la media y desviación estándar del	Modificar el servicio de cálculo de medias y desviaciones de datos para calcular los datos del transformador						4	
valor eficaz de la tensión, la frecuencia, el factor de potencia, la potencia activa, la potencia reactiva, la potencia aparente y la potencia de distorsión.	Crear la vista del panel para mostrar los datos del transformador							4

Tabla 2. Sprint Backlog del segundo sprint

En la *Figura 13* se muestra el progreso por días del segundo sprint.

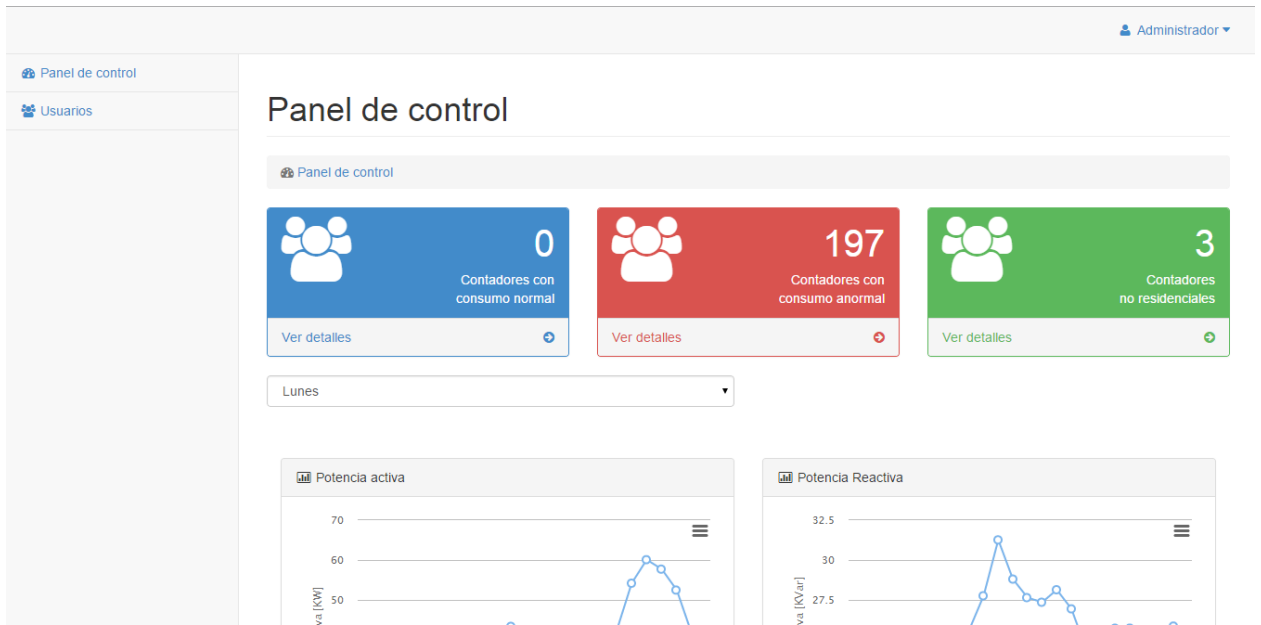
Figura 13. Burndown chart del segundo sprint



Fuente: Autor

5.2.3.3. Implementación y pruebas. La clasificación de los medidores fue realizada en la vista del panel de control donde también se muestran los datos del transformador. En la *Figura 14* se muestra lo realizado en estos 2 requisitos. La distorsión fue hallada según las formulas expresadas y se obtuvieron datos como los mostrados en la *Figura 15*. Los datos fueron verificados por los estudiantes de Ingeniería Eléctrica que están realizando simultáneamente el trabajo de grado (MEZA & ROSALES, 2015).

Figura 14. Panel de control – prototipo #2



Fuente: Autor

Figura 15. Potencia de distorsión de un usuario – prototipo #2

⚡ Potencia de distorsión		
Hora: 16:00		
Máximo: 0.2469205549 [KVar]		
Desviación estándar: 0.0239687179482 [KVar]		
Hora: 08:00		
Mínimo: 0.0220202217089 [KVar]		
Desviación estándar: 0.00992286015355 [KVar]		
Hora	Potencia de distorsión promedio [KVA]	Desviación estándar [KVA]
00:00	0.0543205381799	0.02
01:00	0.0255725541838	0.02
02:00	0.0494195173979	0.01
03:00	0.0414999602284	0.02
04:00	0.0465214454009	0.01
05:00	0.0674381796364	0.02
06:00	0.0247949228326	0.01

Fuente: Autor

5.2.4. Tercer sprint

5.2.4.1. Requisitos

- Calcular y mostrar gráficamente por día de la semana las energías activa, reactiva y de distorsión para los medidores y el transformador en estudio.
- Hacer escalable el cálculo de las medias y desviaciones estándar para los datos de los medidores usando MapReduce.
- Realizar un módulo de simulación de datos del consumo de energía eléctrica.

5.2.4.2. Prototipo #3. En este prototipo se incluyen conceptos de escalabilidad del cálculo de los datos de los consumos de energía eléctrica de los usuarios, debido a que se prevé un incremento en la cantidad de datos recolectados a mediano y largo plazo.

Para el cálculo de la energía se usa el método de trapecios para hallar la integral de cada contador en cada día basándose en el consumo por horas de la potencia.

Entre las opciones que se plantean en el marco teórico para MapReduce (Hadoop, Octo.py, Mincemeat.py) fue escogida Mincemeat.py debido a su facilidad de instalación, poco peso, escalabilidad, confiabilidad de resultados y facilidad de implementación, Octo.py fue descartada debido a que ha dejado de tener soporte desde el 2008 y Hadoop aunque tiene una confiabilidad y escalabilidad superior a las demás opciones requiere de un proceso engorroso de instalación en los nodos esclavos y en el servidor.

Para simular los datos se tomó como base el usuario promedio con sus desviaciones estándar y se realizó una función aleatoria basada en una distribución normal para arrojar los datos simulados de nuevos usuarios y medidores.

En la *tabla 3* se muestran las actividades realizadas para satisfacer los requisitos del tercer sprint del proyecto.

Requisito	Tarea	Intensidad en horas por día						
		Día 1	Día 2	Día 3	Día 4	Día 5	Día 6	Día 7
Calcular y mostrar gráficamente por día de la semana las	Modificar la vista de usuario para agregar la gráfica de las	2						

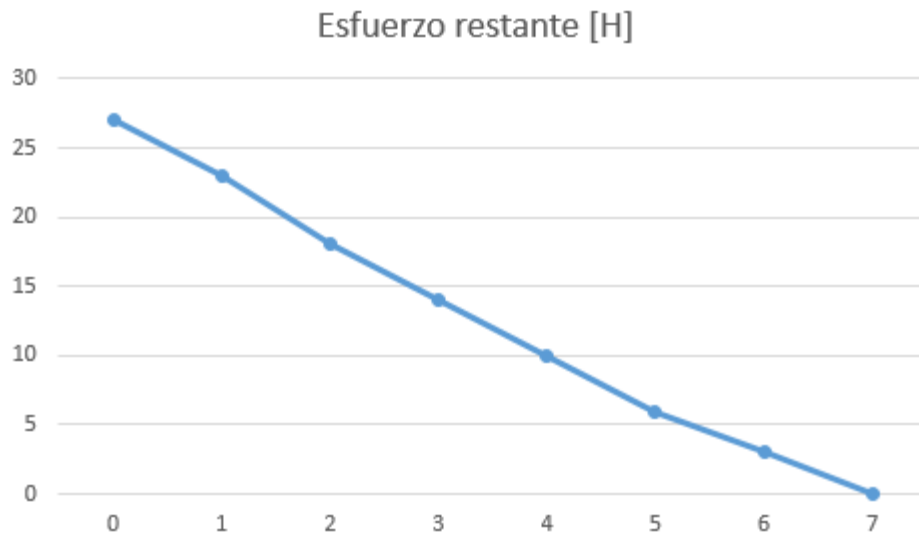
energías activa, reactiva y de distorsión para los medidores y el transformador en estudio.	energías activa, reactiva y de distorsión							
	Modificar la vista del panel para agregar la gráfica de la energía activa, reactiva y de distorsión al transformador	2						
	Realizar servicio para calcular y mostrar la energía activa por medidor y para el transformador		5					
Hacer escalable el cálculo de las medias y desviaciones estándar para los	Exportar los datos de la base de datos a CSV					2		

datos de los medidores usando MapReduce	Realizar un script en python usando MapReduce Mincemeat.py para calcular medias y desviaciones estándar de los medidores			4	4			
	Realizar un script en bash para medir el tiempo de ejecución del script en python					2		
Realizar un módulo de simulación de datos del consumo de energías eléctricas	Realizar un script en python para simular datos del consumo de energías eléctricas de los usuarios						3	3

Tabla 3. Sprint Backlog del tercer sprint

En la *Figura 16* se muestra el tiempo usado en satisfacer los requisitos del tercer sprint.

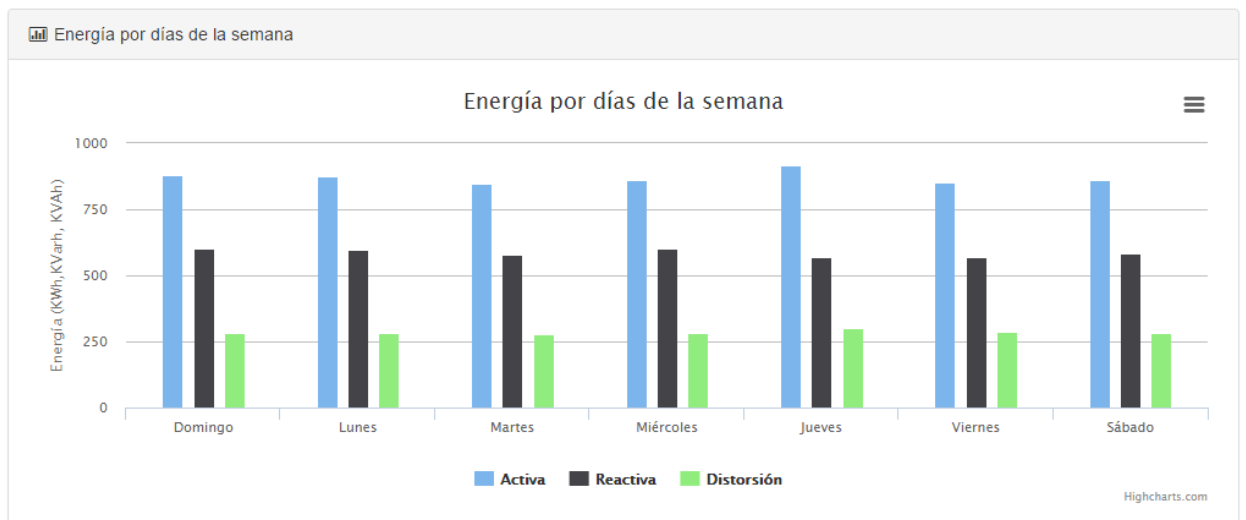
Figura 16. Burndown chart del tercer sprint



Fuente: Autor

5.2.4.3. Implementación y pruebas. La energía activa, reactiva y de distorsión es mostrada gráficamente en el comienzo de la presentación de datos de usuarios y para el transformador, como se puede apreciar en la Figura 17. Para la simulación y el desarrollo del script en MapReduce los resultados fueron corroborados con los datos en la base de datos. Los datos fueron verificados por los estudiantes de Ingeniería Eléctrica que realizan otro trabajo de grado simultáneo (Meza & Rosales, 2015).

Figura 17. Energía por días de la semana – prototipo #3



Fuente: Autor

5.2.5. Cuarto sprint

5.2.5.1. Requisitos

- Optimizar los scripts de simulación y de MapReduce para reducir el consumo de memoria.
- Realizar un resumen de los datos de cada medidor en una tabla que pueda ser descargada en Excel.

- Calcular el usuario promedio para el transformador en estudio.
- Habilitar un módulo donde se explique la forma de cargar nuevos datos.

5.2.5.2. Prototipo #4. Debido a que la cantidad de datos que se manejan pueden ocasionar problemas con la RAM se crean o se separan en archivos de máximo 500.000 registros al momento de simularlos o de prepararlos para el procesado.

Para poder tomarlos en el procesado con MapReduce se itera sobre los archivos con los datos, de esta manera solo se mantiene en memoria el contenido de un archivo a la vez.

Para hacer el resumen se seleccionan los datos procesados y almacenados en la base de datos y se muestran. A su vez creó una funcionalidad que permite exportar los datos en Excel.

También se muestra un módulo donde se realiza un llamado a la acción para cargar nuevos datos y procesarlos de manera fácil para alguien con conocimientos básicos de MySQL y phpMyAdmin.

En la *Tabla 4* se muestran las actividades necesarias para satisfacer los requisitos del cuarto sprint.

Requisito	Tarea	Intensidad en horas por día						
		Día 1	Día 2	Día 3	Día 4	Día 5	Día 6	Día 7
Optimizar los scripts de simulación y de MapReduce para reducir el consumo de memoria.	Modificar el script de simulador para que produzca una salida de varios archivos de máximo 500.000 registros	2						
	Modificar el script de MapReduce para iterar sobre archivos en el datasource	2						
Realizar un resumen de los datos de cada medidor en una tabla que pueda	Crear vista para el resumen		3					
	Crear servicio para listar el resumen			2				

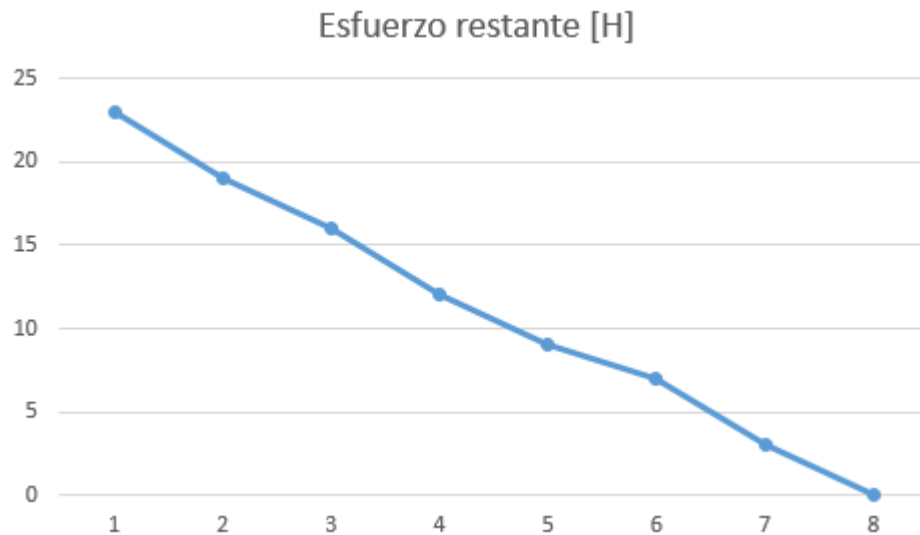
ser descargada en Excel.	Crear servicio para exportar a Excel			2				
Calcular el usuario promedio para el transformado r en estudio.	Crear vista para el usuario promedio				3			
	Crear el servicio que liste los datos de consumo del usuario promedio					2		
	Crear servicio para procesar los datos de todos los medidores y calcular los datos del usuario promedio						4	
Habilitar un módulo donde se explique la	Crear vista de sincronizació n de nuevos							3

forma de datos								
cargar nuevos datos.								

Tabla 4. Sprint Backlog del cuarto sprint

En la *Figura 18* se muestra los tiempos dedicados a satisfacer los requisitos del cuarto sprint.

Figura 18. Burndown chart del cuarto sprint



Fuente: Autor

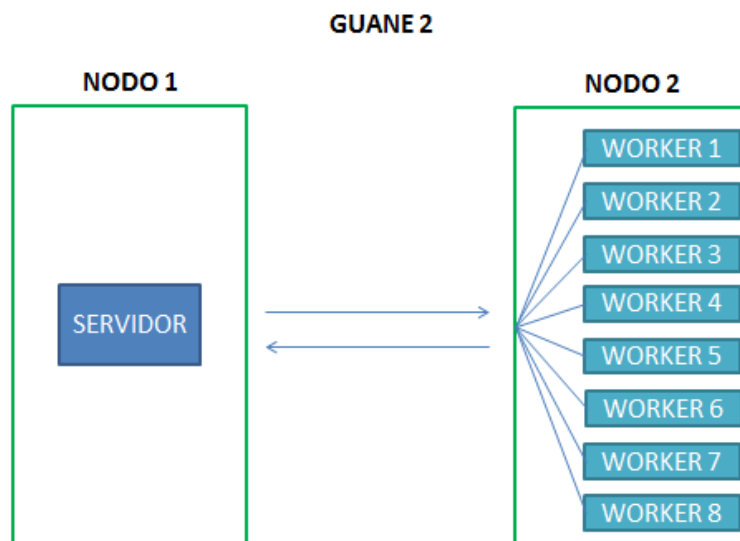
5.2.5.3. Implementación y pruebas. El desarrollo del resumen de los datos de los medidores de los usuarios y cálculo del usuario promedio usando la fórmula (5) fue validado por los estudiantes de Ingeniería Eléctrica (MEZA & ROSALES, 2014) quienes realizan un trabajo de grado simultáneo a este. En las figuras 6 y 7 se muestran los resultados. El módulo de sincronización de datos fue realizado para que luego de realizar los preparativos descritos en el mismo, se realice el

cálculo de los datos necesarios para alimentar la información de la plataforma, en la *Figura 9* se muestra una vista del módulo de sincronización de datos.

5.3. PRUEBAS DE ESCALABILIDAD

5.3.1. Arquitectura de pruebas. Las pruebas de escalabilidad del script en MapReduce fueron realizadas en el Guane 2, donde se usó un nodo como servidor y múltiples clientes en otro nodo como *workers*; cada nodo usado está conformado por 24 procesadores Intel Xeon E5645 que cuentan con 2.4GHz de procesamiento, en la *Figura 19* se puede apreciar el gráfico que muestra lo explicado de manera gráfica para una arquitectura de pruebas para 8 clientes, pero se usaron de 1 a 8 clientes en las pruebas para medir la escalabilidad.

Figura 19. Arquitectura de pruebas para 8 clientes



Fuente: Autor

5.3.2. Simulación de datos. Para la simulación de los datos de C.E.R. fue necesario tomar como base los resultados de los datos de usuarios reales y para hallar los datos de las energías activa, reactiva, de distorsión, el valor eficaz de la tensión, la frecuencia, el factor de potencia, las potencias activa, reactiva, aparente y de distorsión del usuario promedio según IEC 61000-4-30 (IEC, 2004) es necesario usar la siguiente ecuación:

$$Ah = \sqrt{\frac{\sum_{i=1}^N Bih}{N}} \quad (5)$$

Donde Bih representa la variable del medidor i en la hora h, N es el número de medidores asociados al transformador en estudio y Ah es la variable del usuario en la hora h.

Luego de tener los datos del usuario promedio también fue hallada la desviación estándar de cada uno de los datos y se desarrolló un script que crea datos C.E.R. siguiendo una distribución normal, estos datos son almacenados en archivos de máximo 500.000 registros cada uno.

Para ejecutar este script es necesario primero configurar los siguientes parámetros: el tamaño de los archivos, la frecuencia de toma de datos y el número de registros deseados.

5.3.3. Implementación. La ejecución del script en el nodo maestro se realiza escribiendo el siguiente comando:

```
/opt/anaconda/bin/python consumosusuario.py > solucion.txt
```

De esta manera se almacenan los resultados en el archivo solucion.txt, para la ejecución de los clientes es necesario ejecutar un script (*Figura 20*) para que los clientes sean ejecutados casi al mismo tiempo, debido a que la implementación de MapReduce de mincemeat es tolerante a fallos los clientes pueden conectarse y

desconectarse en cualquier momento sin afectar el resultado final, pero si afecta en el tiempo de cómputo de los datos.

Figura 20. Script en bash para ejecución de múltiples clientes

```

1 #!/bin/sh
2
3 echo "start"
4 date +"%s.%N"
5 python mincemeat.py -p electrical123 guane15&
6 # python mincemeat.py -p electrical123 guane15&
7 # python mincemeat.py -p electrical123 guane15&
8 # python mincemeat.py -p electrical123 guane15&
9 # python mincemeat.py -p electrical123 guane15&
10 # python mincemeat.py -p electrical123 guane15&
11 # python mincemeat.py -p electrical123 guane15&
12 # python mincemeat.py -p electrical123 guane15&

```

Fuente: Autor

5.3.4. Resultados. Los resultados de las pruebas de escalabilidad pueden ser observados en la *Tabla 5*.

# Registros	# Clientes	Tiempo 1 [s]	Tiempo 2 [s]	Tiempo medio [s]
30.000	1	7,6475	9,9350	8,7913
30.000	2	7,8714	4,9753	6,4233
30.000	3	7,3408	4,2987	5,8198
30.000	4	5,7995	5,4052	5,6023
30.000	5	6,3712	4,9725	5,6719
30.000	6	8,5946	6,0014	7,2980
30.000	7	6,4216	5,7513	6,0865
30.000	8	9,0979	6,2443	7,6711
50.000	1	14,3584	12,2639	13,3111
50.000	2	11,4994	9,6040	10,5517

50.000	3	13,3853	9,9692	11,6773
50.000	4	14,4931	8,3996	11,4464
50.000	5	12,1837	9,7540	10,9689
50.000	6	14,9659	9,6512	12,3086
50.000	7	13,7935	13,8286	13,8110
50.000	8	15,8653	11,0626	13,4639
75.000	1	15,2643	16,0949	15,6796
75.000	2	9,6253	9,7834	9,7044
75.000	3	8,9241	8,9271	8,9256
75.000	4	7,5781	7,8747	7,7264
75.000	5	7,9631	7,6278	7,7955
75.000	6	8,8011	8,3845	8,5928
75.000	7	9,1544	8,8195	8,9870
75.000	8	9,0638	9,2463	9,1551
100.000	1	16,2728	16,2502	16,2615
100.000	2	11,2957	11,1769	11,2363
100.000	3	10,0925	10,1474	10,1199
100.000	4	11,0742	10,2999	10,6871
100.000	5	10,0361	11,1791	10,6076
100.000	6	11,8766	10,9579	11,4172
100.000	7	14,9803	13,4080	14,1941
100.000	8	14,4430	14,4022	14,4226
222.955	1	30,0151	28,9167	29,4659
222.955	2	21,9392	20,9917	21,4654
222.955	3	20,6995	20,9660	20,8328
222.955	4	23,0053	21,0103	22,0078
222.955	5	21,8197	21,9423	21,8810
222.955	6	25,3258	24,3641	24,8449
222.955	7	27,7155	26,3137	27,0146

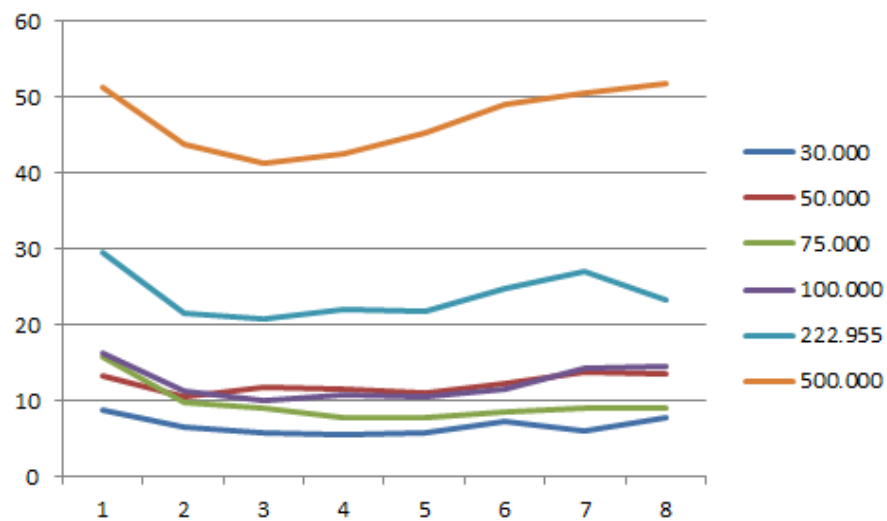
222.955	8	23,5849	23,0370	23,3110
500.000	1	49,9487	52,9024	51,4255
500.000	2	44,6044	43,1651	43,8848
500.000	3	40,6279	42,1813	41,4046
500.000	4	43,3257	41,7612	42,5435
500.000	5	44,9638	45,7777	45,3708
500.000	6	50,3609	47,6656	49,0133
500.000	7	49,9862	51,1708	50,5785
500.000	8	50,8249	52,6924	51,7586
2.000.000	1	168,3711	178,9538	173,6625
2.000.000	2	109,0932	111,3310	110,2121
2.000.000	3	104,4444	100,8865	102,6655
2.000.000	4	83,9182	88,7570	86,3376
2.000.000	5	87,6744	83,8957	85,7851
2.000.000	6	90,9030	83,8061	87,3545
2.000.000	7	82,8825	92,6767	87,7796
2.000.000	8	90,7556	85,6126	88,1841
5.000.000	1	401,0787	457,6052	429,3419
5.000.000	2	253,8117	244,9289	249,3703
5.000.000	3	203,2940	201,0972	202,1956
5.000.000	4	193,0077	183,5415	188,2746
5.000.000	5	166,5837	201,4420	184,0129
5.000.000	6	158,9150	156,1646	157,5398
5.000.000	7	150,2954	158,3086	154,3020
5.000.000	8	160,0809	157,7570	158,9189
10.000.000	1	405,2550	797,6118	601,4334
10.000.000	2	479,6972	478,6886	479,1929
10.000.000	3	389,0985	389,1701	389,1343
10.000.000	4	333,0780	318,8939	325,9859

10.000.000	5	305,2977	306,8670	306,0823
10.000.000	6	295,0362	301,2874	298,1618
10.000.000	7	275,6956	289,7390	282,7173
10.000.000	8	276,9682	258,7466	267,8574

Tabla 5. Tabla de escalabilidad del procesamiento de datos C.E.R. con MapReduce

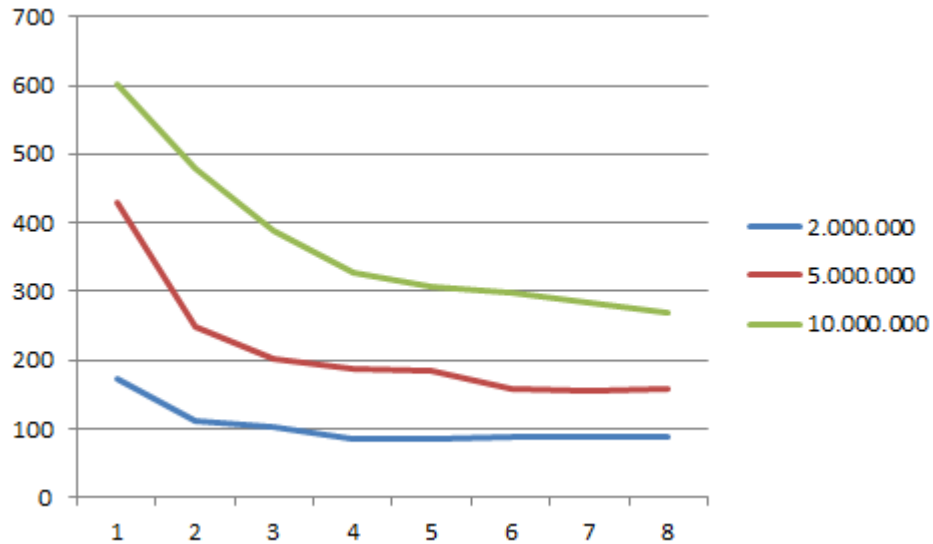
En las figuras 21 y 22 se muestran respectivamente las gráficas de escalabilidad para volúmenes de datos C.E.R. menores y volúmenes de datos C.E.R. mayores respectivamente.

Figura 21. Curva de escalabilidad para volúmenes de datos C.E.R. menores



Fuente: Autor

Figura 22. Curva de escalabilidad para volúmenes de datos C.E.R. mayores



Fuente: Autor

Aunque la plataforma creada procesa de manera escalable cuando volúmenes de datos C.E.R. mayores, esta escalabilidad se ve afectada debido a los siguientes factores:

- Cantidad de datos: la solución de problemas con MapReduce está pensada para trabajar con muchos datos, por lo que cuantos más datos existan, mejor se comportará la curva de escalabilidad.
- Algoritmo *Shuffle*: MapReduce usa un algoritmo *Shuffle* en la mitad del proceso entre el Map y el Reduce, este algoritmo puede ser implementado de muchas maneras para permitir que todas las salidas con igual clave del Map sean procesadas por el Reduce de un solo cliente, pero debido a que se usó `mincemeat.py`, el algoritmo *Shuffle* no está tan optimizado como lo está en Hadoop.

- Transporte de datos: debido a que las pruebas fueron realizadas en nodos diferentes, el transporte de datos consume una porción del tiempo de ejecución del script.
- Lectura de datos: debido a que es posible que se cuente con una memoria limitada al momento de procesar datos, la fuente de datos tiene que ser particionada en archivos más pequeños, lo que significa un aumento en el tiempo de procesamiento debido a operaciones de abrir, leer, procesar y cerrar cada archivo.

6. CONCLUSIONES

- Se procesaron 222.955 datos C.E.R. pertenecientes a usuarios del barrio La Feria en el municipio de Piedecuesta, Santander y se mostraron resultados para el análisis adecuado según lo requerido por los grupos de investigación GISEL y CPS.
- Se investigó las diferentes opciones con las cuales se pueden procesar los datos C.E.R. usando MapReduce y se escogió la librería en Python mincemeat.py para realizar un procesamiento escalable de datos C.E.R.
- Se desarrolló una plataforma que permite procesar de manera escalable usando la librería de MapReduce en Python mincemeat.py y mostrar los datos procesados en tablas y gráficos para facilitar el análisis de los comportamientos de los usuarios según los consumos que se registran en los medidores eléctricos.
- Se creó un script en Python que permite la simulación de datos C.E.R. de usuarios ficticios según los parámetros que provea el administrador de la plataforma y además permite dividir los datos en múltiples archivos para evitar problemas de memoria al momento de procesarlos.
- Se realizaron mediciones y se evaluaron los tiempos de ejecución de la plataforma al momento de procesar datos C.E.R. con diferente número de clientes y diferentes volúmenes de datos, dentro de los que se encuentran los 222.995 datos de usuarios reales del barrio La Feria – Piedecuesta y también volúmenes de datos mayores.

- Se documentó la instalación y uso de plataforma para facilitar uso de cada uno de sus módulos a los usuarios, dentro de lo que se encuentra la forma de cargar nuevos datos para procesarlos y analizarlos en los diferentes módulos que contiene la plataforma.
- Se solucionó un problema real usando una parte de los conocimientos aprendidos durante la carrera de Ingeniería de Sistemas, por lo que sirvió como un acercamiento a la aplicación de los conocimientos adquiridos en la universidad en la futura vida profesional.

7. RECOMENDACIONES

- Cambiar el algoritmo de procesamiento escalable que usa mincemeat.py para que se adapte al uso de Hadoop que es una herramienta más poderosa que la librería que se usó en este proyecto, además garantiza un procesamiento más rápido que el que posee la plataforma actualmente.
- Adaptar la plataforma para que trabaje con datos de múltiples transformadores eléctricos a la vez, debido a que en este momento solo se puede procesar información de datos C.E.R. pertenecientes a un transformador.
- Implementar algoritmos de minería de datos para descubrir nueva información que sea de valor para la toma de decisiones por parte del usuario de la plataforma.
- Mejorar el algoritmo de simulación de datos C.E.R. para garantizar una mayor precisión y semejanza a los datos C.E.R. de usuarios reales.

BIBLIOGRAFÍA

APACHE HADOOP. MapReduce Tutorial. {En línea}. {6 de Diciembre, 2014}. Disponible en: (hadoop.apache.org/docs/r1.2.1/mapred_tutorial.html).

COULOURIS, George, DOLLIMORE, Jean, KINDBERG, Tim, y BLAIR, Gordon. Distributed System: Concepts and design (5th Edition). Boston: Addison-Wesley, 2011.

DEAN, Jeffrey y GHEMAWAT, Sanjay. MapReduce: Simplified Data Processing on Large Clusters. Symposium on Operating Systems Design & Implementation (Diciembre, 2004). San Francisco, USA.

DIELS, Tim. Octopy: Easy MapReduce for Python. {En línea}. {6 de Diciembre, 2014}. Disponible en (code.google.com/p/octopy/).

FAIRLEY, Michael. Mincemeat: Lightweight MapReduce in python. {En línea}. {6 de Diciembre, 2014}. Disponible en: (github.com/michaelfairley/mincemeatpy)

GALLEGO, Luis, FRANCISQUINI, Aislan, GÓMEZ, Oscar y PADILHA Antonio. Estimación de curvas de demanda de consumidores, transformadores de distribución y alimentadores primarios en sistemas de distribución. En: Scientia et Technica Año XIII, No. 35 (2007).

INSTITUTO COLOMBIANO DE NORMAS TÉCNICAS Y CERTIFICACIÓN. Norma Técnica Colombiana- NTC 5001/2008. Calidad de la potencia eléctrica. Límites y metodología de evaluación en punto de conexión común. Bogotá, ICONTEC, 2008.

INTERNATIONAL ELECTROTECHNICAL COMMISSION. IEC 61000 Electromagnetic compatibility (EMC): Testing and measurement techniques. Power quality measurement methods. Geneva, Suiza, IEC, 2004.

JARDINI, José, TAHAN, Carlos, UN, Se, GOUVEA, M., FIGUEIREDO F. Daily Load Profiles for Residential, Commercial and Industrial Low Voltage Consumers. En: IEEE Transaction on Power Delivery, Vol. 15; No. 1 (2000).

MARQUES, Guilherme. Curvas típicas de carga para o planejamento operacional do sistema de distribuição. São Paulo, 2006. Dissertação de Mestre em Engenharia. Universidade de São Paulo.

MEZA, Marcelo y ROSALES, Ray. Caracterización de la carga eléctrica residencial ubicada en el sector La Feria – Piedecuesta. Bucaramanga, 2015. Trabajo de investigación (Pregrado en ingeniería eléctrica) Universidad Industrial de Santander. Facultad de ingenierías físico-mecánicas.

ORDÓÑEZ, Gabriel. Conferencias Mediciones Electricas II. Bucaramanga, Colombia, 2006.

RAMÍREZ, Xiomara, PEREA, Ángela, CAICEDO, Gladys y CASTRO Ferley. Estimación de curvas de carga en transformadores eléctricos mediante redes neuronales. En: Ingeniería y competitividad, Vol.; 10. No. 2 (2008); p. 75 - 86.

SCHWABER, Ken y SUTHERLAND, Jeff. The Scrum Guide. {En línea}. {5 de Diciembre de 2014}. Disponible en: (www.scrum.org/Portals/0/Documents/Scrum%20Guides/Scrum_Guide.pdf)

SERWAY, Raymond y JEWETT, John. Física para ciencias e ingeniería con física moderna. 7 edición. Vol. 2. México: Cengage Learning, 2009.

ANEXOS

Anexo A. Manual de uso de la plataforma

PRE-REQUISITOS

La plataforma fue desarrollada usando como lenguajes base PHP, Python, HTML5, CSS3, JavaScript y para la base de datos se usó MySQL y archivos CSV para almacenar los datos de la aplicación. El servidor que aloje la plataforma debe contar con las siguientes especificaciones:

- Un servidor Apache
- PHP 5.3.0 o superior instalado
- MySQL 5.1.37 o superior instalado
- Python 2.7 instalado
- Extensión Mcrypt para PHP

Se recomienda la instalación de XAMPP para cubrir las especificaciones del servidor Apache, PHP y MySQL. La plataforma también necesita del framework de PHP CodeIgniter 2.2.0 y de Mincemeat.py 0.1.4, pero estos ya se encuentran incluidos dentro de la plataforma. Temporalmente también se necesita de un computador con varios procesadores o núcleos para realizar el procesamiento de datos de manera escalable cuando existan muchos datos.

INSTALACIÓN

- Copiar la carpeta `plataforma` y `mapreduce` en el servidor Apache
- Cargar la base de datos en MySQL
- Agregar las siguientes tablas en SQL

```
CREATE TABLE IF NOT EXISTS `bd_contadores` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `contador` text NOT NULL,  
  `consumo` text NOT NULL,  
  `usuario` text NOT NULL,  
  PRIMARY KEY (`id`)  
)
```

```
CREATE TABLE IF NOT EXISTS `bd_datos` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `nocontador` text NOT NULL,  
  `mes` text NOT NULL,  
  `dia` text NOT NULL,  
  `hora` text NOT NULL,  
  `tension_media` text NOT NULL,  
  `tension_desviacion` text NOT NULL,  
  `frecuencia_media` text NOT NULL,  
  `frecuencia_desviacion` text NOT NULL,  
  `factorp_media` text NOT NULL,  
  `factorp_desviacion` text NOT NULL,  
  `activa_media` text NOT NULL,  
  `activa_desviacion` text NOT NULL,  
  `reactiva_media` text NOT NULL,
```

```

`reactiva_desviacion` text NOT NULL,
`aparente_media` text NOT NULL,
`aparente_desviacion` text NOT NULL,
`distorsion_media` text NOT NULL,
`distorsion_desviacion` text NOT NULL,
PRIMARY KEY (`id`)
)

```

```

CREATE TABLE IF NOT EXISTS `bd_energia` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `tipo` text NOT NULL,
  `contador` text NOT NULL,
  `dia` text NOT NULL,
  `valor` text NOT NULL,
  PRIMARY KEY (`id`)
)

```

```

CREATE TABLE IF NOT EXISTS `bd_resumen` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `contador` text NOT NULL,
  `dia` text NOT NULL,
  `energiap` text NOT NULL,
  `energiaq` text NOT NULL,
  `vprom` text NOT NULL,
  `vmin` text NOT NULL,
  `vmax` text NOT NULL,
  `perm` text NOT NULL,
  `fpmax` text NOT NULL,
  `fpmin` text NOT NULL,
  `pmax` text NOT NULL,

```

```

`pmin` text NOT NULL,
`qmax` text NOT NULL,
`qmin` text NOT NULL,
PRIMARY KEY (`id`)
)

CREATE TABLE IF NOT EXISTS `bd_usuarios` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `usuario` text NOT NULL,
  `password` text NOT NULL,
  `periodicidad` text NOT NULL,
  PRIMARY KEY (`id`)
)

```

- En la tabla `bd_usuarios` agregar los usuarios que deben tener acceso a la plataforma con una clave por defecto que debe ser modificada en la pestaña cambio de contraseña de la plataforma.
- Modificar el archivo ubicado en `/plataforma/application/config/database.php`, establecer los datos de la base de datos que se va a utilizar para procesar los datos.

PROCESADO DE DATOS

Para procesar los datos se cuenta con 2 opciones, una para cuando existen pocos datos y otra para cuando existen muchos datos. Para la opción de pocos datos (menores de ~1.000.000) se siguen los siguientes pasos:

- En el módulo de sincronizar nuevos datos, clic en Sincronizar
- En el módulo resumen, clic en Sincronizar resumen

Para cuando los datos existentes son superiores a ~1.000.000 es conveniente realizar el procesamiento con más de un cliente para acelerar el tiempo de procesamiento, para esto se siguen los siguientes pasos:

- En el Nodo Maestro se ejecuta el siguiente comando `python consumosxusuario.py > datos.txt`
- En los nodos esclavos se ejecuta `python mincemeat.py -p clave hostname`. Donde la clave por defecto es `electrica123` y si el maestro y los clientes se ejecutan en la misma máquina el `hostname` es igual a `localhost`
- El archivo `datos.txt` se pasa a la ubicación `/plataforma/datos.txt` y se alimenta la base de datos con ese archivo que está en formato JSON.

INGRESO A LA PLATAFORMA

Una vez configurado todo lo anteriormente descrito, se puede acceder a la plataforma, pero es necesario poseer un usuario y contraseña. En la *Ilustración 1* se muestra la pantalla de inicio de sesión.

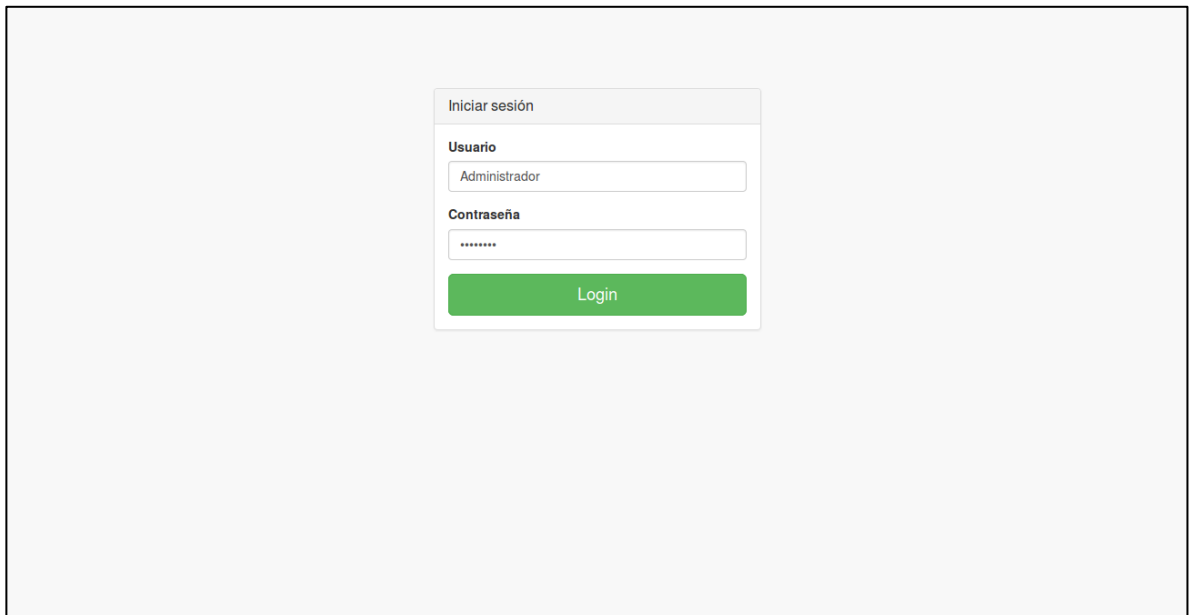


Ilustración 1: Inicio de sesión

CAMBIAR CONTRASEÑA

Para cambiar contraseña:

- Clic en Adimistrador (ver *Ilustración 2*)
- Clic en configuraciones

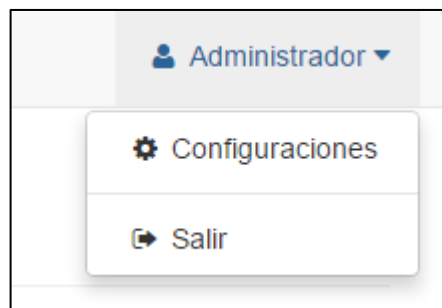


Ilustración 2: Menú desplegable

- Se escribe la antigua y nueva contraseña para cambiarla
- Clic en guardar (ver *Ilustración 3*)

Ilustración 3: Configuraciones

CERRAR SESIÓN

- Clic en Administrador
- Clic en salir (ver *Ilustración 2*)

MÓDULO PANEL DE CONTROL

Este módulo muestra una clasificación por tipo de consumo de los medidores asociados al transformador en estudio, los medidores se dividen en medidores con consumo normal que corresponde a los usuarios que en todo momento tienen un registro de tensión, frecuencia y factor de potencia dentro de los rangos normales según NTC 5001/2008 (ICONTEC, 2008), medidores con consumo anormal que son los usuarios que no están dentro de los rangos de NTC 5001/2008 (ICONTEC, 2008) y medidores no residenciales.

Se muestra también tablas y gráfica por día de la semana de la energía activa, energía reactiva, energía de distorsión, potencia activa, potencia reactiva, potencia aparente, porcentaje de uso del transformador y potencia de distorsión correspondientes al transformador en estudio; de los datos anteriores se halla también los máximos y mínimos con sus respectivas horas (ver *Ilustración 4*).

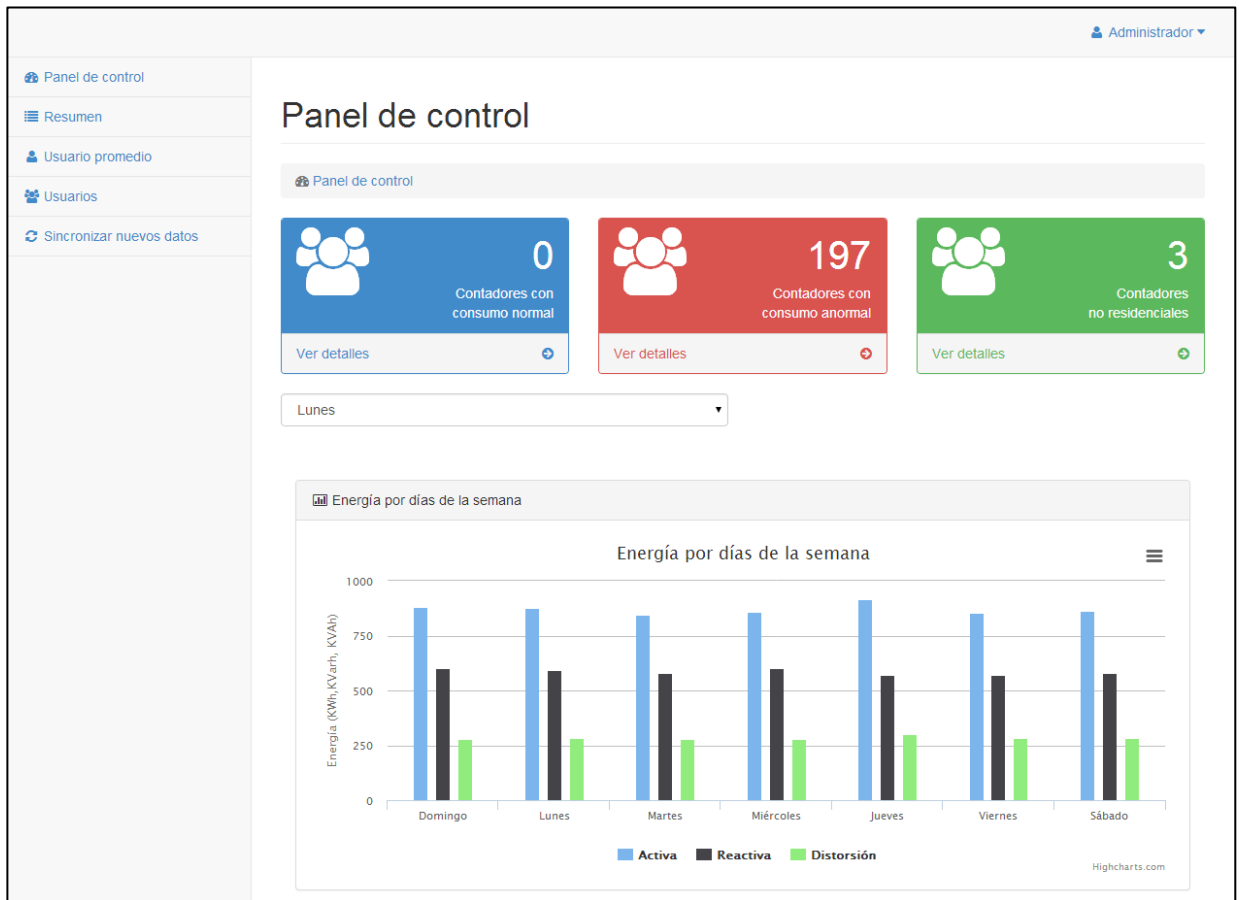


Ilustración 4: Panel de control

Al cambiar el día de la semana del selector los datos se actualizan al día seleccionado, si se selecciona los medidores con consumo normal, se listan los

medidores con consumo normal como se puede ver en la *Ilustración 5*; además permite la búsqueda de medidor por identificador de usuario.

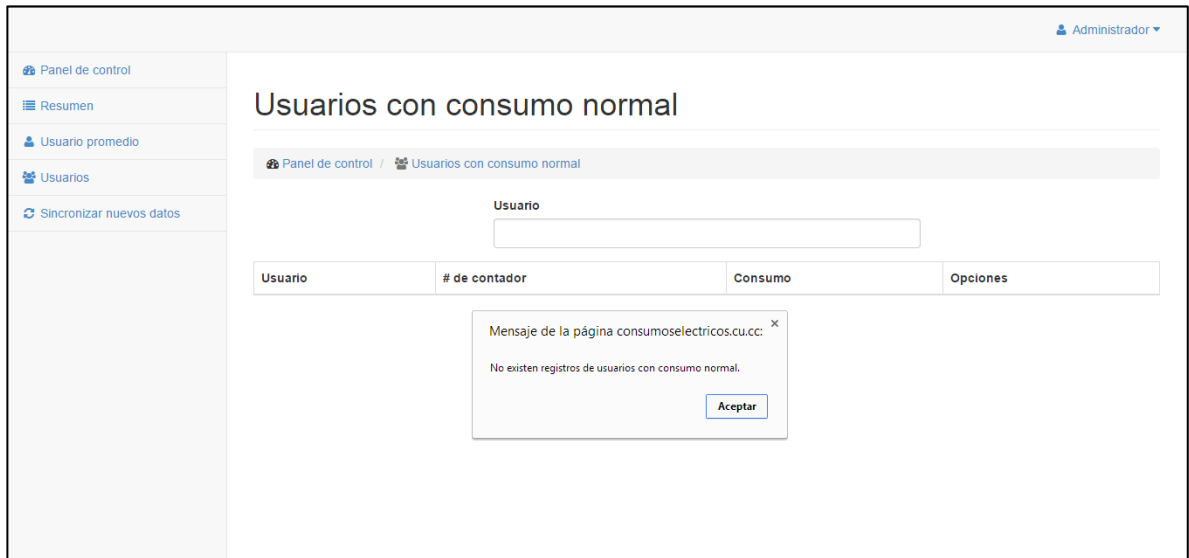


Ilustración 5: Usuarios con consumo normal

Si se seleccionan los medidores con consumo no residencial, estos se listan y se permite la búsqueda por el identificador de usuario (ver *Ilustración 6*).

Panel de control / Medidores con consumo no residencial

Usuario

Usuario	# de medidor	Consumo	Opciones
01	33	Contador no residencial	Ver
02	123	Contador no residencial	Ver
L99998	209	Contador no residencial	Ver

consumoselectricos.cu.cc/web/dashboard

Ilustración 6: Medidores con consumo no residencial

Si se selecciona los medidores con consumo anormal se listan los medidores con consumo anormal y además se clasifican por tipo de anomalía: por tensión, frecuencia y por factor de potencia, en ese orden de prioridad que al seleccionar alguno se listan los datos (ver *Ilustración 7*, *Ilustración 8*, *Ilustración 9* y *Ilustración 10*); además permite la búsqueda de medidor por identificador de usuario.



Ilustración 7: Usuarios con consumo anormal



Ilustración 8: Usuarios con consumo anormal por tensión

Administrador ▾

Panel de control

Resumen

Usuario promedio

Usuarios

Sincronizar nuevos datos

Usuarios con consumo anormal por frecuencia

Panel de control / Usuarios con consumo anormal / Usuarios con consumo anormal por frecuencia

Usuario

Usuario	# de contador	Consumo	Opciones
568112	17	Frecuencia anormal	Ver
1260957	20	Frecuencia anormal	Ver
218074	35	Frecuencia anormal	Ver
1117599	38	Frecuencia anormal	Ver
218935	45	Frecuencia anormal	Ver
1265573	56	Frecuencia anormal	Ver

Ilustración 9: Usuarios con consumo anormal por frecuencia

Administrador ▾

Panel de control

Resumen

Usuario promedio

Usuarios

Sincronizar nuevos datos

Usuarios con consumo anormal por factor de potencia

Panel de control / Usuarios con consumo anormal / Usuarios con consumo anormal por factor de potencia

Usuario

Usuario	# de contador	Consumo	Opciones
218090	1	Factor de potencia anormal	Ver
218091	2	Factor de potencia anormal	Ver
218095	5	Factor de potencia anormal	Ver
218094	6	Factor de potencia anormal	Ver
218086	9	Factor de potencia anormal	Ver
218087	10	Factor de potencia anormal	Ver

Ilustración 10: Usuarios con consumo anormal por factor de potencia

MÓDULO DE RESUMEN

Se calculan y permite exportar a Excel un reporte por día de la semana los datos de energía activa, energía reactiva, tensión promedio, tensión máxima, tensión mínima, porcentaje de permanencia en factor de potencia, frecuencia máxima, frecuencia mínima, potencia activa máxima, potencia activa mínima, potencia reactiva máxima y potencia reactiva mínima por cada usuario asociado al transformador en estudio, además cuando se cargan nuevos datos se tiene que sincronizar resumen para actualizar los datos de resumen (ver *Ilustración 11*).

Contador		Energía P	Energía Q	Vprom	Vmax (Hora)	Vmin (Hora)	%Perm en fp	fpmax (Hora)	fpmin (Hora)	Pmax (Hora)	Pmin (Hora)	Qmax (Hora)	Qmin (Hora)
1	Domingo	7.44	1.11	126.67	130.60	121.92	37.50	0.96	0.71	0.76	0.01	0.11	0.00
1	Lunes	5.63	1.15	125.50	129.91	119.77	16.67	0.95	0.65	0.84	0.03	0.12	0.02
1	Martes	8.21	0.98	125.34	128.99	122.22	37.50	0.96	0.73	0.74	0.08	0.07	0.02
1	Miércoles	5.76	1.07	125.39	130.75	119.22	29.17	0.94	0.72	0.57	0.02	0.09	0.00
1	Jueves	7.11	1.11	125.33	130.88	120.70	45.83	0.97	0.73	0.74	0.03	0.10	0.01
1	Viernes	5.98	0.90	125.24	129.20	121.62	41.67	0.96	0.74	0.69	0.02	0.07	0.00
1	Sábado	4.36	1.00	127.19	131.02	123.26	8.33	0.92	0.70	0.47	0.02	0.10	0.00
2	Domingo	3.02	1.89	126.67	130.63	121.86	8.33	0.93	0.51	0.30	0.01	0.24	0.00
2	Lunes	3.49	1.85	125.50	129.93	119.87	12.50	0.94	0.59	0.35	0.03	0.17	0.00
2	Martes	3.73	1.71	125.32	129.00	122.25	12.50	0.92	0.66	0.37	0.06	0.13	0.00
2	Miércoles	2.95	1.62	125.34	130.75	119.22	4.17	0.94	0.63	0.27	0.01	0.16	0.00

Ilustración 11: Tabla resumen

MÓDULO USUARIO PROMEDIO

Este módulo clasifica por días de la semana o un total de los resultados, procesados en tablas y gráficas para los usuarios asociados al transformador de las cantidades promedio de: energía activa, energía reactiva, energía de distorsión, tensión, frecuencia, factor de potencia, potencia activa, potencia reactiva, potencia aparente y potencia de distorsión. El cálculo de los datos de usuario promedio está basado en IEC 61000-4-30 (IEC, 2004); además se hallan los máximos y mínimos de los datos con su respectiva hora (Ver Ilustración 12).

Los datos en rojo quiere decir que no se ajustan a IEC 61000-4-30 (IEC, 2004).

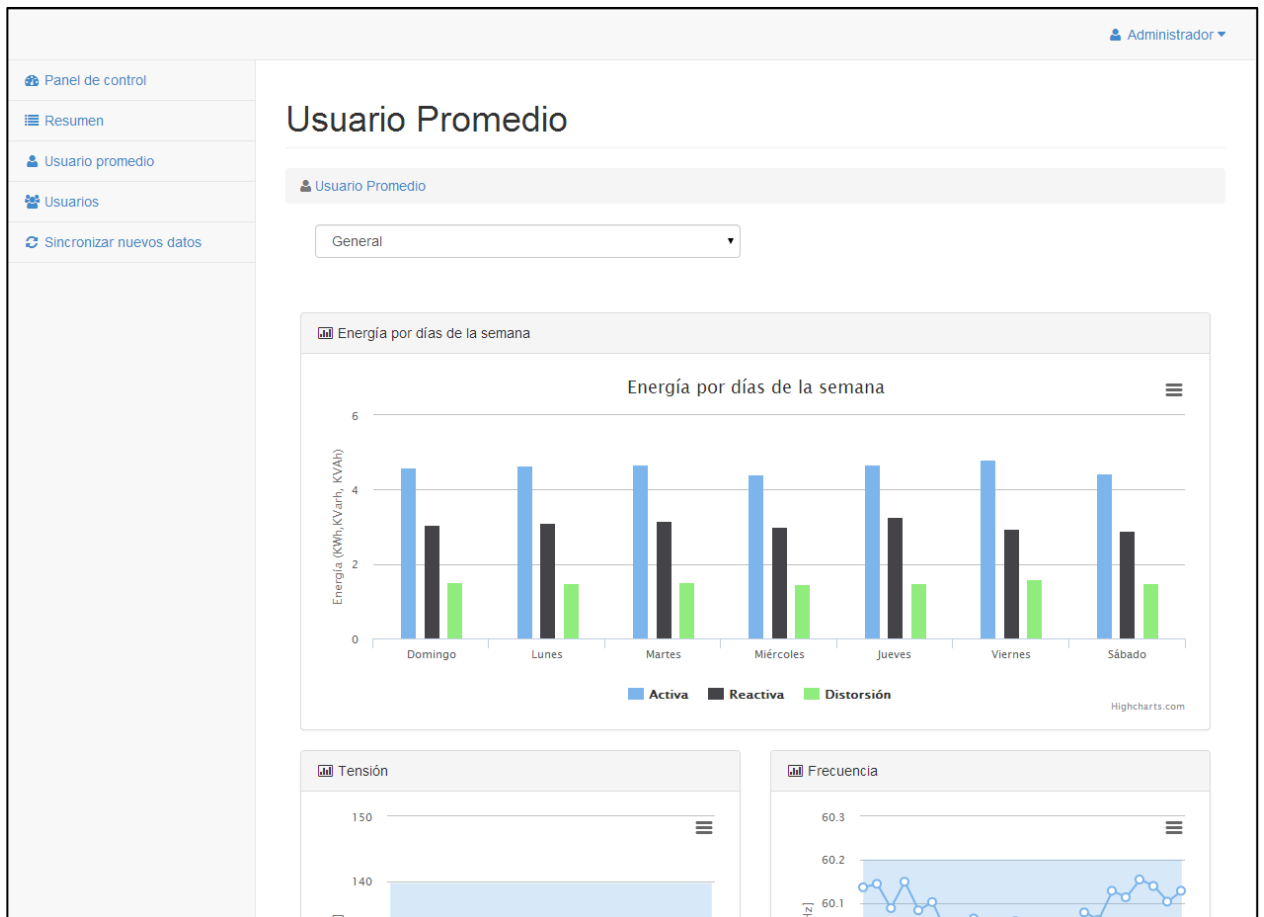


Ilustración 12: Usuario promedio

MÓDULO USUARIOS

Se muestra un listado de todos los medidores y usuarios que están asociados al transformador en estudio y se muestra la clasificación a la que pertenece, donde se puede realizar una búsqueda por usuario y ver los resultados de los datos procesados del usuario seleccionado (ver *Ilustración 13*).

Los datos de todos los usuarios son promediados por día de la semana y hora para almacenar la media y desviación estándar de cada dato, que es mostrado en forma gráfica y en tablas (ver *Ilustración 14*). El consumo eléctrico de los usuarios se comporta como una distribución normal (JARDINI et al, 2000).

Administrador ▾

Panel de control

Resumen

Usuario promedio

Usuarios

Sincronizar nuevos datos

Usuarios

Usuarios

Usuario

Usuario	# de contador	Consumo	Opciones
218090	1	Factor de potencia anormal	Ver
218091	2	Factor de potencia anormal	Ver
573662	3	Tensión anormal	Ver
218096	4	Tensión anormal	Ver
218095	5	Factor de potencia anormal	Ver
218094	6	Factor de potencia anormal	Ver
218084	7	Tensión anormal	Ver
218085	8	Tensión anormal	Ver
218086	9	Factor de potencia anormal	Ver
218087	10	Factor de potencia anormal	Ver
218088	11	Factor de potencia anormal	Ver

Ilustración 13: Usuarios

Administrador ▾

Panel de control

Resumen

Usuario promedio

Usuarios

Sincronizar nuevos datos

Usuario

Usuarios / Usuario

Contador: 1
Usuario: 218090
Dirección: CLL 10 3 70 LA FERIA

Lunes ▾

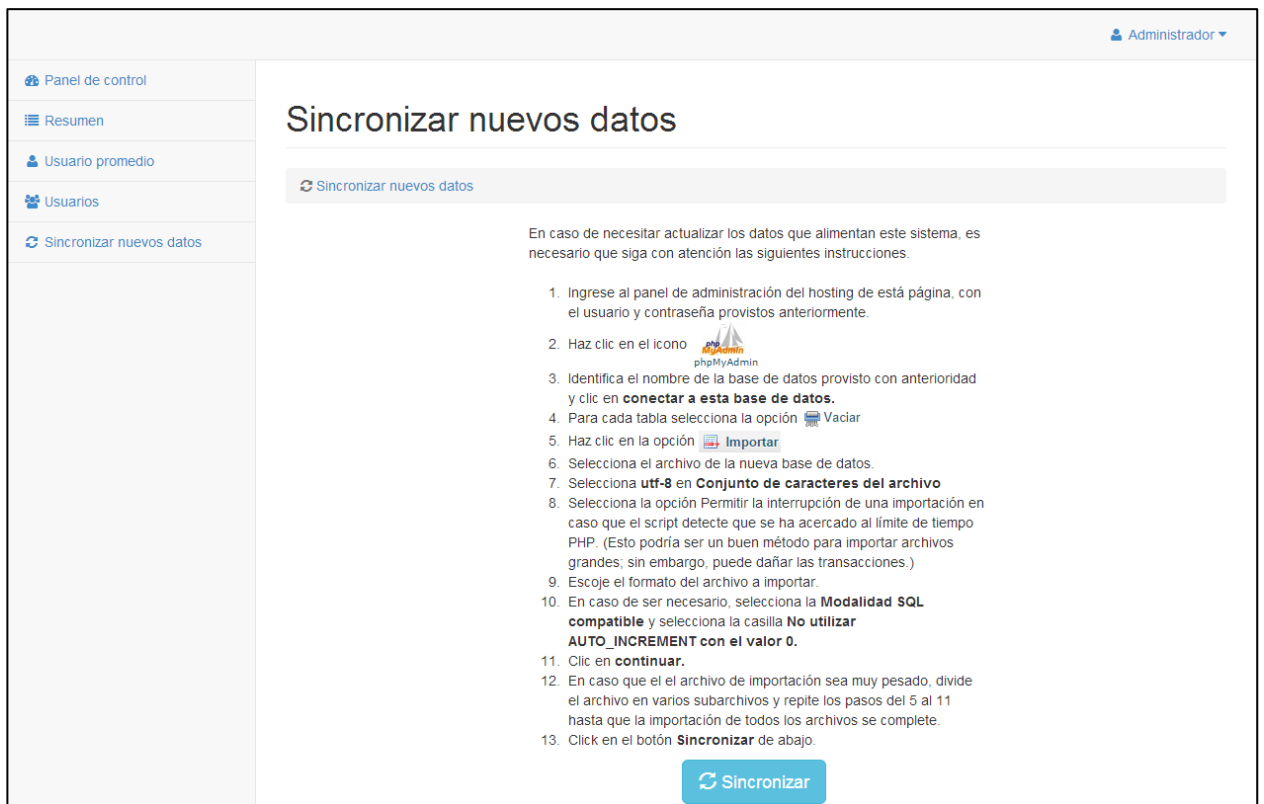
Energía por días de la semana

Día	Energía (Wh, KVAh)
Domingo	4.5
Lunes	6.5
Martes	7.5
Miércoles	6.5
Jueves	8.5
Viernes	6.5
Sábado	7.5

Ilustración 14: Usuario

MÓDULO DE SINCRONIZACIÓN DE NUEVOS DATOS

Se muestra información guía para establecer los requisitos que se deben tener y los pasos a recurrir para procesar correctamente nuevos datos recolectados en los medidores de los medidores de los usuarios y se realiza un llamado a la acción para procesar los datos una vez finalizado los pasos (ver *Ilustración 15*).



Administrador ▾

Panel de control

Resumen

Usuario promedio




Usuarios

Sincronizar nuevos datos

Sincronizar nuevos datos

Sincronizar nuevos datos

En caso de necesitar actualizar los datos que alimentan este sistema, es necesario que siga con atención las siguientes instrucciones.

1. Ingrese al panel de administración del hosting de esta página, con el usuario y contraseña provistos anteriormente.
2. Haz clic en el icono  phpMyAdmin
3. Identifica el nombre de la base de datos provisto con anterioridad y clic en **conectar a esta base de datos**.
4. Para cada tabla selecciona la opción  Vaciar
5. Haz clic en la opción  Importar
6. Selecciona el archivo de la nueva base de datos.
7. Selecciona **utf-8** en **Conjunto de caracteres del archivo**
8. Selecciona la opción Permitir la interrupción de una importación en caso que el script detecte que se ha acercado al límite de tiempo PHP. (Esto podría ser un buen método para importar archivos grandes; sin embargo, puede dañar las transacciones.)
9. Elige el formato del archivo a importar.
10. En caso de ser necesario, selecciona la **Modalidad SQL compatible** y selecciona la casilla **No utilizar AUTO_INCREMENT con el valor 0**.
11. Clic en **continuar**.
12. En caso que el archivo de importación sea muy pesado, divide el archivo en varios subarchivos y repite los pasos del 5 al 11 hasta que la importación de todos los archivos se complete.
13. Click en el botón **Sincronizar** de abajo.

Sincronizar

Ilustración 15: Sincronizar nuevos datos

Anexo B. Script MapReduce consumosxusuario.py

```
#!/usr/bin/env python
import mincemeat
import glob
#import fileinput
#datos = []
#for line in fileinput.input(['datos/consumos.txt']):
#    datos.append(line)
#datasource = dict(enumerate(datos))

all_files = glob.glob('datos/*.txt')

def file_contents(file_name):
    f = open(file_name)
    try:
        datos = []
        for line in f:
            datos.append(line)
        return datos
    finally:
        f.close()

# The data source can be any dictionary-like object
datasource = dict((file_name, file_contents(file_name))
                  for file_name in all_files)

def mapfn(k, val):
    import datetime
```

```

for v in val:
    linea = v.replace('\"', '').split(",")
    fecha = linea[0]
    hora = linea[1]
    hora = hora[:2]+":00"
    nocontador = linea[2]
    activainst = linea[5]
    reactivainst = linea[6]
    voltaje = linea[7]
    corriente = linea[8]
    fp = linea[9]
    frecuencia = linea[10]
    dia = datetime.datetime.strptime(fecha, "%Y-%m-%d").weekday()
    clave = str((nocontador, dia, hora))
    valor = (activainst, reactivainst, voltaje, corriente, fp,
frecuencia)
    yield clave, valor

def reducefn(key, list_of_values):
    reactiva = activa = aparente = frecuencia = tension = fp =
distorsion = 0
    dreactiva = dactiva = daparente = dfrecuencia = dtension =dfp=
ddistorsion = 0

    for valor in list_of_values:
        activa += float(valor[0])
        reactiva += float(valor[1])
        tension += float(valor[2])
        aparente += (float(valor[2])*float(valor[3]))/1000

```

```

fp += float(valor[4])
frecuencia += float(valor[5])
if((((float(valor[2])*float(valor[3]))/1000)**2
float(valor[0])**2 - float(valor[1])**2)>0):
    distorsion
    ((float(valor[2])*float(valor[3]))/1000)**2 - float(valor[0])**2
- float(valor[1])**2)**(0.5)
else:
    distorsion += 0

activa = activa/len(list_of_values)
reactiva = reactiva/len(list_of_values)
tension = tension/len(list_of_values)
aparente = aparente/len(list_of_values)
fp = fp/len(list_of_values)
frecuencia = frecuencia/len(list_of_values)
distorsion = distorsion/len(list_of_values)
for valor in list_of_values:
    dactiva += (float(valor[0])-activa)**2
    dreactiva += (float(valor[1])-reactiva)**2
    dtension += (float(valor[2])-tension)**2
    daparente += ((float(valor[2])*float(valor[3]))/1000-
aparente)**2
    dfp += (float(valor[4])-fp)**2
    dfrecuencia += (float(valor[5])-frecuencia)**2
    if((((float(valor[2])*float(valor[3]))/1000)**2
float(valor[0])**2 - float(valor[1])**2)>0):

```

```

        ddistorsion += (
((float(valor[2])*float(valor[3]))/1000)**2 - float(valor[0])**2
- float(valor[1])**2)**(0.5) - distorsion)**2
    else:
        ddistorsion += 0

    dactiva = (dactiva/len(list_of_values))**(0.5)
    dreactiva = (dreactiva/len(list_of_values))**(0.5)
    dtension = (dtension/len(list_of_values))**(0.5)
    daparente = (daparente/len(list_of_values))**(0.5)
    dfp = (dfp/len(list_of_values))**(0.5)
    dfrecuencia = (dfrecuencia/len(list_of_values))**(0.5)
    ddistorsion = (ddistorsion/len(list_of_values))**(0.5)

    val =
str((activa,dactiva,reactiva,dreactiva,tension,dtension,aparente,d
aparente,fp,dfp,frecuencia,dfrecuencia,distorsion,ddistorsion, ''))
    return val

s = mincemeat.Server()
s.datasource = datasource
s.mapfn = mapfn
s.reducefn = reducefn

results = s.run_server(password="electrica123")
print results

```