

2016

## Modelo computacional para manejo educativo de la respuesta a la demanda de energía eléctrica en el hogar

Juan Sebastián Rincón Castro  
*Universidad de La Salle, Bogotá*

Manuel Sebastián Bernal Díaz  
*Universidad de La Salle, Bogotá*

Follow this and additional works at: [https://ciencia.lasalle.edu.co/ing\\_automatizacion](https://ciencia.lasalle.edu.co/ing_automatizacion)



Part of the [Computational Engineering Commons](#), [Computer Engineering Commons](#), and the [Power and Energy Commons](#)

---

### Citación recomendada

Rincón Castro, J. S., & Bernal Díaz, M. S. (2016). Modelo computacional para manejo educativo de la respuesta a la demanda de energía eléctrica en el hogar. Retrieved from [https://ciencia.lasalle.edu.co/ing\\_automatizacion/125](https://ciencia.lasalle.edu.co/ing_automatizacion/125)

This Trabajo de grado - Pregrado is brought to you for free and open access by the Facultad de Ingeniería at Ciencia Unisalle. It has been accepted for inclusion in Ingeniería en Automatización by an authorized administrator of Ciencia Unisalle. For more information, please contact [ciencia@lasalle.edu.co](mailto:ciencia@lasalle.edu.co).

MODELO COMPUTACIONAL PARA MANEJO EDUCATIVO DE LA RESPUESTA  
A LA DEMANDA DE ENERGÍA ELÉCTRICA EN EL HOGAR

JUAN SEBASTIÁN RINCÓN CASTRO

Código 45121017

MANUEL SEBASTIÁN BERNAL DÍAZ

Código 45111022

UNIVERSIDAD DE LA SALLE  
FACULTAD DE INGENIERÍA  
INGENIERIA EN AUTOMATIZACIÓN  
BOGOTÁ D.C.

2016

MODELO COMPUTACIONAL PARA MANEJO EDUCATIVO DE LA RESPUESTA  
A LA DEMANDA DE ENERGÍA ELÉCTRICA EN EL HOGAR

JUAN SEBASTIÁN RINCÓN CASTRO  
MANUEL SEBASTIÁN BERNAL DÍAZ

Monografía para optar por el título de  
Ingeniero en Automatización

Directora  
DIANA JANETH LANCHEROS CUESTA  
Ph.D. Ingeniería

UNIVERSIDAD DE LA SALLE  
FACULTAD DE INGENIERÍA  
INGENIERIA EN AUTOMATIZACIÓN  
BOGOTÁ  
2016

Nota de aceptación:

---

---

---

---

---

---

Firma Asesor  
Ph.D. Diana Janeth Lancheros Cuesta

---

Firma del jurado  
Ing. Geovanny Marulanda

---

Firma del jurado  
Ing. Maximiliano Bueno

Bogotá D.C.

## AGRADECIMIENTOS

Agradezco primeramente a Dios por bendecirme y darme la oportunidad de llegar hasta donde actualmente estoy, luego de esto agradecer inmensamente a mi madre por el apoyo que me ha brindado durante toda mi vida, ya que sin ella nada de esto hubiese sido posible. Gracias por su compañía, sus consejos, sus ayudas, pero ante todo gracias por nunca dejarme solo y siempre sentir que está a mi lado respaldando todas mis ideas y locuras. De igual manera, gracias a la universidad de la Salle y a cada uno de los ingenieros que fueron parte de mi proceso de aprendizaje durante cada uno de los semestres por los que pase. A cada una de las personas que me rodearon en la universidad he hicieron que esta fuese una gran etapa de mi vida. Y por último, pero no menos importante, a mi compañero de tesis con el cual logramos culminar esta etapa de la manera más exitosa.

**Sebastián Rincón**

## AGRADECIMIENTOS

*“Best, you've got to be the best  
You've got to change the world  
And use this chance to be heard  
Your time is now”*

*Muse. Butterflies and Hurricanes*

A mis padres y hermanos, por creer en mí, por brindarme su apoyo incondicional día tras día y no dejarme desistir para alcanzar mis sueños, como el de ser ingeniero, un camino que no es fácil pero lleno de recompensas y enseñanzas que me permitieron crecer personal y profesionalmente.

A mi novia y mis amigos por su constante apoyo, paciencia y ganas de ayudarme a salir adelante, haciendo a esta etapa de mi vida única e inolvidable.

A todos los ingenieros, docentes y compañeros que hicieron parte de este proceso, especialmente a la ing. Diana Lancheros, al ing. Geovanny Marulanda, a mi amigo y compañero Sebastián Rincón con quien trabajamos hasta el cansancio por este proyecto, por este sueño.

**Manuel Sebastián Bernal Díaz**

## TABLA DE CONTENIDO

Pág.

RESUMEN.....	13
INTRODUCCIÓN.....	14
GLOSARIO.....	15
OBJETIVOS.....	16
1. TRABAJOS RELACIONADOS.....	17
2. MARCO TEÓRICO.....	21
2.1 CONSUMO ELÉCTRICO.....	21
2.2 DEMANDA DE ENERGÍA ELÉCTRICA.....	22
2.2.1 CURVA DE CARGA.....	23
2.3 CLASIFICACIÓN DE LOS PROGRAMAS DE LA RESPUESTA A LA DEMANDA DE ENERGÍA ELÉCTRICA.....	24
2.3.1 Programas basados en incentivos.....	24
2.3.2 Programas Basados en Tiempo de Uso.....	26
2.4 MODELO COMPUTACIONAL.....	28
2.5 APLICACIÓN WEB.....	28
2.6 APLICACIÓN MÓVIL.....	29
2.7 DISEÑO DE SOFTWARE.....	32
2.7.1 Bases de datos.....	32
2.7.2 Casos de uso.....	33
2.7.3 Algoritmos.....	35
2.8 MARCO LEGAL Y NORMATIVO.....	36
3. ANTECEDENTES.....	37
4. SISTEMA POUPA WATT.....	45
5. DISEÑO DEL SISTEMA COMPUTACIONAL.....	48
5.1 REQUERIMIENTOS DEL SISTEMA.....	48
5.1.1 Requerimientos del usuario.....	48
5.1.2 Requerimientos software.....	52
5.2 DISEÑO DEL SISTEMA.....	58
5.2.1 Usuarios.....	58

5.2.2	Casos de uso .....	59
5.2.3	BASES DE DATOS .....	78
5.2.4	ALGORITMOS .....	86
6.	DESARROLLO E IMPLEMENTACIÓN.....	99
6.1	DESARROLLO E IMPLEMENTACIÓN APLICACIÓN WEB .....	99
6.1.1	Plataformas de desarrollo .....	100
6.1.2	Aplicación web .....	103
6.1.3	Funcionamiento aplicación web .....	103
6.2	DESARROLLO E IMPLEMENTACIÓN APLICACIÓN MÓVIL.....	120
6.2.1	Plataformas de desarrollo .....	121
6.2.2	Aplicación móvil.....	122
6.2.3	Funcionamiento aplicación móvil .....	124
6.3	Comunicación aplicación móvil con aplicación web.....	135
7.	VALIDACIONES .....	138
7.1	Aplicación web.....	139
7.1.1	Consulta consumo.....	139
7.1.2	Algoritmos modelos respuesta a la demanda .....	144
7.1.3	Juego .....	149
7.2	Aplicación móvil .....	151
7.2.1	Validación de Login .....	152
7.2.2	Validación de Consumo .....	154
7.2.3	Validación de Históricos .....	157
7.2.4	Validación de Juego.....	159
7.2.4	Posibles Errores .....	163
7.3	Nuevo caso de estudio .....	163
7.3.1	Consulta consumo del sensor 1 .....	164
7.3.2	Consulta consumo del sensor 2.....	167
7.3.3	Consulta consumo del sensor 3.....	170
7.3.4	Consulta consumo del sensor 4.....	173
7.3.5	Consulta consumo del sensor 5.....	175
7.3.6	Respuesta del Modelo .....	178
8.	CONCLUSIONES .....	181
9.	BIBLIOGRAFÍA.....	183



10.	ANEXO A .....	188
11.	ANEXO B .....	215
12.	ANEXO C .....	393
13.	ANEXO D .....	406

## LISTA DE TABLAS

	Pág.
Tabla 1. Resumen trabajos relacionados.....	20
Tabla 2. Ventajas y desventajas app nativas.....	30
Tabla 3. Ventajas y desventajas Web app.....	31
Tabla 4. Ventas y desventajas Web app nativa.....	31
Tabla 5. Leyes y normas del proyecto.....	36
Tabla 6. Requerimientos de capacidad.....	49
Tabla 7. Requerimientos de restricción.....	51
Tabla 8. Requerimientos funcionales.....	53
Tabla 9. Requerimientos no funcionales de interfaz.....	56
Tabla 10. Requerimientos no funcionales de recursos.....	57
Tabla 11. Requerimientos no funcionales de seguridad.....	58
Tabla 12. CU001 Registrar usuario.....	61
Tabla 13. CU002 Registrar Facturas.....	62
Tabla 14. CU003 Consultar información personal.....	62
Tabla 15. CU004 Consultar información de facturación.....	63
Tabla 16. CU005 Consultar preferencias.....	63
Tabla 17. CU006 Consultar consumo.....	64
Tabla 18. CU007 Modificar información del usuario.....	64
Tabla 19. CU008 Modificar información de facturación.....	65
Tabla 20. CU009 Modificar preferencias.....	65
Tabla 21. CU010 Juego plataforma.....	66
Tabla 22. CU011 Eliminar Perfil.....	66
Tabla 23. CU012 Ejecutar algoritmo modelo programa respuesta a la demanda.....	67
Tabla 24. CU013 Habilitar y deshabilitar usuarios.....	67
Tabla 25. CU014 Consultar información de los usuarios.....	68
Tabla 26. CU015 Modificar información de los usuarios.....	69
Tabla 27. CU016 Eliminar usuario.....	69
Tabla 28. CU017 Cambiar tipo de usuario.....	70
Tabla 29. CU018 Consultar parámetros del sistema.....	71
Tabla 30. CU019 Modificar programas de respuesta a la demanda.....	71
Tabla 31. CU020 Modificar sensores.....	72
Tabla 32. CU021 Modificar electrodomésticos.....	72
Tabla 33. CU022 Adicionar programas de respuesta a la demanda.....	73
Tabla 34. CU023 Adicionar sensores.....	73
Tabla 35. CU024 Adicionar electrodoméstico.....	74
Tabla 36. CU025 Registrar usuario.....	75
Tabla 37. CU026 Consultar información personal.....	76
Tabla 38. CU027 Consultar preferencias.....	77
Tabla 39. CU028 Consultar consumo.....	77
Tabla 40. CU029 Juego plataforma.....	78

Tabla 41. Tabla sensores. ....	81
Tabla 42. Tabla usuarios. ....	81
Tabla 43. Tabla rol. ....	82
Tabla 44. Tabla facturación. ....	82
Tabla 45. Tabla preferencias usuario aplicación móvil. ....	83
Tabla 46. Tabla preferencias usuario aplicación web. ....	83
Tabla 47. Tabla de los programas de respuesta a la demanda. ....	84
Tabla 48. Tabla electrodomésticos. ....	84
Tabla 49. Tabla sensores. ....	84
Tabla 50. Tabla ciudades. ....	85
Tabla 51. Tabla localidades. ....	85
Tabla 52. Tabla barrios. ....	85
Tabla 53. Pantalla Login aplicación web. ....	104
Tabla 54. Pantalla Home aplicación web. ....	105
Tabla 55. Pantalla noticias aplicación web. ....	106
Tabla 56. Pantalla perfil aplicación web. ....	107
Tabla 57. Pantalla Preferencias aplicación web. ....	108
Tabla 58. Pantalla facturas aplicación web. ....	109
Tabla 59. Pantalla Consumo aplicación web. ....	110
Tabla 60. Pantallas Modelos aplicación web. ....	111
Tabla 61. Pantalla Juego aplicación web. ....	113
Tabla 62. Pantalla login Administrador aplicación web. ....	114
Tabla 63. Pantalla home administrador aplicación web. ....	115
Tabla 64. Pantalla usuarios aplicación web. ....	116
Tabla 65. Pantalla sensores aplicación web. ....	117
Tabla 66. Pantalla Electrodomésticos aplicación web. ....	118
Tabla 67. Pantalla Programas aplicación web. ....	119
Tabla 68. Inicio aplicación móvil. ....	124
Tabla 69. Registro de nuevo usuario en la aplicación móvil. ....	125
Tabla 70. Registro exitoso en aplicación móvil. ....	126
Tabla 71. Registro de preferencias en aplicación móvil. ....	127
Tabla 72. Home aplicación móvil. ....	128
Tabla 73. Alcance del proyecto aplicación móvil. ....	129
Tabla 74. Tutorial aplicación móvil. ....	130
Tabla 75. Apagar paga en aplicación móvil. ....	131
Tabla 76. Menú aplicación móvil. ....	132
Tabla 77. Consulta consumo aplicación móvil. ....	133
Tabla 78. Históricos en aplicación móvil. ....	134
Tabla 79. Juego aplicación móvil. ....	135

## LISTA DE FIGURAS

	Pág.
Figura 1. Demanda de energía .....	22
Figura 2. Ejemplo de una curva de carga .....	23
Figura 3. Diagrama de casos de uso .....	35
Figura 4. Bosquejo proyecto VRIT .....	39
Figura 5. Sistema Poupa Watt .....	46
Figura 6. Fases de desarrollo del sistema .....	46
Figura 7. Parámetros de entrada del modelo computacional.....	47
Figura 8. Interacción usuarios y sistema.....	59
Figura 9. Diagrama casos de uso en aplicación web.....	60
Figura 10. Diagrama casos de uso aplicación móvil .....	75
Figura 11. Diagrama de conexiones .....	79
Figura 12. Diagrama entidad-relación.....	80
Figura 13. Diagrama de flujo Oferta a la Demanda.....	87
Figura 14. Diagrama de flujo tiempo de uso. ....	88
Figura 15. Diagrama de flujo juego .....	89
Figura 16. Diagrama de flujo variante Oferta a la Demanda.....	96
Figura 17. Aplicación web.....	100
Figura 18. Entorno de desarrollo Sublime Text.....	101
Figura 19. Mapa de navegación aplicación web. ....	103
Figura 20. Aplicación móvil. ....	120
Figura 21. Entorno de desarrollo Android Studio. ....	122
Figura 22. Mapa de navegación aplicación móvil. ....	123
Figura 23. Vista previa aplicación .....	123
Figura 24. Registro en aplicación móvil .....	136
Figura 25. Datos registrados en la base de datos.....	136
Figura 26. Perfil en aplicación web. ....	137
Figura 27. Datos servidores .....	138
Figura 28. Usuario sin sensores asignados. ....	139
Figura 29. Usuario con sensores asignados.....	140
Figura 30. Grafica consumo vacía. ....	141
Figura 31. Datos almacenados .....	142
Figura 32. Grafica consumo.....	143
Figura 33. Usuario sin preferencias registradas.....	144
Figura 34. Tiempo de uso como preferencia .....	145
Figura 35. Valor de reducción .....	146
Figura 36. Respuesta tiempo de uso .....	146
Figura 37. Oferta a la demanda como preferencia.....	147
Figura 38. Ingreso de horas para ejecución de oferta a la demanda. ....	148
Figura 39. Datos sensores para sumatoria. ....	149
Figura 40. Reducción alcanzada en el juego. ....	150
Figura 41. Reducción excedida en el juego. ....	151
Figura 42. Datos en base de datos de aplicación móvil. ....	152

Figura 43. Caso 1 login aplicación móvil.....	153
Figura 44. Caso 2 Login aplicación móvil .....	153
Figura 45. Caso 1 Validación de consumo en aplicación móvil. ....	154
Figura 46. Caso 2 Validación de consumo en aplicación móvil. ....	155
Figura 47. Datos de consumo a graficar .....	156
Figura 48. Caso 3 Validación consumo en aplicación móvil. ....	157
Figura 49. Caso 1 Validación de históricos en aplicación móvil.....	158
Figura 50. Caso 2 Validación de históricos aplicación móvil.....	159
Figura 51. Caso 1 Validación de juego aplicación móvil. ....	160
Figura 48. Caso 2 Validación juego en aplicación móvil. ....	161
Figura 53. Caso 3 Validación juego en aplicación móvil. ....	162
Figura 54. Caso 4 Validación juego, aplicación móvil. ....	163
Figura 55. Sensores en los servidores.....	164
Figura 56. Consulta consumo para el sensor 1.....	165
Figura 57. Datos del sensor 1 en el servidor .....	165
Figura 58. Grafica consulta sensor 1 .....	166
Figura 59. Consulta consumo para el sensor 2.....	167
Figura 60. Datos del sensor 2 en el servidor .....	168
Figura 61. Gráfica consulta sensor 2 .....	169
Figura 62. Consulta consumo para el sensor 3.....	170
Figura 63. Datos del sensor 3 en el servidor .....	171
Figura 64. Gráfica consulta 3.....	172
Figura 65. Consulta consumo para el sensor 4.....	173
Figura 66. Datos del sensor 4 en el servidor .....	174
Figura 67. Gráfica consulta sensor 4 .....	174
Figura 68. Consulta consumo para el sensor 5.....	176
Figura 69. Datos del sensor 5 en el servidor .....	177
Figura 70. Gráfica consulta sensor 5 .....	177
Figura 71. Valor de reducción nuevo caso de estudio .....	178
Figura 72. Resultados nuevo caso de estudio .....	180

## RESUMEN

En el constante consumo de energía eléctrica en los hogares, oficinas y demás, es importante un monitoreo para verificar y cuantificar el gasto por electrodoméstico y equipo electrónico dentro del consumo diario, dependiendo de la hora en la que se desee utilizar dicho aparato. Una estrategia pedagógica es la mejor forma de aprendizaje para el consumidor cotidiano, usuario final del sistema para conocer sus gastos y mejorar sus hábitos de consumo.

Debido a las tecnologías disponibles y el uso de teléfonos celulares inteligentes y tabletas, la forma más versátil de llegar a los consumidores es a través de una aplicación móvil de interfaz sencilla y práctica, que de manera didáctica provea las herramientas necesarias para participar en un programa de respuesta a la demanda.

Los hogares según su consumo pueden hacer parte de diferentes programas de respuesta a la demanda, por eso, a través de los medidores inteligentes de energía eléctrica se adquirió información del consumo de un hogar, para establecer un comportamiento y posteriormente las horas pico y valle en dicha casa. Una vez hecho esto y basándose en programas de respuesta a la demanda, se asignó uno al usuario garantizando reducir el costo de su consumo.

Todo esto fue hecho a partir de un algoritmo que, con los datos adquiridos por los medidores inteligentes, cuantificó y almacenó el consumo de dicha casa en una base de datos y por medio de esta disponibles en la aplicación móvil, que con los datos de consumo y a través de una aplicación didáctica, cada usuario aprende de su consumo y como reducirlo por medio de un programa de respuesta a la demanda previamente diseñado.

## INTRODUCCIÓN

Según la Unidad de Planeación Minero Energética (UPME) en su último informe cuatrimestral en el cual se muestran las proyecciones de la demanda de energía eléctrica y potencia máxima en Colombia, muestra que actualmente el consumo se encuentra en crecimiento teniendo al 2016 una tasa de 3,6% anual, pero para marzo del 2017 se encuentra en un 3,8% anual. (Unidad de Planeamiento Minero Energética, 2016)

Este notable crecimiento se debe a varios factores como el incremento en la temperatura en varias regiones del país, esto lleva a que en los hogares aumente el uso de aire acondicionado o ventiladores. La demanda de energía eléctrica en los hogares seguirá en aumento debido al crecimiento poblacional, por ello se busca generar nuevos hábitos de consumo a través de los programas de la respuesta a la demanda, mediante una aplicación web y una aplicación móvil, donde el usuario de manera educativa pueda interactuar en diferentes escenarios de uso cotidiano y monitorear el consumo de su hogar.

Para esto, se desarrolló un modelo computacional para programas de respuesta a la demanda, con capacidad de almacenamiento y análisis de datos para monitoreo de consumo energético en el hogar, permitiendo la cuantificación del consumo y aprendiendo el comportamiento en el transcurso del día para establecer las horas “pico” y “valle” ajustando un programa de respuesta a la demanda de acuerdo a la capacidad de pago, pues actualmente se presentan horas en que el sistema energético se encuentra en su máximo de capacidad aumentando los costos de los consumidores y a través de un programa de respuesta a la demanda estos costos pueden ser reducidos de manera significativa, sin dejar atrás el beneficio al medio ambiente al reducir las emisiones de contaminantes y menor consumo de recursos naturales.

## GLOSARIO

**Aplicación Móvil:** Las aplicaciones son a los teléfonos móviles lo que el software es a los computadores, esto quiere decir que es un software diseñado para funcionar en dispositivos móviles. Tienen 3 grupos para ser clasificadas, nativas, web e híbrida. (Mercado, 2014)

**Aplicación Web:** Una aplicación Web es un sitio Web que contiene páginas con contenido sin determinar, parcialmente o en su totalidad. El contenido final de una página se determina sólo cuando el usuario solicita una página del servidor Web. Dado que el contenido final de la página varía de una petición a otra en función de las acciones del visitante, este tipo de página se denomina página dinámica. (Adobe Systems Incorporated and its licensors, 2015)

**Medidores inteligentes (Smart Meters):** Miden la cantidad de energía que se usa para la facturación y se comunican con la red para compartir información sobre el uso de energía con la empresa prestadora del servicio. (California Public Utilities Commission and California Energy Commission, 2014)

**Modelo computacional:** Es la integración de conocimientos científicos y tecnológicos con modelos matemáticos para el diseño e implementación de un programa de cómputo (Carrillo , 2009)

**Respuesta a la demanda:** Es la reducción en el consumo de la energía en respuesta a períodos de alta demanda (California Public Utilities Commission and California Energy Commission, 2014)

**Sistema de información:** Conjunto de procedimientos, operaciones, funciones y difusión de datos o información para apoyar los diferentes procesos en que son aplicados. (Universidad del Cauca, 2015)

**Smart Grid:** Integración de tecnologías para la información y comunicación de los diferentes procesos de la industria eléctrica. (Palomino, 2014)



## OBJETIVOS

### Objetivo general

- Desarrollar un modelo computacional con fines educativos orientado al usuario en el uso responsable y reducido de energía eléctrica según el programa de respuesta a la demanda.

### Objetivos específicos

- Determinar los parámetros de entrada del modelo computacional en términos de energía eléctrica.
- Diseñar una base de datos que permita el manejo de la información dentro del modelo.
- Diseñar una interfaz HCI amigable y adaptable a los usuarios.
- Implementar el algoritmo de caracterización de la demanda del proyecto presentado a la VRIT titulado “Construcción de un sistema de medición inteligente de energía eléctrica de bajo costo, orientado a programas residenciales de respuesta de la demanda para usuarios de bajos recursos”.

## 1. TRABAJOS RELACIONADOS

En el 2012 se publicó el artículo “**Design of demand response programs in emerging countries**” (Martinez & Rudnick, 2012). En el cual para los países latinoamericanos o emergentes se diseñaron diferentes programas para la producción y ahorro de energía eléctrica, en estos se muestra que al ser implementados se puede lograr una reducción en el consumo de energía eléctrica de hasta un 25%, junto con el uso de smart grids y smart meters.

Los programas de respuesta a la demanda, al ser personalizados necesitan ser flexibles y adaptarse al usuario donde van a ser aplicados como se explica en el artículo “**Demand Response Programs Definition Using Demand Price Elasticity to Define Consumers Aggregation for an Improved Remuneration Structure**” (Faria & Zita, 2013). En este se muestra que la flexibilidad de precios en los diferentes programas permite que se ajusten al usuario y al aplicarlos se tenga un sistema de remuneración según su participación, siendo estos dos puntos clave para el desarrollo del proyecto.

La reducción del consumo de energía eléctrica por parte de los usuarios de respuesta a la demanda, permite a la empresa generadora y distribuidora como lo muestra el artículo “**Optimal Demand Response Based on Utility Maximization in Power Networks**” (Li, Chen, & Low, 2011). Dar una nueva distribución de la energía eléctrica ya generada, pues se propone el almacenamiento de la energía eléctrica no consumida, para una nueva distribución en los sistemas fluctuantes identificados con las smart grids y smart meters.

De los artículos anteriores, se concluye que los programas de respuesta a la demanda por sí solos no son suficientes, para cumplir con su objetivo de manera adecuada necesitan de complementos como los smart meters y las smart grids; o de cargas y sistemas de energía inteligentes como en el artículo “**Demand Side Management: Demand Response, Intelligent Energy Systems, and Smart Loads**” (Palensky & Dietrich, 2011). Este expone que la optimización en generación

y distribución de energía es común e importante, pero el manejo eficiente de la demanda de energía no está tan avanzado como debería, para esto se necesita un sistema que permita llevar un manejo adecuado de los programas de respuesta a la demanda, esto incluye el uso de smart meters, smart grids, cargas inteligentes y por supuesto un sistema de información adecuado para que funcione de manera adecuada y todo en conjunto sea un sistema inteligente.

Lo anterior se ve aplicado en la universidad de La Salle donde a través de un proyecto de investigación de la VRIT se desarrolló un trabajo de grado titulado **“Monitoring system for the consumption of electric energy in a household”** (Pérez, 2014). En el que se diseñó un sistema de información que muestra en tiempo real el consumo y costo de este para diferentes aparatos electrónicos, a través de unos sensores desarrollados dentro del proyecto que cumplen la función de los smart meters.

La importancia de que la recolección de datos sea en tiempo real es resaltada en el artículo **“Real-Time Demand Response Model”** (Conejo, Morales, & Baringo, 2010). Donde se plantea un modelo para cuantificar en tiempo real el consumo y un programa de la respuesta a la demanda que recoge datos 24 horas para posteriormente minimizar el consumo, pues la recolección continua de datos da paso a que el programa de respuesta a la demanda escogido por el usuario tenga información actualizada y por lo tanto la identificación de picos es constante permitiendo identificar y desplazar las cargas que los producen.

Los programas de respuesta a la demanda y sus complementos para la reducción energética, al ser enfocados a las residencias tienen un efecto favorable, como se menciona en el artículo **“The Effects of Residential Energy Efficiency on Electric Demand Response Programs”** (Jewell, 2014). Para este se diseñó un modelo computacional donde se simuló 24 casas, encontrando que el sistema de calefacción no permite un buen desempeño de dichos programas, de modo que identificando y desplazando dicha carga según el programa de respuesta a la

demanda los hogares pueden reducir la carga en la zona. Pero no solo la calefacción produce dicho efecto, por eso la importancia de la recolección de datos en cada hogar para la identificación de las diferentes cargas y su desplazamiento, permitiendo la reducción de consumo en zonas residenciales.

Para aplicar todo esto al proyecto se quiso que las aplicaciones fueran educativas y de fácil manejo del usuario, como en el artículo **“Un método de desarrollo de aplicaciones educativas hipermedia”** (Valencia, 2015). Donde se proponen la descripción del problema, definición de requerimientos y términos del contrato, diseño educativo, diseño computacional y producción, como fases a tener en cuenta para el desarrollo del sistema.

El sistema debe ser desarrollado bajo una metodología para su éxito, esta fue tomada del artículo **“Metodología para el desarrollo de aplicaciones educativas en ambientes multimedios”** (Bianchini, 2015) donde se resalta la identificación de las condiciones de utilización y usuarios que se deben plantear de manera que el uso de la misma sea intuitivo y natural, además de que su uso sea motivante para el usuario.

Para motivar el uso del sistema se implementaron objetivos y así darle un estilo de juego, como en el artículo **“How gamification motivates visits and engagement for online academic dissemination - An empirical study”** (Ming-Shiou & Chuang, 2016). Donde se resalta que la gamificación es usar mecanismos de juegos o elementos de los mismos en contextos de no juego, o sea impulsar al usuario al uso del sistema a través de objetivos, recompensas y competencias.

La tabla 1 contiene un resumen de los trabajos relacionados donde se muestra el aspecto relevante de cada uno para el desarrollo de este proyecto.

Tabla 1. Resumen trabajos relacionados

<b>Trabajo</b>	<b>Aspectos Relevantes</b>
Design of demand response programs in emerging countries	Uso de smart grids y smart meters
Optimal Demand Response Based on Utility Maximization in Power Networks	Distribución de energía con el uso de sistemas inteligentes.
Demand Response Programs Definition Using Demand Price Elasticity to Define Consumers Aggregation for an Improved Remuneration Structure	Flexibilidad de precios y remuneraciones al usuario.
Demand Side Management: Demand Response, Intelligent Energy Systems, and Smart Loads	Sistemas inteligentes para complementar los programas de respuesta a la demanda.
Monitoring system for the consumption of electric energy in a household	Sistemas de información para el monitoreo del consumo.
Real-Time Demand Response Model	Cuantificar en tiempo real el consumo.
The Effects of Residential Energy Efficiency on Electric Demand Response Programs	Identificación de cargas para programas personalizados.
Un método de desarrollo de aplicaciones educativas hipermedia	Fases para el desarrollo de una aplicación educativa.
Metodología para el desarrollo de aplicaciones educativas en ambientes multimedios	Identificación de las condiciones de utilización y usuarios.
How gamification motivates visits and engagement for online academic dissemination - An empirical study	Mecanismos de juegos o elementos de los mismos en contextos de no juego.

Fuente: Los autores

Una vez revisados todos los trabajos relacionados se da paso al marco teórico en donde se encuentra toda la información con respecto al trabajo realizado.

## **2. MARCO TEÓRICO**

En este capítulo se dan a conocer los diferentes conceptos y temas relacionados con el desarrollo del proyecto. Los temas centrales del capítulo son: características del consumo eléctrico, programas de respuesta a la demanda, modelo computacional y diseño de software.

### **2.1 CONSUMO ELÉCTRICO**

Cuando se presenta un excesivo consumo de energía eléctrica, este tiene un efecto negativo desde un punto de vista global, pues puede llevar al colapso de los recursos naturales y contribuyendo a acentuar los efectos del cambio climático (erenovable, 2015).

Por eso es necesaria una gestión de la energía urbana más sostenible, incluyendo la reducción en el consumo y la promoción de fuentes de energía renovables, como una de las líneas de actuación en las políticas de mitigación de los efectos del cambio climático.

El consumo eléctrico es medido en kilo vatios por hora (kWh) y está directamente relacionado con la potencia de un aparato electrónico (erenovable, 2015). Para nuestro proyecto los electrodomésticos, por lo tanto, a mayor tiempo de uso mayor consumo eléctrico.

Cada electrodoméstico para su funcionamiento consume una cantidad de electricidad de acuerdo a la potencia y el tiempo que funcione el mismo, de modo que para conocer el consumo se ha de multiplicar el tiempo de funcionamiento por los watts del electrodoméstico, dicho cálculo ha de ser realizado para cada aparato pues todos tiene un comportamiento diferente.

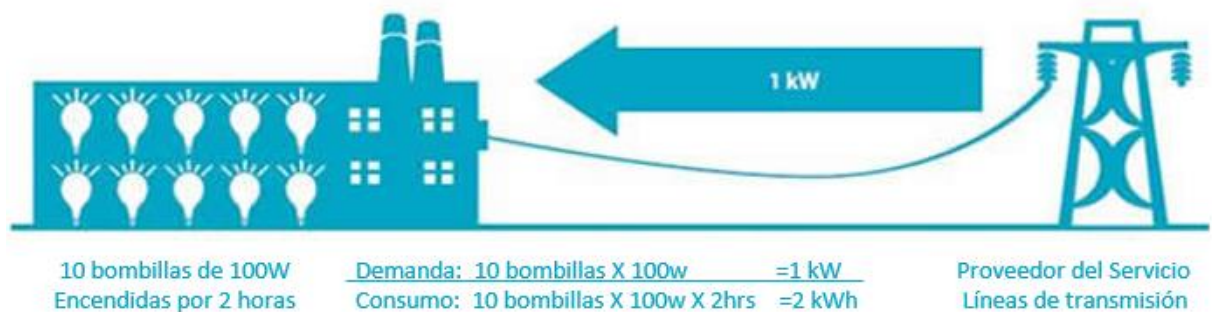
## 2.2 DEMANDA DE ENERGÍA ELÉCTRICA

La demanda de energía eléctrica es la medida que se obtiene en una tasa promedio por el consumo eléctrico que se puede obtener en el hogar o cualquier tipo de instalación (TXU Energy , 2015). Es decir, que entre más aparatos eléctricos se tengan conectados y estén en funcionamiento en un mismo lapso de tiempo, mayor será la demanda de energía que estos mismos exigen y se puede representar por medio de una gráfica llamada curva de carga. Esta demanda es la cual se regula o, en otras palabras, la que se ve evidenciada en el recibo de facturación mensual dado por una empresa reguladora de energía en los hogares e industrias.

La diferencia entre demanda y consumo es que la demanda hace referencia a la cantidad de energía que se necesita en un momento determinado y se mide en kilovatios (KW). Mientras que el consumo es la cantidad de energía que se utiliza durante un período de tiempo determinado y se mide en kilovatio-hora (kWh) (TXU Energy , 2015).

La figura 1 muestra un ejemplo de la demanda de energía que puede tener una industria. Por ejemplo, para 10 bombillas encendidas por 1 hora la demanda es de 1kW para mantener la luz constante en ese tiempo específico.

Figura 1. Demanda de energía



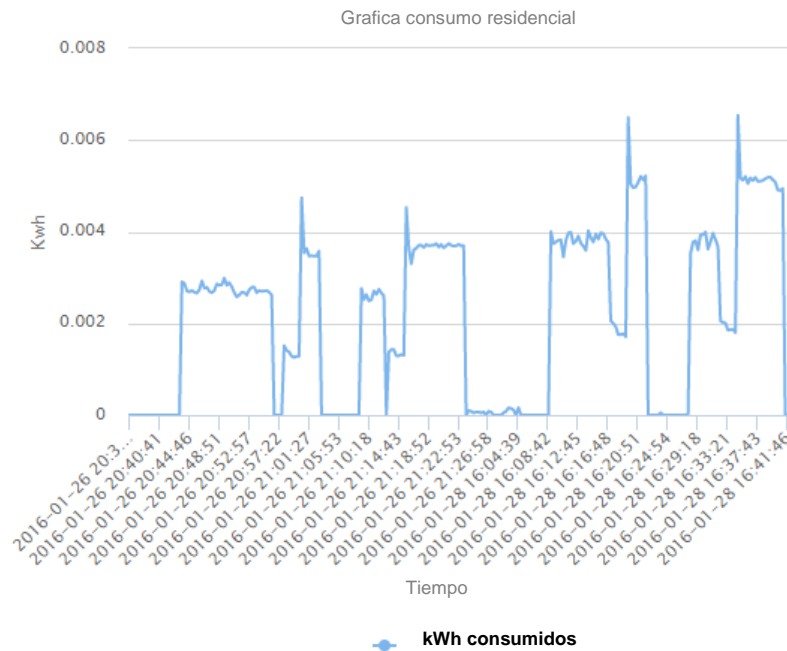
Tomado de (TXU Energy , 2015)

### 2.2.1 CURVA DE CARGA

La curva de carga o de demanda es la representación gráfica de la variación de la carga eléctrica en el tiempo (Facultad de Ciencias Exactas y Tecnología - Universidad Nacional de Tucumán, 2014), dicha curva se ve afectada con la demanda de los consumidores de acuerdo a la hora del día e incluso a semanas o meses del año, esto dependiendo de los hábitos de consumo de los usuarios que se pueden ver afectados por su zona y hasta de las condiciones atmosféricas (Gámiz Juárez, 2014).

En la figura 2 hay momentos del día en que el consumo energético aumenta considerablemente, pues las costumbres de consumo se ven reflejadas en dicha curva y permite determinar el uso final de los consumidores en la demanda máxima del sistema.

Figura 2. Ejemplo de una curva de carga



Fuente: Los Autores



## **2.3 CLASIFICACIÓN DE LOS PROGRAMAS DE LA RESPUESTA A LA DEMANDA DE ENERGÍA ELÉCTRICA**

Debido al importante uso que poseen los programas de respuesta a la demanda, fue necesario crear diferentes tipos de ellos con el fin de satisfacer al usuario y de igual manera encontrar un beneficio para el operador. Es por esto que se determinan dos tipos generales de programas de RD como lo son (Marulanda García, 2014).:

- *Programas Basados en Incentivos (PBI)*
- *Programas Basados en Tiempo de Uso (PBT)*

### **2.3.1 Programas basados en incentivos**

La finalidad de este tipo de programas es lograr incentivar al usuario a vincularse y generar un cambio en el consumo habitual que este posee, buscando así la reducción de consumo en las horas más concurridas, distribuyendo el mismo en otros periodos. Ofreciendo diferentes alternativas en donde el usuario puede llegar a recibir pagos en efectivo, créditos o por otro lado un descuento en su facturación mensual. Pero todo esto depende de la reducción que el usuario realice y del mismo modo el empeño que este disponga. Los tipos de PBI son (Marulanda García, 2014).:

#### **2.3.1.1 Oferta a la demanda**

En este programa el operador de red realiza una oferta en un mercado mayorista, posteriormente el usuario en base a la oferta del operador realiza una oferta de reducción de carga, indicando de manera clara y precisa, que cortes y/o reducciones puede realizar y el precio por megavatio hora al cual realizarán dicha reducción. La oferta es aceptada por el operador única y exclusivamente si es menor el costo de reducción que el de generación al suplir el total de la demanda.

### **2.3.1.2 Control Directo de Carga**

Por otro lado, este es diferente al ya mencionado debido a que para este tipo de programa es el operador quien realiza cortes remotos en la prestación de energía cuando este considere conveniente. El usuario es notificado del corte a realizar con poco tiempo de antelación ya que se hace cuando el consumo del sector es muy elevado en horas de alta demanda y puede generar cierto tipo de inconvenientes o sobresaltos en la producción.

### **2.3.1.3 Emergencia**

Para este programa es necesario que se presente una situación en particular en donde el operador lo clasifica como momento de emergencia y es ahí en donde todos los participantes a este deben de realizar una disminución notable en el consumo, pero llevar a cabo esta reducción es decisión de cada uno de los usuarios ya que no es obligatoria y por lo tanto no se incurre en sanciones.

### **2.3.1.4 Demanda Interrumpible**

En este punto es donde el operador le informa al usuario una reducción que debe de realizar ante una contingencia que se presenta, es necesario que se realice la disminución puesto que el operador está en el derecho de realizar ciertos cortes de energía de manera remota según lo acuerde con el respectivo usuario. Diferencia de los demás en este caso en donde se llegue a incumplir el acuerdo pactado entre las dos partes se puede llegar a realizar multas económicas.

### **2.3.1.5 Demanda como Recurso de Capacidad**

Para este caso, los usuarios deben establecer un valor determinado de reducción en el momento en que ocurra una contingencia con el fin de lograr superarla y de esta manera tener un equilibrio en esos momentos críticos que se presentan de manera inesperada. En este programa el operador no puede

realizar cortes ni nada por el estilo debido al pacto establecido con el usuario, pero de igual manera el incumplimiento de este genera sanciones económicas.

#### **2.3.1.6 Reserva Rodante**

Este tipo de programa se implementa cuando se tienen desbalances entre la generación de energía y la demanda que solicitan los usuarios, es por esto que se debe de tener una rápida respuesta a esto para generar un equilibrio y volver todo a la normalidad, se busca dar ofertas y soluciones a los problemas que pueda haber en el momento de suministrar la energía en los diferentes puntos.

#### **2.3.1.7 Reserva no Rodante**

Es muy similar a la mencionada anteriormente puesto que lo que busca es general un balance entre la energía generada y entregada con la demanda que se tenga en ese instante de tiempo, la pequeña diferencia es que el anterior tenía una respuesta rápida a estos desbalances, en cambio en este tipo de programa la respuesta puede tener cierto tipo de retrasos dependiendo del tipo del evento.

### **2.3.2 Programas Basados en Tiempo de Uso**

Estos programas tienen una perspectiva diferente a los PBI ya que en este caso los usuarios no tienen ni incentivos ni penalidades, lo que se busca es aplanar o reducir la curva de demanda mediante metodologías dinámicas que fijan precios y tarifas. Una vez fijados son comunicados periódicamente a los usuarios, para que estén en capacidad de programar su consumo y tomar decisiones respecto al precio de la energía (Marulanda García, 2014).

Los diferentes tipos de programas de forma PBT en los que pueden participar los usuarios son los siguientes (Marulanda García, 2014).:

### **2.3.2.1 Precio Pico Crítico**

Para este programa es necesario que se estipulen dos tipos de precio, de los cuales el usuario es notificado con poco tiempo de antelación. El primero es el precio de pico crítico, que funciona ante condiciones de emergencia o precios elevados del mercado durante un número limitado de horas al día y por otro lado está el precio ordinario en los periodos normales de consumo.

### **2.3.2.2 Precio Pico Crítico con Control**

Es muy parecido al anterior programa explicado, puesto que es la combinación de las modalidades de Precio Pico Crítico y el Control Directo de Carga. Es decir, que la metodología es la misma que el anterior en donde se estipulan los dos precios, pero en el momento en el que el operador decida realizar un corte remoto debido a una contingencia o precios elevados, está en todo el derecho de hacerlo.

### **2.3.2.3 Descuento de Tiempo Pico**

Para este programa se tiene un valor fijo para todo el año que no cambia a pesar de tener días pico de consumo y demás, pero el usuario puede llegar a recibir cierto tipo de descuento cuando este realice una reducción notable de en su carga según su consumo promedio en los días que se cataloguen como días pico o críticos. Este lo que busca es que se pueda reducir de manera voluntaria sin perder la confiabilidad.

### **2.3.2.4 Precio en Tiempo Real**

Este es uno de los programas más curiosos ya que el usuario obtiene cierto tipo de beneficio pues el precio ofrecido por el operador es el costo real de la energía en el mercado mayorista, y los participantes reciben boletines con la información necesaria de los precios del día o en su dado casa la hora en la que se encuentra puesto que estos varían dependiendo de la época.

### **2.3.2.5 Precio por Tiempo de Uso**

Para este tipo de programa el operador divide el día en ciertos periodos y se establece diferentes precios dependiendo de la hora en la que se consuma, con el fin de reducir el consumo en las horas pico y trasladarlo a horas de menor demanda. Es claro que en las horas de la noche será más caro el precio y este no corresponderá o se asemejará al precio real que posea el mercado.

Se analizaron los diferentes programas de respuesta a la demanda y las ventajas que ofrecen para el desarrollo del proyecto, las razones de la elección de estos programas se mencionan al final de la sección 3. Serán usados los siguientes:

- Oferta a la demanda
- Descuento de tiempo pico

## **2.4 MODELO COMPUTACIONAL**

Es prácticamente el uso de matemáticas, física o informática para estudiar el comportamiento de sistemas de alta densidad mediante la simulación por computadora o dispositivo móvil, contiene numerosas variables que caracterizan el sistema bajo estudio. La simulación se realiza ajustando estas variables y observando cómo los cambios afectan los resultados pronosticados por el modelo. Los resultados de las simulaciones de los modelos ayudan a realizar predicciones acerca de qué pasará en el sistema real que se está estudiando en respuesta a condiciones cambiantes. Normalmente se utiliza para generar una plataforma de prueba en medio de una investigación con el fin de verificar y corroborar el comportamiento de un sistema como tal. (Bioengineering, 2015)

## **2.5 APLICACIÓN WEB**

La implementación de los programas de respuesta a la demanda en el sistema es por medio de una aplicación Web, que es un sitio Web que contiene páginas con

contenido sin determinar, parcialmente o en su totalidad. El contenido final de una página se determina sólo cuando el usuario solicita una página del servidor Web. Dado que el contenido final de la página varía de una petición a otra en función de las acciones del visitante, este tipo de página se denomina página dinámica. (Adobe Systems Incorporated and its licensors, 2015)

Cuando se utiliza una aplicación web se está trabajando desde una computadora o un dispositivo móvil, pero la mayor parte del procesamiento se hace dentro de una red de servidores. (GCF Global, 2015)

Ventajas de las aplicaciones Web:

- Muchas aplicaciones web son gratuitas.
- Acceso a información en cualquier lugar y momento.
- No hay dependencia de un computador o de algún equipo específico ya que el contenido está almacenado en la web.
- Muchas de las aplicaciones web permiten que varias personas trabajen simultáneamente en ellas.
- Los documentos y archivos no se van a perder ni borrar a menos que se quiera.

## **2.6 APLICACIÓN MÓVIL**

Junto con la aplicación web está la aplicación móvil para la implementación de los PRD, esta es un software o un programa que el usuario puede descargar desde una plataforma destinada para ello y al que puede acceder directamente desde su teléfono o desde algún otro aparato móvil ya sea una Tablet o cualquier otro aparato. (Comercio, 2011)

Existen 3 tipos de aplicaciones dependiendo del método y la finalidad con la que se desarrollan, las cuales son:

- App Nativas: Son aquellas aplicaciones que se desarrollan o se programan de una manera específica dependiendo del sistema operativo el cual la vaya a utilizar, llamado SDK dependiendo si es para iOS (Objective – C), Android (Java) o Windows phone (.Net). Estas aplicaciones no necesitan de una conexión a internet para que estas funcionen, sencillamente se descargan directamente de las tiendas según el sistema operativo y se instala en el dispositivo. (Talent, 2013). En la tabla 2 se muestran las ventajas y las desventajas de las apps nativas.

Tabla 2. Ventajas y desventajas app nativas.

Ventajas	Inconvenientes
<ul style="list-style-type: none"> <li>• Acceso completo al dispositivo</li> <li>• Mejor experiencia del usuario</li> <li>• Visibilidad en APP Store</li> <li>• Envío de notificaciones o “avisos” a los usuarios</li> <li>• La actualización de la app es constante</li> </ul>	<ul style="list-style-type: none"> <li>• Diferentes habilidades / idiomas / herramientas para cada plataforma de destino</li> <li>• Tienden a ser más caras de desarrollar</li> <li>• El código del cliente no es reutilizable entre las diferentes plataformas</li> </ul>

Tomado de (Talent, 2013)

- Web App: Este tipo es completamente diferente al que anteriormente se mencionó puesto que el método o el lenguaje en el cual se programará es más conocido y manejado por los programadores en general, tales como HTML, CSS y JavaScript son las bases de dichas aplicaciones. Las aplicaciones web se ejecutan dentro del propio navegador web del dispositivo a través de una URL, en realidad la gran diferencia con una aplicación nativa es que no necesita instalación. Las apps web móviles son siempre una buena opción si nuestro objetivo es adaptar la web a formato móvil. (Talent, 2013). En la tabla 3 se muestran las ventajas y las desventajas de las Web app.

Tabla 3. Ventajas y desventajas Web app.

Ventajas	Inconvenientes
<ul style="list-style-type: none"> <li>• El mismo código base reutilizable en múltiples plataformas</li> <li>• Proceso de desarrollo más sencillo y económico</li> <li>• No necesitan ninguna aprobación externa para publicarse .</li> <li>• El usuario siempre dispone de la última versión</li> <li>• Pueden reutilizarse sitios “responsive” ya diseñados</li> </ul>	<ul style="list-style-type: none"> <li>• Requiere de conexión a internet</li> <li>• Acceso muy limitado a los elementos y características de hardware del dispositivo</li> <li>• La experiencia del usuario (navegación, interacción...) y el tiempo de respuesta es menor que en una app nativa</li> <li>• Requiere de mayor esfuerzo en promoción y visibilidad</li> </ul>

Tomado de (Talent, 2013)

- Web app Nativa: Estas son conocidas como aplicaciones híbridas, en sí es la combinación entre las dos mencionadas anteriormente, porque son desarrolladas en los mismos lenguajes que el anterior, es decir, JavaScript, CSS y HTML. Lo que facilita el uso de escenarios y plataformas que obtienen gran cantidad de características del dispositivo como tal. En diferencia a la anterior, esta sí tiene la posibilidad de ser descargada directamente de una tienda virtual dependiendo del sistema operativo. (Talent, 2013). En la tabla 4 se muestran las ventajas y las desventajas de las Web apps nativas.

Tabla 4. Ventas y desventajas Web app nativa.

Ventajas	Inconvenientes
<ul style="list-style-type: none"> <li>• Es posible distribuirla en las tiendas de iOS y Android</li> <li>• Instalación nativa pero construida con JavaScript, HTML y CSS</li> <li>• El mismo código base para múltiples plataformas</li> <li>• Acceso a parte del hardware del dispositivo</li> </ul>	<ul style="list-style-type: none"> <li>• Experiencia del usuario más propia de la aplicación web que de la app nativa</li> <li>• Diseño visual no siempre relacionado con el sistema operativo en el que se muestre</li> </ul>

Tomado de (Talent, 2013)



## **2.7 DISEÑO DE SOFTWARE**

Para dar paso al desarrollo de las aplicaciones debe de haber un diseño previo y para esto lo primero que se debe hacer es definir los requisitos del software a desarrollar, pues definen que debe tener y como va a funcionar. A partir de estos se realiza el diseño y tiene las siguientes etapas (Torres, 2015).:

- Diseño de datos: Hace referencia a las estructuras de datos, que se requieren para implementar el software.
- Diseño Arquitectónico: Es la relación que tendrán los principales elementos estructurales en el sistema para su funcionamiento.
- Diseño Procedimental: Transforma los elementos estructurales en una descripción de los diferentes procedimientos que llevará acabo el sistema.

Para este sistema los elementos de cada etapa son: Bases de datos, casos de uso y algoritmos, respectivamente.

### **2.7.1 Bases de datos**

Una base de datos es una herramienta para recopilar y organizar información estructurada y relacionada (Microsoft, 2007), en esta se almacenan grandes cantidades de información y se automatiza el acceso a la mismas lo que facilita la utilización de la misma.

Entre las principales características de los sistemas de base de datos se pueden mencionar:

- Independencia lógica y física de los datos.
- Redundancia mínima.
- Acceso concurrente por parte de múltiples usuarios.
- Integridad de los datos.
- Consultas complejas optimizadas.
- Seguridad de acceso y auditoría.

- Respaldo y recuperación.
- Acceso a través de lenguajes de programación estándar.

Para el uso adecuado de las mismas, las bases de datos cuentan con datos que deben contar una organización lógica y con partes fundamentales de la misma, que son las tablas, formularios, informes, consultas. Esto se logra a través de sistemas gestores de bases de datos, software con capacidad para definir, mantener y utilizar una base de datos. (Berzal Galiano , 2016)

Información a través del tiempo: Cuando se crea una base de datos y se inicia su operación el volumen de la información es mínimo y el desempeño de la aplicación y base de datos es óptima. Sin embargo, conforme pasa el tiempo el volumen de información incrementa y en consecuencia surgen los siguientes problemas:

- Problemas de espacio en disco.
- Incrementos de costo en almacenamiento y procesamiento.
- Problemas de desempeño en las consultas.
- Fallas en el aplicativo afectando la disponibilidad de los datos.
- Problemas con la base de datos y la pérdida de información.

De los problemas anteriores surge la necesidad de hacer mejoras a las bases de datos y/o crear arquitecturas para dar soluciones a los problemas anteriores. Se han implementado varias soluciones a nivel Software y Hardware. (Anguiano Morales, 2014)

### **2.7.2 Casos de uso**

Los casos de uso se utilizan para modelar cómo un sistema funciona, o cómo los usuarios desean que funcione. Consiste en actores y casos de uso, los actores representan usuarios y otros sistemas que interactúan con el sistema y los casos de uso representan el comportamiento del sistema, los escenarios que el sistema

atraviesa en respuesta a un estímulo desde un actor (Popkin Software and Systems, 2002).

Existe el Lenguaje Unificado de Modelado (UML) que prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos significan. Mientras que ha habido muchas notaciones y métodos usados para el diseño orientado a objetos, ahora los modeladores sólo tienen que aprender una única notación (Popkin Software and Systems, 2002).

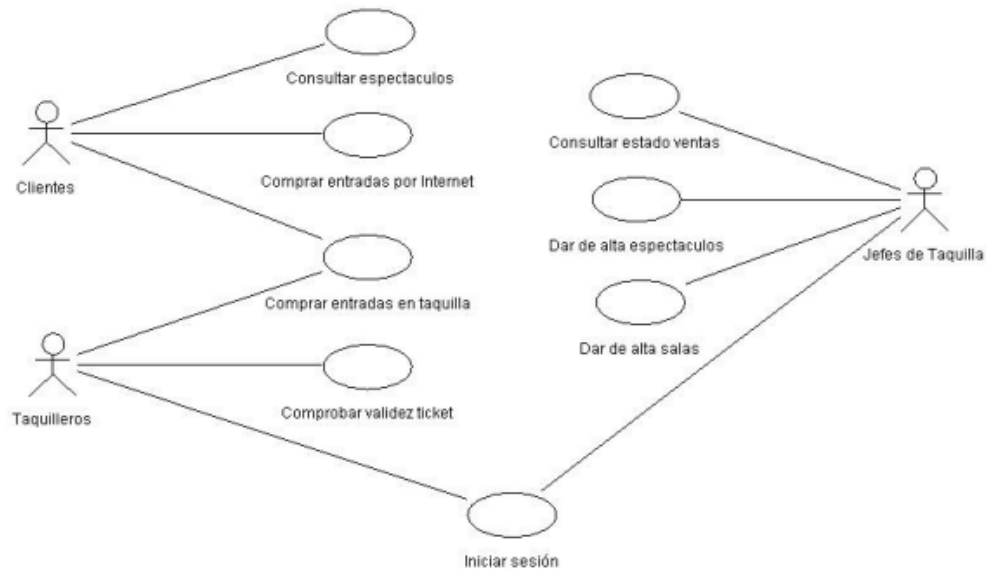
UML ofrece nueve diagramas en los cuales modelar sistemas y representar alguna parte o punto de vista del mismo (Popkin Software and Systems, 2002).:

- Diagramas de Casos de Uso para modelar los procesos.
- Diagramas de Secuencia para modelar el paso de mensajes entre objetos.
- Diagramas de Colaboración para modelar interacciones entre objetos.
- Diagramas de Estado para modelar el comportamiento de los objetos en el sistema.
- Diagramas de Actividad para modelar el comportamiento de los Casos de Uso, objetos u operaciones.
- Diagramas de Clases para modelar la estructura estática de las clases en el sistema.
- Diagramas de Objetos para modelar la estructura estática de los objetos en el sistema.
- Diagramas de Componentes para modelar componentes.
- Diagramas de Implementación para modelar la distribución del sistema.

Los diagramas de casos de uso son la técnica más efectiva y a la vez la más simple para modelar los requisitos del sistema desde la perspectiva del usuario.

En la figura 3 se muestra cómo se desarrolla un diagrama de casos de uso con un ejemplo para un sistema de venta de entradas para espectáculos, donde se involucra a los principales actores y actividades generales.

Figura 3. Diagrama de casos de uso



Tomado de (Hernández Orallo, 2002).

### 2.7.3 Algoritmos

Un algoritmo se puede definir como una secuencia de instrucciones que representan un modelo de solución para determinado tipo de problemas. O bien como un conjunto de instrucciones que realizadas en orden conducen a obtener la solución de un problema (Informática, Universidad Nacional del Nordeste, 2006).

El diseño de un algoritmo se hace por medio de un pseudocódigo que es la técnica que permite expresar la solución a un problema mediante un algoritmo escrito en palabras normales de un idioma, este debe ser preciso, definido, finito y puede tener cero o más elementos de entrada para producir un resultado (Informática, Universidad Nacional del Nordeste, 2006).

Por último, se deben de tener en cuenta dos características más en relación al diseño de software para el funcionamiento óptimo de la aplicación, es por ello que se tiene (Torres, 2015).:

- Diseño preliminar: Donde se centra en la conversión o transformación de los datos y la información que se obtienen anteriormente, como también de la arquitectura del software.
- Diseño detallado: Se encarga de “refinar” y detallar la estructura arquitectónica que posee el programa para las representaciones algorítmicas del software.

## 2.8 MARCO LEGAL Y NORMATIVO

Para el diseño del sistema y la presentación del documento, se tuvo en cuenta una serie de aspectos legales y normativos con los que se determinaron el alcance y naturaleza del proyecto. En la tabla 5 las leyes y normas involucradas en el desarrollo y aplicación del proyecto.

Tabla 5. Leyes y normas del proyecto.

Leyes	Normas
<ul style="list-style-type: none"> <li>• Ley 1581 de 2012</li> <li>• Decreto 1377 de 2013</li> <li>• Ley 1341 de 2009</li> <li>• Decreto 1526 de Julio 24 de 2002</li> <li>• Ley 1273 de 2009</li> </ul>	<ul style="list-style-type: none"> <li>• Norma NTC 1486</li> <li>• Norma NTC ISO 9000</li> <li>• IEE STD 830 – 1998</li> </ul>

Fuente: Los autores.

Es importante ver de dónde nace el proyecto y que le da el enfoque que tiene, por otro lado, explicar de dónde salen los datos usados, es por esto que el siguiente capítulo son los antecedentes.

### 3. ANTECEDENTES

Este proyecto de grado hace parte de un proyecto de investigación de la VRIT titulado **“Construcción de un sistema de medición inteligente de energía eléctrica de bajo costo, orientado a programas residenciales de respuesta de la demanda para usuarios de bajos recursos”** (Marulanda & Lancheros Cuesta, 2015) el cual es necesario conocer para entender las siguientes secciones que comprenden el desarrollo de este proyecto. A continuación, un resumen.

Debido a un incremento en la producción de dióxido de carbono a raíz del aumento en el consumo de energía, un nuevo hábito de consumo podría generar una reducción en esto lo cual daría un aporte en la disminución de gases de efecto invernadero los cuales se encuentran afectando el medio ambiente. (The World Bank, 2014) Según investigaciones es posible reducir esto con cambios pequeños cambios en los hogares que terminan siendo significativos frente a la problemática mencionada anteriormente. (Lancheros Cuesta & Pérez Lara, 2014) El proyecto plantea que bajo los programas de Respuesta de la Demanda puede lograrse la disminución necesaria para observar un cambio dentro del sistema.

Los objetivos son los siguientes:

Objetivo general: Construir un sistema de medición inteligente de bajo costo orientado a la implementación de programas residenciales de respuesta de la demanda, y emplearlo para caracterizar el consumo de energía eléctrica en usuarios de bajos recursos.

Objetivos específicos:

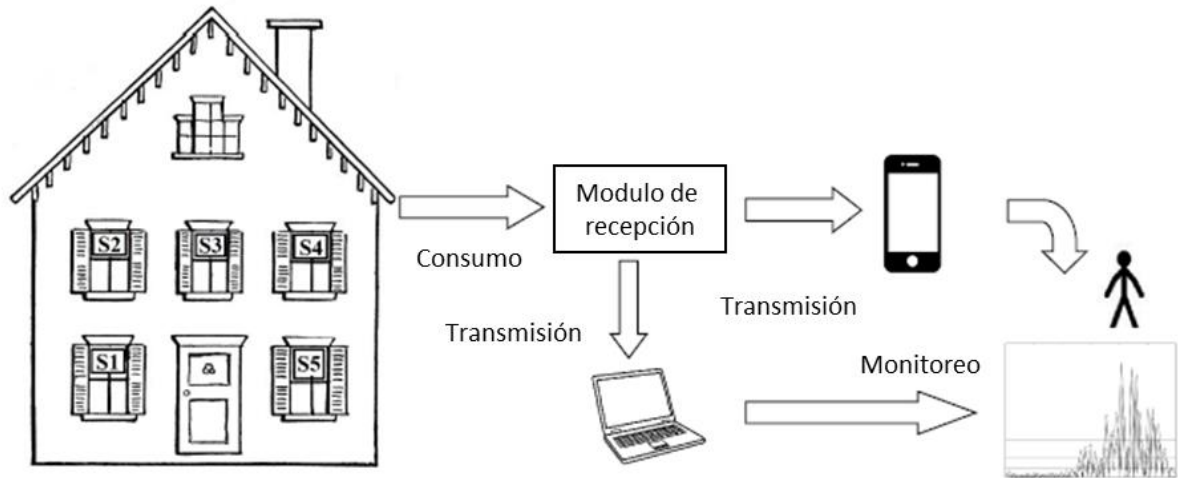
- Caracterizar el consumo típico de un usuario residencial de bajos recursos
- Construir un medidor inteligente de bajo costo, que permita a un usuario residencial de escasos recursos participar de forma activa en un programa de respuesta de la demanda

- Implementar un sistema de información que permita acceder a reportes de consumo de energía eléctrica
- Analizar el consumo de un usuario residencial de bajos recursos ante distintas señales que pueda emplear para participar en un programa de respuesta de la demanda.

Su justificación radica en que a mayor cantidad de programas de la respuesta a la demanda implementados en los hogares se obtendrá un mejor beneficio para el sector eléctrico, debido a indirectamente se estaría generando una reducción en el consumo que podría verse reflejada en la facturación mensual pudiendo así los usuarios tener algunos ingresos económicos por la participación en estos. (Marulanda García, 2014) Lo que este trabajo pretende realizar es diseñar un medidor inteligente que pueda adaptarse a las necesidades importantes que posea el sector eléctrico de nuestro país. Teniendo como principal meta, la oportunidad de brindarles el apoyo para que personas de bajos recursos tengan la capacidad de acceder a estos tipos de programas con el fin de generar un beneficio en cada uno de ellos, parte de lo que busca este proyecto es profundizar en la “investigación orientada a la caracterización del consumo en usuarios residenciales para la identificación de alternativas que incentiven la RD en el país.” (Pérez Lara, 2015)

En la figura 4 se observa cómo se estructura el proyecto en donde se cuenta con 5 medidores distribuidos en la casa, los cuales censan y envían la información al módulo de recepción el cual almacenará todo en la base de datos. Para que posteriormente sea consultado mediante el celular o el computador.

Figura 4. Bosquejo proyecto VRIT



Fuente: (Pérez Lara, 2015)

Su metodología se divide en 4 fases que buscan suplir cada uno de los objetivos pactados en un comienzo, la primera de ellas es:

- **Identificación y Análisis**  
Busca identificar aspectos importantes para la caracterización de los usuarios dentro de la construcción del medidor, basados en la investigación del estado de arte.
- **Construcción del medidor**  
Se construirá el medidor teniendo en cuenta varios aspectos importantes como lo son la selección de sensores, las pruebas de sensado, el análisis de las señales, la amplificación y el tratamiento de la señal, la transmisión y recepción de los datos. Para finalmente realizar las pruebas necesarias de su funcionamiento.
- **Diseño e implementación del sistema de información**  
Entra a ver en cómo se visualizarán los datos tanto de la medición como del sistema de monitoreo, de tal forma que estos puedan presentarse al usuario de manera pedagógica orientada al usuario. Para esto se tuvo en cuenta el análisis de los modelos de casos de uso, el diseño de la base de datos, la



caracterización de ventanas y generación de contenido digital, el diseño del entorno gráfico.

- Validación

En esta fase se realizan todas las pruebas de funcionamiento con el fin de analizar el consumo del usuario residencial. En donde comprende una prueba del medido junto al sistema de información, un análisis de la información para caracterizar el consumo y por ultimo simular el impacto que podría llegar a tener este sistema en la sociedad a gran escala.

Una vez entendido el sistema en su totalidad, se puede ver que uno de esos objetivos es el punto de partida para nuestro proyecto plasmando en un sistema de información los reportes de consumo de energía junto a sus históricos, por otro lado, el algoritmo capaz de generar un consumo efectivo para un ahorro deseado por el usuario, pero antes de esto es necesario saber cómo se inició el proyecto con la construcción del medidor inteligente de bajo costo.

En el proyecto ya descrito, también se realizó el trabajo de grado “**Monitoring system for the consumption of electric energy in a household**”. (Pérez Lara, 2015) Donde se diseñaron los medidores encargados de recolectar todos los datos que serán almacenados en las bases de datos para posteriormente ser tomados por nuestro sistema y realizar el respectivo análisis a cada uno de ellos. Este trabajo se estructuró de la siguiente manera:

Una investigación correspondiente con base de la temática trabajada en donde se recorrió todos los aspectos importantes y relevantes que fuesen necesarios para el desarrollo del trabajo, los conceptos para el desarrollo del trabajo fueron:

- Consumo energético
- Sensor de corriente linear Hall Effect
- Sistemas de información
- Bases de datos

Se desarrolló una metodología de 6 partes las cuales se distribuyen en:

- **Análisis del consumo por nodos**  
Para saber cómo y qué elementos eléctricos y electrónicos monitorear, se debe de hacer un estudio del ambiente en donde se realizarán los monitoreos y posteriormente con dicha información seleccionar el sensor apto para el ambiente y con soporte a los elementos escogidos. Para corroborar la elección del sensor se procede a un test donde se verifica la toma de datos y se realiza la validación de los mismos.
- **Etapas de potencia**  
Ubicar el sensor y ponerlo en funcionamiento no es suficiente para una toma de datos eficaz, por eso el sistema requiere de una etapa de potencia, por eso se realiza un análisis y posterior amplificación de las señales de salida del sensor, para una transmisión y recepción de datos, apoyado esto en un protocolo de comunicación diseñado para la adquisición de datos.
- **Sistema de información**  
La adquisición de datos en sí, no sirve de nada si la información no es clasificada y organizada para su posterior análisis y uso por eso una vez garantizada la conexión entre los sensores y la base de datos se procede al diseño de la base de datos que contempla los casos de uso y la selección del servidor para tener un uso óptimo de los datos.
- **HMI**  
Para el monitoreo y análisis de la información, no es suficiente una base de datos, se necesita de una interfaz gráfica que permita una fácil navegación, por eso se analiza y selecciona una herramienta de programación y diseño para esto.

- Pruebas y aprobación  
Las anteriores etapas son testeadas una a una y en conjunto para comprobar su funcionamiento, si es el correcto se procede a aprobación si no a corrección de errores.
- Documentación  
Redacción de manuales de operación de los diferentes elementos del sistema.

Pasando por la metodología en donde se especifica y se recorre todo el proyecto como tal, en donde se da la planeación y estructuración de las tareas. Se pasó a la etapa del desarrollo, en la cual ya se ve evidenciado cada uno de los puntos anteriores pero puestos en práctica, se planteó de la siguiente forma:

a) Análisis del consumo por nodos

- Con el sensor diseñado se procedió a la toma de datos, siendo estos la corriente consumida por diferentes electrodomésticos, la adquisición de datos se realiza cada segundo.
- Los electrodomésticos utilizados para la adquisición de datos son:
  1. Lavadora
  2. Nevera
  3. Plancha de vapor
  4. Tv LED
  5. Bombillas fluorescentes

Como algunos electrodomésticos cuentan con diferentes modos de uso que pueden reducir o aumentar su consumo, los datos se adquirieron cambiando el modo durante la adquisición.

b) Diseño etapa de potencia y circuitos electrónicos.

- Etapa de potencia  
El sensor al estar compuesto por diferentes elementos se necesita que cada uno sea alimentado de manera adecuada, para así evitar el daño de los mismos y asegurar su funcionamiento, la etapa de potencia

contempla diferentes voltajes de alimentación además de la amplificación de las señales.

- Circuito electrónico.

Diseño del circuito electrónico para etapa de potencia, micro controlador, módulo Wi-Fi y diferentes componentes del sensor.

c) Protocolo de comunicación

El protocolo de comunicación está dividido por tareas.

- Comunicación Wi-Fi.
- Programación arduino (Envío y adquisición de datos).
- Comunicación sensores-base de datos.

d) Sistema de monitoreo

- Diseño e implementación de sistema de monitoreo en tiempo real del consumo de energía eléctrica. Se desarrollaron las siguientes tareas para construir dicho sistema.

1. Análisis general del sistema.
2. Establecer usuarios.
3. Establecer requerimientos de usuarios y sistema.
4. Diseño base de datos.
5. Almacenamiento datos.
6. Diseño página Web.
7. Test y validación.

- Las anteriores etapas son testeadas una a una y en conjunto para comprobar su funcionamiento, si es el correcto se procede a aprobación si no a corrección de errores.

Una vez culminado el proyecto mencionado anteriormente, da pie para iniciar este proyecto, partiendo de la base de datos la cual contiene toda la información de los sensores que será usada por la aplicación o la aplicación web para que el usuario pueda interactuar con ella de manera pedagógica, pero antes de ello se debe de

tener cuenta cuales son los programas de la respuesta a la demanda que serán usados dentro de nuestro sistema, los dos programas son:

- Oferta a la demanda

Se escoge este tipo de programa puesto que es donde el usuario interactúa con el proceso y determina un valor de megavatio por la cantidad de reducción que hará a lo largo del mes, se debe tener en cuenta el valor que el usuario proporcionará porque este debe de ser menor al costo de generación que se necesita para cubrir toda la demanda. Lo sé que busca es que el usuario participe a su criterio y gusto del mismo.

- Descuento de tiempo pico

Por otro lado, este es el otro tipo de programa que se implementará debido a que es donde se premia al usuario al tener una reducción en el consumo base que tiene a lo largo del año. Lo que se busca es que esto motive al usuario a generar un hábito de cambio y de esta manera pueda mejorar su demanda en las horas pico de consumo.

Una vez contextualizado el proyecto se da paso al capítulo del sistema Poupá Watt en donde se evidencian sus fases de diseño y desarrollo.

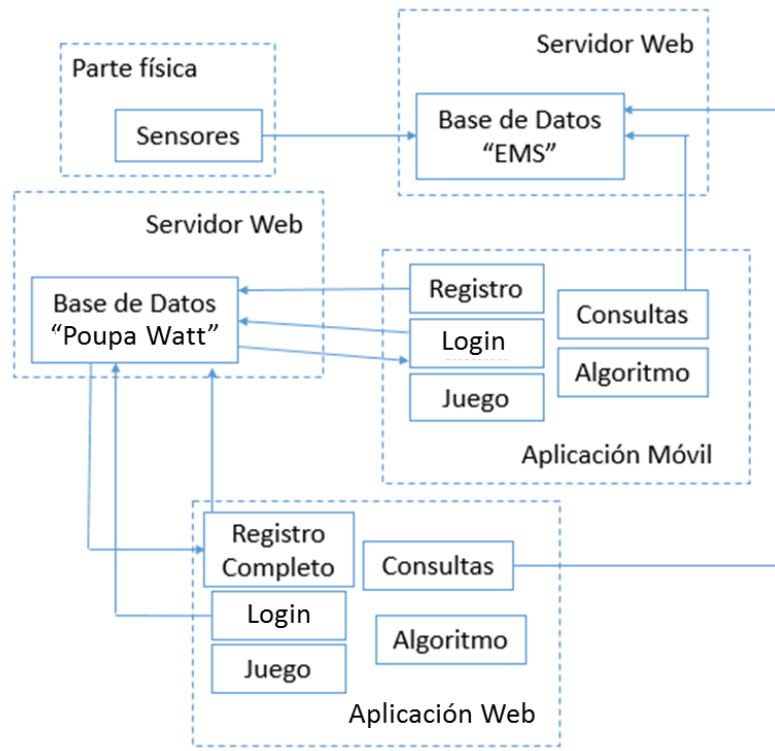
#### **4. SISTEMA POUPA WATT**

El sistema del proyecto tiene por nombre Poupa Watt, poupa palabra en portugués que traduce ahorro, cumpliendo con el objetivo general que es ahorrar energía.

El sistema cuenta con dos bases de datos, una aplicación web y una aplicación móvil, en la figura 5 se presenta el esquema de conexión y funcionamiento de estos elementos. Los datos de consumo (parte física) son recolectados por medio de los sensores, para que estos puedan ser tratados y utilizados en posteriores operaciones deben de ser almacenados; estos son almacenados en la base de datos EMS que es de uso exclusivo para la recolección de los kWh consumidos por los usuarios del sistema. Para que el sistema sea de uso de los diferentes usuarios se utiliza la base de datos Poupa Watt que es de uso exclusivo para el almacenamiento de datos personales y de funcionamiento de los usuarios, además de todos los parámetros del sistema; en estos parámetros se hace manejo de las diferentes partes que componen el sistema, entre estos el manejo de usuarios que corresponde a asignación de sensores para posterior uso de datos de cada usuario.

Para permitir todo este manejo de información, se dispone de una aplicación web, en esta hay interacción del usuario estándar y el usuario administrador donde se tendrá completo manejo de las diferentes funciones del sistema, mientras que la aplicación web está destinada a uso exclusivo de los usuarios estándar donde podrán revisar aspectos básicos como su consumo e interactuar con el juego.

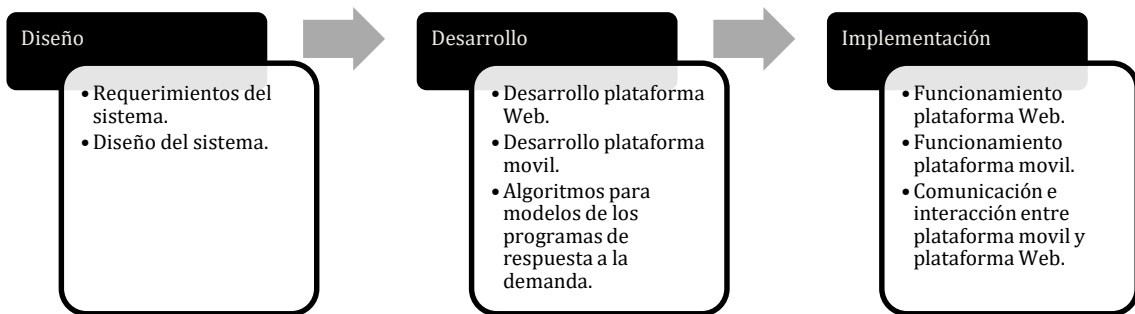
Figura 5. Sistema Poupa Watt



Fuente: Los autores

En las siguientes secciones del documento se dan a conocer los diferentes pasos de creación del sistema, dichos pasos se pueden resumir en 3 fases. Cada una de estas contiene los procedimientos y tareas principales realizadas, como se muestra en la figura 6.

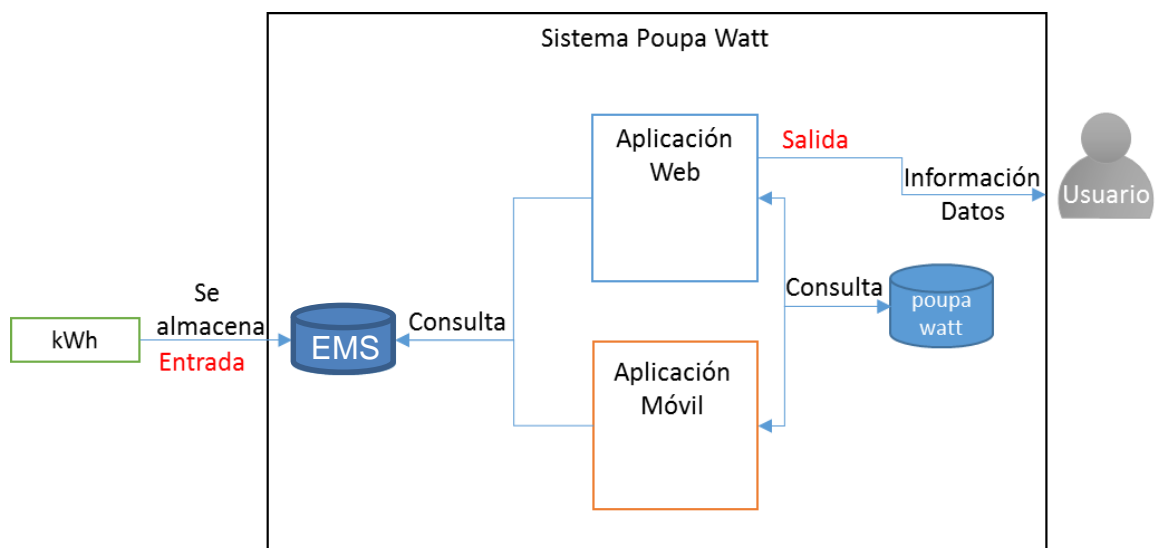
Figura 6. Fases de desarrollo del sistema



Fuente: Los autores.

Para proceder con el diseño y desarrollo del sistema, primero se deben determinar los parámetros de entrada del modelo computacional en términos de energía eléctrica, siendo estos los kWh sensados y recolectados por los smart meters. En la figura 7 se observa la interacción del parámetro de entrada y como se define en el modelo computacional; los kWh sensados son el parámetro de entrada del sistema ya que estos son los que permiten el desarrollo de las diferentes operaciones, estos son parte de la base de datos EMS desde donde son consultados por la aplicación web y la aplicación móvil. Una vez realizadas las diferentes operaciones por los usuarios estos datos son almacenados en la base de datos Poupa Watt para ser posteriormente consultados y observados por los usuarios.

Figura 7. Parámetros de entrada del modelo computacional



Fuente: Los autores

Una vez entendido como es el sistema, se realiza el diseño de todos los diagramas de caso de uso y de todos los requerimientos para la funcionalidad del sistema. Es por eso que el siguiente capítulo se centra en todo el diseño.



## **5. DISEÑO DEL SISTEMA COMPUTACIONAL**

La fase de diseño es fundamental para Poupa Watt pues se hace una descripción completa del sistema a desarrollar, esta se divide en tres partes: el levantamiento de requerimientos y el diseño del sistema de información, determinando los usos que tendrán los diferentes usuarios con el software y su interacción, para posteriormente desarrollar la tercera parte que son los algoritmos que permiten el funcionamiento del sistema.

### **5.1 REQUERIMIENTOS DEL SISTEMA**

Para llevar a cabo un proyecto que implique el desarrollo de software se deben comprender todas las tareas relacionadas con la determinación de necesidades y/o condiciones a satisfacer para el software a desarrollar, teniendo en cuenta los diversos requerimientos del proyecto. A esto se le llama levantamiento de requerimientos. (Arévalo, Linares, Correa, Olarte Parra, & Gonzalez, 2009)

Hay diferentes tipos de requerimientos del sistema, estos son los requisitos del software, que son una descripción del comportamiento del sistema a desarrollar junto con los requisitos no funcionales, que son los que imponen restricciones en el diseño y funcionamiento del sistema además de los requisitos del usuario. (Arévalo, Linares, Correa, Olarte Parra, & Gonzalez, 2009)

Dichos requerimientos están hechos de acuerdo a la norma IEEE Std 830-1998 (IEEE Computer Society, 1998) que describe estructuras posibles y contenido deseable de una especificación de requisitos de software.

#### **5.1.1 Requerimientos del usuario**

En esta sección se describen los requisitos de usuario (UR) del sistema Poupa Watt, dividiéndose en requerimientos de capacidad del sistema y requerimientos de restricción impuestos al sistema por el usuario.

La especificación de requisitos se realiza en una tabla con las siguientes propiedades (Moreno Boiza, 2012):

- **Id:** identificador unívoco de requisito de usuario.
- **Descripción:** descripción de requisito de usuario.
- **Necesidad:** refleja la importancia del requisito para el usuario.
- **Prioridad:** indica la prioridad establecida para un requisito siendo el valor **alta** el de mayor prioridad y **baja** el de menor prioridad.
- **Estabilidad:** indica si el requisito está sujeto a modificaciones.
- **Fuente:** indica la procedencia de la especificación del requisito.

#### 5.1.1.1 *Requerimientos de capacidad*

En la tabla 6 se muestran los requerimientos que hacen referencia a la capacidad del sistema; esta capacidad se refiere a lo que debe ser capaz de hacer.

Tabla 6. Requerimientos de capacidad

<b>Id</b>	<b>Descripción</b>	<b>Necesidad</b>	<b>Prioridad</b>	<b>Estabilidad</b>	<b>Fuente</b>
<b>UR001</b>	Registro de nuevos usuarios.	Esencial	Alta	No modificable	Usuario
<b>UR002</b>	Al registrar un nuevo usuario será obligatorio introducir una contraseña de usuario.	Esencial	Alta	No modificable	Usuario
<b>UR003</b>	Al registrar un nuevo usuario será obligatorio completar información requerida.	Esencial	Alta	No modificable	Usuario
<b>UR004</b>	Permitir habilitar/deshabilitar usuario.	Esencial	Alta	No modificable	Usuario

<b>UR005</b>	Acceso al sistema.	Esencial	Alta	No modificable	Usuario
<b>UR006</b>	Permitir consulta de información.	Esencial	Alta	No modificable	Usuario
<b>UR007</b>	Permitir modificación de información.	Esencial	Alta	No modificable	Usuario
<b>UR008</b>	Permitir agregar información.	Esencial	Alta	No modificable	Usuario
<b>UR009</b>	Permitir cambio de contraseña de usuario.	Esencial	Alta	No modificable	Usuario
<b>UR010</b>	Manejo de usuarios.	Esencial	Alta	No modificable	Usuario
<b>UR011</b>	Mostrar descripción de campos requeridos.	Deseable	Medio	Modificable	Usuario
<b>UR012</b>	Interacción con los programas de respuesta a la demanda.	Esencial	Alta	No modificable	Usuario
<b>UR013</b>	Permitir consultar parámetros del sistema	Esencial	Alta	No modificable	Usuario
<b>UR014</b>	Permitir modificación parámetros del sistema.	Esencial	Alta	No modificable	Usuario
<b>UR015</b>	Permitir adicionar parámetros al sistema.	Esencial	Alta	No modificable	Usuario
<b>UR016</b>	Conexión aplicación móvil con aplicación web.	Esencial	Alta	No modificable	Usuario

Fuente: Los autores.

### **5.1.1.2 Requerimientos de restricción**

La tabla 7 muestra las restricciones impuestas al sistema para su funcionamiento mínimo.

Tabla 7. Requerimientos de restricción.

<b>Id</b>	<b>Descripción</b>	<b>Necesidad</b>	<b>Prioridad</b>	<b>Estabilidad</b>	<b>Fuente</b>
<b>UR017</b>	La aplicación web será de fácil manejo para un usuario relacionado con el entorno Windows, Os X, Linux.	Deseable	Alta	No modificable	Ing. Desarrollo
<b>UR018</b>	La aplicación móvil será de fácil manejo para un usuario relacionado con el entorno Android.	Deseable	Alta	No modificable	Ing. Desarrollo
<b>UR019</b>	La aplicación web funcionará sobre un computador con sistema operativo Windows 7 o superior, Os X 10.6 o superior y Ubuntu 12.04, Debian 7, OpenSuSE 12.2, Fedora Linux 17 o superior.	Esencial	Alta	Modificable	Ing. Desarrollo
<b>UR020</b>	La aplicación móvil funcionará sobre un dispositivo con Android 4.3.1 o superior.	Esencial	Alta	Modificable	Ing. Desarrollo
<b>UR021</b>	La aplicación web funcionará sobre un computador con mínimo un procesador de 1 GHz que soporte SS2, 512 MB de RAM y 1 GB de disco duro.	Esencial	Alta	Modificable	Ing. Desarrollo
<b>UR022</b>	La aplicación móvil funcionará sobre un dispositivo con mínimo un procesador de 1 GHz, 512 MB de	Esencial	Alta	Modificable	Ing. Desarrollo

	RAM y 1 GB de almacenamiento.				
<b>UR023</b>	Los navegadores en que se presentará la plataforma serán, Google Chrome 3.18 o superior y Mozilla Firefox 31 o superior.	Esencial	Alta	Modificable	Ing. Desarrollo
<b>UR024</b>	Para uso de la aplicación web y aplicación móvil será necesario disponer conexión a internet.	Esencial	Alta	No modificable	Ing. Desarrollo
<b>UR025</b>	Se utilizará MySQL como gestor de base de datos.	Esencial	Alta	No modificable	Ing. Desarrollo

Fuente: Los autores.

### 5.1.2 Requerimientos software

En esta sección se describen los requisitos de software del sistema (SR) Poupa Watt, dividiéndose en requerimientos funcionales y requerimientos no funcionales.

La especificación de requisitos se realiza en una tabla con las siguientes propiedades (Moreno Boiza, 2012):

- Identificador: identificador unívoco de requisito software.
- Descripción: descripción de requisito software.
- Necesidad: refleja la importancia del requisito.
- Prioridad: indica la prioridad establecida para un requisito siendo el valor **alto** el de mayor prioridad y **baja** el de menor prioridad.
- Estabilidad: indica si el requisito está sujeto a modificaciones.

- Fuente: indica la procedencia de la especificación del requisito.

### 5.1.2.1 *Requerimientos funcionales*

La tabla 8 muestra los requerimientos funcionales; estos definen qué hace el sistema de acuerdo a los requerimientos de capacidad definidos para el usuario.

Tabla 8. Requerimientos funcionales.

<b>Id</b>	<b>Descripción</b>	<b>Necesidad</b>	<b>Prioridad</b>	<b>Estabilidad</b>	<b>Fuente</b>
<b>SR001</b>	Registro de nuevos usuarios.	Esencial	Alta	No modificable	UR001
<b>SR002</b>	Al registrar un nuevo usuario será obligatorio introducir una contraseña de usuario.	Esencial	Alta	No modificable	UR002
<b>SR003</b>	La contraseña del usuario debe ser mínimo de 8 caracteres con una mayúscula y un número.	Deseable	Alta	Modificable	UR002
<b>SR004</b>	Al crear un nuevo usuario, será estándar por defecto.	Deseable	Alta	Modificable	UR001
<b>SR005</b>	El registro del nuevo usuario, será de forma parcial en la aplicación.	Deseable	Media	Modificable	UR003
<b>SR006</b>	El registro del nuevo usuario en la aplicación web será completo.	Esencial	Alta	No modificable	UR003
<b>SR007</b>	Al registrarse, el usuario debe introducir toda la información solicitada.	Esencial	Alta	No modificable	UR003
<b>SR008</b>	El usuario al solicitar eliminar su perfil, es	Esencial	Alta	Modificable	UR004

	deshabilitado del sistema.				
<b>SR009</b>	Permitir habilitar/deshabilitar usuario por el administrador.	Esencial	Alta	No modificable	UR004
<b>SR010</b>	Permitir consultar información personal del usuario.	Esencial	Alta	No modificable	UR006
<b>SR011</b>	Permitir consultar información de facturación del usuario.	Esencial	Alta	No modificable	UR006
<b>SR012</b>	Permitir consultar preferencias del usuario.	Esencial	Alta	No modificable	UR006
<b>SR013</b>	Permitir consultar consumo del usuario.	Esencial	Alta	No modificable	UR006
<b>SR014</b>	Permitir consultar consumo de los usuarios por parte del administrador.	Esencial	Alta	No modificable	UR010
<b>SR015</b>	Permitir consultar información personal de los usuarios por parte del administrador.	Esencial	Alta	No modificable	UR010
<b>SR016</b>	Permitir consultar información de facturación de los usuarios por parte del administrador.	Esencial	Alta	No modificable	UR010
<b>SR017</b>	Permitir modificar información personal del usuario.	Esencial	Alta	No modificable	UR007
<b>SR018</b>	Permitir modificar información de facturación del usuario.	Esencial	Alta	No modificable	UR007

<b>SR019</b>	Permitir modificar preferencias del usuario.	Esencial	Alta	No modificable	UR007
<b>SR020</b>	Permitir modificar información personal de los usuarios por parte del administrador.	Esencial	Alta	No modificable	UR010
<b>SR021</b>	Permitir modificar información de facturación de los usuarios por parte del administrador.	Esencial	Alta	No modificable	UR010
<b>SR022</b>	Permitir agregar información de facturación.	Esencial	Alta	No modificable	UR008
<b>SR023</b>	Permitir cambio de contraseña de usuario.	Esencial	Alta	No modificable	UR009
<b>SR024</b>	Eliminar usuarios del sistema por el administrador.	Esencial	Alta	No modificable	UR010
<b>SR025</b>	Cambiar tipo de usuario.	Esencial	Alta	No modificable	UR010
<b>SR026</b>	Mostrar descripción de campos requeridos.	Deseable	Media	Modificable	UR011
<b>SR027</b>	Aplicación modelos de los programas de respuesta a la demanda.	Esencial	Alta	No modificable	UR012
<b>SR028</b>	Consultar parámetros del sistema.	Esencial	Alta	No modificable	UR013
<b>SR029</b>	Permitir modificar programas de respuesta a la demanda.	Esencial	Alta	No modificable	UR014
<b>SR030</b>	Permitir modificar asignación de sensores.	Esencial	Alta	No modificable	UR014
<b>SR031</b>	Permitir modificar información de electrodomésticos.	Esencial	Alta	No modificable	UR014



<b>SR032</b>	Permitir adición de programas de respuesta a la demanda.	Esencial	Alta	No modificable	UR015
<b>SR033</b>	Permitir adición de sensores.	Esencial	Alta	No modificable	UR015
<b>SR034</b>	Permitir adición de electrodomésticos.	Esencial	Alta	No modificable	UR015
<b>SR035</b>	Conexión aplicación móvil con aplicación web.	Esencial	Alta	No modificable	UR016

Fuente: Los autores.

### **5.1.2.2 Requerimientos no funcionales de interfaz**

La tabla 9 muestra los requerimientos de interfaz que definen la interacción y comunicación de las interfaces, aplicación móvil y aplicación web del sistema de acuerdo a los requisitos de restricción definidos para el usuario.

Tabla 9. Requerimientos no funcionales de interfaz.

<b>Id</b>	<b>Descripción</b>	<b>Necesidad</b>	<b>Prioridad</b>	<b>Estabilidad</b>	<b>Fuente</b>
<b>SR036</b>	La plataforma Web será de fácil manejo para un usuario relacionado con el entorno Windows, Os X, Linux.	Deseable	Alta	No modificable	UR017
<b>SR037</b>	La aplicación móvil será de fácil manejo para un usuario relacionado con el entorno Android.	Deseable	Alta	No modificable	UR018
<b>SR038</b>	La aplicación web funcionará sobre un computador con sistema operativo Windows 7 o superior, Os X 10.6 o superior y Ubuntu 12.04, Debian 7, OpenSuSE 12.2,	Esencial	Alta	Modificable	UR019

	Fedora Linux 17 o superior.				
<b>SR039</b>	La aplicación móvil funcionará sobre un dispositivo con Android 4.3.1 o superior.	Esencial	Alta	Modificable	UR020
<b>SR040</b>	Los navegadores en que se presentará la plataforma serán, Google Chrome 3.18 o superior y Mozilla Firefox 31 o superior.	Esencial	Alta	Modificable	UR023

Fuente: Los autores.

### **5.1.2.3 Requerimientos no funcionales de recursos**

La tabla 10 muestra los requerimientos de recursos, los cuales definen los recursos mínimos para que la aplicación web y la aplicación móvil funcionen de manera adecuada en los diferentes dispositivos.

Tabla 10. Requerimientos no funcionales de recursos.

<b>Id</b>	<b>Descripción</b>	<b>Necesidad</b>	<b>Prioridad</b>	<b>Estabilidad</b>	<b>Fuente</b>
<b>SR041</b>	La aplicación web funcionará sobre un computador con mínimo un procesador de 1 GHz que soporte SS2, 512 MB de RAM y 1 GB de disco duro.	Esencial	Alta	Modificable	UR022
<b>SR042</b>	La aplicación móvil funcionará sobre un dispositivo con mínimo un procesador de 1 GHz, 512 MB de RAM y 1 GB de almacenamiento.	Esencial	Alta	Modificable	UR023

Fuente: Los autores.

#### **5.1.2.4 Requerimientos no funcionales de seguridad**

La tabla 11 muestra los requisitos de seguridad, estos definen las características de seguridad del sistema.

Tabla 11. Requerimientos no funcionales de seguridad.

<b>Id</b>	<b>Descripción</b>	<b>Necesidad</b>	<b>Prioridad</b>	<b>Estabilidad</b>	<b>Fuente</b>
<b>SR043</b>	Acceso al sistema a usuarios habilitados.	Esencial	Alta	Modificable	UR005
<b>SR044</b>	Las contraseñas manipuladas por el sistema deben ser cifradas por MD5.	Esencial	Alta	Modificable	UR002

Fuente: Los autores.

## **5.2 DISEÑO DEL SISTEMA.**

En el diseño del sistema se da paso a la organización lógica del software, proporcionando un enlace entre los requerimientos y el desarrollo a través de un modelo o representación que será construido posteriormente. Dicho modelo es realizado a través de lenguaje de modelamiento unificado (UML) que permite visualizar, especificar y documentar el diseño del sistema. (Popkin Software and Systems, 2002)

De esta manera se procede a establecer los diferentes usuarios del sistema y con estos dar cumplimiento a los requerimientos.

### **5.2.1 Usuarios**

Los usuarios que interactúan con el sistema son dos, cada uno con funciones, privilegios y acciones diferentes. Por lo tanto, se establecen los siguientes tipos de usuario:

1. Usuario Estándar:

El usuario estándar es la persona que participa en los programas de respuesta a la demanda, además de contar con sensores ya instalados e información recolectada de los mismos, para poder hacer uso pleno del sistema.

2. Usuario Administrador:

El usuario administrador es el encargado del manejo del sistema, tanto de usuarios como parámetros del mismo, esta persona es designada por los directores del proyecto y debe contar con los conocimientos relacionados al sistema para brindar una buena administración.

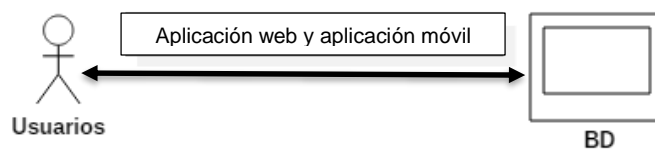
### 5.2.2 Casos de uso

Todo sistema ofrece a sus usuarios una serie de servicios, y un caso de uso, es una secuencia de interacciones entre un sistema y alguien o algo que usa alguno de sus servicios.

Dichos casos son descritos a través de diagramas que sirven para especificar la comunicación y el comportamiento de un sistema mediante su interacción con los usuarios y/u otros sistemas. En estos se evita el empleo de jergas técnicas, prefiriendo en su lugar un lenguaje más cercano al usuario final. (Marco de Desarrollo de la Junta de Andalucía, 2015)

Con base en lo anterior y dando cumplimiento a los diferentes requerimientos, e n la figura 8 se muestra la interacción de los usuarios con la base de datos a través de la aplicación web y la aplicación web.

Figura 8. Interacción usuarios y sistema



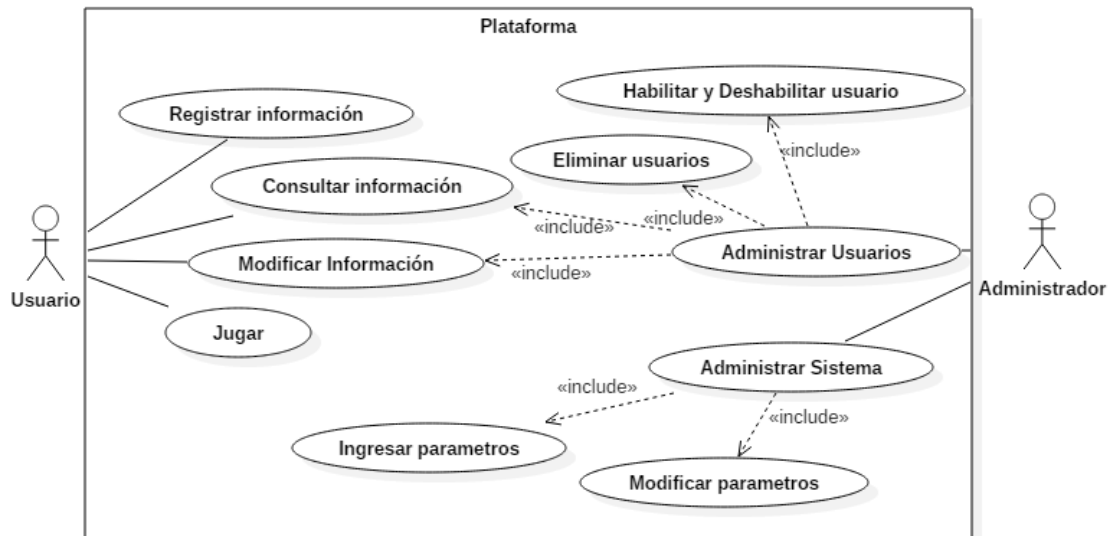
Fuente: Los autores.

Establecida la interacción del sistema, se presentan los diferentes casos de uso para los diferentes usuarios en la aplicación móvil y la aplicación web. En los diagramas y tablas, el usuario estándar será nombrado como “usuario” y el usuario administrador como “administrador”.

### 5.2.2.1 CASOS DE USO APLICACIÓN WEB

La aplicación web interactúa con los dos tipos de usuarios del sistema como se observa en la figura 9 que presenta los casos de uso a nivel general, pues de cada uno se desprenden casos de uso específicos.

Figura 9. Diagrama casos de uso en aplicación web



Fuente: Los autores.

#### 5.2.2.1.1 Casos de uso usuario estándar

Para entender la interacción del usuario estándar con la aplicación web se realizó una descripción de cada uno de los casos a través de las siguientes tablas.

La tabla 12 describe el caso de uso “Registrar Usuario” en la aplicación Web.

Tabla 12. CU001 Registrar usuario.

<b>CU001</b>	<b>Registrar Usuario</b>		
<b>Versión</b>	2.0 (22/06/2016)		
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• SR001</li> <li>• SR002</li> <li>• SR003</li> <li>• SR004</li> </ul>	<ul style="list-style-type: none"> <li>• SR005</li> <li>• SR006</li> <li>• SR007</li> <li>• SR046</li> </ul>	
<b>Precondición</b>	Ser usuario nuevo.		
<b>Descripción</b>	Registrar un nuevo usuario en el sistema.		
<b>Secuencia Normal</b>	Paso	Acción.	
	1	Ir a poopawatt.info	
	2	Dar clic en el botón “crear”.	
	3	Introducir una contraseña con mínimo 8 caracteres con una mayúscula y un número.	
	4	Introducir toda la información personal solicitada.	
	5	Dar clic en el botón “siguiente”	
	6	Introducir toda la información de facturación solicitada.	
	7	Dar clic en el botón “siguiente”.	
	8	Introducir toda la información de preferencias solicitada.	
	9	Dar clic en el botón “finalizar”.	
<b>Postcondición</b>	El usuario queda registrado en el sistema.		
<b>Excepciones</b>	Paso	Acción	
	4	Si el usuario se registra por medio de la aplicación.	
		4.1	Introducir toda la información solicitada.
		4.2	Terminar registro en la aplicación web en la sección “Perfil”.
	6	Si el usuario se registra por medio de la aplicación.	
		6.1	Registrar información de facturación en la aplicación web en la sección “Facturas”
	8	Si el usuario se registra por medio de la aplicación.	
		8.1	Introducir toda la información solicitada.
		8.2	Terminar registro en la aplicación web en la sección “Preferencias”.
	<b>Comentarios</b>	El nuevo usuario es registrado como usuario estándar.	

Fuente: Los autores.

La tabla 13 describe el caso de uso “Registrar Facturas” en la aplicación Web.

Tabla 13. CU002 Registrar Facturas.

<b>CU002</b>	<b>Registrar Facturas</b>	
<b>Versión</b>	2.0 (22/06/2016)	
<b>Dependencias</b>	SR022	
<b>Precondición</b>	Tener una sesión iniciada. Factura no existente.	
<b>Descripción</b>	Registrar una nueva factura.	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Ir a la sección “Factura”
	2	Dar clic en el botón “adicionar factura”.
	3	Introducir toda la información de facturación solicitada.
4	Dar clic en el botón “agregar”	
<b>Postcondición</b>	Se registró una factura nueva al usuario.	
<b>Excepciones</b>	Ninguna	
<b>Comentarios</b>	El usuario puede registrar cuantas facturas quiera. No hay facturas repetidas el sistema.	

Fuente: Los autores.

La tabla 14 describe el caso de uso “Consultar información personal” en la aplicación Web.

Tabla 14. CU003 Consultar información personal.

<b>CU003</b>	<b>Consultar información personal</b>	
<b>Versión</b>	2.0 (22/06/2016)	
<b>Dependencias</b>	SR010	
<b>Precondición</b>	Tener una sesión iniciada.	
<b>Descripción</b>	Consultar información del perfil del usuario.	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Ir a la sección “Perfil”
<b>Postcondición</b>	El usuario queda registrado en el sistema.	
<b>Excepciones</b>	Ninguna.	
<b>Comentarios</b>	En la sección perfil se visualiza toda la información personal del usuario.	

Fuente: Los autores.

La tabla 15 describe el caso de uso “Consultar facturas” en la aplicación Web.

Tabla 15. CU004 Consultar información de facturación.

<b>CU004</b>	<b>Consultar facturas</b>		
<b>Versión</b>	2.0 (22/06/2016)		
<b>Dependencias</b>	SR011		
<b>Precondición</b>	Tener una sesión iniciada.		
<b>Descripción</b>	Consultar facturas del usuario.		
<b>Secuencia Normal</b>	Paso	Acción.	
	1	Ir a la sección "Factura"	
	2	Seleccionar un rango de fechas.	
	3	Dar clic en el botón "buscar"	
<b>Postcondición</b>	Información de factura seleccionada.		
<b>Excepciones</b>	Paso	Acción.	
	2	Si rango de fechas no existente.	
		2.1	Seleccionar un nuevo rango de fechas valido.
<b>Comentarios</b>	En la sección factura, se visualiza la factura seleccionada.		

Fuente: Los autores.

La tabla 16 describe el caso de uso "Consultar preferencias" en la aplicación Web.

Tabla 16. CU005 Consultar preferencias.

<b>CU005</b>	<b>Consultar Preferencias</b>		
<b>Versión</b>	2.0 (22/06/2016)		
<b>Dependencias</b>	SR012		
<b>Precondición</b>	Tener una sesión iniciada.		
<b>Descripción</b>	Consultar preferencias del usuario		
<b>Secuencia Normal</b>	Paso	Acción.	
	1	Ir a la sección "Preferencias"	
<b>Postcondición</b>	Preferencias del usuario.		
<b>Excepciones</b>	Ninguna.		
<b>Comentarios</b>	En la sección factura, se visualizan las preferencias registradas por el usuario.		

Fuente: Los autores.

La tabla 17 describe el caso de uso "Consultar consumo" en la aplicación Web.



Tabla 17. CU006 Consultar consumo.

<b>CU006</b>	<b>Consultar consumo</b>		
<b>Versión</b>	2.0 (22/06/2016)		
<b>Dependencias</b>	SR013		
<b>Precondición</b>	Tener una sesión iniciada.		
<b>Descripción</b>	Consultar facturas del usuario.		
<b>Secuencia Normal</b>	Paso	Acción.	
	1	Ir a la sección "Consultar"	
	2	Seleccionar consumo o histórico	
	3	Seleccionar rango de fechas	
	4	Dar clic en el botón consultar	
<b>Postcondición</b>	Consumo entre rango de fechas seleccionado.		
<b>Excepciones</b>	Paso	Acción.	
	2	Al seleccionar consumo o histórico	
		2.1	Seleccionar precio o kWh
<b>Comentarios</b>	Ninguno.		

Fuente: Los autores.

La tabla 18 describe el caso de uso "Modificar perfil" en la aplicación Web.

Tabla 18. CU007 Modificar información del usuario.

<b>CU007</b>	<b>Modificar Perfil</b>		
<b>Versión</b>	2.0 (22/06/2016)		
<b>Dependencias</b>	SR017		
<b>Precondición</b>	Tener una sesión iniciada.		
<b>Descripción</b>	Modificar información personal del perfil.		
<b>Secuencia Normal</b>	Paso	Acción.	
	1	Ir a la sección "Perfil"	
	2	Seleccionar campo a modificar.	
	3	Introducir nueva información.	
	4	Dar clic en el botón "actualizar"	
<b>Postcondición</b>	Información personal actualizada.		
<b>Excepciones</b>	Paso	Acción.	
	2	El campo documento no es modificable.	
<b>Comentarios</b>	Ninguno.		

Fuente: Los autores.

La tabla 19 describe el caso de uso "Modificar Factura" en la aplicación Web.

Tabla 19. CU008 Modificar información de facturación.

<b>CU008</b>	<b>Modificar Factura</b>	
<b>Versión</b>	2.0 (22/06/2016)	
<b>Dependencias</b>	SR018	
<b>Precondición</b>	Tener una sesión iniciada.	
<b>Descripción</b>	Modificar facturas del usuario.	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Ir a la sección "Factura"
	2	Seleccionar un rango de fechas.
	3	Seleccionar campo a modificar.
	4	Introducir nueva información.
	5	Dar clic en el botón "actualizar"
<b>Postcondición</b>	Factura actualizada.	
<b>Excepciones</b>	Paso	Acción.
	3	Los campos de fechas de facturación no son modificables.
<b>Comentarios</b>	Ninguno.	

Fuente: Los autores.

La tabla 20 describe el caso de uso "Modificar preferencias" en la aplicación Web.

Tabla 20. CU009 Modificar preferencias.

<b>CU009</b>	<b>Modificar Preferencias</b>	
<b>Versión</b>	2.0 (22/06/2016)	
<b>Dependencias</b>	SR019	
<b>Precondición</b>	Tener una sesión iniciada.	
<b>Descripción</b>	Modificar preferencias del usuario.	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Ir a la sección "Preferencias"
	2	Seleccionar campo a modificar.
	3	Introducir nueva información.
	4	Dar clic en el botón "actualizar"
<b>Postcondición</b>	Factura actualizada.	
<b>Excepciones</b>	Paso	Acción.
	2	Los campos programa, reducción y valor no son modificables.
<b>Comentarios</b>	La modificación de los campos de la excepción 2 es función del administrador.	

Fuente: Los autores.

La tabla 21 describe el caso de uso “Jugar” en la aplicación Web.

Tabla 21. CU010 Juego plataforma.

<b>CU010</b>	<b>Jugar</b>	
<b>Versión</b>	2.0 (22/06/2016)	
<b>Dependencias</b>	SR027	
<b>Precondición</b>	Tener una sesión iniciada.	
<b>Descripción</b>	Interactuar con los modelos por medio del juego.	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Ir a la sección “Jugar”
	2	Seleccionar rango de fechas.
	3	Seleccionar electrodomésticos.
	4	Seleccionar rango de horas de operación.
	5	Dar clic en el botón “calcular”
<b>Postcondición</b>	Factura actualizada.	
<b>Excepciones</b>	Ninguna.	
<b>Comentarios</b>	La estructura del juego está sujeta al programa de respuesta a la demanda seleccionado en preferencias.	

Fuente: Los autores.

La tabla 22 describe el caso de uso “Eliminar perfil” en la aplicación Web.

Tabla 22. CU011 Eliminar Perfil.

<b>CU011</b>	<b>Eliminar Perfil</b>	
<b>Versión</b>	2.0 (22/06/2016)	
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• SR008</li> <li>• SR041</li> </ul>	
<b>Precondición</b>	Tener una sesión iniciada.	
<b>Descripción</b>	Eliminar perfil del usuario.	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Ir a la sección “Perfil”
	2	Dar clic en el botón “eliminar perfil”
	3	Confirmar acción en el botón “aceptar”
<b>Postcondición</b>	Perfil deshabilitado	
<b>Excepciones</b>	Ninguna.	
<b>Comentarios</b>	El usuario no puede eliminar su perfil del sistema, con esta acción se deshabilita para que los administradores hagan uso de su información.	

Fuente: Los autores.

La tabla 23 describe el caso de uso “Ejecutar algoritmo” en la aplicación Web.

Tabla 23. CU012 Ejecutar algoritmo modelo programa respuesta a la demanda.

<b>CU012</b>	<b>Ejecutar algoritmo</b>		
<b>Versión</b>	3.0 (20/07/2016)		
<b>Dependencias</b>	SR027		
<b>Precondición</b>	Tener una sesión iniciada.		
<b>Descripción</b>	Ejecución algoritmo modelo del programa de respuesta a la demanda.		
<b>Secuencia Normal</b>	Paso	Acción.	
	1	Ir a la sección “Modelos”	
	2	Dar clic en el botón “Calcular”	
<b>Postcondición</b>	Recomendación para ahorrar		
<b>Excepciones</b>	Paso	Acción	
	2	Cuando en preferencias se seleccionó “Oferta a la demanda”	
		2.1	Activar seleccionar horarios
		2.2	Ingresar horarios en los timepicker
<b>Comentarios</b>	El algoritmo a ejecutar depende de las preferencias del usuario.		

Fuente: Los autores.

#### 5.2.2.1.2 Casos de uso usuario administrador

Para entender la interacción del usuario administrador con la aplicación web se realizó una descripción de cada uno de los casos a través de las siguientes tablas.

La tabla 24 describe el caso de uso “Habilitar y deshabilitar usuarios” en la aplicación Web.

Tabla 24. CU013 Habilitar y deshabilitar usuarios

<b>CU013</b>	<b>Habilitar y deshabilitar usuarios</b>		
<b>Versión</b>	2.0 (22/06/2016)		
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• SR009</li> <li>• SR041</li> </ul>		
<b>Precondición</b>	Tener una sesión iniciada.		
<b>Descripción</b>	Habilitar y deshabilitar el acceso al sistema a los usuarios estándar.		
	Paso	Acción.	

<b>Secuencia Normal</b>	1	Ir a la sección "Usuarios"
	2	Introducir número de documento del usuario en el campo "buscar".
	3	Dar clic en el botón "buscar"
	4	En el campo "estado" seleccionar habilitar o deshabilitar según el caso.
	5	Confirmar acción con el botón "aceptar"
<b>Postcondición</b>	Usuario deshabilitado o habilitado según el caso.	
<b>Excepciones</b>	Ninguna.	
<b>Comentarios</b>	Ninguno.	

Fuente: Los autores.

La tabla 25 describe el caso de uso "Consultar información de los usuarios" en la aplicación Web.

Tabla 25. CU014 Consultar información de los usuarios.

<b>CU014</b>	<b>Consultar información de los usuarios</b>	
<b>Versión</b>	2.0 (22/06/2016)	
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• SR014</li> <li>• SR015</li> <li>• SR016</li> </ul>	
<b>Precondición</b>	Tener una sesión iniciada.	
<b>Descripción</b>	Consultar perfil de los usuarios.	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Ir a la sección "Usuarios"
	2	Introducir número de documento del usuario en el campo "buscar".
	3	Dar clic en el botón "buscar"
	4	Seleccionar perfil del usuario.
<b>Postcondición</b>	Perfil del usuario seleccionado.	
<b>Excepciones</b>	Ninguna.	
<b>Comentarios</b>	Al seleccionar el perfil del usuario para consultar su información personal, también se muestran sus preferencias, sus facturas y su consumo.	

Fuente: Los autores.

La tabla 26 describe el caso de uso "Modificar información de los usuarios" en la aplicación Web.

Tabla 26. CU015 Modificar información de los usuarios.

<b>CU015</b>	<b>Modificar información de los usuarios</b>	
<b>Versión</b>	2.0 (22/06/2016)	
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• SR015</li> <li>• SR016</li> </ul>	
<b>Precondición</b>	Tener una sesión iniciada.	
<b>Descripción</b>	Consultar perfil de los usuarios.	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Ir a la sección "Usuarios"
	2	Introducir número de documento del usuario en el campo "buscar".
	3	Dar clic en el botón "buscar"
	4	Dar clic en el botón "modificar"
	5	Seleccionar sección a modificar
	6	Introducir información en campos a modificar
	7	Dar clic en el botón "actualizar"
<b>Postcondición</b>	Información del usuario modificada.	
<b>Excepciones</b>	Paso	Acción.
	5	Modificar información de perfil, facturación o preferencias
		5.1
<b>Comentarios</b>	Al seleccionar el perfil del usuario para modificar su información personal, también se pueden modificar sus preferencias y sus facturas.	

Fuente: Los autores

La tabla 27 describe el caso de uso "Eliminar usuario" en la aplicación Web.

Tabla 27. CU016 Eliminar usuario.

<b>CU016</b>	<b>Eliminar usuario</b>	
<b>Versión</b>	2.0 (22/06/2016)	
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• SR024</li> <li>• SR041</li> </ul>	
<b>Precondición</b>	Tener una sesión iniciada.	
<b>Descripción</b>	Eliminar usuarios del sistema.	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Ir a la sección "Usuarios"
	2	Introducir número de documento del usuario en el campo "buscar".
	3	Dar clic en el botón "buscar"

	4	Dar clic en el botón “eliminar”
	5	Confirmar acción con el botón “aceptar”
<b>Postcondición</b>	Usuario eliminado del sistema.	
<b>Excepciones</b>	Ninguna.	
<b>Comentarios</b>	El usuario es eliminado definitivamente del sistema.	

Fuente: Los autores

La tabla 28 describe el caso de uso “Cambiar tipo de usuario” en la aplicación Web.

Tabla 28. CU017 Cambiar tipo de usuario.

<b>CU017</b>	<b>Cambiar tipo de usuario</b>	
<b>Versión</b>	2.0 (22/06/2016)	
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• SR025</li> <li>• SR041</li> </ul>	
<b>Precondición</b>	Tener una sesión iniciada.	
<b>Descripción</b>	Cambiar el usuario de tipo estándar a administrador y viceversa.	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Ir a la sección “Usuarios”
	2	Introducir número de documento del usuario en el campo “buscar”.
	3	Dar clic en el botón “buscar”
	4	En el campo “estado” seleccionar habilitar o deshabilitar según el caso.
	5	Confirmar acción con el botón “aceptar”
	6	Dar clic en el botón “modificar”
	7	Seleccionar perfil.
	8	En el campo nivel seleccionar el tipo de usuario.
9	Dar clic en el botón “actualizar”	
<b>Postcondición</b>	Usuario deshabilitado o habilitado según el caso.	
<b>Excepciones</b>	Ninguna.	
<b>Comentarios</b>	Ninguno.	

Fuente: Los autores

La tabla 29 describe el caso de uso “Consultar parámetros del sistema” en la aplicación Web.

Tabla 29. CU018 Consultar parámetros del sistema.

<b>CU018</b>	<b>Consultar parámetros del sistema.</b>			
<b>Versión</b>	2.0 (22/06/2016)			
<b>Dependencias</b>	SR028			
<b>Precondición</b>	Tener una sesión iniciada.			
<b>Descripción</b>	Consultar parámetros del sistema.			
<b>Secuencia Normal</b>	Paso	Acción.		
	1	Ir a la sección que se desea consultar.		
<b>Postcondición</b>	Parámetros sección seleccionada.			
<b>Excepciones</b>	Paso	Acción.		
	1	Las secciones a consultar se encuentran en el menú de la plataforma.		
		1.1	Programas	
		1.2	Sensores	
	1.3	Electrodomésticos		
<b>Comentarios</b>	Ninguno.			

Fuente: Los autores

La tabla 30 describe el caso de uso “Modificar programas de respuesta a la demanda” en la aplicación Web.

Tabla 30. CU019 Modificar programas de respuesta a la demanda

<b>CU019</b>	<b>Modificar programas de respuesta a la demanda</b>		
<b>Versión</b>	2.0 (22/06/2016)		
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• SR029</li> <li>• SR042</li> </ul>		
<b>Precondición</b>	Tener una sesión iniciada.		
<b>Descripción</b>	Modificar parámetros de los programas de respuesta a la demanda.		
<b>Secuencia Normal</b>	Paso	Acción.	
	1	Ir a la sección que “programas”	
	2	Seleccionar el programa a modificar	
	3	Dar clic en el botón “modificar”	
	4	Introducir nuevos parámetros en los campos a modificar.	
	5	Dar clic en el botón actualizar.	
<b>Postcondición</b>	Programa modificado.		
<b>Excepciones</b>	Ninguna.		
<b>Comentarios</b>	Cuando se está modificando un programa aparece la opción de ser eliminado.		

Fuente: Los autores.



La tabla 31 describe el caso de uso “Modificar asignación de sensores” en la aplicación Web.

Tabla 31. CU020 Modificar sensores.

<b>CU020</b>	<b>Modificar asignación de sensores.</b>	
<b>Versión</b>	2.0 (22/06/2016)	
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• SR030</li> <li>• SR042</li> </ul>	
<b>Precondición</b>	Tener una sesión iniciada.	
<b>Descripción</b>	Modificar asignación de sensores.	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Ir a la sección que “sensores”.
	2	Seleccionar sensor.
	3	En el campo electrodoméstico seleccionar el electrodoméstico a asignar al sensor.
	4	En el campo usuario seleccionar usuario a asignar el sensor.
	5	Dar clic en el botón guardar.
	6	Confirmar acción con el botón “aceptar”
<b>Postcondición</b>	Sensor asignado.	
<b>Excepciones</b>	Ninguna.	
<b>Comentarios</b>	No hay límite de sensores por usuario. Cuando se está modificando un sensor, aparece la opción de ser eliminado.	

Fuente: Los autores.

La tabla 32 describe el caso de uso “Modificar parámetros de los electrodomésticos” en la aplicación Web.

Tabla 32. CU021 Modificar electrodomésticos.

<b>CU021</b>	<b>Modificar parámetros de electrodomésticos</b>	
<b>Versión</b>	2.0 (22/06/2016)	
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• SR031</li> <li>• SR042</li> </ul>	
<b>Precondición</b>	Tener una sesión iniciada.	
<b>Descripción</b>	Modificar información de los electrodomésticos.	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Ir a la sección que “electrodomésticos”
	2	Seleccionar el electrodoméstico.

	3	Dar clic en el botón “modificar”
	4	Introducir nuevos parámetros en los campos a modificar.
	5	Dar clic en el botón actualizar.
	6	Confirmar acción con el botón “aceptar”
<b>Postcondición</b>	Parámetros del electrodoméstico modificados.	
<b>Excepciones</b>	Ninguna.	
<b>Comentarios</b>	Cuando se está modificando un electrodoméstico, aparece la opción de ser eliminado.	

Fuente: Los autores.

La tabla 33 describe el caso de uso “Adicionar programas de respuesta a la demanda” en la aplicación Web.

Tabla 33. CU022 Adicionar programas de respuesta a la demanda

<b>CU022</b>	<b>Adicionar programas de respuesta a la demanda</b>	
<b>Versión</b>	2.0 (22/06/2016)	
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• SR032</li> <li>• SR042</li> </ul>	
<b>Precondición</b>	Tener una sesión iniciada.	
<b>Descripción</b>	Adicionar programas de respuesta a la demanda.	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Ir a la sección que “programas”
	2	Dar clic al botón “adicionar”.
	3	Introducir la información solicitada
	4	Dar clic al botón “guardar”
	5	Confirmar acción con el botón “aceptar”
<b>Postcondición</b>	Programa de respuesta a la demanda agregado.	
<b>Excepciones</b>	Ninguna.	
<b>Comentarios</b>	Ninguno.	

Fuente: Los autores.

La tabla 34 describe el caso de uso “Adicionar sensores” en la aplicación Web.

Tabla 34. CU023 Adicionar sensores.

<b>CU023</b>	<b>Adicionar sensores</b>
<b>Versión</b>	2.0 (22/06/2016)
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• SR033</li> </ul>

	• SR043	
<b>Precondición</b>	Tener una sesión iniciada.	
<b>Descripción</b>	Adicionar un nuevo sensor al sistema	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Ir a la sección que “sensores”
	2	Dar clic al botón “adicionar”.
	3	Introducir la información solicitada
	4	Dar clic al botón “guardar”
5	Confirmar acción con el botón “aceptar”	
<b>Postcondición</b>	Sensor nuevo agregado al sistema.	
<b>Excepciones</b>	Ninguna.	
<b>Comentarios</b>	Ninguno.	

Fuente: Los autores

La tabla 35 describe el caso de uso “Adicionar electrodomésticos” en la aplicación Web.

Tabla 35. CU024 Adicionar electrodoméstico.

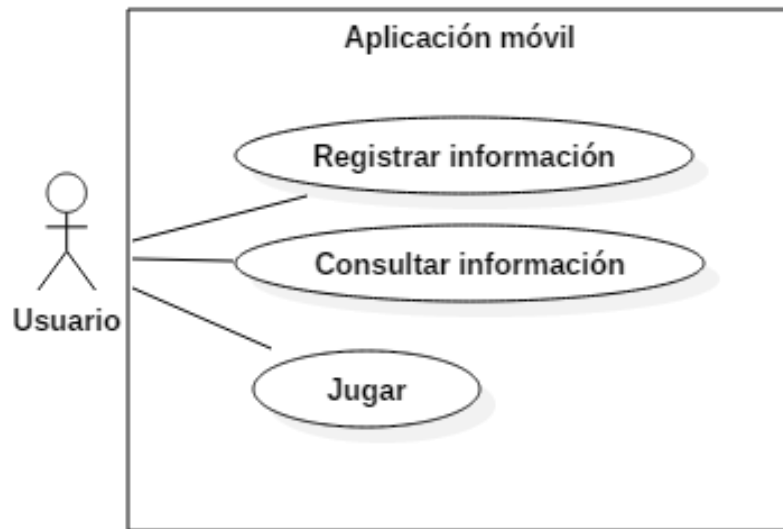
<b>CU024</b>	<b>Adicionar electrodomésticos</b>	
<b>Versión</b>	2.0 (22/06/2016)	
<b>Dependencias</b>	SR034	
<b>Precondición</b>	Tener una sesión iniciada.	
<b>Descripción</b>	Adicionar un nuevo sensor al sistema	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Ir a la sección que “electrodomésticos”
	2	Dar clic al botón “adicionar”.
	3	Introducir la información solicitada
	4	Dar clic al botón “guardar”
5	Confirmar acción con el botón “aceptar”	
<b>Postcondición</b>	Electrodoméstico nuevo agregado al sistema.	
<b>Excepciones</b>	Ninguna.	
<b>Comentarios</b>	Ninguno.	

Fuente: Los autores

### 5.2.2.2 CASOS DE USO APLICACIÓN MÓVIL

En la figura 10 se muestra la interacción exclusiva del usuario estándar con la aplicación móvil y los casos generales de este juntos con los específicos en el desarrollo de la sección.

Figura 10. Diagrama casos de uso aplicación móvil



Fuente: Los autores

#### 5.2.2.2.1 Casos de uso usuario estándar

Para entender la interacción del usuario administrador con la aplicación móvil se realizó una descripción de cada uno de los casos a través de las siguientes tablas.

La tabla 36 describe el caso de uso “Registrar usuario” en la aplicación móvil.

Tabla 36. CU025 Registrar usuario.

<b>CU025</b>	<b>Registrar Usuario</b>	
<b>Versión</b>	2.0 (22/06/2016)	
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• SR001</li> <li>• SR002</li> <li>• SR003</li> <li>• SR004</li> </ul>	<ul style="list-style-type: none"> <li>• SR005</li> <li>• SR006</li> <li>• SR007</li> <li>• SR046</li> </ul>

<b>Precondición</b>	Ser usuario nuevo.	
<b>Descripción</b>	Registrar un nuevo usuario en el sistema.	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Abrir aplicación
	2	Dar clic en el botón “registrarse”
	3	Introducir una contraseña
	4	Introducir toda la información personal solicitada.
	5	Dar clic en el botón “siguiente”.
	6	Introducir toda la información de preferencias solicitada.
	7	Dar clic en el botón “finalizar”.
<b>Postcondición</b>	El usuario queda registrado en el sistema.	
<b>Excepciones</b>	Ninguna.	
<b>Comentarios</b>	El registro es de manera parcial, debe ser terminado en la aplicación web.	

Fuente: Los autores.

La tabla 37 describe el caso de uso “Consultar información personal” en la aplicación móvil.

Tabla 37. CU026 Consultar información personal.

<b>CU026</b>	<b>Consultar información personal</b>	
<b>Versión</b>	2.0 (22/06/2016)	
<b>Dependencias</b>	SR010	
<b>Precondición</b>	Estar en la aplicación con una sesión iniciada.	
<b>Descripción</b>	Consultar información del perfil del usuario.	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Ir a la sección “Perfil”
<b>Postcondición</b>	El usuario queda registrado en el sistema.	
<b>Excepciones</b>	Ninguna.	
<b>Comentarios</b>	En la sección perfil se visualiza toda la información personal del usuario.	

Fuente: Los autores.

La tabla 38 describe el caso de uso “Consultar preferencias del usuario” en la aplicación móvil.

Tabla 38. CU027 Consultar preferencias.

<b>CU027</b>	<b>Consultar preferencias del usuario.</b>	
<b>Versión</b>	2.0 (22/06/2016)	
<b>Dependencias</b>	SR010	
<b>Precondición</b>	Estar en la aplicación con una sesión iniciada.	
<b>Descripción</b>	Consultar información del perfil del usuario.	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Ir a la sección "Preferencias"
<b>Postcondición</b>	El usuario queda registrado en el sistema.	
<b>Excepciones</b>	Ninguna.	
<b>Comentarios</b>	En la sección perfil se visualiza toda la información personal del usuario.	

Fuente: Los autores.

La tabla 39 describe el caso de uso "Consultar consumo" en la aplicación móvil.

Tabla 39. CU028 Consultar consumo.

<b>CU028</b>	<b>Consultar consumo</b>	
<b>Versión</b>	2.0 (22/06/2016)	
<b>Dependencias</b>	SR013	
<b>Precondición</b>	Estar en la aplicación con una sesión iniciada.	
<b>Descripción</b>	Consultar facturas del usuario.	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Ir a la sección "Consultar"
	2	Seleccionar consumo o histórico
	3	Seleccionar rango de fechas
	4	Dar clic en el botón consultar
<b>Postcondición</b>	Consumo entre rango de fechas seleccionado.	
<b>Excepciones</b>	Paso	Acción.
	2	Al seleccionar consumo o histórico
2.1		Seleccionar precio o kWh
<b>Comentarios</b>	Ninguno.	

Fuente: Los autores.

La tabla 40 describe el caso de uso "Jugar" en la aplicación móvil.

Tabla 40. CU029 Juego plataforma.

<b>CU029</b>	<b>Jugar</b>	
<b>Versión</b>	2.0 (22/06/2016)	
<b>Dependencias</b>	SR027	
<b>Precondición</b>	Estar en la aplicación con una sesión iniciada.	
<b>Descripción</b>	Interactuar con los modelos por medio del juego.	
<b>Secuencia Normal</b>	Paso	Acción.
	1	Ir a la sección "Jugar"
	2	Seleccionar rango de fechas.
	3	Seleccionar electrodomésticos.
	4	Seleccionar rango de horas de operación.
	5	Dar clic en el botón "calcular"
<b>Postcondición</b>	Factura actualizada.	
<b>Excepciones</b>	Ninguna.	
<b>Comentarios</b>	La estructura del juego está sujeta al programa de respuesta a la demanda seleccionado en preferencias.	

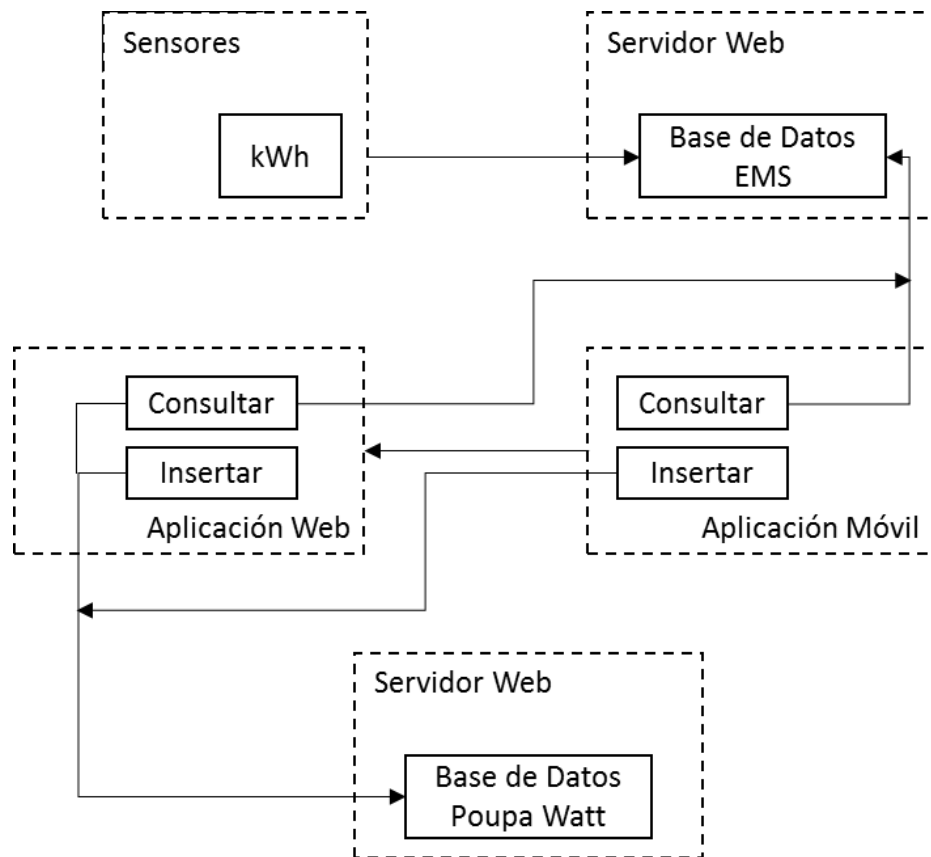
Fuente: Los autores.

### 5.2.3 BASES DE DATOS

El sistema Poupa Watt maneja dos bases de datos diferentes, la primera se llama EMS (Energy Monitoring System) en la cual se gestiona toda la información recolectada por los sensores, la segunda se llama PoupaWatt en la cual se gestiona toda la información de los usuarios y parámetros del sistema.

Para entender cómo se relacionan ambas bases de datos en el sistema, se realizó el diagrama de conexiones de la figura 11 que muestra la interacción de las bases de datos con los sensores, aplicación web y aplicación móvil.

Figura 11. Diagrama de conexiones



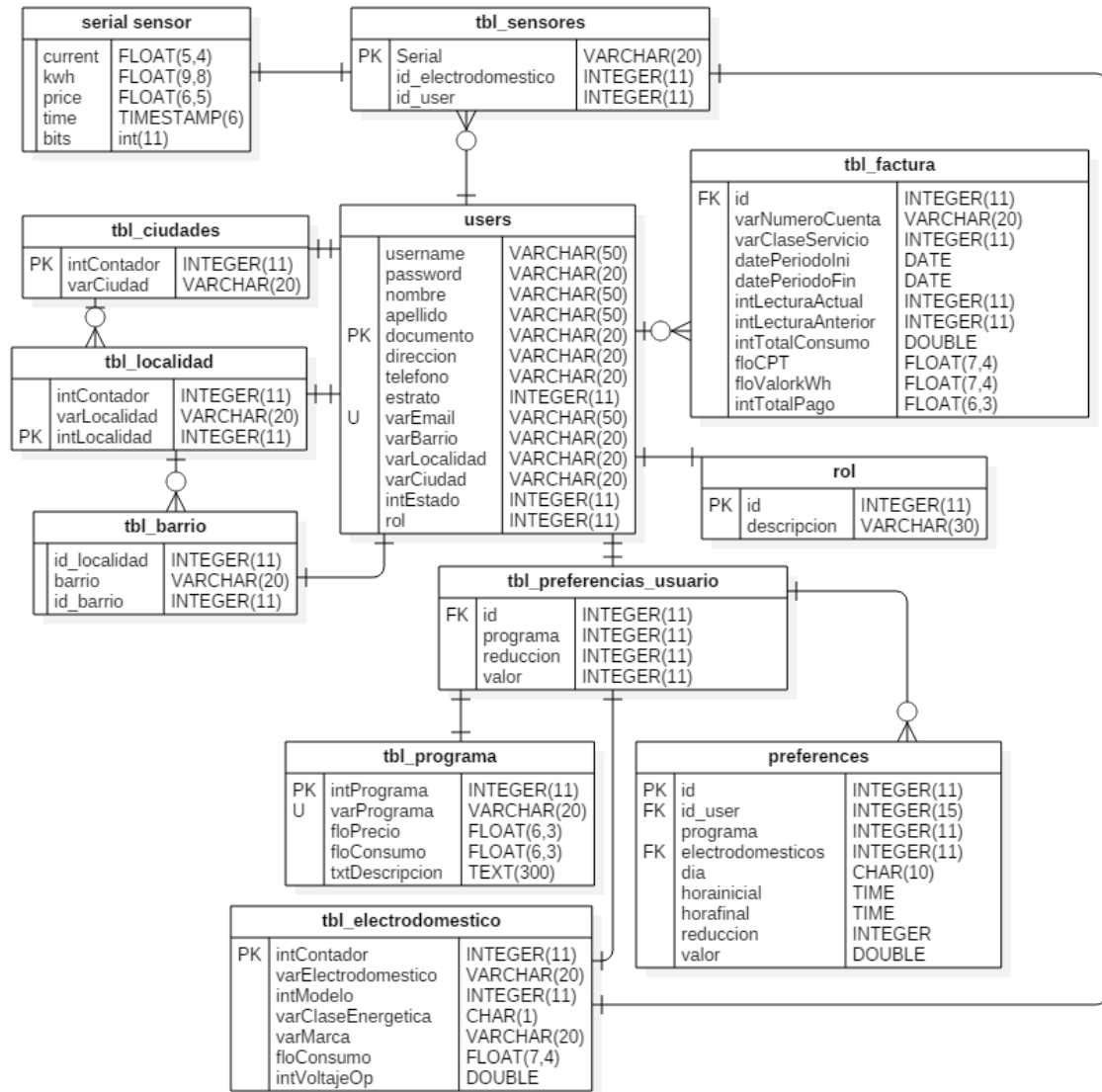
Fuente: Los autores.

Las bases de datos son gestionadas y visualizadas con HeidiSQL, un programa que a través de MySQL permite la gestión y visualización de bases de datos almacenadas en un servidor web.

Para entender el funcionamiento de la base de datos y la relación entre tablas, se realizó el diagrama entidad-relación de la figura 12.



Figura 12. Diagrama entidad-relación



Fuente: Los autores

A continuación, se describe cada base de datos con sus tablas y componentes, dando cumplimiento a los requerimientos establecidos.

### 5.2.3.1 EMS

Esta base de datos contiene toda la información recolectada por los sensores cada 20 segundos (evento) y es almacenada en una tabla estándar. Para identificar la

información de cada sensor se varia el nombre de la tabla en la base de datos, de manera que tiene como nombre el serial del sensor, dicha tabla tiene la estructura de la tabla 41.

Tabla 41.Tabla sensores.

<b>Serial sensor</b>		
<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
<b>current</b>	float	Corriente sensada.
<b>kWh</b>	float	Kilowatt hora sensado.
<b>price</b>	float	Precio del kWh sensado.
<b>time</b>	timestamp	Fecha y hora de sensado.
<b>bits</b>	int	Bits para control.

Fuente: Los autores.

### **5.2.3.2 PoupaWatt**

Esta base de datos contiene toda la información de los usuarios y los parámetros del sistema. A continuación, las tablas y parámetros que la componen.

La tabla 42 contiene la información personal de los usuarios, además de dos campos para control de usuarios. Los datos personales deben ser ingresados por los usuarios en su registro, mientras que los campos para control, intestado y rol son gestionados por parte del administrador.

Tabla 42. Tabla usuarios.

<b>users</b>		
<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
<b>username</b>	varchar	Nombre de usuario.
<b>password</b>	varchar	Contraseña del usuario.
<b>nombre</b>	varchar	Nombres del usuario.
<b>apellido</b>	varchar	Apellidos del usuario.
<b>documento</b>	varchar	Documento de identidad del usuario, llave principal de la tabla
<b>direccion</b>	varchar	Dirección de residencia del usuario.
<b>telefono</b>	varchar	Teléfono del usuario.
<b>estrato</b>	int	Estrato del usuario
<b>varEmail</b>	varchar	Email del usuario

<b>varBarrio</b>	varchar	Barrio residencia del usuario.
<b>varLocalidad</b>	varchar	Localidad residencia del usuario.
<b>varCiudad</b>	varchar	Ciudad de residencia del usuario.
<b>intEstado</b>	int	Usuario habilitado o deshabilitado.
<b>rolrol</b>	int	Rol estándar o administrador.

Fuente: Los autores.

La tabla 43 contiene la información de los roles que se pueden asignar a los usuarios en el sistema, estos fueron definidos y sólo son modificables por el programador del proyecto. Sin embargo, la asignación del rol es hecha por el usuario Administrador.

Tabla 43. Tabla rol.

<b>rol</b>		
<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
<b>id</b>	int	Identificador del rol.
<b>descripcion</b>	varchar	Etiqueta del rol.

Fuente: Los autores.

La tabla 44 contiene las diferentes facturas de energía de la casa de los usuarios, esta información debe ser ingresada por el usuario.

Tabla 44. Tabla facturación.

<b>tbl_factura</b>		
<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
<b>id</b>	int	Documento del usuario.
<b>varNumeroCuenta</b>	varchar	Número de cuenta del usuario.
<b>varClaseServicio</b>	varchar	Clase de servicio del usuario.
<b>datePeriodoIni</b>	date	Fecha periodo inicial de facturación.
<b>datePeriodoFin</b>	date	Fecha periodo final de facturación.
<b>intLecturaActual</b>	int	Lectura actual del medidor.
<b>intLecturaAnterior</b>	int	Lectura anterior del medidor.
<b>intTotalConsumo</b>	int	Consumo total.
<b>floCPT</b>	float	Consumo promedio del trimestre.
<b>floValorkWh</b>	float	Valor kWh
<b>intTotalPago</b>	int	Total a pagar.

Fuente: Los autores.

La tabla 45 contiene la información registrada en las preferencias del usuario por medio de la aplicación móvil de acuerdo a los programas de respuesta a la demanda descritos en el marco teórico.

Tabla 45. Tabla preferencias usuario aplicación móvil.

<b>tbl_preferencias_usuario</b>		
<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
<b>id</b>	int	Identificar único de la preferencia.
<b>programa</b>	int	Documento del usuario.
<b>reduccion</b>	int	Porcentaje de reducción.
<b>valor</b>	double	Valor de la reducción.

Fuente: Los autores.

La tabla 46 contiene la información registrada en las preferencias del usuario por medio de la aplicación web de acuerdo a los programas de respuesta a la demanda descritos en el marco teórico.

Tabla 46. Tabla preferencias usuario aplicación web.

<b>preferencias</b>		
<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
<b>id</b>	int	Identificar único de la preferencia.
<b>id_user</b>	int	Documento del usuario.
<b>programa</b>	int	Identificador programa de respuesta a la demanda
<b>electrodomesticos</b>	int	Identificador electrodoméstico.
<b>dia</b>	char	Día de la semana seleccionado.
<b>horainicial</b>	time	Hora inicio de operación del electrodoméstico.
<b>horafinal</b>	time	Hora finalización de operación del electrodoméstico.
<b>reduccion</b>	int	Reducción kWh semanal
<b>valor</b>	double	Valor reducción kWh semanal

Fuente: Los autores.

La tabla 47 Contiene los programas de respuesta a la demanda seleccionados en el marco teórico y sus parámetros. Estos son ingresados por el administrador.

Tabla 47. Tabla de los programas de respuesta a la demanda.

<b>tbl_programa</b>		
<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
<b>intPrograma</b>	varchar	Identificador programa de respuesta a la demanda.
<b>varPrograma</b>	varchar	Nombres del programa.
<b>floPrecio</b>	varchar	Precio del kWh.
<b>floConsumo</b>	varchar	Consumo promedio del programa.
<b>txtDescripcion</b>	varchar	Descripción del programa de respuesta a la demanda.

Fuente: Los autores.

La tabla 48 Contiene los electrodomésticos que operan en el sistema y sus parámetros. Estos son ingresados por el administrador.

Tabla 48. Tabla electrodomésticos.

<b>tbl_electrodomestico</b>		
<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
<b>intContador</b>	varchar	Identificador electrodoméstico.
<b>varElectrodomestico</b>	varchar	Nombres del electrodoméstico.
<b>intModelo</b>	varchar	Año fabricación.
<b>varClaseEnergetica</b>	varchar	Clase energética de consumo.
<b>varMarca</b>	varchar	Marca.
<b>floConsumo</b>	varchar	Consumo estándar.
<b>intVoltajeOp</b>	int	Voltaje de operación.

Fuente: Los autores.

La tabla 49 Contiene los sensores y sus valores asignados. Estos son ingresados por el administrador.

Tabla 49. Tabla sensores.

<b>tbl_sensores</b>		
<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
<b>Serial</b>	varchar	Serial del sensor.

<b>id_electrodomestico</b>	int	Identificador electrodoméstico asignado.
<b>id_user</b>	int	Documento usuario asignado.

Fuente: Los autores.

La tabla 50 contiene las ciudades disponibles para selección en el registro. Estas son modificables directamente por el programador en la base de datos.

Tabla 50. Tabla ciudades.

<b>tbl_ciudades</b>		
<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
<b>intContador</b>	int	Identificador ciudad
<b>varCiudad</b>	varchar	Nombre ciudad.

Fuente: Los autores.

La tabla 51 Contiene las localidades disponibles para selección en el registro. Estas son modificables directamente por el programador en la base de datos.

Tabla 51. Tabla localidades.

<b>tbl_localidad</b>		
<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
<b>intContador</b>	Int	Identificador de la ciudad.
<b>varLocalidad</b>	varchar	Nombre localidad.
<b>intLocalidad</b>	Int	Identificar de la localidad.

Fuente: Los autores.

La tabla 52 contiene los barrios disponibles para la selección en el registro. Estos son modificables directamente por el programador en la base de datos.

Tabla 52. Tabla barrios.

<b>tbl_barrio</b>		
<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
<b>intLocalidad</b>	int	Identificador localidad.
<b>varBarrio</b>	varchar	Nombre barrio.
<b>intBarrio</b>	int	Identificador barrio.
Fuente: Los Autores		

## **5.2.4 ALGORITMOS**

Una vez definida la estructura del sistema y las plataformas de desarrollo se procede a desarrollar a través de la lógica de programación la interfaz de la aplicación web y la aplicación móvil dando cumplimiento a los requerimientos.

De manera que se muestra el proceso lógico de los algoritmos a través de diagramas de flujo y posteriormente como pseudocódigos para entender qué y cómo se va a programar en el lenguaje de implementación escogido (PHP) para que pueda ser interpretado por el sistema.

Los algoritmos del sistema corresponden a los modelos de los programas de respuesta a la demanda escogidos y el juego.

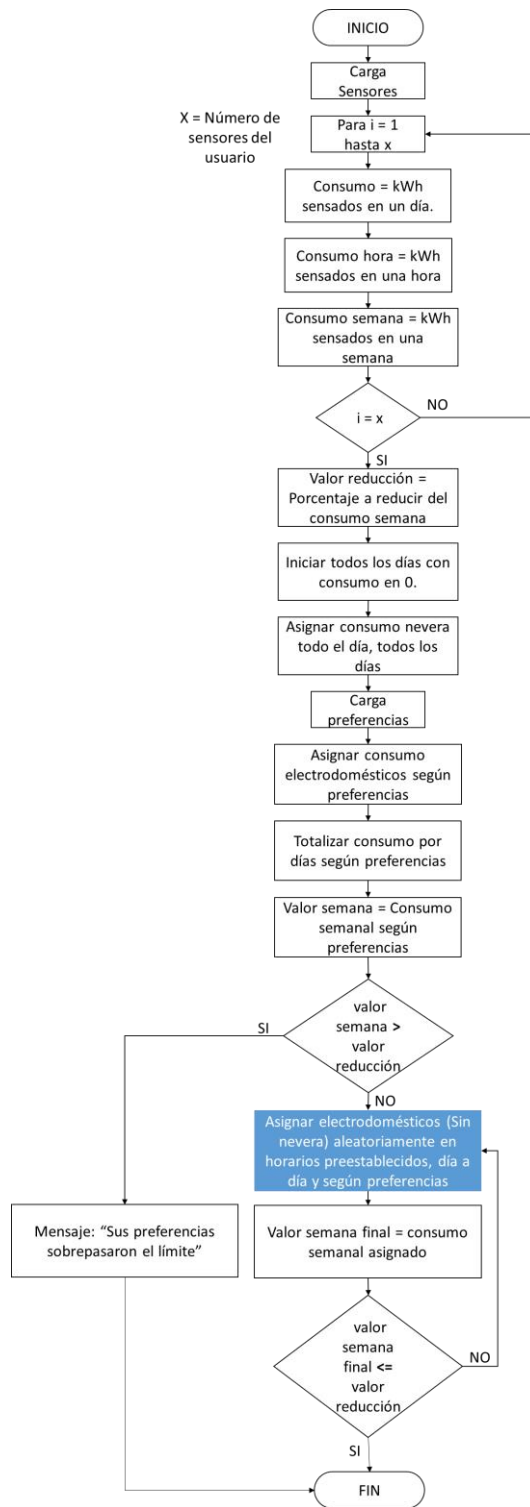
### **5.2.4.1 DIAGRAMAS DE FLUJO**

Para entender el proceso que se lleva a cabo en el sistema, al implementar los algoritmos de los modelos para los programas de respuesta a la demanda, se diseñó un diagrama de flujo donde se ve reflejada la secuencia a seguir, operaciones y decisiones.

Los diagramas son hechos con formas y símbolos de normas ANSI que cuentan con una simbología predeterminada y con significados específicos. (McGraw-Hill Education, 2014)

Se realizaron dos diagramas de flujo, el de la figura 13 para el programa de oferta a la demanda y el de la figura 14 para el programa de tiempo de uso. En ambas figuras hay un recuadro de diferente color, está es la diferencia principal entre los diagramas, los horarios de asignación pues para el caso de la figura 13 estos son en horarios preestablecidos o asignados por el usuario y para el caso de la figura 14 dicha asignación se hace en horarios de menor costo asignados por el administrador.

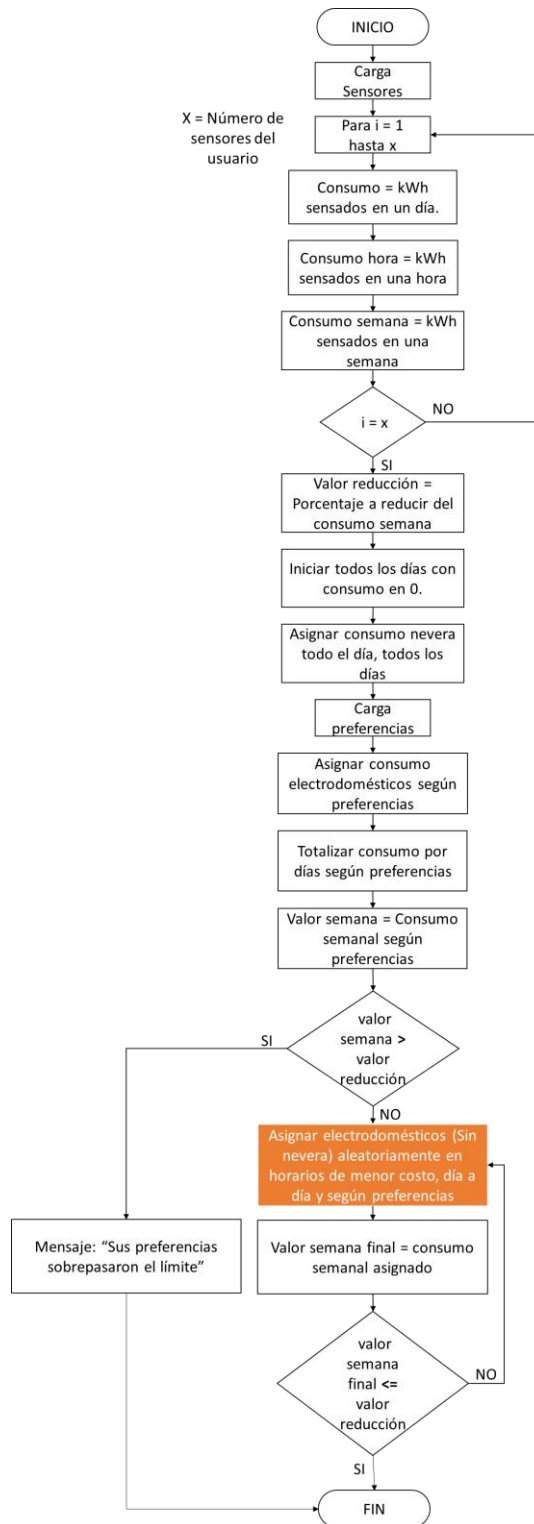
Figura 13. Diagrama de flujo Oferta a la Demanda



Fuente: Los Autores



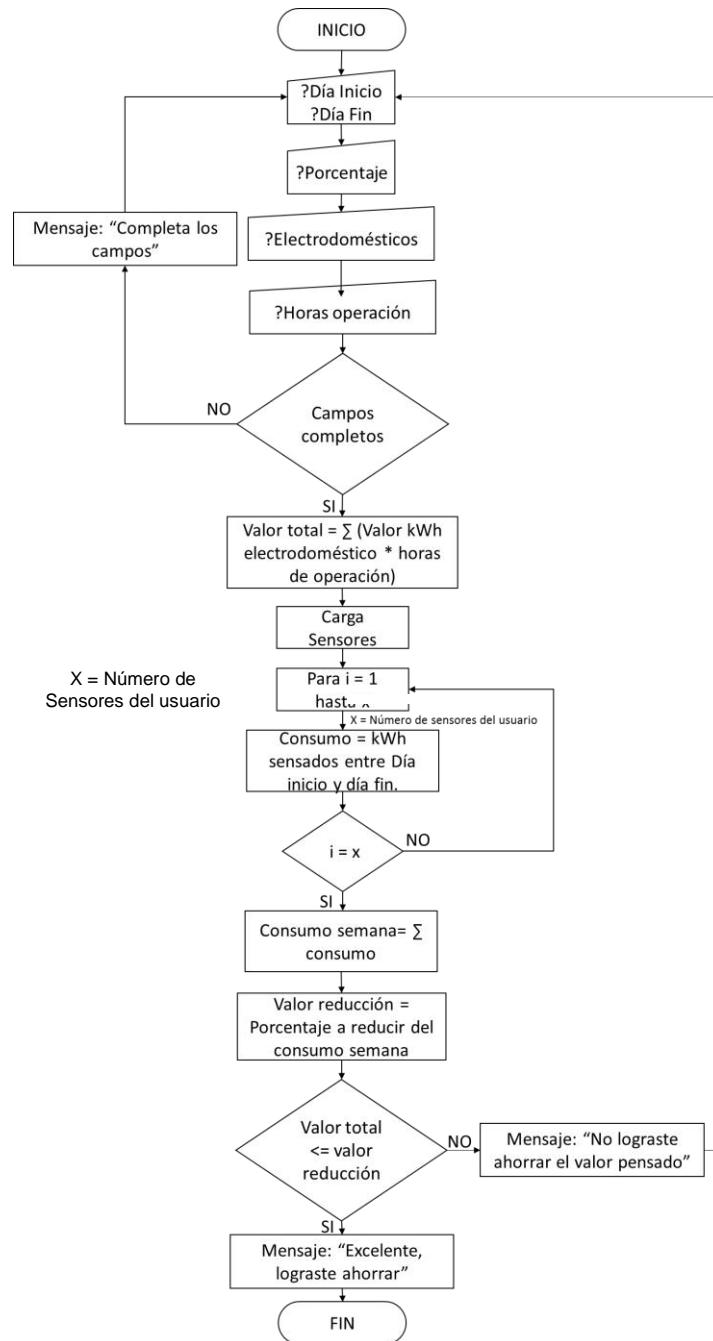
Figura 14. Diagrama de flujo tiempo de uso.



Fuente: Los autores.

También, en la figura 15 se presenta el diagrama de flujo para el algoritmo de funcionamiento del juego.

Figura 15. Diagrama de flujo juego



Fuente: Los autores

### 5.2.4.2 PSEUDOCÓDIGOS

En esta sección, a través de pseudocódigos, se transcriben los diagramas de flujo anteriores, describiendo a mayor detalle los algoritmos implementados que hacen parte del sistema además del juego. Debido a que la sección tiene por objetivo que el lector pueda entender e interpretar el código se excluyen datos sin relevancia para su entendimiento.

El pseudocódigo está dividido en módulos, cada uno de estos cumple con una función dentro de los diferentes algoritmos, a continuación, el primero “extracción datos sensores” que es usado en todos los 3 algoritmos.

#### Extracción datos sensores

*Float consumo\_semana, consumo\_total*

*Int ContadorSensores, DiferenciaDias*

*Varchar id\_user, documento, id*

*Date Fecha\_Inicio, Fecha\_Final*

*Datatable tbl\_sensores, serial*

*Inicio*

*id* ← **documento** del Usuario con sesión iniciada.

**ContadorSensores** ← ConexionPoupaWatt(“Seleccionar contar(serial) desde tbl\_sensores, donde **id\_user=id** ”)

Si **ContadorSensores** = 0

**msgError** ← (“No cuenta con sensores asignados”)

Escribir (**msgError**)

Si no

Leer **Fecha\_inicio, Fecha\_Final**

**DiferenciaDias** ← **Fecha\_Final** – **Fecha\_Inicial**

Si **DiferenciaDias** = 7

Para  $i=1; i \leftarrow \mathbf{ContadorSensores}; i++$

**consumo\_semana** ← ConexionEMS(“Seleccionar suma(kWh) desde serial donde time esté entre **Fecha\_Inicial** y **Fecha\_Final** ”)

**consumo\_total** ← **consumo\_total** + **consumo\_semana**

Fin para

```

        Si no
            msgError ← (“Los días sensados no son suficientes”)
            Escribir (msgError)
        Fin si
    Fin si
Fin

```

El siguiente módulo es el encargado de cargar y asignar las preferencias dentro de los algoritmos de respuesta a la demanda energética.

<b>Carga y asignar preferencias</b>
-------------------------------------

```

Float   kWh,   consumo_total,   valor_a_reducir,   consumo_nevera,
consumo_hora_nevera,
Int reducción, id_electrodomestico, ContadorSensores, días_semana
Array dia, diapref
Date Fecha_Inicio, Fecha_Final, hora_inicio, hora_final
Datatable tbl_sensores, serial(serial_nevera), preferences

```

Inicio

```

reduccion ← ConexionPoupaWatt(“Seleccionar reducción desde preferences
donde id_user = id”)

```

```

valor_a_reducir ← consumo_total-reduccion

```

```

serial_nevera ← ConexionPoupaWatt(“Seleccionar serial desde tbl_sensores
donde id_electrodomestico=1”)

```

```

consumo_nevera ← ConexionEMS(“Seleccionar suma(kWh) desde
serial_nevera donde time esté entre Fecha_Inicial y Fecha_Final”)

```

```

consumo_hora_nevera ← ((consumo_nevera / 7) / 24)

```

```

días_semana ← 7

```

```

dia ← vector 24 posiciones

```

```

para i=1; i ← días_semana; i++

```

```

    dia ← [0 a 23] = consumo_hora_nevera + diapref;

```

```

para b=1; b ← ContadorSensores; b++

```

```

    preferencia ← ConexionPoupaWatt(“Seleccionar hora_inicio
y hora_final desde preferences donde id_user=id y
id_electrodomestico=b”)

```

```

        consumo_preferencia ← ConexionEMS("Seleccionar
        suma(kWh) desde serial donde time=preferencia")
        diapref ← [hora inicio] a [hora final]=(consumo_preferencia)
    Fin para
Fin para
Fin

```

Para el funcionamiento del programa oferta a la demanda se hace uso de los módulos "extracción datos sensores" y "cargar y asignar preferencias" junto con el siguiente módulo que es el encargado de hacer los cálculos correspondientes al programa de respuesta a la demanda.

<b>Cálculos Oferta a la demanda</b>
-------------------------------------

```

Float valor_semana, valor_semana_final, valor_a_reducir, consumo,
consumo_semana
Int dias_semana
Varchar id_user, documento, id, serial_nevera
Array dia

Inicio
    para i=1; i ≤ dias_semana; i++
        valor_semana = valor_semana+dia

        si valor_semana < valor_a_reducir
            hacer dia ← dias_semana ←, [random, excepto
            preferencia]= consumo ← ((consumo_semana / 7) / 24)
            donde consumo ← id_electrodomestico(random)
            valor_semana_final=valor_semana_final+dia
        repetir mientras valor_semana_final ≤ valor_a_reducir
            si no msg ← ("Reducción exitosa!")
                Escribir (msg)
        Fin si
    Fin para
Final

```

Adicionalmente para los cálculos de oferta a la demanda se presenta un complemento, donde los valores random para operación de los electrodomésticos están definidos por rangos en los usuarios están dispuestos a desplazar las cargas, estos son ingresados por el usuario.

### Complemento cálculos Oferta a la demanda

Float *valor\_semana*, *valor\_semana\_final*, *valor\_a\_reducir*, *consumo*,  
*consumo\_semana*  
Int *días\_semana*, *ingresar\_horas*  
Varchar *id\_user*, *documento*, *id*, *serial\_nevera*  
Time *horas\_de\_operacion*  
Array *dia*

Inicio

Si *ingresar\_horas* = 1

Leer *horas\_de\_operacion*

para  $i=1; i \leftarrow \text{días\_semana}; i++$

$\text{valor\_semana} = \text{valor\_semana} + \text{dia}$

si  $\text{valor\_semana} < \text{valor\_a\_reducir}$

hacer  $\text{días\_semana} \leftarrow, [\text{horas\_de\_operacion}] =$

$\text{consumo} \leftarrow ((\text{consumo\_semana} / 7) / 24)$

donde  $\text{consumo} \leftarrow \text{id\_electrodomestico}(\text{random})$

$\text{valor\_semana\_final} = \text{valor\_semana\_final} + \text{dia}$

repetir mientras  $\text{valor\_semana\_final} \leq \text{valor\_a\_reducir}$

si no  $\text{msg} \leftarrow$  (“Reducción exitosa!”)

Escribir (*msg*)

Fin si

Fin para

Si no Ejecutar **Cálculos Oferta a la demanda**

Fin si

Final

Para el funcionamiento del programa tiempo de uso se hace uso de los módulos “extracción datos sensores” y “cargar y asignar preferencias” junto con el siguiente módulo es el encargado de hacer los cálculos correspondientes al programa de respuesta a la demanda.

### Cálculos Tiempo de uso

Float *valor\_semana*, *valor\_semana\_final*, *valor\_a\_reducir*, *consumo*,  
*consumo\_semana*  
Int *días\_semana*  
Varchar *id\_user*, *documento*, *id*, *serial\_nevera*  
Array *dia*

*Inicio*

```
para i=1; i<= dias_semana; i++  
    valor_semana = valor_semana+dia  
  
    si valor_semana < valor_a_reducir  
        hacer dias_semana ←, [random, 6 a 10]= consumo←  
        ((consumo_semana / 7) / 24)  
        donde consumo← id_electrodomestico(random)  
        valor_semana_final=valor_semana_final+dia  
    repetir mientras valor_semana_final <= valor_a_reducir  
        si no msg ← (“¡(“Reducción exitosa!”)  
            Escribir (msg)  
        Fin si  
    Fin para  
Final
```

Para el funcionamiento del juego se hace uso del módulo “extracción datos sensores” y junto con el siguiente módulo que es el encargado de hacer los cálculos correspondientes al programa de respuesta a la demanda.

### Juego

```
Float valor_semana, valor_semana_final, valor_a_reducir, consumo,  
consumo_semana  
Int dias_semana  
Varchar id_user, documento, id, serial_nevera  
Array dia
```

*Inicio*

```
Leer dia_inicio, dia_fin, porcentaje, electrodomestico1,  
electrodomestico2, electrodomestico3, electrodomestico4,  
electrodomestico5, horasoperacionelec1, horasoperacionelec2,  
horasoperacionelec3, horasoperacionelec4,  
horasoperacionelec5  
  
dia_semana=7  
para i=1; i<=dia_semana; i++  
    valor_total = valor total + (consumo_total*horasoperacion)  
  
para i=horasoperacionelec1; i<=horasoperacionelec5; i++  
    horasoperacion = horasoperacionelec
```

```

        Fin para
    Fin para

    Si valor_total <= consumo_total
        msg ← (“¡“Excelente, lograste ahorrar!”)
        Escribir (msg)
    si no msg ← (“No lograste ahorrar el valor pensado”)
        Escribir (msg)
    Fin si
Fin

```

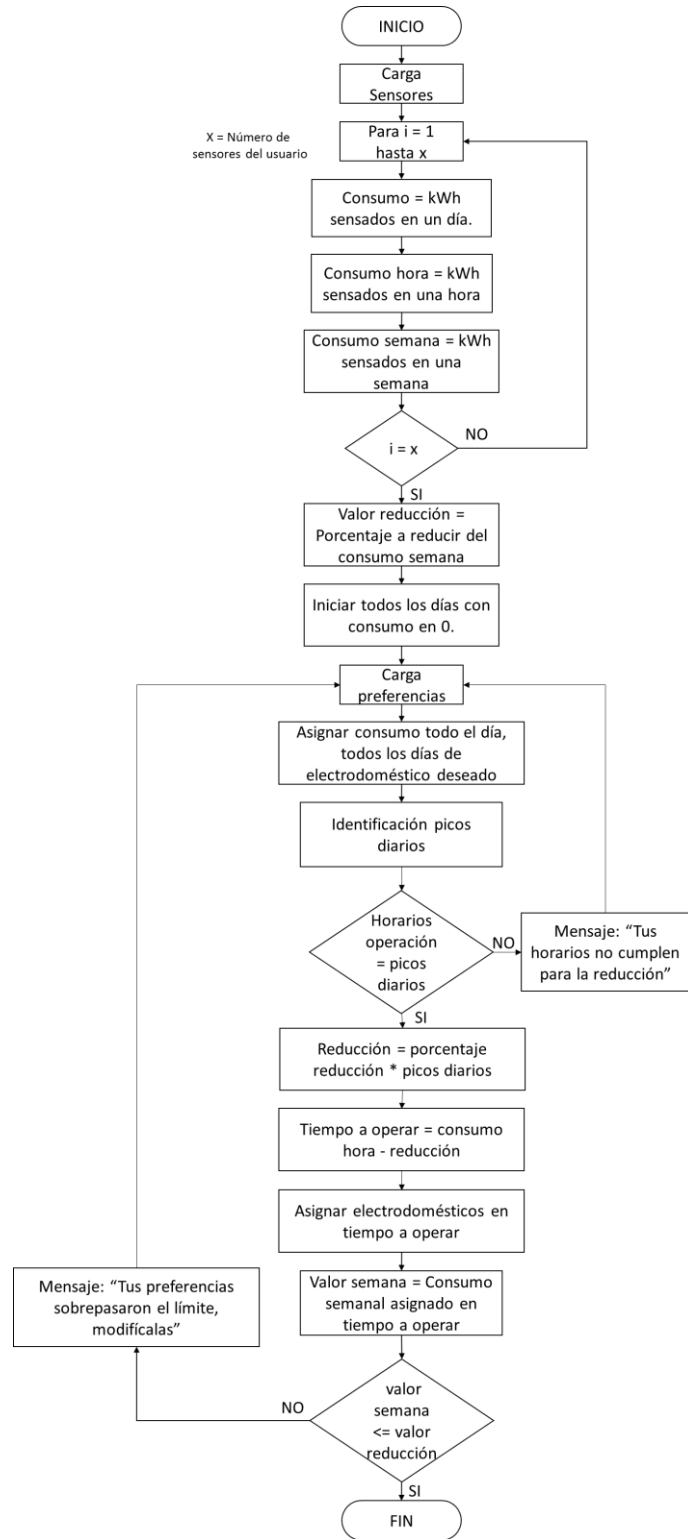
### 5.2.4.3 TRABAJOS FUTUROS

Los anteriores diagramas de flujo y pseudocódigos corresponden a los algoritmos implementados en la plataforma. Adicionalmente se plantea un nuevo diagrama de flujo y su correspondiente pseudocódigo como variante para Oferta a la demanda para implementación en trabajos futuros dentro de la plataforma.

Para este las preferencias serán horarios de cuando el usuario está dispuesto a mover sus cargas, a continuación, en la figura 16 el diagrama de flujo para esta variante.



Figura 16. Diagrama de flujo variante Oferta a la Demanda



Fuente: Los autores

Para la implementación del pseudocódigo de esta variable, se necesita de un módulo previo que identifica los picos para su posterior uso.

### Identificación Picos

*Float kWh*

*Double promedio\_curva, rango\_curva, picos\_curva*

*Varchar id\_user, documento, id, serial\_nevera*

*Inicio*

*promedio\_curva* ← *ConexionEMS*("Seleccionar avg(kWh) desde serial  
donde time BETWEEN interval 7 Day AND Now")

*rango\_curva* ← *ConexionEMS*("Seleccionar StDev(kWh) desde serial  
donde time BETWEEN interval 7 Day AND Now")

*picos\_curva* ← *ConexionEMS*("Seleccionar Max(kWh) desde serial  
donde time BETWEEN interval 7 Day AND Now")

*Si picos\_curva > rango\_curva*

*serial*                    *picos\_diarios* ← *ConexionEMS*("Seleccionar Suma(time) desde  
donde kWh => rango\_curva")

*si no msg* ← ("Error")

*Escribir (msg)*

*Fin si*

*Fin*

Para la implementación del pseudocódigo de esta variable, se hace uso de los módulos "extracción datos sensores", "cargar y asignar preferencias" e "identificación picos".

### Variación Oferta a la Demanda

*Float consumo\_hora*

*Int reduccion, porcentaje\_reduccion*

*Double picos\_diarios, valor\_semana*

*Varchar id\_user, documento, id, serial\_nevera*

*Array dia*

*Inicio*

*para i=1; i<=picos\_diarios; i++*

**Si Preferencia <= picos\_diarios**  
**reduccion=porcentaje\_reduccion\*picos\_diarios**  
**tiempo\_operacion← consumo\_hora – reduccion**

**para i=1; i< dias\_semana; i++**  
**valor\_semana = valor\_semana+dia**  
**hacer dia= [tiempo\_operacion] ← consumo**  
**Fin para**

**Si valor\_semana <= reduccion**  
**msg ← (“Tus preferencias sobrepasaron el límite”)**  
**Escribir (msg)**

**Si no**  
**Resultado← dia**  
**Escribir (Resultado)**

**Fin para**

Diseñado el sistema, se da paso al desarrollo de este para posteriormente implementarlo y verlo en funcionamiento, por esto que el capítulo siguiente es el encargado del desarrollo e implementación donde se dan a conocer los software utilizados, mapas de navegación y demás requerimientos.

## **6. DESARROLLO E IMPLEMENTACIÓN**

El propósito de Desarrollo de Software es la realización sistemática de las actividades de planeación, diseño, codificación, pruebas, lanzamiento de productos de software nuevos cumpliendo con los requisitos especificados y con las normativas de seguridad de información. (GUERRERO, ERAZO, MIRANDA, & ANTE, 2011)

Una vez definido el diseño del sistema, se procede al desarrollo del mismo, donde se tiene que dar cumplimiento a los diferentes casos de uso además de la implementación de los algoritmos para los programas de respuesta a la demanda.

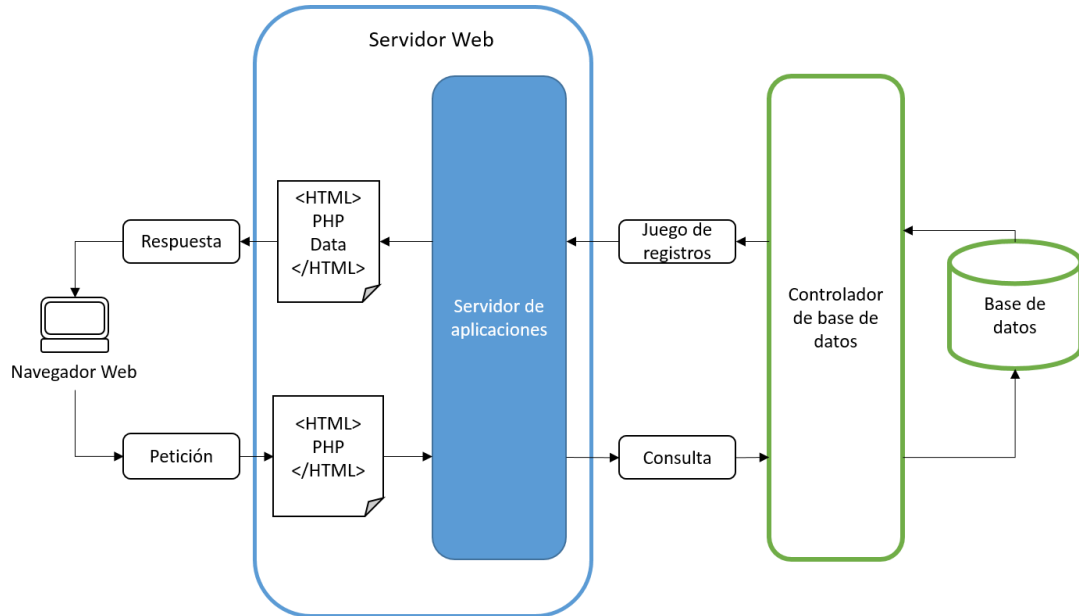
El desarrollo del sistema se dividió en dos partes, la primera es el desarrollo de la aplicación web donde el sistema tiene completo manejo y por último la aplicación móvil enfocada al usuario estándar.

### **6.1 DESARROLLO E IMPLEMENTACIÓN APLICACIÓN WEB**

El sistema se desarrolló como una aplicación web para que cualquier persona pueda acceder a ella, pues a través de la web se pueden brindar todos los servicios necesarios sin necesidad de tener instalaciones y/o configuraciones como en el caso de las aplicaciones de escritorio.

En la figura 17 se observa la estructura básica de una aplicación web y a partir de esta se eligieron las plataformas de desarrollo para que cada etapa fuera funcional y compatible con la siguiente.

Figura 17. Aplicación web



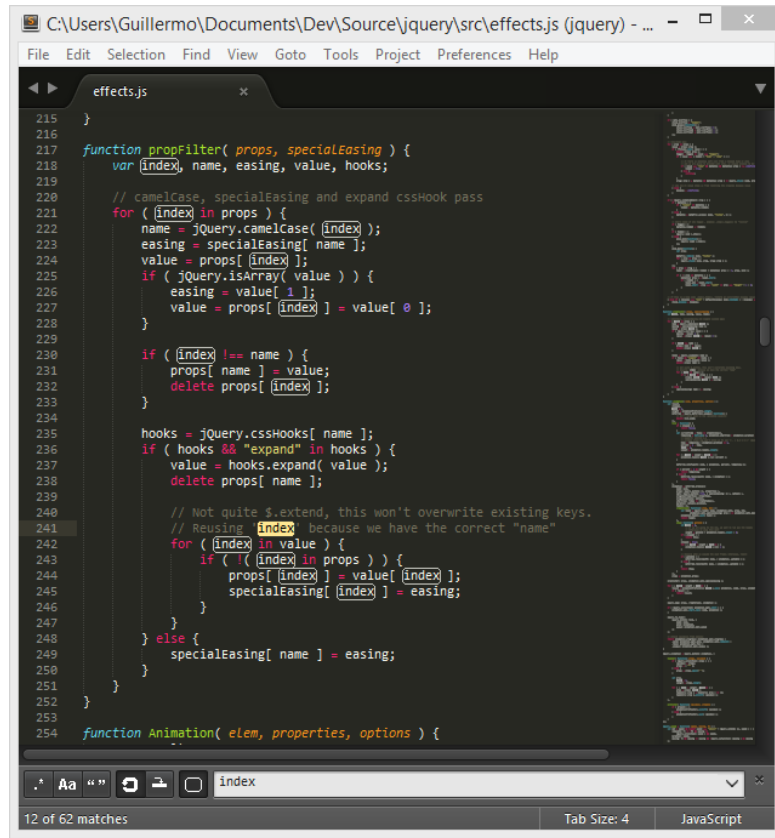
Tomado de (Adobe Systems Incorporated and its licensors, 2015)

### 6.1.1 Plataformas de desarrollo

El servidor web es el encargado de almacenar los archivos que permiten la visualización de la aplicación, comunicación con la base de datos y dentro del servidor de aplicaciones se almacena los archivos que permiten que sea dinámica.

Para la creación de los diferentes archivos de funcionamiento y permitir las funciones ya mencionadas en el servidor web, se utilizó como programa de desarrollo Sublime Text, un editor de código multiplataforma que soporta un gran número de diferentes lenguajes (C, C++, C#, CSS, D, Erlang, HTML, Groovy, Haskell, HTML, Java, JavaScript, LaTeX, Lisp, Lua, Markdown, Matlab, OCaml, Perl, PHP, Python, R, Ruby, SQL, TCL, Textile y XML) su entorno de programación se observa en la figura 18 y este es propicio para la programación.

Figura 18. Entorno de desarrollo Sublime Text.



```
215 }
216
217 function propFilter( props, specialEasing ) {
218   var index, name, easing, value, hooks;
219
220   // camelCase, specialEasing and expand cssHook pass
221   for ( index in props ) {
222     name = jQuery.camelCase( index );
223     easing = specialEasing[ name ];
224     value = props[ index ];
225     if ( jQuery.isArray( value ) ) {
226       easing = value[ 1 ];
227       value = props[ index ] = value[ 0 ];
228     }
229
230     if ( index !== name ) {
231       props[ name ] = value;
232       delete props[ index ];
233     }
234
235     hooks = jQuery.cssHooks[ name ];
236     if ( hooks && "expand" in hooks ) {
237       value = hooks.expand( value );
238       delete props[ name ];
239     }
240
241     // Not quite $.extend, this won't overwrite existing keys.
242     // Reusing 'index' because we have the correct "name"
243     for ( index in value ) {
244       if ( ! ( index in props ) ) {
245         props[ index ] = value[ index ];
246         specialEasing[ index ] = easing;
247       }
248     }
249     else {
250       specialEasing[ name ] = easing;
251     }
252   }
253 }
254 function Animation( elem, properties, options ) {
```

Fuente: (Sublime HQ Pty Ltd, 2016)

Para cumplir con las diferentes funciones que debe llevar a cabo la aplicación web se utilizaron diferentes lenguajes de programación, cada uno con una funcionalidad y objetivo diferente dentro de la aplicación.

El primero, es el encargado de la estructura en la parte visual, este es HTML5 un lenguaje que se emplea para el desarrollo de páginas de internet. Está compuesto por una serie de etiquetas que el navegador interpreta y da forma en la pantalla. HTML dispone de etiquetas para imágenes, hipervínculos que nos permiten dirigirnos a otras páginas, saltos de línea, listas, tablas, etc. (González Gutiérrez, 2006). Junto con HTML se tiene también el CSS que es el encargado de la presentación de la página ya que es utilizado por parte de los navegadores web de

internet y por los programadores web informáticos para elegir multitud de opciones de presentación como colores, tipos y tamaños de letra, etc. (Sierra, 2006)

Para realizar las diferentes operaciones que se diseñaron para el sistema, es necesario utilizar un lenguaje que se pueda incrustar en la estructura ya diseñada, para esto se eligió el PHP que es el acrónimo de Hipertext Preprocesor. Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación. (Alvarez, Desarrollo Web, 2001)

Un lenguaje del lado del servidor es aquel que se ejecuta en el servidor web, justo antes de que se envíe la página a través de Internet al cliente. Las páginas que se ejecutan en el servidor pueden realizar accesos a bases de datos, conexiones en red, y otras tareas para crear la página final que verá el cliente. (Alvarez, Desarrollo Web, 2001)

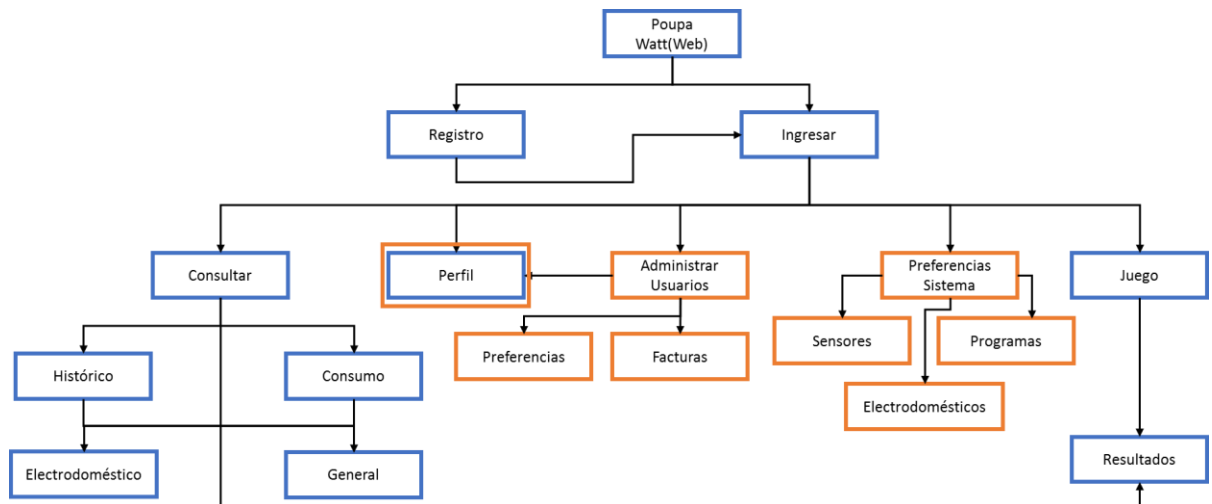
Una vez estructurada la aplicación, con la apariencia deseada y funcional, a través de PHP se hace conexión e interacción con la base de datos para llevar a cabo las diferentes operaciones de los usuarios. Para que las bases de datos se puedan conectar y funcionar de manera adecuada se utilizó el lenguaje SQL que permite las diferentes operaciones dentro de las bases, este es el lenguaje estándar ANSI/ISO de definición, manipulación y control de bases de datos relacionales. Es un lenguaje declarativo: sólo hay que indicar qué se quiere hacer por medio de sentencias. (Casillas Santillán, Ginestà, & Pérez Mora, 2007)

Como el sistema cuenta con más de una base de datos, tablas y campos, es necesario un gestor que permita el manejo y visualización de las mismas. Este es MySQL un sistema gestor de bases de datos (SGBD, DBMS por sus siglas en inglés) muy conocido y ampliamente usado por su simplicidad y notable rendimiento, disponible para múltiples plataformas. (Casillas Santillán, Ginestà, & Pérez Mora, 2007).

## 6.1.2 Aplicación web

La aplicación web se diseñó para que su navegación fuera intuitiva de manera que usuarios con o sin relación a este tipo de plataformas puedan navegar y hacer uso de la misma sin necesidad de un manual de usuario o de ayuda externa, de manera que se desarrolló para que así lo fuera. En la figura 19 está el mapa de navegación de la aplicación web, del cual se partió para el desarrollo de la interfaz gráfica y algoritmos de funcionamiento.

Figura 19. Mapa de navegación aplicación web.



Fuente: Los autores

## 6.1.3 Funcionamiento aplicación web

En el desarrollo de esta sección se encuentran las diferentes pantallas para los usuarios del sistema y una descripción básica de las mismas para conocer su interacción.

### 6.1.3.1 Usuario estándar

En la tabla 53 se encuentra la pantalla de login, esta es lo primero que ve cualquier usuario que ingrese a poupawatt.info.



Tabla 53. Pantalla Login aplicación web.

Pantalla
 <p>¡BIENVENIDO A POUPA WATT!</p> <p>email</p> <p>password</p> <p>Ingresar</p> <p>No estas registrado? <a href="#">Crear una cuenta</a></p> <p><a href="#">Acerca de Poupawatt</a></p>
Descripción
<p>Pantalla Login: En esta pantalla se ingresa a la aplicación web llenado los campos de “Email” y “Password”.</p>

Fuente: Los autores.

En la tabla 54 luego de hacer login se encuentra la pantalla home del usuario estándar.

Tabla 54. Pantalla Home aplicación web.

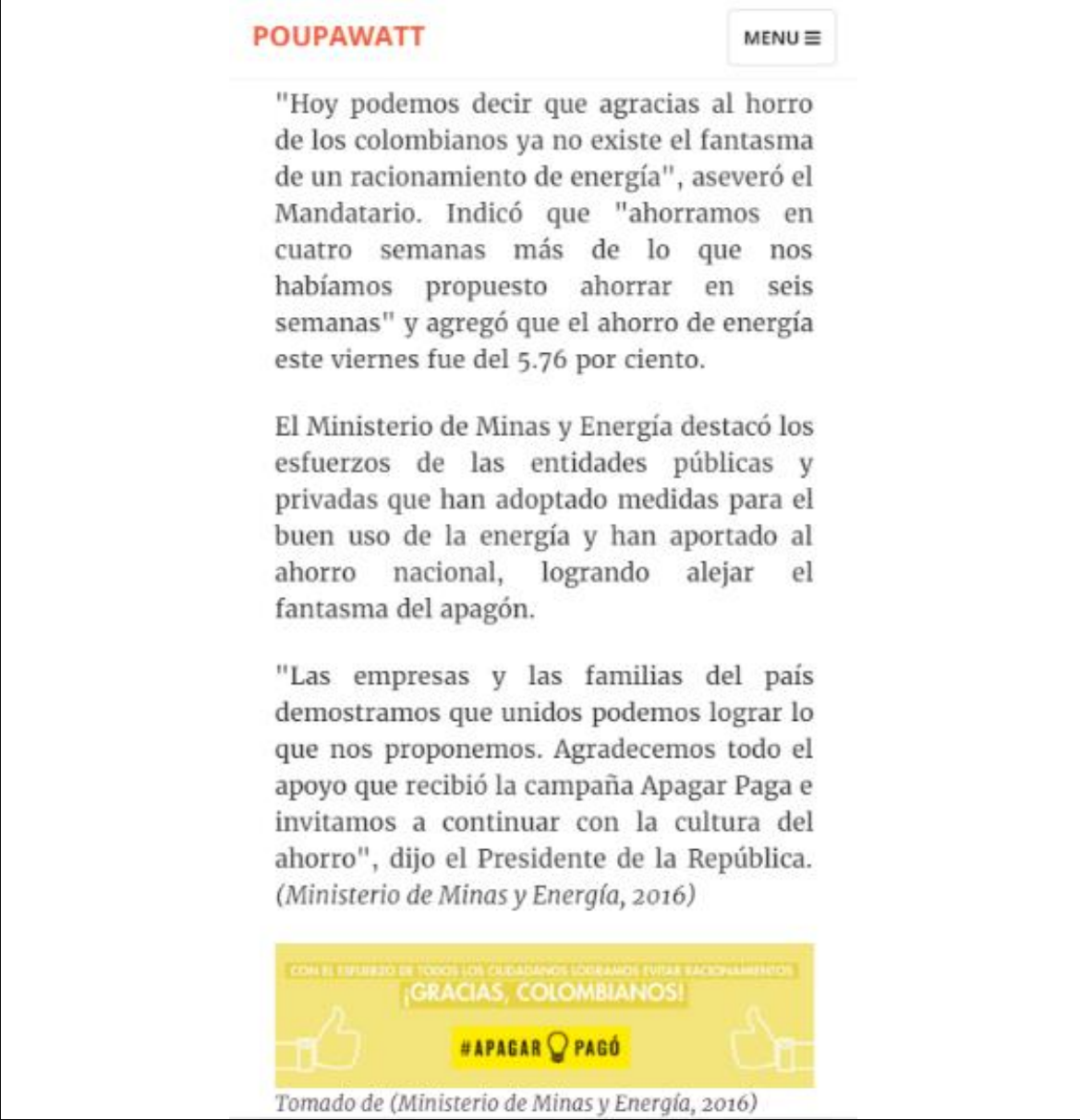
Pantalla

Descripción
<p>Pantalla Home: Es la pantalla que se le muestra al usuario una vez entra al sitio, permitiéndole navegar por las diferentes secciones de la aplicación web.</p>

Fuente: los autores.

La tabla 55 contiene la pantalla noticias, información relevante para el usuario en su tarea de ahorrar energía.

Tabla 55. Pantalla noticias aplicación web.

Pantalla
 <p><b>POUPAWATT</b> MENU ≡</p> <p>"Hoy podemos decir que gracias al horro de los colombianos ya no existe el fantasma de un racionamiento de energía", aseveró el Mandatario. Indicó que "ahorramos en cuatro semanas más de lo que nos habíamos propuesto ahorrar en seis semanas" y agregó que el ahorro de energía este viernes fue del 5.76 por ciento.</p> <p>El Ministerio de Minas y Energía destacó los esfuerzos de las entidades públicas y privadas que han adoptado medidas para el buen uso de la energía y han aportado al ahorro nacional, logrando alejar el fantasma del apagón.</p> <p>"Las empresas y las familias del país demostramos que unidos podemos lograr lo que nos proponemos. Agradecemos todo el apoyo que recibió la campaña Apagar Paga e invitamos a continuar con la cultura del ahorro", dijo el Presidente de la República. <i>(Ministerio de Minas y Energía, 2016)</i></p> <p>CON EL ESFUERZO DE TODOS LOS CIUDADANOS LOGRAMOS EVITAR RACIONAMIENTOS ¡GRACIAS, COLOMBIANOS! #APAGAR PAGÓ</p> <p><i>Tomado de (Ministerio de Minas y Energía, 2016)</i></p>
Descripción
<p>Pantalla Noticias: Esta presenta información relevante, referente al enfoque del proyecto.</p>

Fuente: Los autores.

La tabla 56 contiene la pantalla del perfil del usuario, donde se encuentra el manejo de información personal.

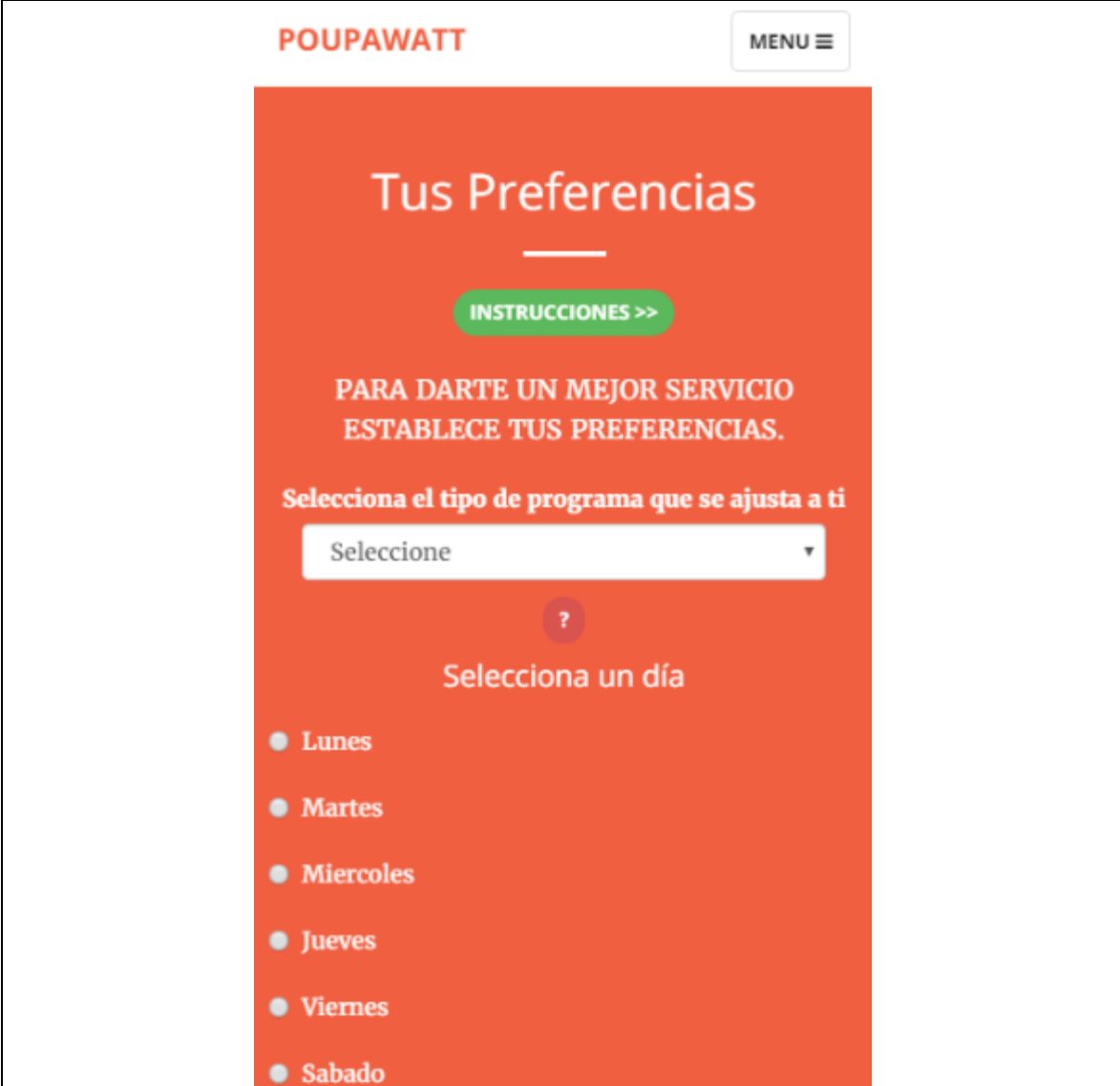
Tabla 56. Pantalla perfil aplicación web.

Pantalla
Descripción
Pantalla Perfil. Se observa y modifica los datos personales registrados.

Fuente: Los autores.

La tabla 57 contiene la pantalla de preferencias, fundamentales para el uso pleno de la aplicación web.


Tabla 57. Pantalla Preferencias aplicación web.

Pantalla
 <p>POUPAWATT MENU</p> <h2>Tus Preferencias</h2> <p>INSTRUCCIONES &gt;&gt;</p> <p>PARA DARTE UN MEJOR SERVICIO ESTABLECE TUS PREFERENCIAS.</p> <p>Selecciona el tipo de programa que se ajusta a ti</p> <p>Selecciona</p> <p>?</p> <p>Selecciona un día</p> <ul style="list-style-type: none"><li>Lunes</li><li>Martes</li><li>Miercoles</li><li>Jueves</li><li>Viernes</li><li>Sabado</li></ul>
Descripción
Pantalla Preferencias: presenta toda la información relevante para la ejecución de los algoritmos de los modelos.

Fuente: Los autores.

La tabla 58 contiene las diferentes facturas del usuario y la información de las mismas.

Tabla 58. Pantalla facturas aplicación web.

Pantalla

Descripción
Pantalla tus facturas: Presenta toda la información importante para llevar el control de las facturas de los usuarios.

Fuente: Los autores.

La tabla 59 contiene la pantalla de consumo, esta permite observar el comportamiento del usuario.

Tabla 59. Pantalla Consumo aplicación web.

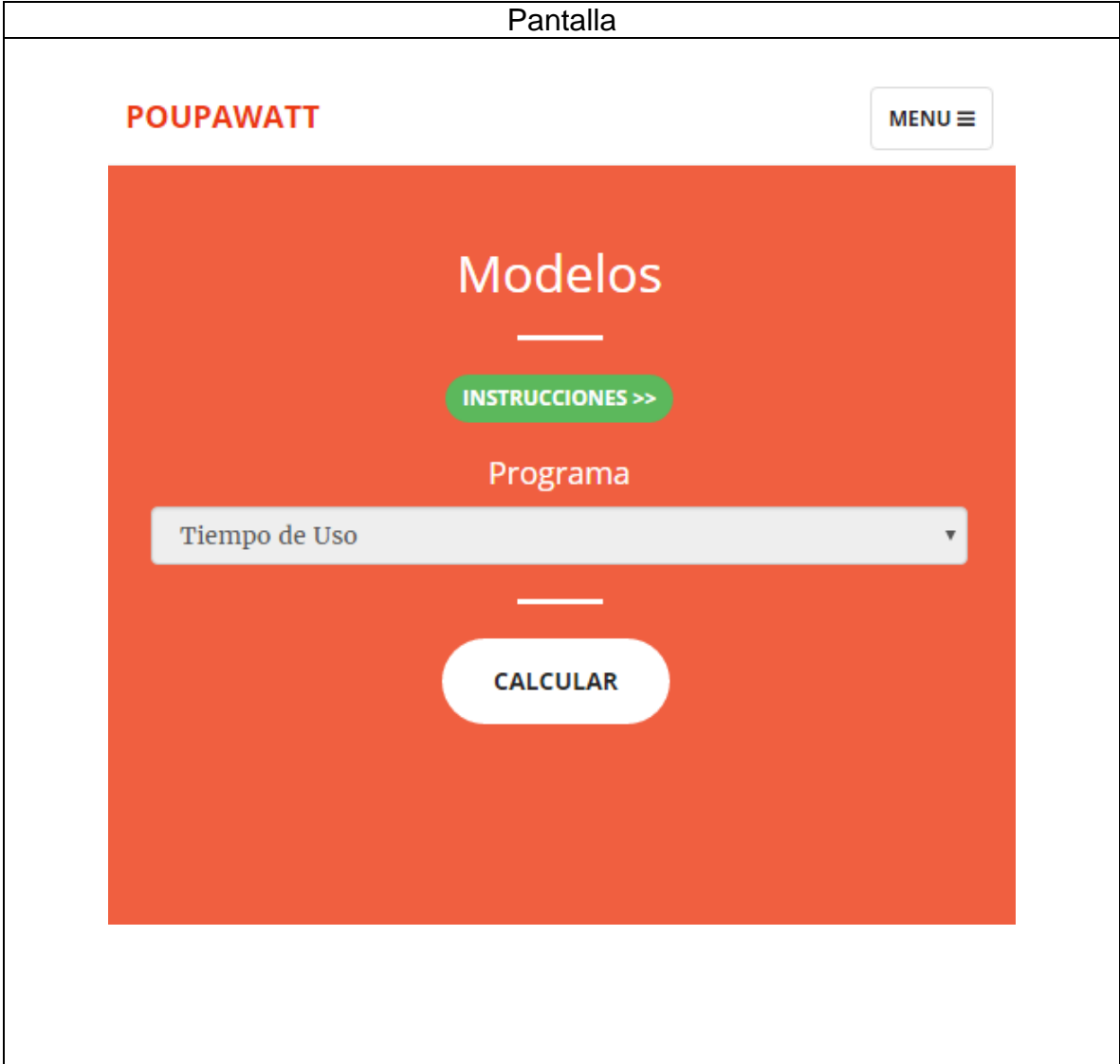
Pantalla

Descripción
Pantalla revisa tu consumo: se puede observar el comportamiento del consumo y totalizarlo en el rango de fechas seleccionado.

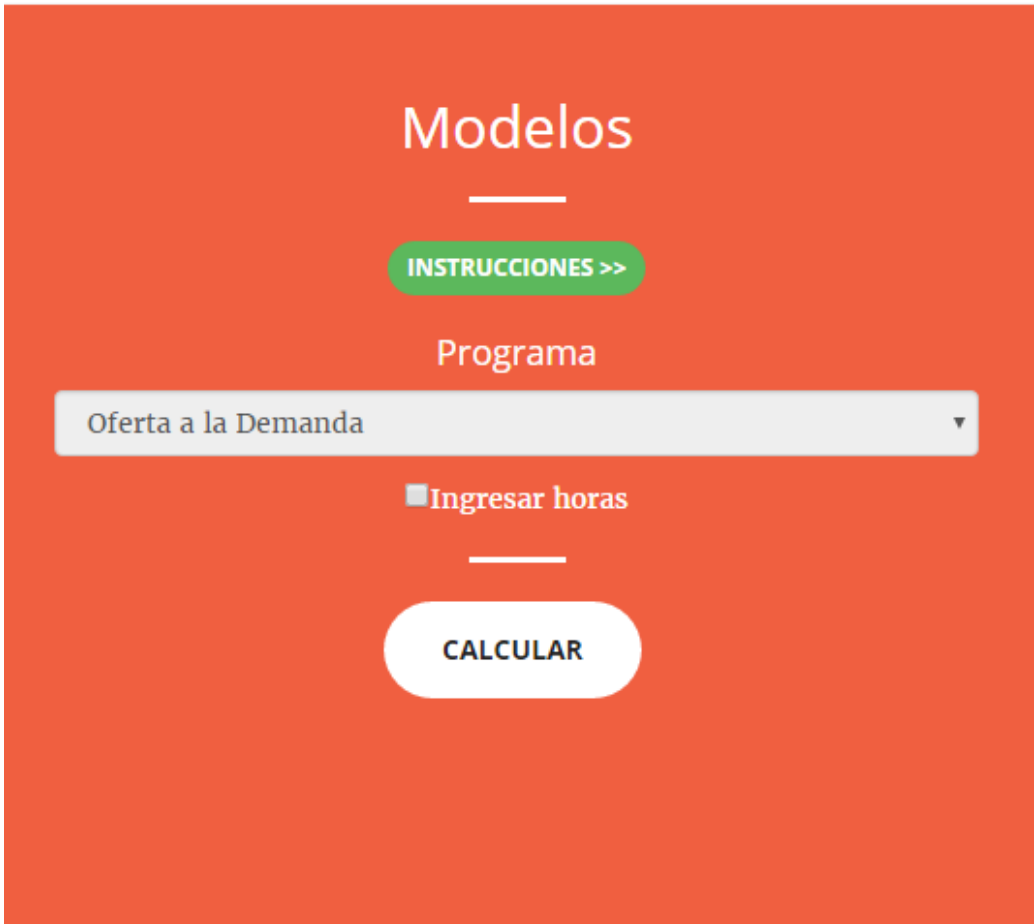
Fuente: Los autores.

La tabla 60 muestra la pantalla de ejecución de los algoritmos de respuesta a la demanda.

Tabla 60. Pantallas Modelos aplicación web.







<p style="text-align: center;">Descripción</p>
<p>Pantalla Modelos: se observa la ejecución del algoritmo del modelo del programa de respuesta de la demanda seleccionado por el usuario anteriormente en preferencia y la posible selección de las horas de ejecución para lograr la reducción en el consumo.</p>

Fuente: Los autores.

La tabla 61 contiene la pantalla del juego para el usuario y su consumo.

Tabla 61. Pantalla Juego aplicación web.

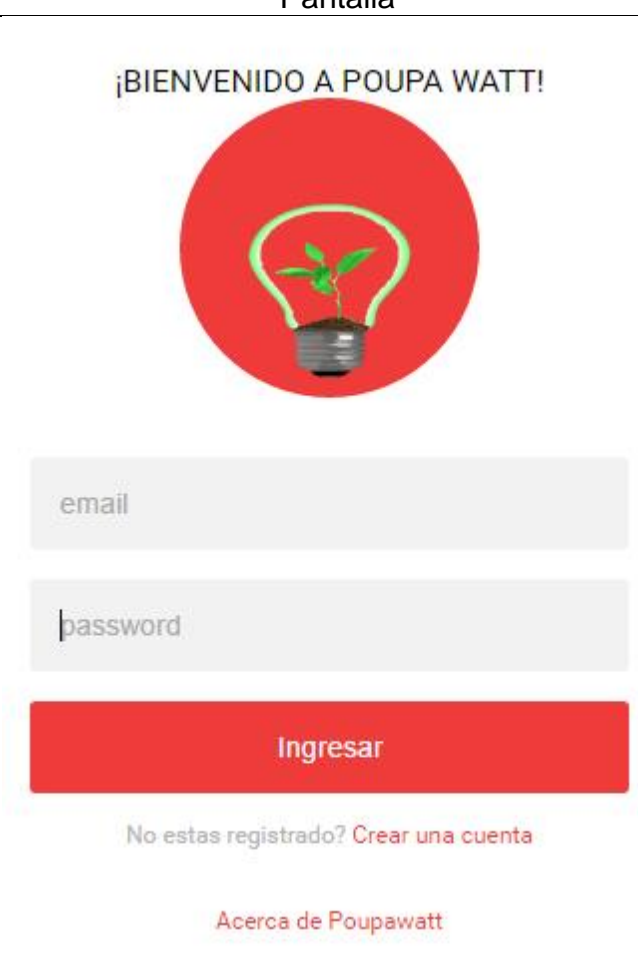
Pantalla

Descripción
Pantalla Juego: se observan los campos requerido para que el usuario calcule cuantas veces quiera, sus horarios de operación de lo electrodomesticos.

Fuente: los autores.

### 6.1.3.2 Usuario administrador

En la tabla 62, al igual que en el usuario estándar la pantalla de login es lo primero que se muestra para quien accede a poupawatt.info sin importar que usuario sea.

Tabla 62. Pantalla login Administrador aplicación web.

Pantalla
 <p>¡BIENVENIDO A POUPA WATT!</p> <p>email</p> <p>password</p> <p>Ingresar</p> <p>No estas registrado? <a href="#">Crear una cuenta</a></p> <p><a href="#">Acerca de Poupawatt</a></p>
Descripción
Pantalla Login: En esta pantalla se ingresa a la aplicación web llenado los campos de "Email" y "Password".

Fuente: Autores.

En la tabla 63 se muestra el home del administrador una vez ha hecho el login.


Tabla 63. Pantalla home administrador aplicación web.

Pantalla
 <p><b>POUPAWATT</b> MENU</p> <h1>HOLA MANUEL</h1> <p>Esta es la página del administrador, donde podrás modificar los parametros del sistema además del manejo de usuarios</p> <p><b>EMPEZAR</b></p> <h2>Manejo de usuarios</h2> <p>En esta sección podrás agregar, modificar y eliminar usuarios.</p>
Descripción
<p>Pantalla Home: Es la pantalla que se le muestra al administrador cada vez que entra al sitio, permitiéndole navegar por las diferentes secciones de la aplicación web.</p>

Fuente: Los autores.

La tabla 64 muestra el manejo de usuario del administrador.

Tabla 64. Pantalla usuarios aplicación web.

Pantalla
 <p><b>POUPAWATT</b> MENU ≡</p> <h2>Usuarios</h2> <p>INSTRUCCIONES &gt;&gt;</p> <p>Documento</p> <p>SELECCIONAR</p> <p><b>Username:</b> Username</p> <p><b>Email:</b> Email</p> <p><b>Nombre:</b> Nombre</p> <p><b>Apellido:</b> Apellido</p> <p><b>Documento:</b> Documento</p> <p><b>Dirección:</b> Direccion</p> <p>Seleccione un Estado ▾</p> <p><b>Teléfono:</b> Telefono</p>
Descripción
Pantalla Usuarios: como se observa permite identificar cualquier usuario registrado y realizar modificaciones en sus datos.

Fuente: Los autores.

La tabla 65 muestra la pantalla de manejo de usuarios de sensores.

Tabla 65. Pantalla sensores aplicación web.

Pantalla

Descripción
Pantalla Sensores: como se observa permite agregarle los sensores a los diferentes usuarios.

Fuente: Los autores.

La tabla 66 muestra la pantalla de manejo de electrodomésticos.


Tabla 66. Pantalla Electrodomésticos aplicación web.

Pantalla
 <p><b>POUPAWATT</b> MENU</p> <h2>Electrodomésticos</h2> <p>INSTRUCCIONES &gt;&gt;</p> <p>Electrodoméstico: Televisor</p> <p>Electrodoméstico: Nevera</p> <p>Electrodoméstico: Bombillas</p> <p>Electrodoméstico: Plancha</p> <p>Electrodoméstico: Lavadora</p> <p>Electrodoméstico</p> <p>Modelo</p> <p>Clase Energética</p> <p>marca</p> <p>Consumo</p> <p>Voltaje Op</p> <p>AGREGAR ELECTRODOMESTICO</p>
Descripción
Pantalla Electrodomesticos: permite agregas o modificar los electrodomesticos que hacen parte del sistema.

Fuente: Los autores.

La tabla 67 muestra la pantalla para el manejo de PRD.

Tabla 67. Pantalla Programas aplicación web.

Pantalla
 <p>POUPAWATT</p> <p>MENU</p> <h2>Programas</h2> <p>INSTRUCCIONES &gt;&gt;</p> <p>Programa: Oferta a la Demanda</p> <p>Programa: Tiempo de Uso</p> <p>Programa</p> <p>Precio</p> <p>Consumo</p> <p>Descripción</p> <p>AGREGAR PROGRAMA</p>
Descripción
Pantalla Programas: permite agregar o modificar los programas de respuesta a la demanda.

Fuente: Los autores.



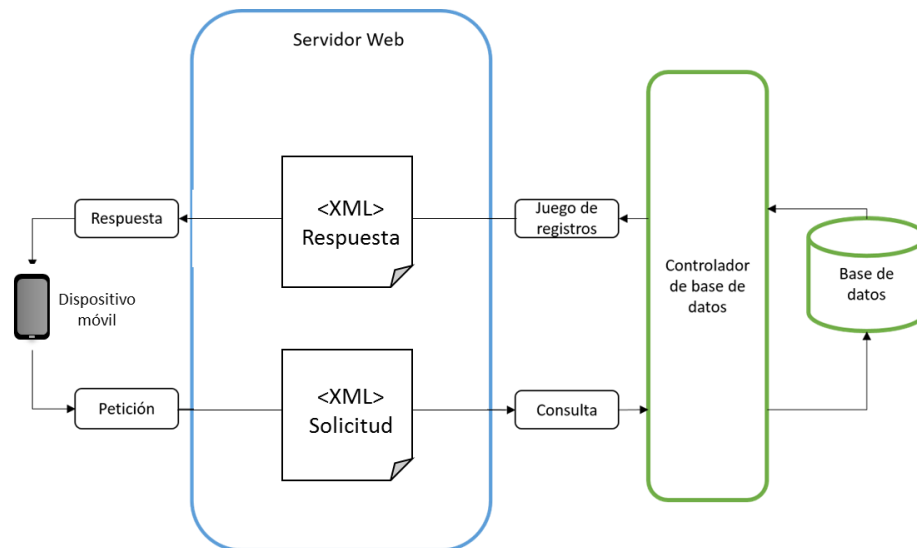
## 6.2 DESARROLLO E IMPLEMENTACIÓN APLICACIÓN MÓVIL

El sistema al estar enfocado en el usuario estándar, debe facilitar el acceso y uso del mismo, por eso se desarrolló una aplicación móvil de uso exclusivo para dicho usuario, continuando así con el uso de los servicios web.

Cabe destacar que, aunque todas las aplicaciones son programas, no todos los programas son aplicaciones. Existe multitud de software en el mercado, pero sólo se denomina así a aquel que ha sido creado con un fin determinado, para realizar tareas concretas. No se consideraría una aplicación, por ejemplo, un sistema operativo, ni una suite, pues su propósito es general. (Master Magazine, 2016)

En la figura 20 se observa la estructura básica de una aplicación web y partiendo de esta se eligieron las plataformas de desarrollo para que cada etapa fuera funcional y compatible con la siguiente.

Figura 20. Aplicación móvil.



Fuente: Los autores

### **6.2.1 Plataformas de desarrollo**

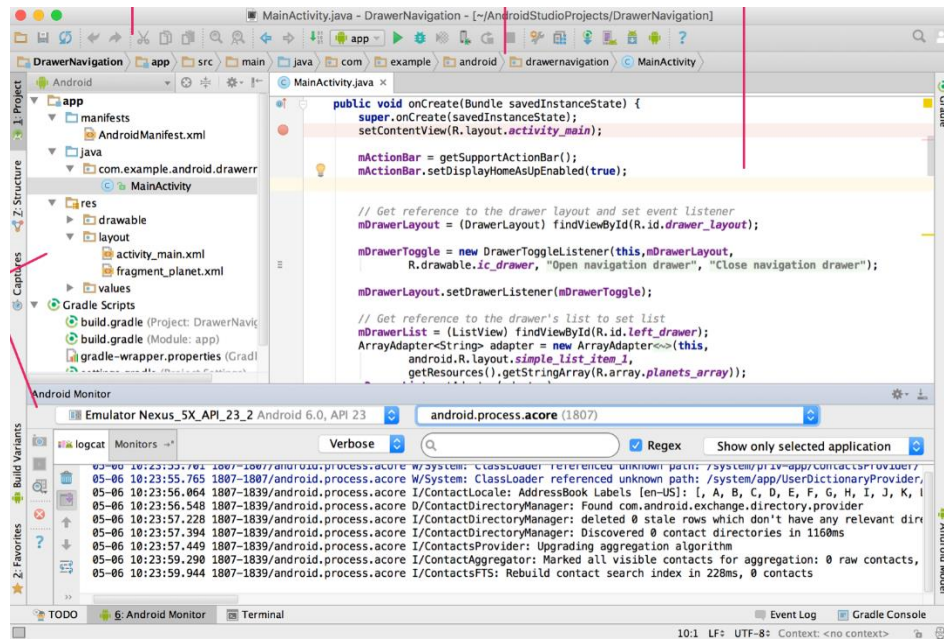
El servidor web es el encargado de almacenar los archivos que permiten el funcionamiento de la aplicación, comunicación con la base de datos.

Para la creación de los diferentes archivos de comunicación y dinámica de la aplicación, se utilizó como programa de desarrollo Sublime Text al igual que en la aplicación web, dichos archivos de la aplicación móvil utilizaron diferentes lenguajes de programación, cada uno con una funcionalidad y objetivo diferente dentro de la aplicación.

Para realizar las diferentes operaciones de comunicación con la base de datos y operaciones de la aplicación es necesario utilizar un lenguaje que se pueda incrustar en la estructura ya diseñada, para esto se eligió el PHP que es el acrónimo de Hipertext Preprocesor. Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación. (Alvarez, Desarrollo Web, 2001)

En el desarrollo de la estructura, funcionamiento interno y visualización de la aplicación se utilizó Android Studio, una herramienta para editar y compilar código para el desarrollo de aplicaciones de cualquier dispositivo Android, pues tiene la capacidad de renderización en tiempo real, viene incorporado con una consola de desarrollador la cual provee de consejos de optimización, ayuda para la traducción, estadísticas de uso y genera una refactorización específica de Android y arreglos rápidos como se puede observar en la figura 21. (Google Developers, 2013)

Figura 21. Entorno de desarrollo Android Studio.



Fuente (Google Developers, 2013)

Esta plataforma de desarrollo utiliza Java un lenguaje orientado a objetos de propósito general compilado e interpretado. Todo programa en Java ha de compilarse y el código que se genera bytecodes es interpretado por una máquina virtual. De este modo se consigue la independencia de la máquina, el código compilado se ejecuta en máquinas virtuales que si son dependientes de la plataforma. (Belmonte Fernández, 2005)

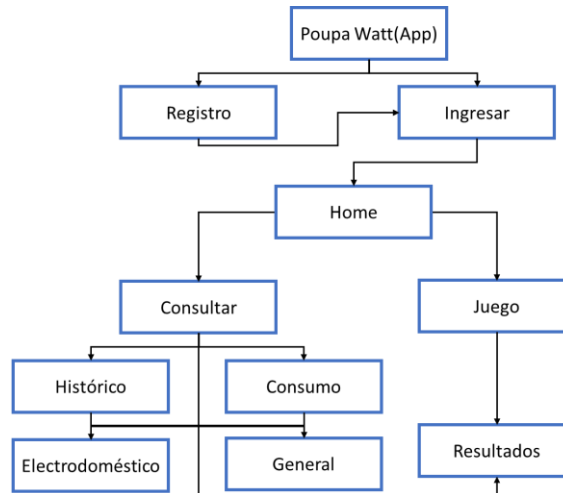
Por último, la aplicación móvil al estar alojada en el mismo servidor que la aplicación web el manejo de bases de datos es el mismo.

## 6.2.2 Aplicación móvil

La aplicación móvil se diseñó para que su navegación fuera intuitiva de manera que usuarios con o sin relación a este tipo de plataformas puedan navegar y hacer uso de la misma sin necesidad de un manual de usuario o de ayuda externa, de manera que se desarrolló para que así lo fuera. En la figura 22 está el mapa de navegación

de la aplicación móvil, del cual se partió para el desarrollo de la interfaz gráfica y estructura de la aplicación.

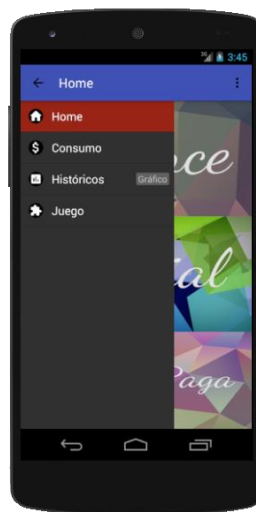
Figura 22. Mapa de navegación aplicación móvil.



Fuente: Los autores

En la figura 23 se observa el menú de la aplicación dando cumplimiento al mapa de navegación.

Figura 23. Vista previa aplicación




Fuente: Los autores

### 6.2.3 Funcionamiento aplicación móvil

De acuerdo a los casos de uso de la sección 5.2.2 y el mapa de navegación de la aplicación móvil de la tabla 68, en los cuales se identifica la interacción con el usuario final se desarrolló la interfaz gráfica y se implementó de la siguiente manera.


Tabla 68. Inicio aplicación móvil.

Pantalla	Descripción
	Pantalla inicio aplicación móvil: Para hacer uso de la plataforma móvil es necesario tener una sesión de usuario estándar iniciada, por esto al iniciar la aplicación la primera pantalla es la de validación de usuario.

Fuente: Los autores.

La tabla 69 muestra la pantalla de registro dentro de la aplicación móvil.


Tabla 69. Registro de nuevo usuario en la aplicación móvil

Pantalla	Descripción
	<p>Pantalla Registro de nuevo usuario en la aplicación móvil: En caso de que no se cuente con una cuenta de usuario estándar, en la pantalla de la tabla 68 se encuentra una opción para crear una nueva cuenta con el botón “registrarse”.</p>

Fuente: Los autores

La tabla 70 muestra la pantalla de un registro exitoso del usuario.

Tabla 70. Registro exitoso en aplicación móvil.

Pantalla	Descripción
	<p>Pantalla Registro exitoso en aplicación móvil: En la pantalla de la tabla 69, se observan una serie de campos, que fueron especificados en los requisitos y casos de uso, estos campos deben ser completados para así crear un nuevo registro y una vez completado esto aparece una confirmación de registro.</p>

Fuente: Los autores

La tabla 71 muestra la pantalla de registro de preferencias en la aplicación móvil.

Tabla 71. Registro de preferencias en aplicación móvil.


Pantalla	Descripción
	<p>Pantalla registro de preferencias en aplicación móvil: Una vez los datos personales se han registrado y se ha creado la nueva cuenta de usuario, se procede a registrar las preferencias del usuario para así dar por terminado el registro de un nuevo usuario.</p>

Fuente: Los autores

La tabla 72 muestra la pantalla del home una vez se hizo login.




Tabla 72. Home aplicación móvil.

Pantalla	Descripción
	<p>Pantalla Home aplicación móvil: Una vez terminado el registro, la aplicación se redirige a la pantalla de la tabla 68 para iniciar sesión con el nuevo usuario. Al iniciar sesión el usuario es llevado al “home” de la aplicación donde se presentan tres opciones.</p>

Fuente: Los autores

La tabla 73 contiene la pantalla con la información del proyecto en la aplicación móvil.


Tabla 73. Alcance del proyecto aplicación móvil.

Pantalla	Descripción
	<p>Pantalla alcance del proyecto aplicación móvil: En la tabla 72, se observan 3 opciones diferentes, estas son el alcance, tutorial y apagar paga. La primera, alcance presenta la información general del proyecto.</p>

Fuente: Los autores

La tabla 74 muestra la pantalla del tutorial de la aplicación móvil.

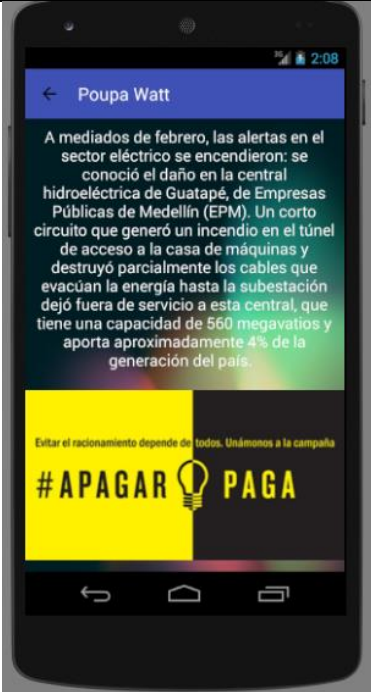
Tabla 74. Tutorial aplicación móvil

Pantalla	Descripción
	<p>Pantalla tutorial aplicación móvil: La segunda opción es el tutorial, en la cual se encuentra una explicación rápida y fácil de entender de como navegar en la aplicación.</p>

Fuente: Los autores

La tabla 75 muestra la pantalla de noticias e información relevante para el usuario.


Tabla 75. Apagar paga en aplicación móvil.

Pantalla	Descripción
	<p>Pantalla apagar paga en aplicación móvil: La tercera y última opción de la sección “Home” es “apagar paga”, donde se encontrará información acerca de los programas de respuesta a la demanda a nivel nacional.</p>

Fuente: Los autores

La tabla 76 muestra la pantalla con el menú de la aplicación móvil y sus diferentes secciones.

Tabla 76. Menú aplicación móvil

Pantalla	Descripción
	<p>Pantalla menú aplicación móvil: Para navegar por las demás secciones de la aplicación se encuentra un menú deslizable (Tabla 76), el cual contiene las diferentes secciones de la aplicación móvil.</p>

Fuente: Los autores

La tabla 77 muestra la pantalla para consulta de consumo en la aplicación móvil.


Tabla 77. Consulta consumo aplicación móvil.

Pantalla	Descripción
	<p>Pantalla consulta consumo aplicación móvil: En el menú deslizable de la tabla 76, se encuentra la sección “Consumo” que permite al usuario consultar cuanto ha consumido en cada uno de sus sensores y electrodomésticos entre dos fechas.</p> <p>En la tabla 77, para realizar la consulta del consumo primero se selecciona si se quiere consultar en precio o kWh, luego se procede a seleccionar el sensor asociado al electrodoméstico. Posteriormente se selecciona el rango de fechas y se presiona el botón consultar arrojando el resultado como se evidencia en la imagen.</p>

Fuente: Los autores

La tabla 78 contiene la pantalla para la consulta de históricos en la aplicación móvil.


Tabla 78. Históricos en aplicación móvil

Pantalla	Descripción
	<p>Pantalla de históricos en aplicación móvil: La siguiente opción del menú es “históricos” en el cual se puede hacer una consulta del consumo de manera gráfica.</p> <p>En la tabla 78 se muestra de manera gráfica el consumo, al igual que en la sección “consumo” se procede a seleccionar lo que se desea visualizar, precio o kWh, el sensor a observar y el rango de fechas, una vez hecho esto se presiona el botón graficar y se observa el consumo seleccionado como en la imagen.</p>

Fuente: Los autores

La tabla 79 muestra la pantalla con el juego dentro de la aplicación móvil.

Tabla 79. Juego aplicación móvil

Pantalla	Descripción
	<p>Pantalla Juego aplicación móvil: la última opción que se presenta en el menú es el juego, que permite al usuario a través de selección de horarios y electrodomésticos aprender a reducir su consumo.</p> <p>En la tabla 79, se observan las instrucciones y el visual del juego, al igual que el resultado del mismo que varía según lo escogido por el usuario.</p>

Fuente: Los autores

### 6.3 Comunicación aplicación móvil con aplicación web

Para un uso pleno del sistema Poupa Watt el usuario debe de estar en capacidad de interactuar con las dos aplicaciones, sin importar en cual se registró o realizó alguna acción.

En la figura 24 se encuentra el registro de la aplicación móvil, con el que posteriormente hará login y uso de la aplicación web demostrando así su conexión.



Figura 24. Registro en aplicación móvil



Fuente: Los autores

Los datos anteriores quedan registrados en la base de datos como se observa en la figura 25 y posteriormente con estos se hace login en la aplicación web.

Figura 25. Datos registrados en la base de datos.

Srincon19	sebastian091994	Sebastian	Rincon	1010217283
Calle 69 # 99 - 33	3143232830	3	sebas.rincon19@hotmail.com	

Fuente: Los autores

Con estos mismos datos se hace login en la aplicación web para comprobar su conexión. En la figura 26 se encuentra la visualización del perfil en la plataforma y su información, con la que se compara que sea la misma registrada en la aplicación móvil.

Figura 26. Perfil en aplicación web.

The image shows a web profile form titled "Perfil" on a red background. At the top center, there is a green button labeled "INSTRUCCIONES >>". The form is organized into two columns of input fields. The left column contains: Username (Srincon19), Email (sebas.rincon19@hotmail.com), Nombre (Sebastian), Apellido (Rincon), Documento (1010217283), and Dirección (Calle 69 # 99 - 33). The right column contains: Teléfono (3143232830), Estrato (Estrato 3), Ciudad (Bogota), Localidad (Usaquen), and Barrio (La Floresta De La Sabana). Each field is a light gray input box with a red border.

Field	Value
Username:	Srincon19
Email:	sebas.rincon19@hotmail.com
Nombre:	Sebastian
Apellido:	Rincon
Documento:	1010217283
Dirección:	Calle 69 # 99 - 33
Teléfono:	3143232830
Estrato:	Estrato 3
Ciudad:	Bogota
Localidad:	Usaquen
Barrio:	La Floresta De La Sabana

Fuente: Los autores

Como se puede observar, los datos registrados coinciden en ambas aplicaciones mostrando así la comunicación entre las mismas.

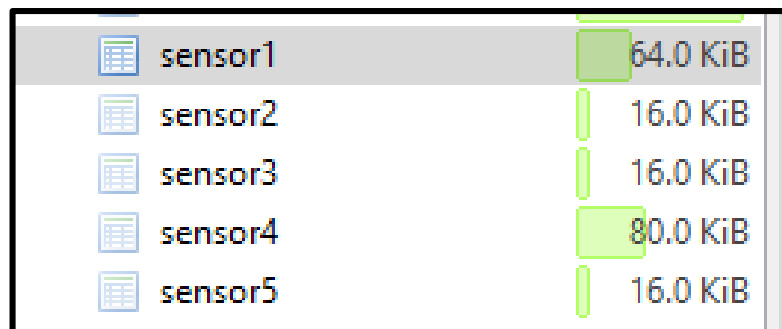
Por último, las validaciones donde se muestra con datos reales el comportamiento del sistema en todos los aspectos que puede llegar a trabajar, junto con los diferentes escenarios que se le pueden presentar al usuario.

## 7. VALIDACIONES

Las validaciones son muy importantes dentro del proceso de comprobación del funcionamiento de la aplicación móvil y la aplicación web, puesto que aseguran y confirman que el software funcionará bajo las especificaciones dadas desde un comienzo. Se busca analizar todos los casos que puede llegar a tomar el usuario y los posibles errores que se presenten en el momento de ejecutarla.

A continuación, las validaciones respectivas para la aplicación móvil y la aplicación web. Dentro de un caso de estudio en una vivienda de estrato 3, en donde se instalaron 5 sensores con el fin de obtener datos reales para el análisis de nuestros modelos y de todas las opciones que posee nuestro sistema. En la figura 27 se observa que en los servidores se encuentran almacenados datos para cada uno de los 5 sensores:

Figura 27. Datos servidores



sensor1	64.0 KiB
sensor2	16.0 KiB
sensor3	16.0 KiB
sensor4	80.0 KiB
sensor5	16.0 KiB

Fuente: Los Autores

Las fechas de estos datos se encuentran entre el 25 de enero del 2016 al 31 de enero del 2016, en donde cada sensor almaceno información en varios días dentro de este rango. La primera validación que se hará será la consulta de los históricos donde el usuario sabrá cuanto ha consumido en kilo vatio hora o en precio real.

## 7.1 Aplicación web

Las validaciones de la aplicación web, están enfocadas al funcionamiento de la consulta del consumo, funcionamiento juegos y funcionamiento algoritmos.

### 7.1.1 Consulta consumo

Cuando se consulta el consumo se generan dos casos, cuando el usuario cuenta y cuando no cuenta con sensores asignados.

En la figura 28 se observa que el usuario no cuenta con sensores asignados, por lo tanto, no puede hacer uso de dicha sección. Para poder hacer uso de esta, los sensores deben ser asignados al usuario por parte del administrador junto con un electrodoméstico para este.

Figura 28. Usuario sin sensores asignados.



Fuente: Los autores

En la figura 29 se observa la sección activada, una vez el usuario tiene sensores asignados por parte del administrador ya puede hacer uso de esta.

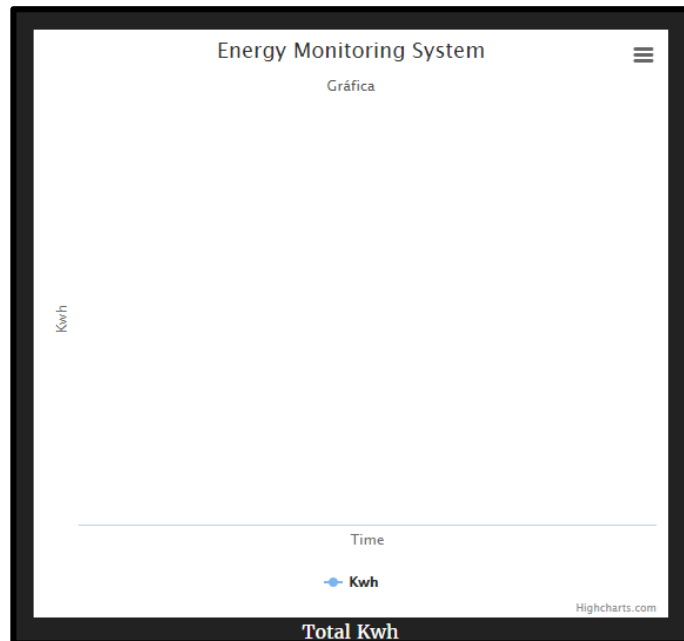
Figura 29. Usuario con sensores asignados.

The screenshot shows the POUPAWATT mobile application interface. At the top, the brand name "POUPAWATT" is displayed in red on the left, and a "MENU" button with a hamburger icon is on the right. The main heading is "Revisa tu consumo" in white. Below this is a green button labeled "INSTRUCCIONES >>". The form consists of several white input fields on a dark background: a dropdown menu for "Seleccione Sensor", another dropdown menu for "Select an option", a date field with the placeholder "dd/mm/aaaa", a time field with the placeholder "--:-- -----", another date field with the placeholder "dd/mm/aaaa", and a final time field with the placeholder "--:-- -----". At the bottom of the form is a white button labeled "CONSULTAR".

Fuente: Los autores

En este caso los sensores asignados al usuario recolectaron datos entre el 25 de enero de 2016 al 28 de enero de 2016. Sí el usuario selecciona un rango de fechas y horas no válido la gráfica estará en blanco como en la figura 30, en ese caso las fechas seleccionadas fueron del 21 de enero de 2016 al 24 de enero de 2016.

Figura 30. Grafica consumo vacía.



Fuente: Los autores

Cuando el usuario selecciona los rangos dentro de las fechas de funcionamiento de sus sensores puede observar su comportamiento y totalizar el consumo del rango, como se observa en la figura 30, la fecha seleccionada fue el 26 de enero de 2016 de las 20:36 a las 21:25 del mismo día. Es decir, el sistema tiene la capacidad de analizar 150 datos puesto que entre esas dos horas hay 3000 segundos y los sensores suben un dato cada 20 segundos. En la figura 31 se observa parte de los datos que serán adquiridos por el sistema y analizados para graficar y mostrar el consumo total en ese intervalo de tiempo.

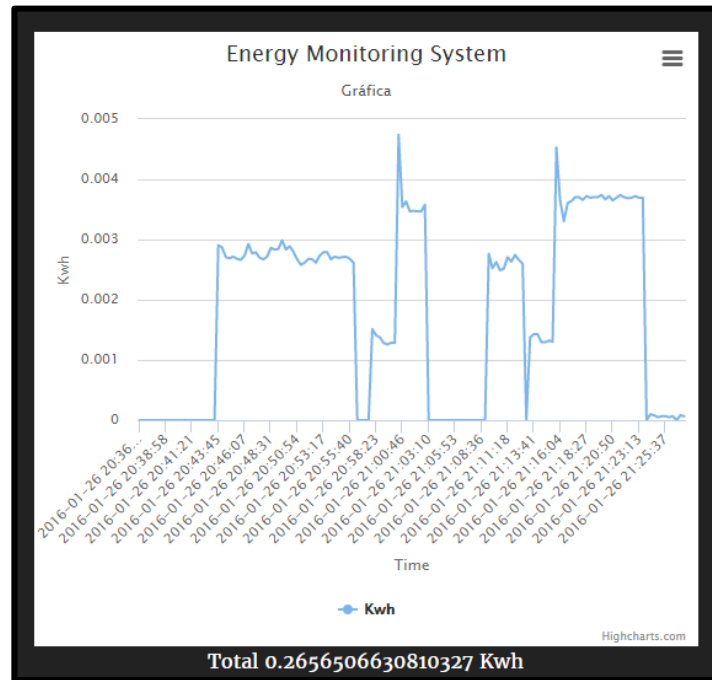
Figura 31. Datos almacenados

current	kwh	price	time	bits
0.0911	0.0000598163	0.0245127	2016-08-01 17:07:52	16
0.0571	0.0000374919	0.0153642	2016-08-01 17:08:12	14
0.0741	0.0000486541	0.0199384	2016-08-01 17:08:32	15
0.0741	0.0000486541	0.0199384	2016-08-01 17:08:53	15
0.1081	0.0000709785	0.029087	2016-08-01 17:09:53	17
1.5191	0.000997441	0.408751	2016-08-01 17:10:13	100
1.5531	0.00101977	0.4179	2016-08-01 17:10:34	102
1.5021	0.000986279	0.404177	2016-08-01 17:10:54	99
1.4511	0.000952792	0.390454	2016-08-01 17:11:59	96
1.4341	0.00094163	0.38588	2016-08-01 17:12:15	95
1.4001	0.000919306	0.376731	2016-08-01 17:12:35	93
1.4511	0.000952792	0.390454	2016-08-01 17:12:55	96
1.4171	0.000930468	0.381306	2016-08-01 17:13:16	94

Fuente: Los Autores

Una vez el sistema tome estos datos, realizará la sumatoria de las 150 casillas de la columna de kWh o de la columna price dependiendo de lo que el usuario seleccione. La cual se verá reflejada en la parte inferior de la gráfica como se observa en la figura 32.

Figura 32. Grafica consumo



Fuente: Los autores

En la figura 32 se totaliza el consumo en kWh del tiempo seleccionado obteniendo un total de 0.2656 kWh además de observar su comportamiento, permitiéndole al usuario identificar por sí mismo los horarios de consumo pico para el electrodoméstico asignado a dicho sensor. El usuario determina el consumo en ese intervalo de tiempo y puede tomar acción en ello para que mediante los algoritmos pueda generar un ahorro y obtener un beneficio de este.

En la gráfica se observa que el eje X representa el consumo en kWh que posee un intervalo desde 0.001 hasta 0.005 y del mismo modo, en el eje Y se encuentran las fechas seleccionadas por el usuario como lo son el 26/01/2016 siendo las 20:36 hasta el mismo 26/01/2016 a las 21:25.



### 7.1.2 Algoritmos modelos respuesta a la demanda

La ejecución de los algoritmos, es la base del proyecto, pues en estos se le dan opciones al usuario para que sin dejar de hacer las cosas como le gusta pueda ahorrar y reducir su consumo de energía eléctrica.

De manera que sí el usuario no cuenta con preferencias aún registradas, como se observa en la figura 33 es dirigido inmediatamente a registrarlas para hacer uso pleno de la aplicación web

Figura 33. Usuario sin preferencias registradas.



Fuente: Los autores

Cuando el usuario cuenta con preferencias ya asignadas se pueden presentar 3 casos, el primero cuando en estas escogió el Tiempo de uso como programa de respuesta a la demanda como es el caso de la figura 34.

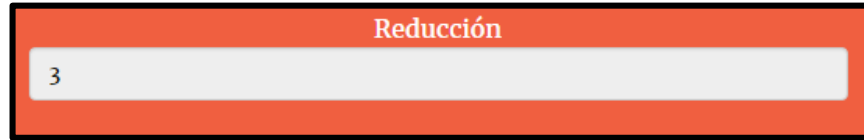
Figura 34. Tiempo de uso como preferencia



Fuente: Los autores

En la figura 35 se identifica que el usuario desea realizar una reducción de 3 KWh hora en la siguiente semana, es por ello que usa el modelo con el fin de que este le genere toda la planeación necesaria para que se cumpla el objetivo que posee este usuario.

Figura 35. Valor de reducción

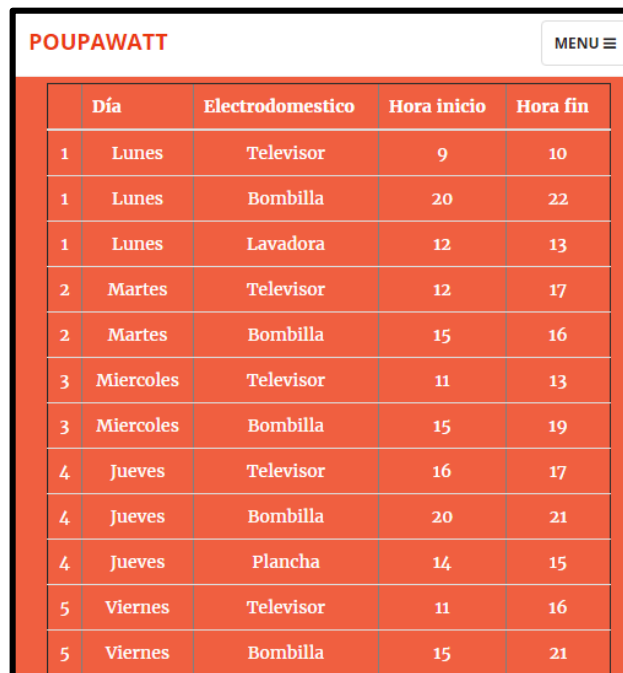


A screenshot of a web form with a red border. The title 'Reducción' is centered at the top in white text. Below it is a white input field containing the number '3'.

Fuente: Los Autores

Una vez teniendo ese valor de reducción que el usuario desea, el modelo procede a calcular y mostrar en pantalla la manera en que el usuario debe de manejar los electrodomésticos que se encuentran registrados en la plataforma. En la figura 36 se detalla que para que el usuario pueda reducir 3 kilo vatio hora a la semana debe de encender el televisor en las horas de la mañana y apagarlo en la tarde y en la noche; la bombilla reducir una poco el consumo de esta teniéndola prendida alrededor de 2 a 3 horas por día. Tanto la plancha como la lavadora se realizan 1 vez a la semana teniendo una duración de 1 hora en este caso, los Lunes para la lavadora y los Jueves para la plancha.

Figura 36. Respuesta tiempo de uso



A screenshot of a mobile application interface. At the top, the text 'POUPAWATT' is displayed in red, and a 'MENU' button with a hamburger icon is on the right. Below is a table with a red background and white text, showing a weekly schedule of appliance usage.

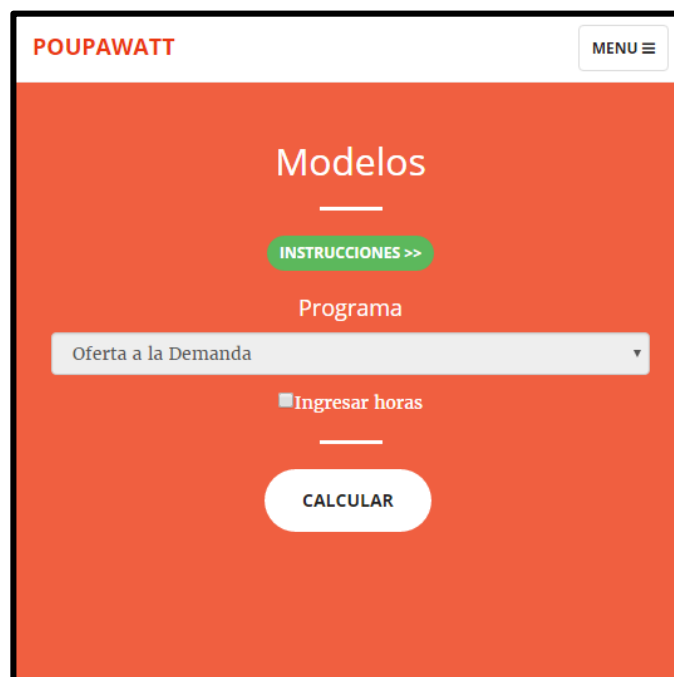
	Día	Electrodomestico	Hora inicio	Hora fin
1	Lunes	Televisor	9	10
1	Lunes	Bombilla	20	22
1	Lunes	Lavadora	12	13
2	Martes	Televisor	12	17
2	Martes	Bombilla	15	16
3	Miercoles	Televisor	11	13
3	Miercoles	Bombilla	15	19
4	Jueves	Televisor	16	17
4	Jueves	Bombilla	20	21
4	Jueves	Plancha	14	15
5	Viernes	Televisor	11	16
5	Viernes	Bombilla	15	21

Fuente: Los autores

En la figura 36 la planeación semanal se presenta como una tabla en la que se asigna para cada día los electrodomésticos con unas horas de operación recomendadas. Si estas horas de operación son respetadas, se asegura lograr el valor de reducción indicado en las preferencias del usuario. El algoritmo le muestra que electrodomésticos puede prender y en que horario debe de hacerlo. Debido a que este es el programa de Tiempo de uso, el modelo le da prioridad a las horas valles que se encuentran ubicadas en las horas de la mañana y en la tarde.

El siguiente caso se da cuando el usuario seleccionó previamente en sus preferencias el programa de respuesta a la demanda, como se observa en la figura 37.

Figura 37. Oferta a la demanda como preferencia



The screenshot shows the POUPAWATT application interface. At the top left is the logo 'POUPAWATT' and at the top right is a 'MENU' button with a hamburger icon. The main content area has a red background. The title 'Modelos' is centered at the top. Below it is a green button labeled 'INSTRUCCIONES >>'. Underneath is the label 'Programa' above a dropdown menu that currently displays 'Oferta a la Demanda'. Below the dropdown is a checkbox labeled 'Ingresar horas'. At the bottom of the form is a white button with the text 'CALCULAR'.

Fuente: Los autores.

Al ejecutar este algoritmo se pueden presentar dos opciones, la primera la ejecución normal y la segunda siendo el tercer caso, cuando el usuario selecciona ingresar horas. La respuesta de los dos se presenta de la misma manera, pero difiere en que al seleccionar ingresar horas como se observa en la figura 38 el usuario ingresa las

horas en que está dispuesto a desplazar su carga, mientras que de la otra manera dicha horas son asignadas aleatoriamente.

Figura 38. Ingreso de horas para ejecución de oferta a la demanda.

The screenshot shows the POUPAWATT web interface. At the top left is the logo "POUPAWATT" and at the top right is a "MENU" button with a hamburger icon. Below the logo, the word "Programa" is displayed above a dropdown menu currently showing "Oferta a la Demanda". Underneath the dropdown is a checked checkbox labeled "Ingresar horas". The interface then lists three appliance categories, each with two input fields for "Hora operación inicial" and "Hora operación Final":

- Televisor**: Two white input fields for initial and final operation hours.
- Lavadora**: Two white input fields for initial and final operation hours.
- Bombilla**: Two white input fields for initial and final operation hours.

Fuente: Los autores

Para ambos casos la respuesta se presenta igual que en la figura 36, pero con los horarios de operación respectivos según el programa escogido.

### 7.1.3 Juego

Cuando se ejecuta el juego, es necesario llevar a cabo las instrucciones que se muestran en pantalla para su correcto funcionamiento, para esta parte se toman los datos de los sensores como se muestran en la figura 39 para realizar la sumatoria dentro del intervalo dado por el usuario y de esta manera tener el consumo límite con el que podrá interactuar.

Figura 39. Datos sensores para sumatoria.

kwh	price	time
0.0000598163	0.0245127	2016-08-01 17:07:52
0.0000374919	0.0153642	2016-08-01 17:08:12
0.0000486541	0.0199384	2016-08-01 17:08:32
0.0000486541	0.0199384	2016-08-01 17:08:53
0.0000709785	0.029087	2016-08-01 17:09:53
0.000997441	0.408751	2016-08-01 17:10:13
0.00101977	0.4179	2016-08-01 17:10:34
0.000986279	0.404177	2016-08-01 17:10:54

kwh	price	time
0.000140906	0.0566148	2016-07-27 21:44:16
0.000140906	0.0566148	2016-07-27 21:43:18
0.000140906	0.0566148	2016-07-27 21:41:16
0.000140906	0.0566148	2016-07-27 21:39:54
0.000140906	0.0566148	2016-07-27 21:37:14
0.000140906	0.0566148	2016-07-27 21:29:10
0.000145371	0.0584087	2016-07-27 21:49:37
0.000145371	0.0584087	2016-07-27 21:49:18

kwh	price	time
0.00623888	2.55669	2016-07-26 12:46:09
0.00632595	2.59237	2016-07-26 12:46:29
0.000104465	0.0428098	2016-07-26 12:46:49
0.0000598163	0.0245127	2016-07-26 12:47:29
0.0000486541	0.0199384	2016-07-26 12:47:50
0.0000598163	0.0245127	2016-07-26 12:48:10
0.0000598163	0.0245127	2016-07-26 12:48:31
0.0000486541	0.0199384	2016-07-26 12:49:54

kwh	price	time
0.000265923	0.106845	2016-07-25 12:48:25
0.000270388	0.108639	2016-07-25 12:48:46
0.000261458	0.105051	2016-07-25 12:49:06
0.000270388	0.108639	2016-07-25 12:49:26
0.000270388	0.108639	2016-07-25 12:49:46
0.000270388	0.108639	2016-07-25 12:50:26
0.000265923	0.106845	2016-07-25 12:50:46
0.000270388	0.108639	2016-07-25 12:51:06

Fuente: Los Autores

Una vez hecho esto se pueden presentar dos casos, el primero cuando el usuario con las horas de operación dadas, en las fechas especificadas logró la reducción que ingreso como se observa en la figura 40. El usuario deseaba reducir el 10% del consumo total, cuando el juego calcula ese valor le pide al usuario que ingrese los valores para luego arrojar una respuesta.

Figura 40. Reducción alcanzada en el juego.

The image shows a mobile application interface with a dark red background. At the top, there is a white notification box with the text "Excelente! Lograste ahorrar" and a close button (X). Below the notification, there is a red button labeled "CERRAR". The main content area contains several input fields for time selection, each with a label above it: "Hora Final 1" (with the value "04:00 p. m."), "Hora Inicio 2", "Hora Final 2", "Hora Inicio 3", and "Hora Final 3". At the bottom, there is a grey button labeled "CALCULAR".

Fuente: Los autores

Para el caso de la figura 40, al televisor se le asignaron 2 horas de operación en la noche y una en la tarde, a la nevera se le asignó permanente operación, las bombillas 4 horas de operación en la noche, la plancha una hora en la tarde y por último la nevera hora y media en horas de la tarde, para un plazo de una semana como se recomienda en las instrucciones. De esta forma si el usuario decide consumir tal cual como ingresó esos valores puede generar ese ahorro del 10% del consumo total.

El segundo caso que se puede presentar es que el usuario con las horas de operación indicadas, supere la reducción deseada como se observa en la figura 41, en ese caso tiene la opción de volver a calcular su consumo.

Figura 41. Reducción excedida en el juego.

The image shows a mobile application interface with a dark red background. At the top, a white notification box contains the text "No lograste ahorrar el valor pensado!" and a red "CERRAR" button. Below this, the form is organized into three sections, each with a title and a corresponding input field:

- Hora Final 1:** Input field containing "04:00 p. m."
- Hora Inicio 2:** Input field containing "--:--:--"
- Hora Final 2:** Input field containing "--:--:--"
- Hora Inicio 3:** Input field containing "--:--:--"
- Hora Final 3:** Input field containing "--:--:--"

At the bottom center, there is a grey "CALCULAR" button.

Fuente: Los autores

La figura 41 hace caso a un exceso, pues se doblaron los horarios de operación de la figura anterior, superando así el valor de reducción propuesto.

## 7.2 Aplicación móvil

La aplicación móvil al tener menos secciones para interacción tendrá validaciones de todas sus opciones de navegación, a continuación, las diferentes validaciones de la misma. Al igual que con la plataforma web se tomaron los datos almacenados en los servidores de Poupa Watt para cada uno de los 5 sensores registrados para el caso de estudio de la casa de estrato 3.



### 7.2.1 Validación de Login

Esta es la primera validación que se realiza debido a que es la primera layout con la que interactúa el usuario, esta parte consiste en ingresar los datos como el usuario y la contraseña para poder acceder a todo el contenido de la aplicación. Para la validación de esta parte se presentan dos casos el caso 1 en donde se introduce los datos erróneos y el caso 2 donde los datos son correctos según los que están en la base de datos. En la Figura 42 se muestran los datos almacenados en el servidor que son con los que la aplicación hará la validación:

Figura 42. Datos en base de datos de aplicación móvil.

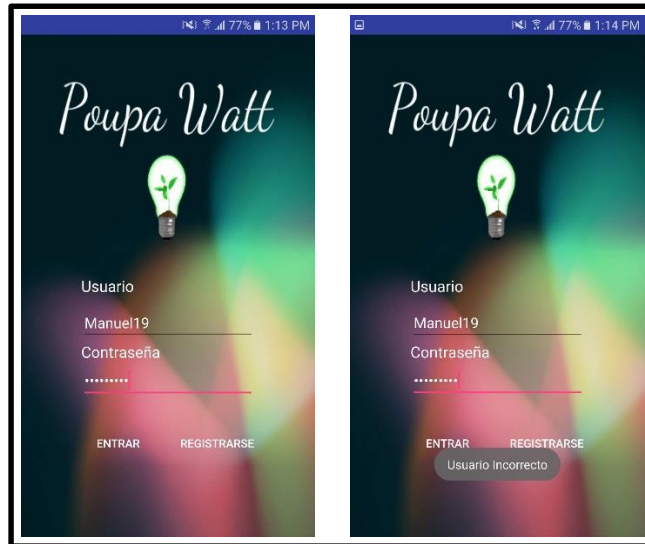
intNivel	username	password
1	Manuel	Abc123456
2	Srincon19	sebaatian091994

Fuente: Los autores

- Caso 1

Es cuando el usuario por error de digitación o por olvido introduce los datos de manera errónea y esto el sistema lo detecta y le informa al usuario que los datos son incorrectos por esta razón no puede acceder al contenido de la aplicación, así como se muestra en la figura 43.

Figura 43. Caso 1 login aplicación móvil

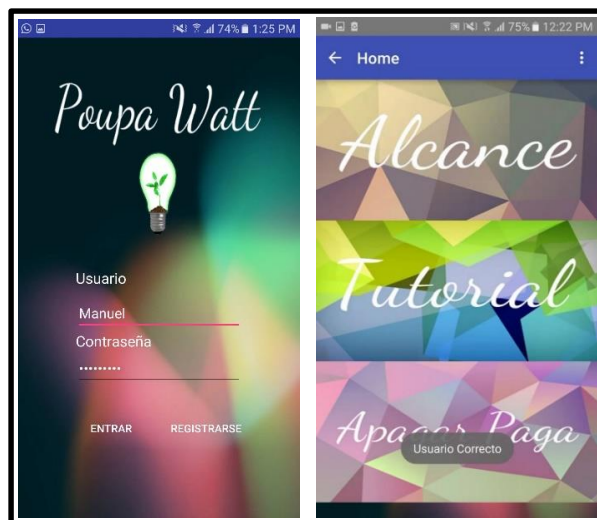


Fuente: Los autores

- Caso 2

Por el otro lado este caso es cuando el usuario ingresa correctamente los datos, es decir, se encuentra registrado en las bases de datos de la aplicación y tiene puede acceder a todo el contenido que le ofrece la aplicación como lo muestra la figura 44.

Figura 44. Caso 2 Login aplicación móvil



Fuente: Los autores

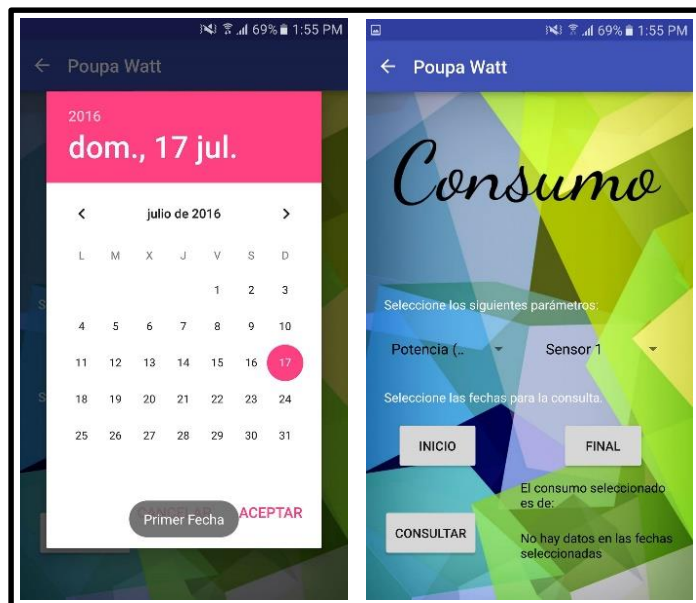
## 7.2.2 Validación de Consumo

Una de las opciones que tiene el usuario dentro de la aplicación es consultar su consumo almacenado por los sensores en nuestro servidor, para esto se deben de dar ciertos parámetros necesarios para que se pueda generar la búsqueda correspondiente para validar el funcionamiento se tiene tres casos el primero cuando se introducen fechas en donde no hay datos registrado, otro cuando se desea consultar el consumo por precio y por ultimo cuando se desea consultar el consumo en kWh.

- Caso 1

Este caso es cuando el usuario dentro de los parámetros de ingreso selecciona dos fechas completamente diferentes a las que los sensores registraron en el servidor, para esta validación se tienen datos solo del mes de enero de 2016 es decir, que toda fecha fuera de este mes es invalida hasta que no se registren nuevamente datos en un nuevo margen. En la figura 45 se muestra como reacciona el sistema ante un caso de estos.

Figura 45. Caso 1 Validación de consumo en aplicación móvil.



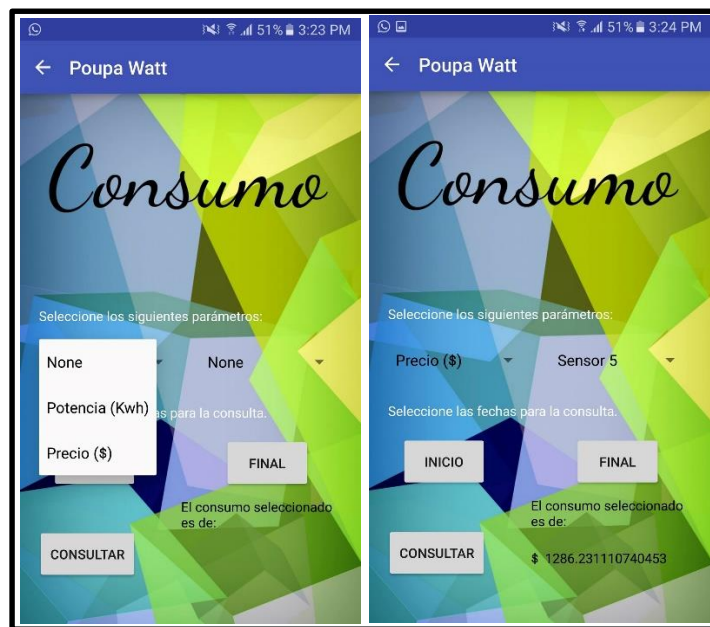
Fuente: Los autores

Se ingresó como parámetro la fecha de 17 de Julio de 2016 en donde el sistema reconoce que no hay datos registrados y por esto le informa que no existen datos en las fechas seleccionadas y por ello no se puede mostrar el consumo del usuario.

- Caso 2

El usuario puede consultar cuanto le costará el consumo que lleva hasta la fecha correspondiente que introduzca, para esto también necesita los parámetros básicos como en todos los casos, pero en uno de ellos tiene la posibilidad de consultar el consumo en Potencia o en Precio como se muestra en la figura 46.

Figura 46. Caso 2 Validación de consumo en aplicación móvil.



Fuente: Los autores

Al seleccionar precio como se muestra en la figura 43 se observa que en la parte inferior izquierda el usuario obtiene que entre las fechas seleccionadas es decir entre el 25 de enero de 2016 y el 30 de enero de 2016 en sensor 5 a medido el consumo correspondiente a \$1286 pesos colombianos. Es decir, el sistema tiene la

capacidad de analizar 150 datos puesto que entre esas dos horas hay 3000 segundos y los sensores suben un dato cada 20 segundos. En la figura 47 se observa parte de los datos que serán adquiridos por el sistema y analizados para graficar y mostrar el consumo total en ese intervalo de tiempo.

Figura 47. Datos de consumo a graficar

current	kwh	price	time	bits
0.1081	0.0000709785	0.029087	2016-07-27 19:47:43	17
0.0741	0.0000486541	0.0199384	2016-07-27 19:48:04	15
0.0911	0.0000598163	0.0245127	2016-07-27 19:48:24	16
0.0571	0.0000374919	0.0153642	2016-07-27 19:48:44	14
0.0741	0.0000486541	0.0199384	2016-07-27 19:49:24	15
0.0571	0.0000374919	0.0153642	2016-07-27 19:49:45	14
0.0741	0.0000486541	0.0199384	2016-07-27 19:50:05	15
0.0741	0.0000486541	0.0199384	2016-07-27 19:50:34	15
0.0741	0.0000486541	0.0199384	2016-07-27 19:51:08	15
0.0401	0.0000263297	0.0107899	2016-07-27 19:51:36	13
0.0571	0.0000374919	0.0153642	2016-07-27 19:51:46	14

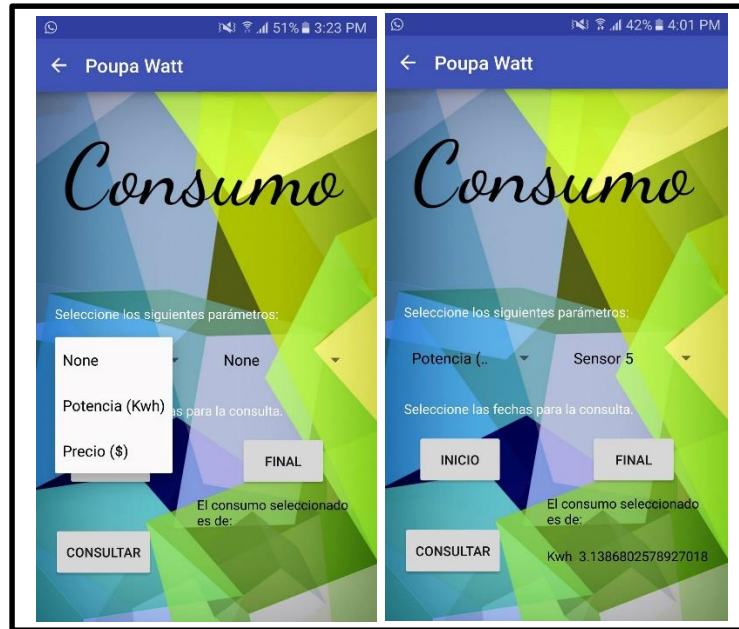
Fuente: Los Autores

Basados en los datos mostrados en la figura 47 se muestra el consumo tanto para el Caso 2 en donde la figura 46 muestra el resultado de consultar deseando saber el consumo en el precio colombiano. Como para el Caso 3 que se muestra a continuación donde se busca saber el consumo en la unidad de kWh.

- Caso 3

El usuario puede consultar cuanto es el consumo que lleva hasta la fecha correspondiente que introduzca, para esto también necesita los parámetros básicos como en todos los casos, pero en uno de ellos tiene la posibilidad de consultar el consumo en Potencia o en Precio como se muestra en la Figura 48.

Figura 48. Caso 3 Validación consumo en aplicación móvil.



Fuente: Los autores

La Figura 48 evidencia que el usuario obtiene el consumo en kWh almacenado entre las fechas seleccionadas es decir entre el 25 de enero de 2016 y el 30 de enero de 2016 en sensor 5. Es decir, el sistema tiene la capacidad de analizar 150 datos puesto que entre esas dos horas hay 3000 segundos y los sensores suben un dato cada 20 segundos. Y de esta manera arroja el resultado que este medidor a sensado un consumo de 3.13 Kwh.

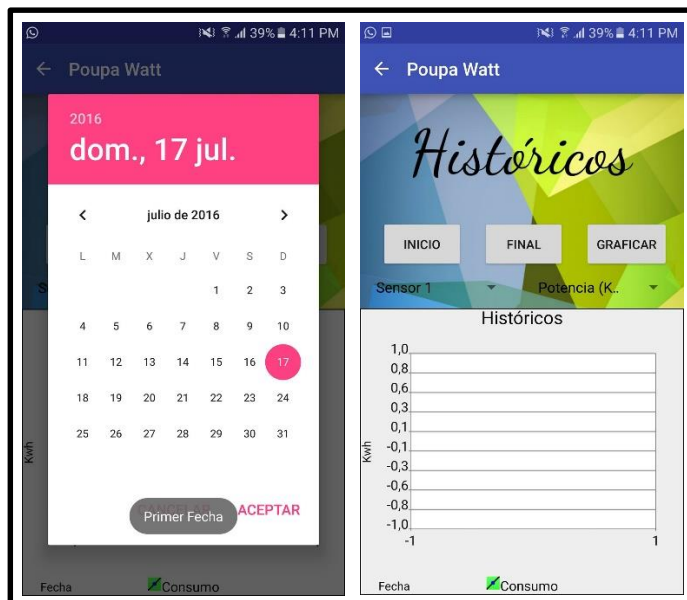
### 7.2.3 Validación de Históricos

El sistema puede graficar los históricos de consumo tomados de los datos almacenados correspondientes a cada usuario. El por eso que el esto es otra de las opciones posibles dentro del menú, para validar la graficada existen 2 casos, el primero cuando se introducen fechas en donde no hay datos registrado y por ultimo cuando se desea graficar en KWh.

- Caso 1

Este caso es cuando el usuario dentro de los parámetros de ingreso selecciona dos fechas completamente diferentes a las que los sensores registraron en el servidor, para esta validación se tienen datos solo del mes de enero de 2016 es decir, que toda fecha fuera de este mes es invalida hasta que no se registren nuevamente datos en un nuevo margen. En la figura 49 se muestra ver cómo reacciona el sistema ante este caso.

Figura 49. Caso 1 Validación de históricos en aplicación móvil.



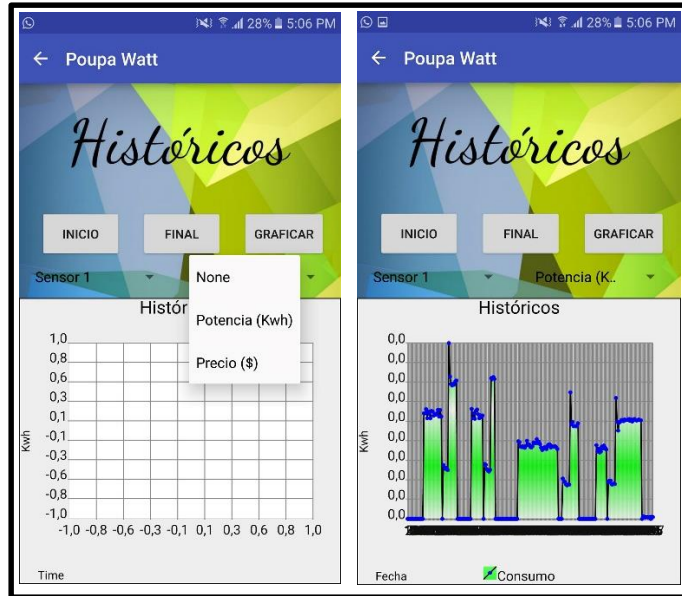
Fuente: Los autores.

Se ingresó como parámetro la fecha de 17 de Julio de 2016 en donde el sistema reconoce que no hay datos registrados y por esto no grafica ningún histórico.

- Caso 2

Si el usuario introduce bien los parámetros puede generar el histórico del consumo en kWh partiendo de las fechas que son las más importantes, así como muestra la Figura 50.

Figura 50. Caso 2 Validación de históricos aplicación móvil.



Fuente: Los autores

El usuario tiene la posibilidad de escoger entre todos los sensores que tenga registrados para que pueda observar la gráfica correspondiente a cada uno de ellos. En este caso se observa la gráfica de datos para el sensor 1 del día 25 de Enero de 2015, los valores del eje X se observan como 0.0 debido a que los sensores midieron valores entre 0.0001 y 0.005. Para el eje Y se encuentran las fechas seleccionadas por el usuario como lo son el 25/01/2016 al 26/01/2016.

#### 7.2.4 Validación de Juego

Esta parte es donde el usuario tiene la mayor libertad de utilizar la aplicación puesto que lo que se busca con el juego es que el usuario partiendo de un porcentaje que desee busca la mejor manera para llegar a él, ingresando horas como el crea convenientes en donde usaría cada electrodoméstico. Acá se podría presentar varios casos el primero cuando el usuario no ingresa ninguna hora solo selecciona los electrodomésticos, el segundo cuando el usuario selecciona dos o más veces el



mismo electrodoméstico, la tercera un usuario que ingrese un valor muy elevado de porcentaje, el cuarto cuando un usuario ingresa un valor muy bajo.

- Caso 1

El usuario tiene la capacidad de escoger entre los 5 electrodomésticos registrados y debe de introducir las horas de operación en las que trabajarían. Tiene 3 posibilidades de horas para ingresar, pero en este caso es como si solo el usuario selecciona los electrodomésticos y no las horas, en la figura 51 se muestra cual sería el comportamiento del sistema.

Figura 51. Caso 1 Validación de juego aplicación móvil.



Fuente: Los autores

Debido a que no se ingresó ninguna hora, el sistema toma esos campos vacíos como un 0, dando como resultado un ahorro muy grande puesto que sería como si no se conectaría ningún electrodoméstico en ningún momento.

- Caso 2

En este caso como el usuario debe de seleccionar los electrodomésticos para asignarle una hora a cada uno puede que seleccione dos electrodomésticos iguales dentro de los spinners y el comportamiento del sistema sería el de la figura 48.

Figura 52. Caso 2 Validación juego en aplicación móvil.



Fuente: Los autores

Sin importar si repitió un electrodoméstico, el sistema lo toma y lo suma al otro registrado. Y de igual forma le arroja un resultado dependiendo de las horas de operación que haya seleccionado.

- Caso 3

El porcentaje que desea reducir el usuario es muy importante porque de ahí parte el juego por ende debe de ser un porcentaje que realmente pueda alcanzarse para que pueda tener un resultado exitoso, en este caso se pondrá un porcentaje muy elevado la figura 53 muestra la respuesta de la aplicación.

Figura 53. Caso 3 Validación juego en aplicación móvil.



Fuente: Los autores

Debido a que fue un 80% el que se introdujo era muy complicado lograr que los electrodomésticos lograran el objetivo, es por esto que sale el mensaje el cual dice que no logró ahorrar el valor pensado, además consumió más de lo que se estaba esperando.

- Caso 4

El porcentaje que desea reducir el usuario es muy importante porque de ahí parte el juego por ende debe de ser un porcentaje que realmente pueda alcanzarse para que pueda tener un resultado exitoso, en este caso se pondrá un porcentaje muy bajo la figura 54 muestra la respuesta de la aplicación.

Figura 54. Caso 4 Validación juego, aplicación móvil.



Fuente: Los autores.

Por el contrario en este caso se logra el objetivo puesto que se quería reducir el 10% de consumo actual que tiene el usuario, y con las horas de operación registradas puede lograrlo y generar un ahorro de 0.677 KWh.

### 7.2.4 Posibles Errores

Dentro de las validaciones hechas se puede evidenciar que la aplicación puede presentar error cuando el usuario no introduzca ningún valor, puesto que no tiene los parámetros base para actuar y no sabe qué acción realizar lo que la lleva a entrar en conflicto y dejar de funcionar. Es por eso que siempre debe de haber al menos un parámetro de los solicitados para que esta no entre en conflicto.

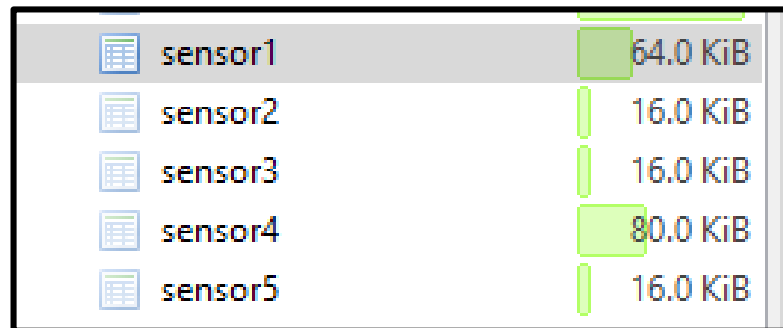
### 7.3 Nuevo caso de estudio

Se realizó un nuevo caso de estudio, en donde se instalaron 5 sensores en una casa de estrato 3 con el fin de obtener datos del consumo real en un hogar con una

familia común. Estos datos eran enviados a nuestros servidores para que posteriormente fuesen usados por nuestro sistema Poupa Watt.

En la figura 55 podemos observar que en los servidores se encuentran almacenados datos para cada uno de los 5 sensores.

Figura 55. Sensores en los servidores



sensor1	64.0 KiB
sensor2	16.0 KiB
sensor3	16.0 KiB
sensor4	80.0 KiB
sensor5	16.0 KiB

Fuente: Los Autores

Las fechas de estos datos se encuentran entre el 26 de Julio del 2016 al 1 de Agosto del 2016, en donde cada sensor almaceno información en varios días dentro de este rango. De esta manera se validará cada uno de estos sensores con el fin de mostrar que los nuevos datos son tomados tanto por la aplicación como por la plataforma web.

### 7.3.1 Consulta consumo del sensor 1

El usuario tiene la posibilidad de escoger el sensor que desee para conocer el consumo correspondiente a las fechas que desee insertar en la figura 56 se puede observar la elección tomada, en donde se muestra los históricos y el consumo del sensor 1 para el día 01 de Agosto de 2016 desde la 1:00 am hasta las 11:00 pm.

Figura 56. Consulta consumo para el sensor 1

sensor1

kWh

01/08/2016

01:00

01/08/2016

23:00

CONSULTAR

Fuente: Los Autores

De esta manera el sistema realiza una búsqueda dentro de los servidores con el fin de encontrar los datos dentro del rango ingresado por el usuario. De esta manera en la figura 57 se detallan los datos que encontraría el sistema para general los históricos y el consumo total.

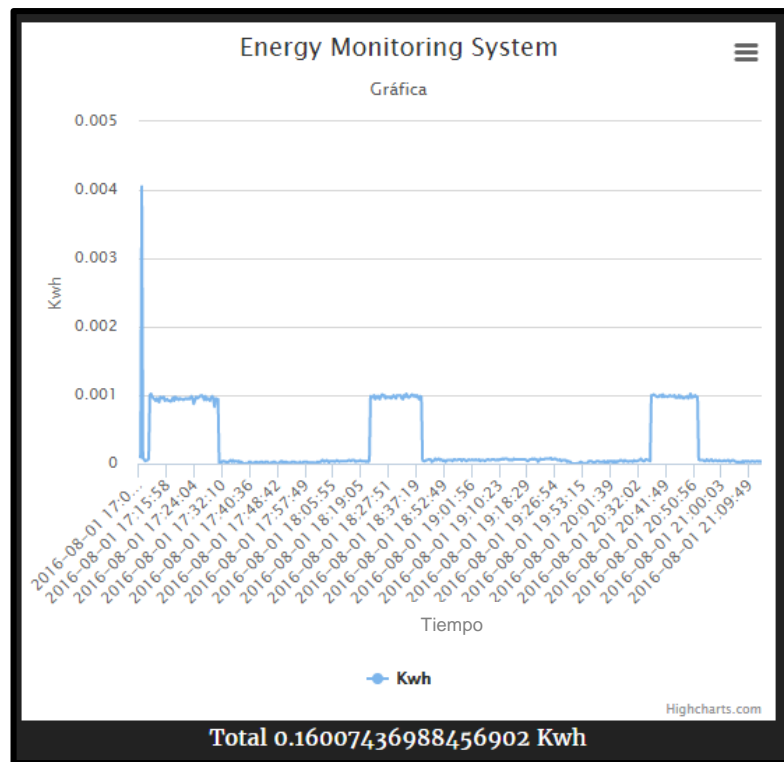
Figura 57. Datos del sensor 1 en el servidor

kwh	price	time
0.0000151675	0.00621562	2016-08-01 19:52:13
0.0000263297	0.0107899	2016-08-01 19:52:33
0.0000151675	0.00621562	2016-08-01 19:52:53
0.00000400526	0.00164136	2016-08-01 19:53:15
0.00000400526	0.00164136	2016-08-01 19:53:35
0.00000400526	0.00164136	2016-08-01 19:53:54
0.00000400526	0.00164136	2016-08-01 19:54:14
0.0000151675	0.00621562	2016-08-01 19:54:34
0.0000374919	0.0153642	2016-08-01 19:54:55
0.0000374919	0.0153642	2016-08-01 19:55:15
0.0000374919	0.0153642	2016-08-01 19:55:35
0.0000263297	0.0107899	2016-08-01 19:55:55
0.0000151675	0.00621562	2016-08-01 19:56:35
0.0000151675	0.00621562	2016-08-01 19:56:55
0.0000263297	0.0107899	2016-08-01 19:57:15
0.0000374919	0.0153642	2016-08-01 19:57:36

Fuente: Los autores

Teniendo así el soporte para realizar la gráfica de consumo y el cálculo de la cantidad de kWh que fueron usados en esas fechas dando como resultado los históricos de la figura 58.

Figura 58. Grafica consulta sensor 1



Fuente: Los autores

En la figura 58 se totaliza el consumo en kWh del tiempo seleccionado obteniendo un total de 0.16007 kWh además de observar su comportamiento, permitiéndole al usuario identificar por sí mismo los horarios de consumo pico para el electrodoméstico asignado a dicho sensor. El usuario determina el consumo en ese intervalo de tiempo y puede tomar acción en ello para que mediante los algoritmos pueda generar un ahorro y obtener un beneficio de este.

En la gráfica se puede observar el eje X representa el consumo en KWh que posee un intervalo desde 0.001 hasta 0.005 y del mismo modo en el eje Y se encuentran

las fechas seleccionadas por el usuario como lo son el 01/08/2016 siendo las 17:06 hasta el mismo 01/08/2016 a las 21:14.

### 7.3.2 Consulta consumo del sensor 2

En el momento en que el usuario desee conocer el consumo del sensor 2 debe de ingresar las correspondientes fechas en que consumió, en la figura 59 se puede observar la elección tomada, en donde se muestra los históricos y el consumo del sensor 2 para el día 27 de Julio de 2016 desde la 7:00 pm hasta las 10:00 pm.

Figura 59. Consulta consumo para el sensor 2



El formulario de consulta de consumo para el sensor 2 contiene los siguientes campos:

- Un menú desplegable con el valor "sensor2".
- Un menú desplegable con el valor "kWh".
- Un campo de texto con el valor "27/07/2016".
- Un campo de texto con el valor "19:00".
- Un campo de texto con el valor "27/07/2016".
- Un campo de texto con el valor "22:00".
- Un botón de acción etiquetado como "CONSULTAR".

Fuente: Los Autores

De esta manera el sistema realiza una búsqueda dentro de los servidores con el fin de encontrar los datos dentro del rango ingresado por el usuario. De esta manera en la figura 60 se detallan los datos que encontraría el sistema para general los históricos y el consumo total.



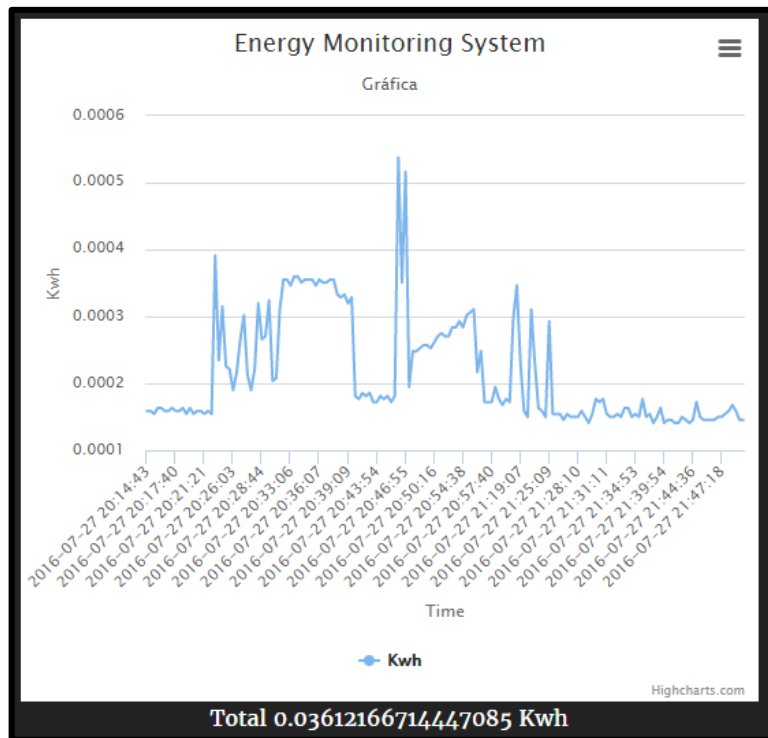
Figura 60. Datos del sensor 2 en el servidor

kwh	price	time
0.000140906	0.0566148	2016-07-27 21:44:16
0.000140906	0.0566148	2016-07-27 21:43:18
0.000140906	0.0566148	2016-07-27 21:41:16
0.000140906	0.0566148	2016-07-27 21:39:54
0.000140906	0.0566148	2016-07-27 21:37:14
0.000140906	0.0566148	2016-07-27 21:29:10
0.000145371	0.0584087	2016-07-27 21:49:37
0.000145371	0.0584087	2016-07-27 21:49:18
0.000145371	0.0584087	2016-07-27 21:46:37
0.000145371	0.0584087	2016-07-27 21:46:17
0.000145371	0.0584087	2016-07-27 21:45:57
0.000145371	0.0584087	2016-07-27 21:45:37
0.000145371	0.0584087	2016-07-27 21:44:36
0.000145371	0.0584087	2016-07-27 21:43:56

Fuente: Los autores

Teniendo así el soporte para realizar la gráfica de consumo y el cálculo de la cantidad de kWh que fueron usados en esas fechas dando como resultado los históricos de la figura 61.

Figura 61. Gráfica consulta sensor 2



Fuente: Los autores

En la figura 61 se totaliza el consumo en kWh del tiempo seleccionado obteniendo un total de 0.036121 kWh además de observar su comportamiento, permitiéndole al usuario identificar por sí mismo los horarios de consumo pico para el electrodoméstico asignado a dicho sensor. El usuario determina el consumo en ese intervalo de tiempo y puede tomar acción en ello para que mediante los algoritmos pueda generar un ahorro y obtener un beneficio de este.

En la gráfica se puede observar el eje X representa el consumo en KWh que posee un intervalo desde 0.0001 hasta 0.0006 y del mismo modo en el eje Y se encuentran las fechas seleccionadas por el usuario como lo son el 27/07/2016 siendo las 19:00 hasta el mismo 27/07/2016 a las 22:00.

### 7.3.3 Consulta consumo del sensor 3

En el momento en que el usuario desee conocer el consumo del sensor 3 debe de ingresar las correspondientes fechas en que consumió, en la figura 62 se puede observar la elección tomada, en donde se muestra los históricos y el consumo del sensor 3 para el día 26 de Julio de 2016 desde la 12:00 pm hasta las 3:00 pm.

Figura 62. Consulta consumo para el sensor 3



The image shows a web form with a dark background and white input fields. At the top, there is a dropdown menu with 'sensor3' selected. Below it is another dropdown menu with 'kWh' selected. The form contains six input fields: the first two are for the date '26/07/2016', the next two are for the start time '12:00', and the last two are for the end time '15:00'. At the bottom center, there is a button labeled 'CONSULTAR'.

Fuente: Los Autores

De esta manera el sistema realiza una búsqueda dentro de los servidores con el fin de encontrar los datos dentro del rango ingresado por el usuario. De esta manera en la figura 63 se detallan los datos que encontraría el sistema para general los históricos y el consumo total.

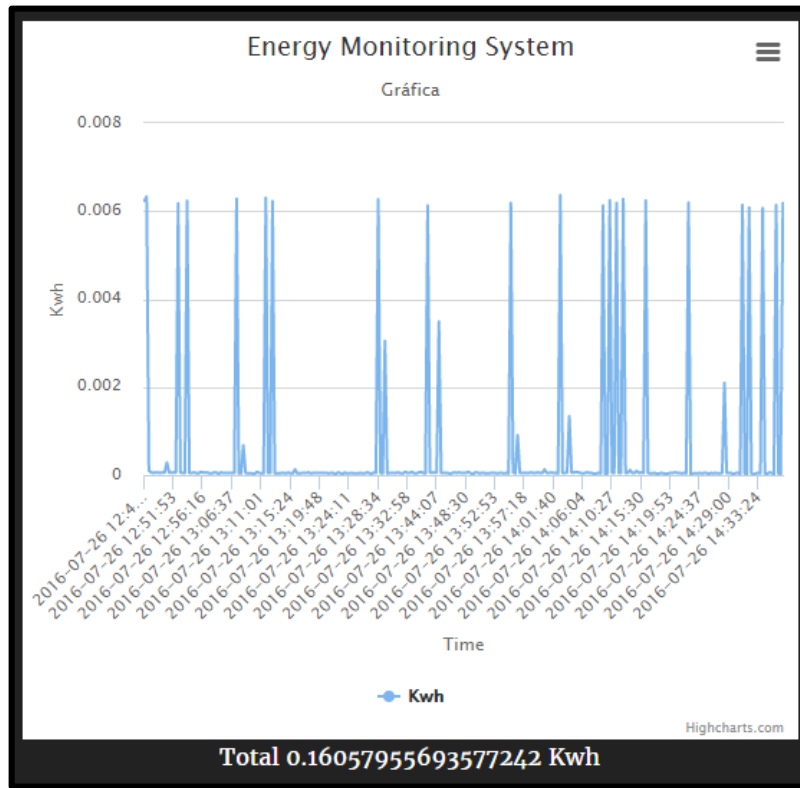
Figura 63. Datos del sensor 3 en el servidor

kwh	price	time
0.0000486541	0.0199384	2016-07-26 14:28:40
0.0000486541	0.0199384	2016-07-26 14:29:00
0.0000486541	0.0199384	2016-07-26 14:29:21
0.0000151675	0.00621562	2016-07-26 14:29:41
0.0000486541	0.0199384	2016-07-26 14:30:01
0.0000486541	0.0199384	2016-07-26 14:30:22
0.0000263297	0.0107899	2016-07-26 14:30:42
0.00613842	2.51553	2016-07-26 14:31:02
0.0000374919	0.0153642	2016-07-26 14:31:22
0.0000263297	0.0107899	2016-07-26 14:31:42
0.00607815	2.49082	2016-07-26 14:32:02
0.0000374919	0.0153642	2016-07-26 14:32:23
0.0000151675	0.00621562	2016-07-26 14:32:43

Fuente: Los autores

Teniendo así el soporte para realizar la gráfica de consumo y el cálculo de la cantidad de kWh que fueron usados en esas fechas dando como resultado los históricos de la figura 64.

Figura 64. Gráfica consulta 3



Fuente: Los autores

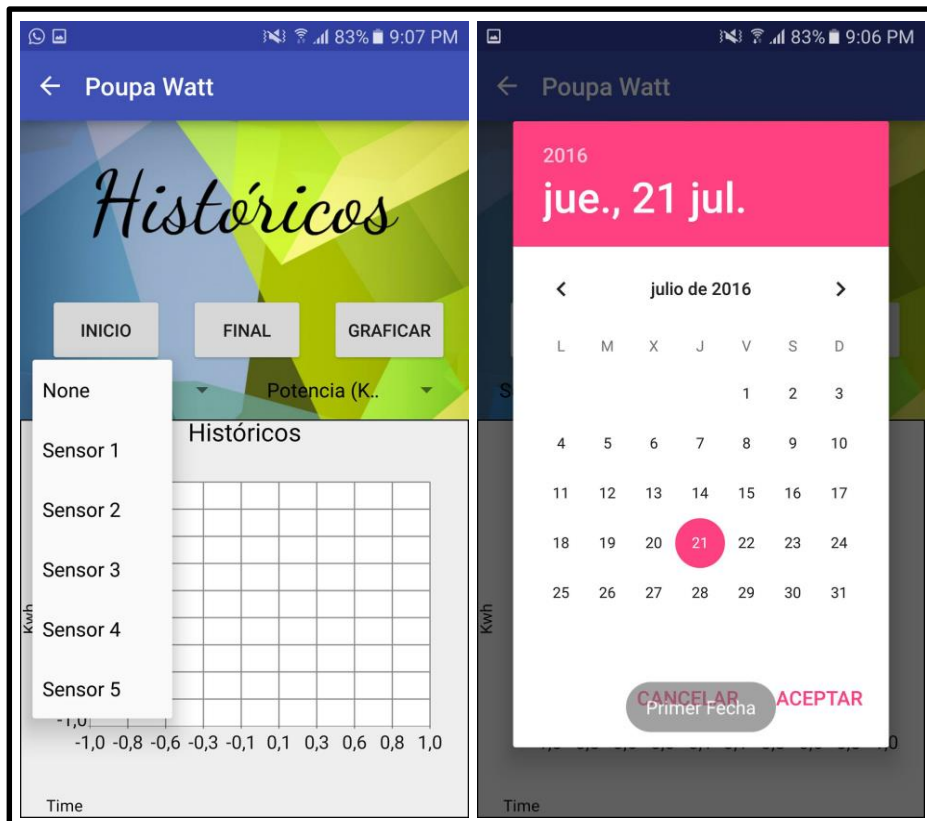
En la figura 64 se totaliza el consumo en kWh del tiempo seleccionado obteniendo un total de 0.16057 kWh además de observar su comportamiento, permitiéndole al usuario identificar por sí mismo los horarios de consumo pico para el electrodoméstico asignado a dicho sensor. El usuario determina el consumo en ese intervalo de tiempo y puede tomar acción en ello para que mediante los algoritmos pueda generar un ahorro y obtener un beneficio de este.

En la gráfica se observa el eje X representa el consumo en kWh que posee un intervalo desde 0.002 hasta 0.008 y del mismo modo en el eje Y se encuentran las fechas seleccionadas por el usuario como lo son el 26/07/2016 siendo las 12:00 hasta el mismo 26/07/2016 a las 15:00.

### 7.3.4 Consulta consumo del sensor 4

En el momento en que el usuario desee conocer el consumo del sensor 4 debe de ingresar las correspondientes fechas en que consumió, en la figura 65 se puede observar la elección tomada, en donde se muestra los históricos y el consumo del sensor 4 para el día 21 de Julio de 2016.

Figura 65. Consulta consumo para el sensor 4



Fuente: Los Autores

De esta manera el sistema realiza una búsqueda dentro de los servidores con el fin de encontrar los datos dentro del rango ingresado por el usuario. De esta manera en la figura 66 se detallan los datos que encontraría el sistema para general los históricos y el consumo total.

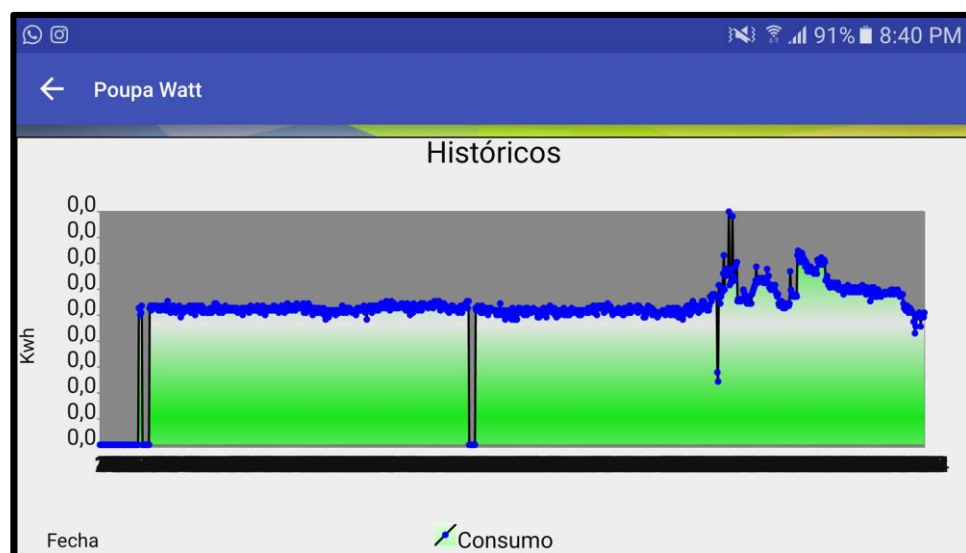
Figura 66. Datos del sensor 4 en el servidor

kwh	price	time
0.0000486541	0.0199384	2016-07-26 14:28:40
0.0000486541	0.0199384	2016-07-26 14:29:00
0.0000486541	0.0199384	2016-07-26 14:29:21
0.0000151675	0.00621562	2016-07-26 14:29:41
0.0000486541	0.0199384	2016-07-26 14:30:01
0.0000486541	0.0199384	2016-07-26 14:30:22
0.0000263297	0.0107899	2016-07-26 14:30:42
0.00613842	2.51553	2016-07-26 14:31:02
0.0000374919	0.0153642	2016-07-26 14:31:22
0.0000263297	0.0107899	2016-07-26 14:31:42
0.00607815	2.49082	2016-07-26 14:32:02
0.0000374919	0.0153642	2016-07-26 14:32:23
0.0000151675	0.00621562	2016-07-26 14:32:43

Fuente: Los autores

Teniendo así el soporte para realizar la gráfica de consumo y el cálculo de la cantidad de kWh que fueron usados en esas fechas dando como resultado los históricos de la figura 67.

Figura 67. Gráfica consulta sensor 4



Fuente: Los autores

En la figura 67 se observa el comportamiento, permitiéndole al usuario identificar por sí mismo los horarios de consumo pico para el electrodoméstico asignado a dicho sensor. En este caso para la aplicación móvil, por el tipo de layout fue complicado el hacer compactas las fechas dentro de la interfaz. Es por eso que se observa en la parte de abajo una línea negra que representa el intervalo de tiempo seleccionado. El usuario determina el consumo en ese intervalo de tiempo y puede tomar acción en ello para que mediante los algoritmos pueda generar un ahorro y obtener un beneficio de este.

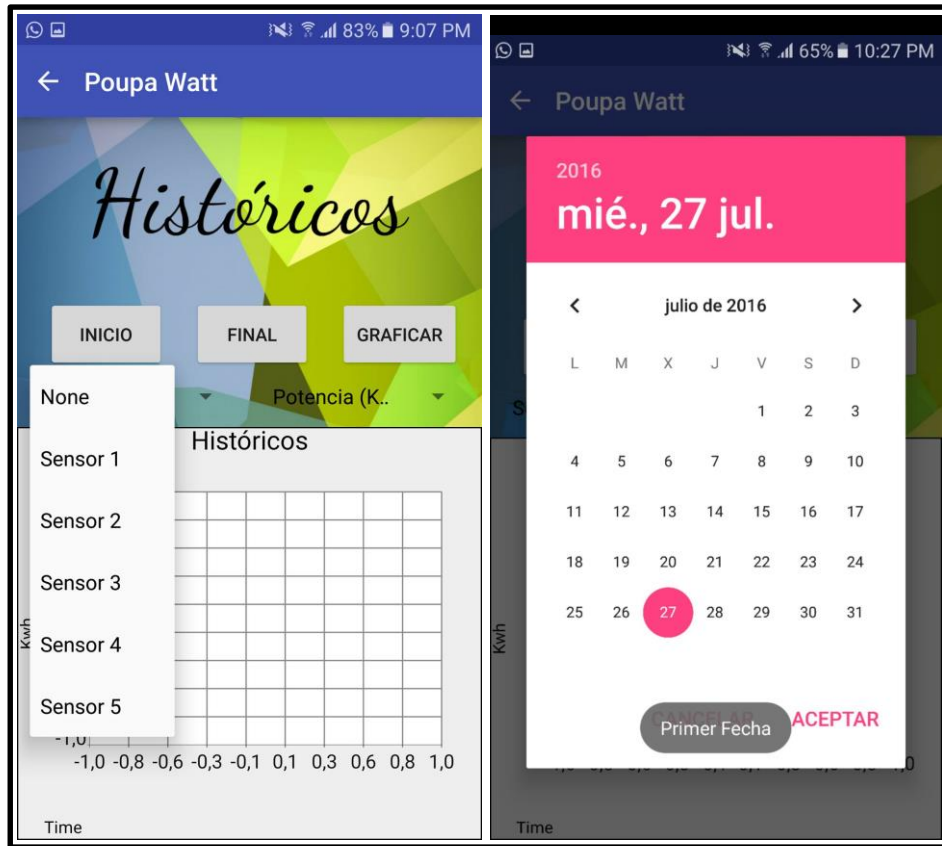
En la gráfica se observa el eje X representa el consumo en kWh que posee un intervalo que supera los 3 decimales después del cero por ende no se alcanza a detallar bien la escala, pero se representa desde 0.002 hasta 0.008 y del mismo modo en el eje Y se encuentran las fechas seleccionadas por el usuario como lo son el 21/07/2016.

### **7.3.5 Consulta consumo del sensor 5**

En el momento en que el usuario desee conocer el consumo del sensor 5 debe de ingresar las correspondientes fechas en que consumió, en la figura 68 se puede observar la elección tomada, en donde se muestra los históricos y el consumo del sensor 5 para el día 27 de Julio de 2016.



Figura 68. Consulta consumo para el sensor 5



Fuente: Los Autores

De esta manera el sistema realiza una búsqueda dentro de los servidores con el fin de encontrar los datos dentro del rango ingresado por el usuario. De esta manera en la figura 69 se detallan los datos que encontraría el sistema para general los históricos y el consumo total.

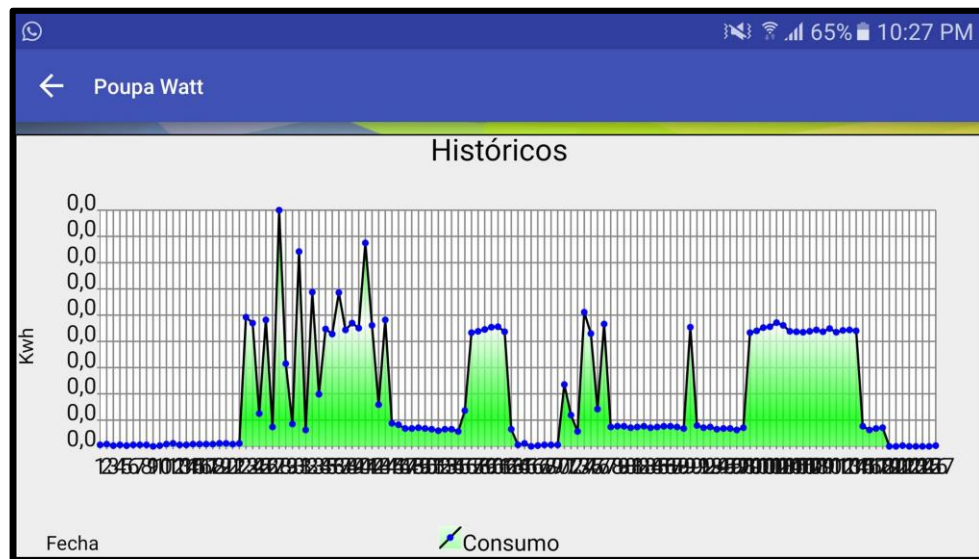
Figura 69. Datos del sensor 5 en el servidor

kwh	price	time
0.00171681	0.70355	2016-07-27 20:39:03
0.00058444	0.239503	2016-07-27 20:39:24
0.00186021	0.762315	2016-07-27 20:39:44
0.000316547	0.129721	2016-07-27 20:40:04
0.000327709	0.134295	2016-07-27 20:40:25
0.000327709	0.134295	2016-07-27 20:40:47
0.000305385	0.125147	2016-07-27 20:41:09
0.000316547	0.129721	2016-07-27 20:41:25
0.000327709	0.134295	2016-07-27 20:41:45
0.000305385	0.125147	2016-07-27 20:42:28
0.000316547	0.129721	2016-07-27 20:42:46
0.000327709	0.134295	2016-07-27 20:43:07
0.000327709	0.134295	2016-07-27 20:43:26

Fuente: Los autores

Teniendo así el soporte para realizar la gráfica de consumo y el cálculo de la cantidad de kWh que fueron usados en esas fechas dando como resultado los históricos de la figura 70.

Figura 70. Gráfica consulta sensor 5



Fuente: Los autores

En la figura 70 se observa el comportamiento, permitiéndole al usuario identificar por sí mismo los horarios de consumo pico para el electrodoméstico asignado a dicho sensor. En este caso para la aplicación móvil, por el tipo de layout fue complicado el hacer compactas las fechas dentro de la interfaz. Es por eso que se observa en la parte de abajo una línea negra que representa el intervalo de tiempo seleccionado. El usuario determina el consumo en ese intervalo de tiempo y puede tomar acción en ello para que mediante los algoritmos pueda generar un ahorro y obtener un beneficio de este.

En la gráfica se puede observar el eje X representa el consumo en KWh que posee un intervalo que supera los 3 decimales después del cero por ende no se alcanza a detallar bien la escala, pero se representa desde 0.002 hasta 0.007 y del mismo modo en el eje Y se encuentran las fechas seleccionadas por el usuario como lo son el 27/07/2016.

### 7.3.6 Respuesta del Modelo

Una vez el usuario haya visto los consumos respectivos a cada uno de los sensores que le fue asignado tiene la posibilidad de ejecutar los algoritmos los cuales le dan su planeación semanal de como consumir para generar un ahorro en ella. Para esto es necesario tomar como valor de partida el valor de la reducción que fue guardado en las preferencias, así como lo muestra la figura 71.

Figura 71. Valor de reducción nuevo caso de estudio



Fuente: Los autores

Una vez se tiene el valor de reducción y se ejecuta el modelo, este busca una manera para que el usuario obtenga un horario adecuado de uso para los electrodomésticos que han sido registrados dentro de la plataforma web. En este caso de estudio, con estos nuevos datos y teniendo como preferencia el programa de Tiempo de uso, la figura 71 muestra el resultado arrojado por el modelo con el cual el usuario puede generar un nuevo hábito de consumo.

De esta manera, este usuario debe de tener en cuenta que se le respetó los horarios pactados como preferencia que son aquellos que aparecen completos, mientras que los otros son los generados matemáticamente. Se muestra que si el usuario apaga los televisores en la noche también podría prenderlo en la tarde para tener un buen uso y empezar así a generar ahorro. También se puede ver que el modelo sugiere el hecho de planchar entre semana a las 10 a.m. es decir en la jornada mañana para obtener un mayor beneficio; la lavadora esta para el día Viernes a medio día con máximo na duración de 1 hora de funcionamiento.

De igual forma si el usuario consume con este horario o menos asegura que se está cumpliendo de reducir 3 kWh en una semana, tal cual se pactó cuando se llenaban las preferencias. El modelo busca brindarle la oportunidad al usuario de modificar sus hábitos bajo sus preferencias con el fin de generar un ahorro económico para ellos y sus familias.

Figura 72. Resultados nuevo caso de estudio

	Día	Electrodomestico	Hora inicio	Hora fin
1	Lunes	Televisor	18:00:00	20:00:00
1	Lunes	Televisor	20	23
1	Lunes	Bombilla	19	22
2	Martes	Televisor	18:00:00	22:00:00
2	Martes	Televisor	15	17
2	Martes	Bombilla	18	23
2	Martes	Plancha	10	11
3	Miercoles	Televisor	20	23
3	Miercoles	Bombilla	21	23
4	Jueves	Televisor	21	23
4	Jueves	Bombilla	18	19
5	Viernes	Televisor	15	16
5	Viernes	Bombilla	21	23
5	Viernes	Lavadora	12	13
6	Sabado	Televisor	21	23
6	Sabado	Bombilla	21	22
7	Domingo	Televisor	17	20
7	Domingo	Bombilla	21	22

Fuente: Los autores

## 8. CONCLUSIONES

- Se desarrolló un modelo computacional orientado al usuario con el fin de que este tenga un uso responsable y reducido de energía eléctrica según un programa de respuesta a la demanda, basado en modelos educativos para que tenga un manejo fácil pero óptimo.
- El parámetro de entrada del sistema son los kWh sensados por los Smart Meters ya que este parámetro hace parte de los datos recolectados para facturación sin importar la región, de modo que hace funcional al sistema sin importar las demás condiciones.
- Para el manejo de la información del modelo se diseñaron dos bases de datos, de manera que la recolección de datos fuera aislada de la aplicación de la misma, una vez recolectados los modelos funcionan sin ser afectados por el usuario permitiéndole configurar unos parámetros base desde la segunda base de datos para la ejecución personal de los mismos.
- Con el fin de garantizar una HCI amigable, la aplicación móvil y la aplicación web fueron diseñadas de manera que la navegación por las mismas fuese de manera intuitiva y fácil de entender.
- El sistema operativo Android fue seleccionado como plataforma para la primera versión de la aplicación ya que en el mercado de los dispositivos móviles posee una alta demanda.
- La aplicación web se diseñó como una “responsive web” para que todos los usuarios tengan acceso a esta sin importar el dispositivo desde el cual se conecten.
- El algoritmo diseñado para el proyecto de la VRIT no se implementó dentro del sistema computacional Poupa Watt debido a la incompatibilidad de softwares utilizados, como solución a esto se diseñó un algoritmo que le permite al usuario tener diferentes opciones con el fin de generar una reducción en su consumo.

- Para dar un buen manejo del sistema en los programas de respuesta a la demanda, se propone para trabajos futuros involucrar como un nuevo actor a la empresa prestadora del servicio.
- Los algoritmos le permiten al usuario realizar una planeación semanal para el uso correcto de los electrodomésticos con el fin de generar un ahorro aplicando cualquiera de los dos tipos de programa a la respuesta de la demanda de energía utilizados dentro de la aplicación. Teniendo así la posibilidad de cambiar sus hábitos y de esta manera evidenciar los resultados en la facturación mensual del hogar.
- La interfaz que posee el sistema es amigable con el usuario y permite que navegue por todas las pestañas utilizando el 100% de la aplicación y de la plataforma. Teniendo como soporte una manual de usuario para que aquel que tenga alguna duda o no logre entender alguna sección pueda aclararla.
- En conclusión Poupa Watt permite que un usuario consumidor de energía eléctrica pueda registrarse y conocer dos de los programas de la respuesta de energía. En donde puede consultar su consumo y a la vez generar la gráfica de los históricos. El juego que posee tanto la plataforma web como la aplicación móvil le permite al usuario interactuar con los electrodomésticos y las horas de operación con el fin de buscar un ahorro en el consumo.

## 9. BIBLIOGRAFÍA

- Ministro de Educación Nacional de Colombia. (24 de Julio de 2002). *Ministro de Educación Nacional de Colombia*. Obtenido de [http://www.mineducacion.gov.co/1621/articles-86088\\_archivo\\_pdf.pdf](http://www.mineducacion.gov.co/1621/articles-86088_archivo_pdf.pdf)
- Adobe Systems Incorporated and its licensors. (2015). *Adobe Support*. Obtenido de <https://helpx.adobe.com/es/dreamweaver/using/web-applications.html>
- Alvarez, M. A. (01 de Mayo de 2001). *Desarrollo Web*. Obtenido de <http://www.desarrolloweb.com/articulos/392.php>
- Alvarez, M. A. (22 de Enero de 2015). *Desarrolloweb.com*. Obtenido de <http://www.desarrolloweb.com/articulos/algoritmo-caracteristicas-ejemplos.html>
- Anguiano Morales, J. D. (30 de Junio de 2014). *IBM developerWorks*. Obtenido de Características y tipos de bases de datos: [https://www.ibm.com/developerworks/ssa/data/library/tipos\\_bases\\_de\\_datos/](https://www.ibm.com/developerworks/ssa/data/library/tipos_bases_de_datos/)
- Aranaz Tudela , J. (Enero de 2009). *Universidad Carlos III de Madrid*. Obtenido de e-Archivo: [http://e-archivo.uc3m.es/bitstream/handle/10016/6506/PFC\\_Jaime\\_Aranaz\\_Tudela\\_2010116132629.pdf?sequence=1&isAllowed=y](http://e-archivo.uc3m.es/bitstream/handle/10016/6506/PFC_Jaime_Aranaz_Tudela_2010116132629.pdf?sequence=1&isAllowed=y)
- Arévalo, O. E., Linares, S. P., Correa, L. F., Olarte Parra, J., & Gonzalez, H. A. (09 de Junio de 2009). *Departamento de Ingeniería de Sistemas e Industrial*. Obtenido de Página del Ing. Ismael Castañeda Fuentes: [dis.unal.edu.co/~icasta/consejero/Requerimientos\\_Norma\\_ieee830.doc](http://dis.unal.edu.co/~icasta/consejero/Requerimientos_Norma_ieee830.doc)
- Belmonte Fernández, O. (06 de Junio de 2005). *Universitat Jaume I*. Obtenido de Introducción a la Simulación Informática: <http://www3.uji.es/~belfern/pdidoc/IX26/Documentos/introJava.pdf>
- Berzal Galiano , F. (2016). *elvex*. Obtenido de <http://elvex.ugr.es/idbis/db/docs/intro/B%20Bases%20de%20Datos.pdf>
- Bianchini, A. (24 de Septiembre de 2015). *Metodología para el desarrollo de aplicaciones educativas en ambientes multimedios*. Obtenido de Universidad Simón Bolívar: <http://ldc.usb.ve/~abianc/Adelaide/mmm.pdf>
- Bioengineering, N. I. (07 de Septiembre de 2015). *National Institute of Biomedical Imaging and Bioengineering*. Obtenido de <http://www.nibib.nih.gov/espanol/temas-cientificos/modelado-computacional>
- California Public Utilities Commission and California Energy Commission. (2014). *Energy UPgrade California*. Obtenido de <http://www.energyupgradeca.org/sp/learn/energy-basics/what-is-demand-response>
- Carrillo , A. (Noviembre de 2009). *Grupo de Geofísica Computacional*. Obtenido de <http://mmc.geofisica.unam.mx/acl/Archivos/Conferencias/091107/Presentacion.pdf>
- Casillas Santillán, L. A., Ginestà, M. G., & Pérez Mora, Ó. (2007). *Universitat Oberta de Catalunya*. Obtenido de Databases: <http://ocw.uoc.edu/computer->



- science-technology-and-multimedia/bases-de-datos/bases-de-datos/P06\_M2109\_02151.pdf
- Comercio, C. F. (Septiembre de 2011). *Información para Consumidor*. Obtenido de <https://www.consumidor.ftc.gov/articulos/s0018-aplicaciones-moviles-que-son-y-como-funcionan>
- Conejo, A. J., Morales, J. M., & Baringo, L. (2010). Real-Time Demand Response Model. *Smart Grid, IEEE Transactions on (Volume:1 , Issue: 3 )*, 236 - 242.
- Congreso de Colombia. (05 de Enero de 2009). *Ley 1273 Protección de Información*. Obtenido de <http://www.alcaldiabogota.gov.co/sisjur/normas/Norma1.jsp?i=34492>
- Congreso de Colombia. (17 de Octubre de 2012). *Ley Estatutaria 1581*. Obtenido de <http://wsp.presidencia.gov.co/Normativa/Leyes/Documents/LEY%201581%20DEL%2017%20DE%20OCTUBRE%20DE%202012.pdf>
- Congreso de la Republica. (30 de Julio de 2009). *Ley 1341 Diario Oficial*. Obtenido de <http://registrotic.mintic.gov.co:9090/registrotic/media/LEY%201341%20DEL%2030%20DE%20JULIO%20DE%202009.pdf?phpMyAdmin=72f735a548e7be308271189719fff242>
- erenovable. (18 de Agosto de 2015). *Erenovable*. Obtenido de <http://erenovable.com/medicion-del-consumo-electrico/>
- Escofet, C. M. (2007). *Universitat Oberta de Catalunya*. Obtenido de Databases: [http://ocw.uoc.edu/computer-science-technology-and-multimedia/bases-de-datos/bases-de-datos/P06\\_M2109\\_02149.pdf](http://ocw.uoc.edu/computer-science-technology-and-multimedia/bases-de-datos/bases-de-datos/P06_M2109_02149.pdf)
- Facultad de Ciencias Exactas y Tecnología - Universidad Nacional de Tucumán. (19 de Octubre de 2014). *Facet*. Obtenido de <http://catedras.facet.unt.edu.ar/centraleselectricas/wp-content/uploads/sites/19/2014/10/Curvas-de-Carga-y-Generaci%C3%B3n.pdf>
- Faria, P., & Zita, V. (2013). Demand Response Programs Definition Using Demand Price Elasticity to Define Consumers Aggregation for an Improved Remuneration Structure. *Innovative Smart Grid Technologies Europe (ISGT EUROPE), 2013 4th IEEE/PES*, 1-5.
- Gámiz Juárez, M. d. (2014). *ESTIMACIÓN DE CURVA DE CONSUMO ELÉCTRICO A FUTURO DE AGREGACIONES DE SUMINISTROS ELÉCTRICOS*. UNIVERSIDAD DE SEVILLA: ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA. Obtenido de [http://bibing.us.es/proyectos/abreproy/90059/fichero/TFG\\_80154348S.pdf](http://bibing.us.es/proyectos/abreproy/90059/fichero/TFG_80154348S.pdf)
- GCF Global. (2015). *GCFGlobal*. Obtenido de GCFAprendeLibre.org: [http://www.gcfaprendelibre.org/tecnologia/curso/informatica\\_basica/aplicaciones\\_web\\_y\\_todo\\_acerca\\_de\\_la\\_nube/2.do](http://www.gcfaprendelibre.org/tecnologia/curso/informatica_basica/aplicaciones_web_y_todo_acerca_de_la_nube/2.do)
- González Gutiérrez, E. (2006). *Aprender a programar*. Obtenido de [http://aprenderaprogramar.com/index.php?option=com\\_content&view=article&id=435:ique-es-y-para-que-sirve-html-el-lenguaje-mas-importante-para-](http://aprenderaprogramar.com/index.php?option=com_content&view=article&id=435:ique-es-y-para-que-sirve-html-el-lenguaje-mas-importante-para-)

crear-paginas-webs-html-tags-cu00704b&catid=69:tutorial-basico-programador-web-html-desde-cero&Itemid=192

Google Developers. (16 de Mayo de 2013). Google I/O 2013 - What's New in Android Developer Tools. Mountain View, California , Estados Unidos.

GUERRERO, G. L., ERAZO, L. R., MIRANDA, P. F., & ANTE, W. A. (2011). *Universidad del Cauca*. Obtenido de <http://artemisa.unicauca.edu.co/~leydierazo/ProyectoSW/ProcesoDeDesarrollo.pdf>

Hernández Orallo, E. (26 de Octubre de 2002). *Universitat Politècnica de València*. Obtenido de Página personal Enrique Hernández Orallo: <http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>

ICONTEC. (22 de Diciembre de 2005). *CEICMO*. Obtenido de [http://www.ceicmo.com/resources/documents/NTC\\_ISO\\_9000-2005.pdf](http://www.ceicmo.com/resources/documents/NTC_ISO_9000-2005.pdf)

ICONTEC. (2008). *Norma Técnica Colombiana 1486*. Obtenido de [http://www.unipamplona.edu.co/unipamplona/portallG/home\\_15/recursos/01\\_general/09062014/n\\_icontec.pdf](http://www.unipamplona.edu.co/unipamplona/portallG/home_15/recursos/01_general/09062014/n_icontec.pdf)

IEEE Computer Society. (25 de Junio de 1998). *Departamento de Ingeniería de Sistemas e Industrial*. Obtenido de [http://dis.unal.edu.co/~icasta/consejero/Requerimientos\\_Norma\\_ieee830.pdf](http://dis.unal.edu.co/~icasta/consejero/Requerimientos_Norma_ieee830.pdf)

Informatica, Universidad Nacional del Nordeste. (09 de Mayo de 2006). *Facultad de Ingenieria*. Obtenido de Universidad Nacional del Nordeste: [http://ing.unne.edu.ar/pub/informatica/Alg\\_diag.pdf](http://ing.unne.edu.ar/pub/informatica/Alg_diag.pdf)

JamieKeene. (20 de Abril de 2012). *The Verge*. Obtenido de Unity Technologies marks one million developers for its game development tools: <http://www.theverge.com/2012/4/10/2938040/unity-game-tools-one-million-registered-developers>

Jewell, W. (2014). The Effects of Residential Energy Efficiency on Electric Demand Response Programs . *System Sciences (HICSS), 2014 47th Hawaii International Conference on* , 2363 - 2372 .

Lancheros Cuesta, D. J., & Pérez Lara, D. A. (2014). *Sistema de monitoreo para el consumo de energía eléctrica en el hogar*. Bogotá: Trabajo de pregrado Unisalle.

Li, N., Chen, L., & Low, S. H. (2011). Optimal Demand Response Based on Utility Maximization in Power Networks. *Power and Energy Society General Meeting, 2011 IEEE*, 1-8.

Mantilla, M. C., Camargo Ariza , L. L., & Medina Delgado , B. (2013). *Metodología para el desarrollo de aplicaciones móviles*. Magdalena: Universidad de Magdalena.

Marco de Desarrollo de la Junta de Andalucía. (29 de Octubre de 2015). *Marco de Desarrollo de la Junta de Andalucía*. Obtenido de Guía para la redacción de casos de uso.: <http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/416>

- Martinez, V. J., & Rudnick, H. (2012). Design of demand response programs in emerging countries. *Power System Technology (POWERCON), 2012 IEEE International Conference on*, 1-6.
- Marulanda García, G. A. (2014). *Impacto de un programa de respuesta de la demanda eléctrica en el sector de gas natural*. Universidad Tecnológica de Pereira, Facultad de Ingenierías . Pereira: Tesis Magister.
- Marulanda, G. A., & Lancheros Cuesta, D. J. (2015). *Construcción de un sistema de medición inteligente de energía eléctrica de bajo costo orientado a programas residenciales de respuesta de la demanda para usuarios de bajos recursos*. Bogota: Proyecto de investigación de la universidad de La Salle.
- Master Magazine. (2016). *Master Magazine*. Obtenido de <http://www.mastermagazine.info/termino/3874.php>
- McGraw-Hill Education. (2014). *AIS in the Business World*. McGraw-Hill Education.
- Mercado, I. (20 de Junio de 2014). *Community Managers Latinoamérica*. Obtenido de <http://communitymanagerslatam.com/que-es-una-aplicacion-movil-y-para-que-sirve/>
- Microsoft. (2007). *Office*. Obtenido de Soporte : <https://support.office.com/es-es/article/Conceptos-b%C3%A1sicos-sobre-bases-de-datos-a849ac16-07c7-4a31-9948-3c8c94a7c204>
- Ming-Shiou, K., & Chuang, T.-Y. (2016). How gamification motivates visits and engagement for online academic dissemination – An empirical study. *Computers in Human Behavior, Volume 55*, 16-27.
- Moreno Boiza, V. (Octubre de 2012). *Universidad Carlos III de Madrid*. Obtenido de e-Archivo Principal: <http://e-archivo.uc3m.es/bitstream/handle/10016/16046/PFCVanesaMorenoBoiza.pdf?sequence=1>
- Palensky, P., & Dietrich, D. (2011). Demand Side Management: Demand Response, Intelligent Energy Systems, and Smart Loads. *Industrial Informatics, IEEE Transactions on (Volume:7 , Issue: 3 )*, 381 - 388.
- Palomino, J. R. (8 de Abril de 2014). *SlideShare*. Obtenido de <http://es.slideshare.net/jrpalomino/sg-demand-response>
- Pérez Lara, D. A. (2015). *MONITORING SYSTEM FOR THE CONSUMPTION OF ELECTRIC ENERGY IN A HOUSEHOLD*. Bogotá D.C.
- Pérez, D. A. (2014). *Monitoring system for the consumption of electric energy in a household*. Bogotá.
- Popkin Software and Systems. (17 de Octubre de 2002). *TLDP-ES*. Obtenido de <http://es.tldp.org/Tutoriales/doc-modelado-sistemas-UML/doc-modelado-sistemas-uml.pdf>
- Presidencia de la Republica. (21 de Junio de 2013). *Decreto 1377 Ministerio de Comercio, Industria y Turismo*. Obtenido de <http://wsp.presidencia.gov.co/Normativa/Decretos/2013/Documents/JUNIO/27/DECRETO%201377%20DEL%2027%20DE%20JUNIO%20DE%202013.pdf>

- Sierra, M. (2006). *Aprende a programar*. Obtenido de [http://www.aprenderaprogramar.com/index.php?option=com\\_attachments&task=download&id=500](http://www.aprenderaprogramar.com/index.php?option=com_attachments&task=download&id=500)
- Sosa, O. (2014). *Tzec*. Obtenido de <http://tzec.com/files/capitulo-diseno-visual-para-ux.pdf>
- Sublime HQ Pty Ltd. (09 de Febrero de 2016). *Sublime Text*. Obtenido de <http://docs.sublimetext.info/en/latest/intro.html>
- Talent, L. (20 de Febrero de 2013). *Lance Talent Blog*. Obtenido de <http://www.lancetalent.com/blog/tipos-de-aplicaciones-moviles-ventajas-inconvenientes/>
- The World Bank. (14 de 07 de 2014). Obtenido de <http://wdi.worldbank.org/table/3.8>
- Torres, M. (2015). *Indalog*. Obtenido de <http://indalog.ual.es/mtorres/LP/FundamentosDiseno.pdf>
- TXU Energy . (07 de Septiembre de 2015). *TXU Energy* . Obtenido de <https://www.txu.com/es/property-management/customer-care/understanding-demand.aspx>
- Unidad de Planeamiento Minero Energética. (2016). *PROYECCIÓN DE LA DEMANDA DE ENERGÍA ELÉCTRICA Y POTENCIA MÁXIMA EN COLOMBIA*. Colombia: Ministerio de Minas y Energía. Obtenido de [http://www.siel.gov.co/siel/documentos/documentacion/Demanda/UPME\\_Proyeccion\\_Demanda\\_Energia\\_Electrica\\_Junio\\_2016.pdf](http://www.siel.gov.co/siel/documentos/documentacion/Demanda/UPME_Proyeccion_Demanda_Energia_Electrica_Junio_2016.pdf)
- Universidad del Cauca. (2015). *Facultad de Ciencias Contables, Económicas y Administrativas. Unicauca*. Obtenido de <http://fccea.unicauca.edu.co/old/siconceptosbasicos.htm>
- Valencia, M. E. (17 de Septiembre de 2015). *Universidad del Valle / Cali, Colombia*. Obtenido de <http://eisc.univalle.edu.co/materias/multimedia/material/metdesaplichm.pdf>

## 10. ANEXO A

### Manual de usuario Aplicación Web

Figura A1. Logo aplicación web.



Fuente: Los autores.

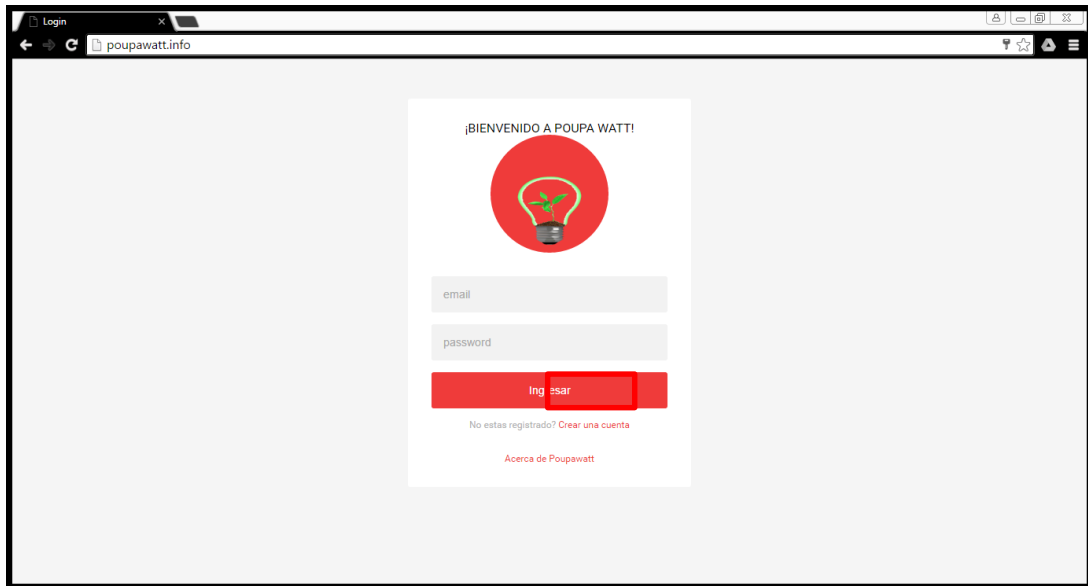
#### **A1. Descripción de la aplicación web**

Poupawattt es una aplicación web, a la cual se accede por medio del dominio [poupawatt.info](http://poupawatt.info) para los navegadores Google Chrome 3.18 o superior y Mozilla Firefox 31 o superior. Permite al usuario consultar noticias relevantes respecto al consumo responsable de energía, crear un perfil el cual incluye su información personal, la cual la cual puede modificar en el momento que desee; también le permite al usuario seleccionar sus preferencias en cuanto a programa, el día, el electrodoméstico, hora de inicio, hora de finalización, reducción y costo por el que se desea llevar la reducción; le permite al usuario llevar un control de sus facturas, al igual que de su consumo, presenta diferentes modelos los cuales arrojan unos horarios de operación para reducir el consumo y por último el usuario encontrara en el cual podrá una y otra vez cuadrar sus horarios para reducir el consumo.

#### **A2. Navegación por la aplicación.**

## A.2.1. LOGIN.

Figura A2. Pantalla Login.

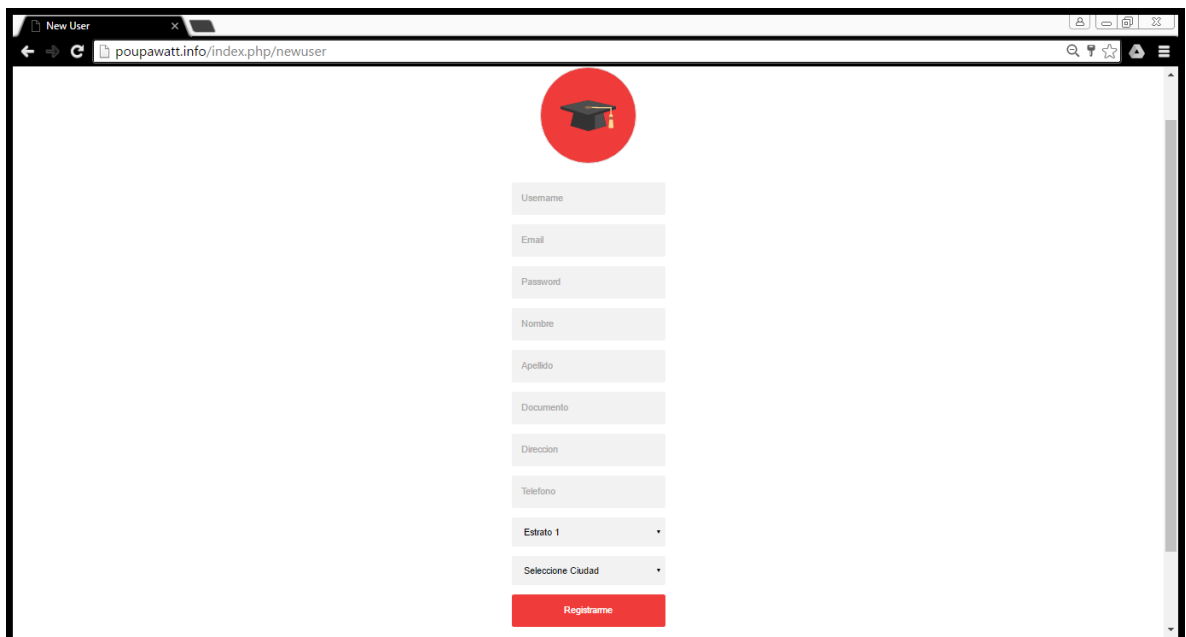


Fuente: Los autores.

Al momento de ingresar a la página web Poupawatt.info lo primero que aparece es el login para hacer uso del sistema, para esto se debe introducir los datos correspondientes para acceder. Si el usuario no posee una cuenta, este cuenta con la posibilidad de crear una, dirigiéndose al enlace “Crear una cuenta” que lo llevara a una nueva ventana.

## A.2.2. CREAR NUEVA CUENTA

Figura A3. Pantalla crear cuenta.

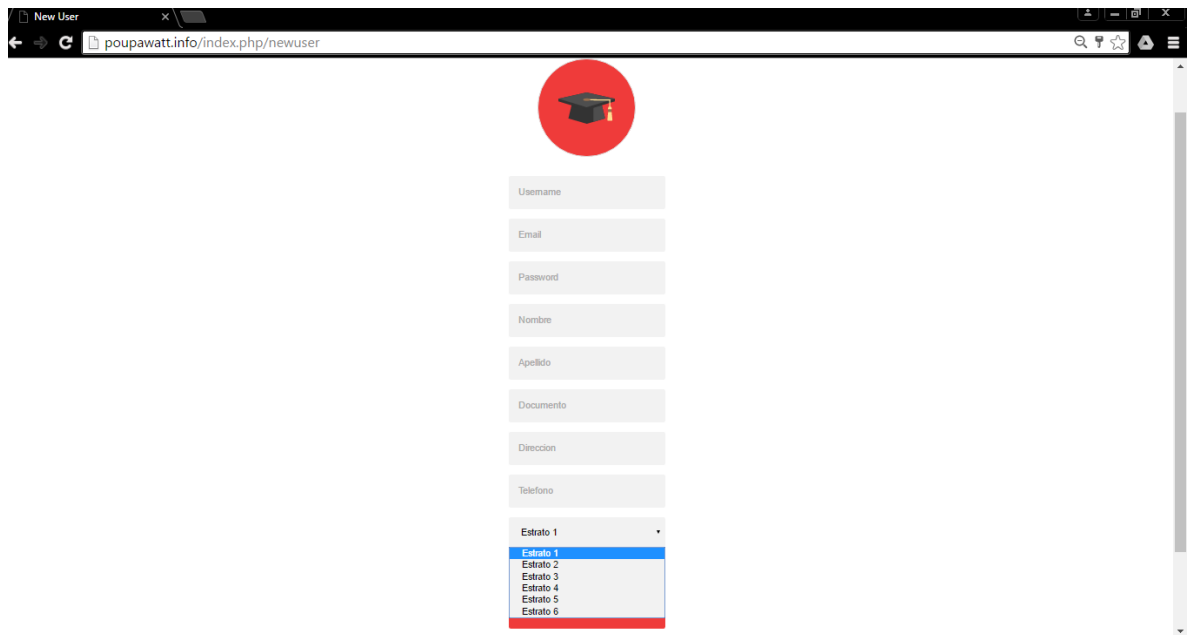


The image shows a web browser window with the address bar displaying 'poupawatt.info/index.php/newuser'. The page content features a red circular logo with a graduation cap icon at the top center. Below the logo is a registration form with the following fields: Username, Email, Password, Nombre, Apellido, Documento, Direccion, Telefono, Estrato 1 (a dropdown menu), and Seleccione Ciudad (a dropdown menu). At the bottom of the form is a red button labeled 'Regístrate'.

Fuente: Los autores.

Para crear una nueva cuenta se requiere que el usuario llene el formulario con su datos principales (nombre, apellido, documento, entre otros...) para que estos sean registrados en la base de datos de la aplicación web. Una vez que el usuario completo los datos que se le solicitan, debe dar clic en registrarse. Al momento de registrar sus datos, debe tener en cuenta la selección del estrato.

Figura A4. Pantalla Crear nueva cuenta 2.



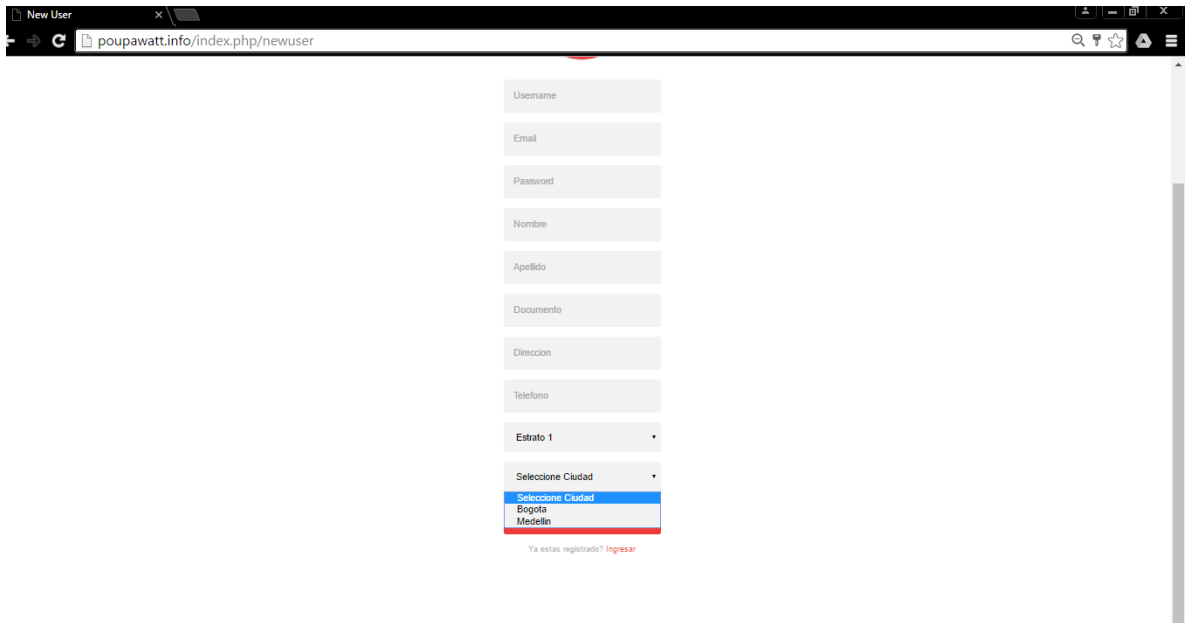
The screenshot displays a web browser window with the title 'New User' and the address bar containing 'poupawatt.info/index.php/newuser'. The main content area features a registration form with a red circular icon of a graduation cap at the top. Below the icon are several input fields: Username, Email, Password, Nombre, Apellido, Documento, Direccion, and Telefono. At the bottom of the form is a dropdown menu labeled 'Estrato' with a downward arrow. The dropdown is open, showing a list of options: Estrato 1 (highlighted in blue), Estrato 2, Estrato 3, Estrato 4, Estrato 5, and Estrato 6.

Fuente: Los autores.

AL igual de la selección de la ciudad debido a que de esta se desplegaran las localidades.



Figura A5. Pantalla Crear nueva cuenta 3.



The image shows a web browser window with the address bar displaying "poupawatt.info/index.php/newuser". The page contains a registration form with the following fields: Username, Email, Password, Nombre, Apellido, Documento, Direccion, Telefono, Estrato 1 (with a dropdown arrow), and Seleccione Ciudad (with a dropdown arrow). The "Seleccione Ciudad" dropdown menu is open, showing two options: "Elegota" and "Medellin". Below the form, there is a link that says "Ya estas registrado? Ingresar".

Fuente: Los autores.

Al seleccionar la localidad a la cual pertenece, se desplegara la lista de los barrios que la conforman.

Figura A6. Pantalla Crear nueva cuenta 4.

Username

Email

Password

Nombre

Apellido

Documento

Direccion

Telefono

Estrato 4

Bogota

Seleccione

Seleccione

Usaquen

Chapinero

Regístrate

Ya estas registrado? [Ingresar](#)

Fuente: Los autores.

Finalmente selecciona el barrio al cual pertenece, para completa su registro.

Figura A7. Pantalla crear nueva cuenta 5.

Username

Email

Password

Nombre

Apellido

Documento

Direccion

Telefono

Estrato 4

Bogota

Chapinero

Seleccione

Seleccione

Chico Reservado

Bellavista

Regístrate

Ya estas registrado? [Ingresar](#)

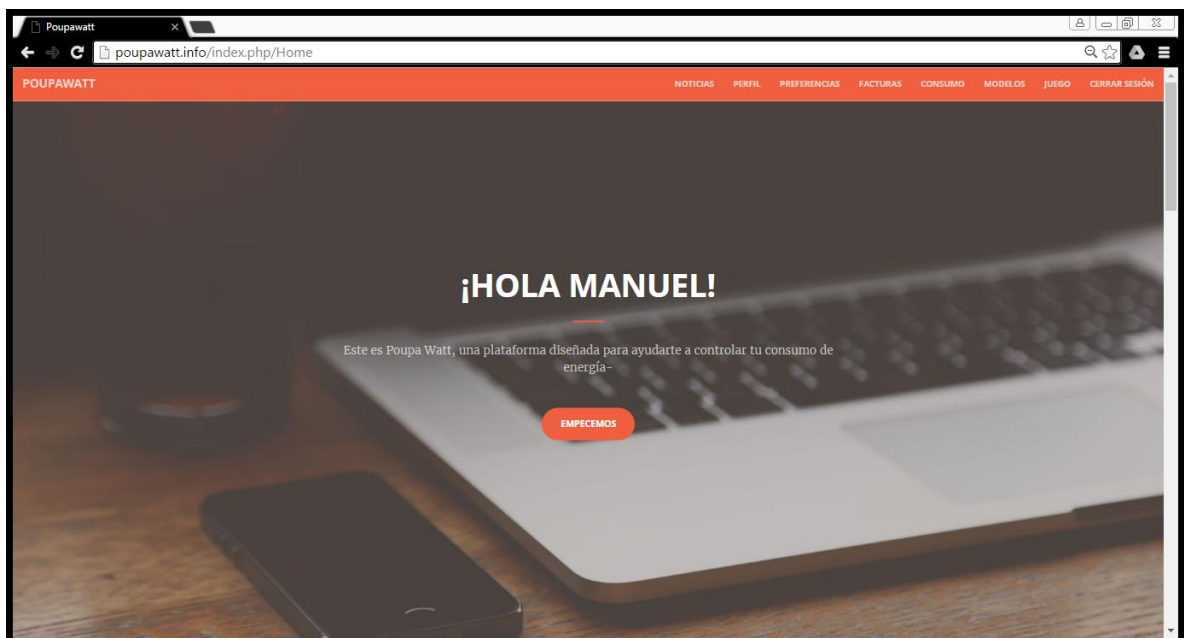
Fuente: Los autores.

Al momento de completar el registro, se puede acceder a la aplicación web e interactuar con ella.

### A.2.3. HOME

Después de completar el registro y acceder a la aplicación web, la aplicación nos redirige a la ventana principal (Home), donde el usuario podrá interactuar con las diferentes opciones que esta cuenta. Una vez en el Home, el usuario tiene la posibilidad de seleccionar una de las 7 opciones que allí se encuentran planteadas que son: Noticias, Perfil, Preferencias, Facturas, Consumo, Modelos y Juego.

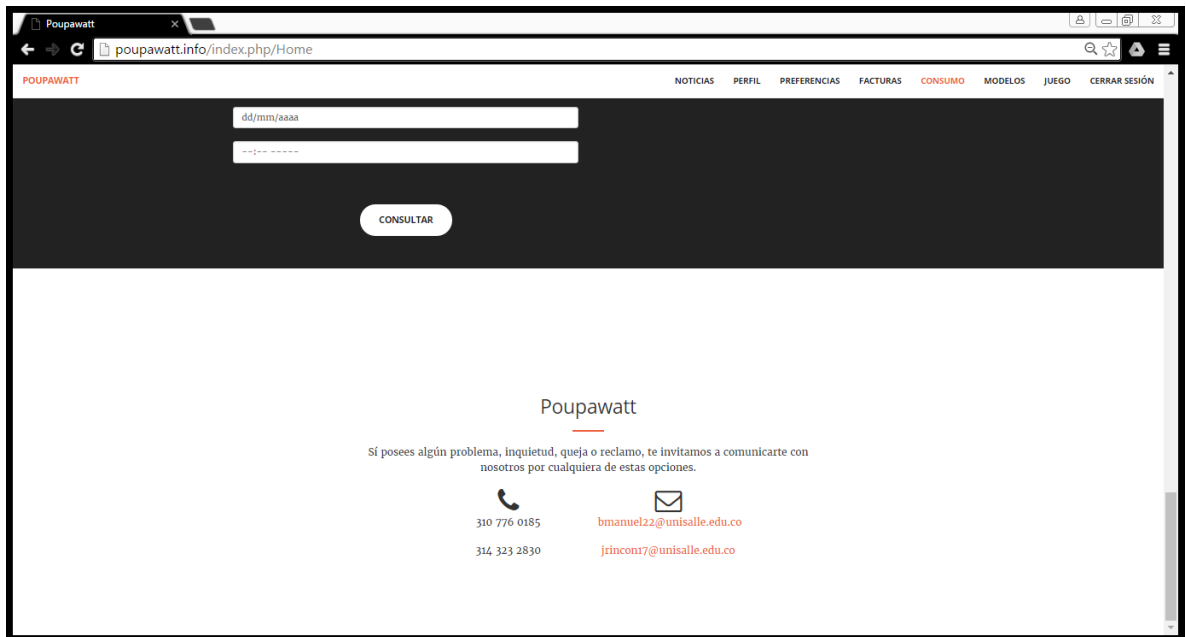
Figura A8. Pantalla Home.



Fuente: Los autores.

Cada una de estas opciones contiene información para el usuario. Al igual el home cuenta con información de contacto en caso de algún problema o inquietud que se presente.

Figura A9. Pantalla home contacto.



#### A.2.4. NOTICIAS

Es una pequeña sección con la que cuenta la aplicación web, donde se muestran noticias como la del proyecto que implanto el estado para generar una reducción en el consumo de energía eléctrica en los hogares colombianos, el cual se encuentra muy cercano a los objetivos de la aplicación.

Figura A10. Pantalla Noticias.



Fuente: Los autores.

### A.2.5. PERFIL

En esta opción el usuario tiene la posibilidad de modificar todos sus datos personales en el momento que quiera.

Figura A11. Pantalla perfil.

POUPAWATT

PREFERENCIAS FACTURAS CONSUMO MODELOS JUEGO CERRAR SESIÓN

### Perfil

[INSTRUCCIONES >>](#)

Username:	Manuel	Teléfono:	777
Email:	manuelbrdi@hotmail.com	Estrato:	Estrato 1
Nombre:	Manuel	Ciudad:	Bogota
Apellido:	Bernal	Localidad:	Usaquen
Documento:	1075671730	Barrio:	Canima
Dirección:	kk		

[MODIFICAR](#) [CAMBIAR CONTRASEÑA](#)

Fuente: Los autores.

Para realizar cualquier modificación, debe dar clic en “Modificar” lo cual le dará el acceso para hacer los cambios que desee.

Figura A12. Pantalla Perfil 2.

POUPAWATT

PREFERENCIAS FACTURAS CONSUMO MODELOS JUEGO CERRAR SESIÓN

### Perfil

[INSTRUCCIONES >>](#)

Username:	Manuel	Teléfono:	777
Email:	manuelbrdi@hotmail.com	Estrato:	Estrato 1
Nombre:	Manuel	Ciudad:	Bogota
Apellido:	Bernal	Localidad:	Usaquen
Documento:	1075671730	Barrio:	Canima
Dirección:	kk		

[CONFIRMAR](#) [CANCELAR](#)

Fuente: Los autores.

Al momento de estar conforme con los cambios registrados, debe dar clic en “confirmar”.

Esta opción de perfil, también presenta la posibilidad de modificar la contraseña para esto debe dar clic en “Cambiar contraseña”.

Figura A13. Pantalla perfil 3.

POUPAWATT

PREFERENCIAS FACTURAS CONSUMO MODELOS JUEGO CERRAR SESIÓN

### Perfil

[INSTRUCCIONES >>](#)

Username:	Manuel	Teléfono:	777
Email:	manuelbrdi@hotmail.com	Estrato:	Estrato 1
Nombre:	Manuel	Ciudad:	Bogota
Apellido:	Bernal	Localidad:	Usaquen
Documento:	1075671730	Barrio:	Canima
Dirección:	kk		

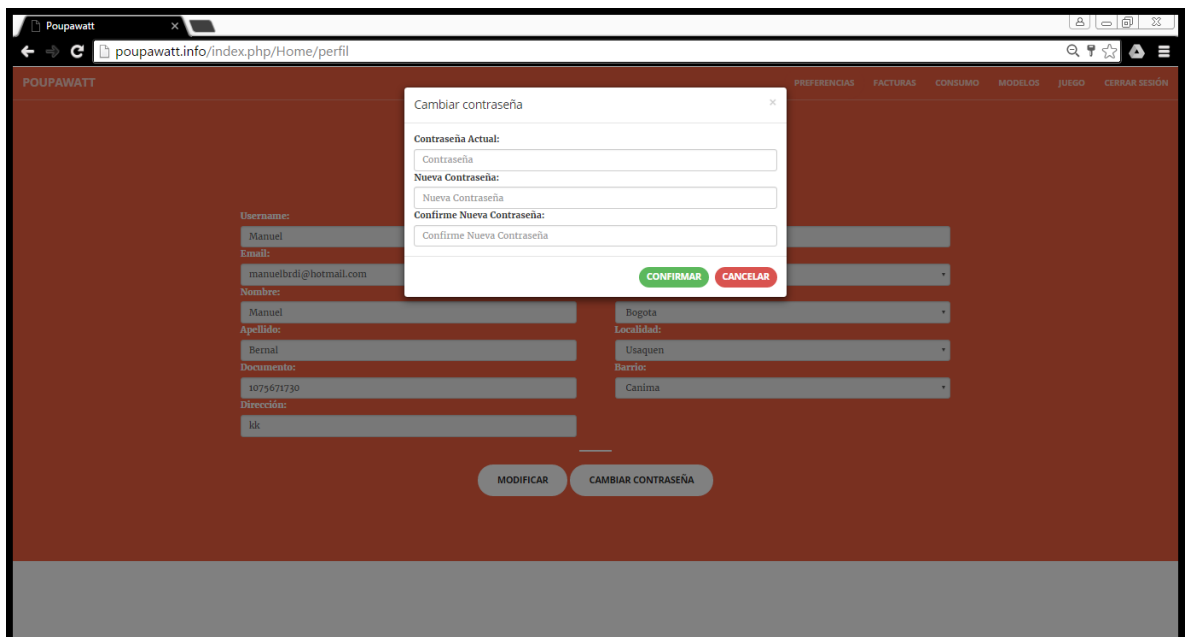
[MODIFICAR](#) [CAMBIAR CONTRASEÑA](#)

Fuente: Los autores.

Al acceder a este enlace, nos presenta una ventana emergente en la cual puede hacer el cambio de la contraseña de una forma muy sencilla, solo se debe ingresar la clave actual, la nueva clave que desee y confirmación.



Figura A14. Pantalla perfil 4.



Fuente: Los autores.

En el momento en que realice los cambio y este conforme con ellos, debe dar clic en el botón “confirmar” de lo contrario debe dar clic en el botón “Cancelar”.

#### **A.2.6. PREFERENCIAS**

En esta opción podrá registrar y modificar todas sus preferencias para que el sistema funcione de manera adecuada para usted. Lo primero que se debe seleccionar es el tipo de programa que se ajuste a usted.

Figura A15. Pantalla preferencias.

The screenshot shows a web browser window with the URL `poupawatt.info/index.php/Home/preferencias`. The page title is "Tus Preferencias". At the top right, there is a navigation menu with links: "PERFIL", "FACTURAS", "CONSUMO", "MODELOS", "JUEGO", and "CERRAR SESIÓN".

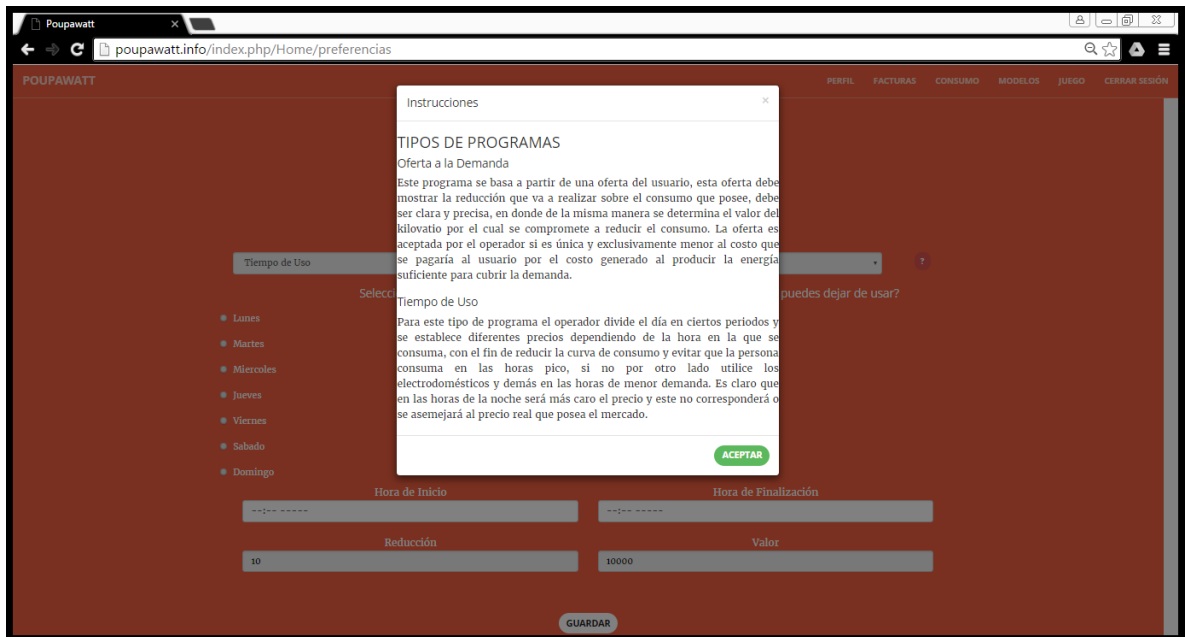
The main content area has a red background and contains the following elements:

- A green button labeled "INSTRUCCIONES >>".
- A heading: "PARA DARTÉ UN MEJOR SERVICIO ESTABLECE TUS PREFERENCIAS."
- A dropdown menu labeled "Tiempo de Uso" with the instruction "Selecciona el tipo de programa que se ajusta a ti". A white box highlights a question mark icon on the right side of the dropdown.
- Two columns of radio button options:
  - Left column: "Selecciona un día" with options: Lunes, Martes, Miércoles, Jueves, Viernes, Sábado, Domingo.
  - Right column: "Que electrodoméstico no puedes dejar de usar?" with options: Televisor, Nevera, Bombillas, Plancha, Lavadora.
- Two input fields: "Hora de Inicio" and "Hora de Finalización", both containing "-----".
- Two input fields: "Reducción" (containing "10") and "Valor" (containing "10000").
- A "GUARDAR" button at the bottom center.

Fuente: Los autores.

Para conocer los tipos de programas y escoger el que mejor se acomode a usted debe dar clic en el signo de interrogación “?”, allí se desplegara una ventana emergente contándole sobre estos programas.

Figura A16. Pantalla preferencias 2.



Fuente: Los autores.

Luego de tener claro que programa utilizara lo seleccionara y continuara en la selección de los días para la ejecución de los modelos.

Figura A17. Pantalla preferencias 3.

POUPAWATT

PERFIL FACTURAS CONSUMO MODELOS JUEGO CERRAR SESIÓN

### Tus Preferencias

[INSTRUCCIONES >>](#)

PARA DARTÉ UN MEJOR SERVICIO ESTABLECE TUS PREFERENCIAS.

Selecciona el tipo de programa que se ajusta a ti

Seleccione

- Seleccione
- Oferta a la Demanda
- Tiempo de Uso

Televisor

Nevera

Bombillas

Plancha

Lavadora

Lunes

Martes

Miercoles

Jueves

Viernes

Sabado

Domingo

Hora de Inicio

Hora de Finalización

Reducción

Valor

[GUARDAR](#)

Fuente: Los autores.

Una vez seleccionado el tipo de programa se prosigue llenando los siguientes campos que ahí se solicitan que son: día, electrodoméstico, hora de inicio, hora finalización, reducción (Numero de KWh que se desea reducir), costo para la ejecución de los modelos.

Figura A18 Pantalla preferencias 4.

POUPAWATT

Tus Preferencias

PERFIL FACTURAS CONSUMO MODELOS JUEGO CERRAR SESIÓN

INSTRUCCIONES >>

PARA DARTÉ UN MEJOR SERVICIO ESTABLECE TUS PREFERENCIAS.

Selecciona el tipo de programa que se ajusta a ti

Oferta a la Demanda

Selecciona un día

Que electrodoméstico no puedes dejar de usar?

- Lunes
- Martes
- Miércoles
- Jueves
- Viernes
- Sábado
- Domingo

- Televisor
- Nevera
- Bombillas
- Plancha
- Lavadora

Hora de Inicio

06:00 p. m.

Hora de Finalización

08:00 p. m.

Reducción

10

Valor

6000

GUARDAR

Fuente: Los autores.

Una vez completado los datos solicitado y conforme con estos, debe dar clic en guardar.

### A.2.7. FACTURAS.

En esta opción el usuario podrá llevar un control sobre sus facturas. Puede registrar cuantas facturas desee, solo es necesario llenar los datos que allí se solicitan.

Figura A19. Pantalla Facturas.

The screenshot shows a web browser window with the URL `poupawatt.info/index.php/Home/facturas`. The page has a red header with the logo 'POUPAWATT' on the left and a navigation menu on the right containing: 'PREFERENCIAS', 'CONSUMO', 'NOTICIAS', 'MODELOS', 'JUEGO', 'AJUSTES', and 'CERRAR SESIÓN'. The main content area is titled 'Tus Facturas' and contains the message 'No tienes facturas agregadas!'. Below this message is a form with the following fields: 'Numero Cuenta', 'Seleccione Clase de Servicio' (a dropdown menu), 'Periodo Facturado Inicio' (with a date format 'dd/mm/aaaa'), 'Periodo Facturado Final' (with a date format 'dd/mm/aaaa'), 'Lectura Actual', 'Lectura Anterior', 'Total Consumo', 'CPT', 'Valor kWh', and 'Total Pago'. At the bottom of the form is a button labeled 'AGREGAR FACTURA'.

Fuente: Los autores.

Una vez completado todos los campos requeridos como: Clase de servicio, periodo facturado, CPT (consumo promedio trimestral), total a pagar, entre otros. Debe dar clic en “agregar factura”.

Figura A20. Pantalla facturas 2.

The screenshot shows a web browser window with the URL `poupawatt.info/index.php/Home/facturas`. The page has a red header with the logo 'POUPAWATT' and navigation links: 'PREFERENCIAS', 'CONSUMO', 'NOTICIAS', 'MODELOS', 'JUEGO', 'AJUSTES', and 'CERRAR SESIÓN'. The main content area is titled 'Tus Facturas' and contains a form for adding a bill. The form includes the following fields:

- Factura Número: 1001896-0
- 1001896-0
- Residencial
- Periodo Facturado Inicio: 10/06/2016
- Periodo Facturado Final: 11/07/2016
- 27632
- 27634
- 8
- 21-333
- 429.84
- 3,429
- AGREGAR FACTURA

Fuente: Los autores.

Una vez registrada la factura, esta aparecerá del lado izquierdo de la pantalla, si da clic en ella, se desplegará la información de esa factura, podrá realizar modificaciones sobre esta información, cuando allá terminado con los cambios, debe dar clic en “actualizar”.

Figura A21. Pantalla facturas 3.

The screenshot shows a web browser window with the URL `poupawatt.info/index.php/Home/facturas`. The page title is "Tus Facturas" and the breadcrumb is "Factura Número: 2016-06-10". The form contains the following fields:

1001896-0
Residencial
Periodo Facturado Desde
10/06/2016
Periodo Facturado Hasta
11/07/2016
27632.0
27632.0
8
28-333
429.84
3.470

At the bottom of the form are two buttons: "ACTUALIZAR" and "ELIMINAR".

Fuente: Los autores.

Con esta opción también podrá eliminar facturas que ya no desee, para esto debe dar clic en “eliminar”.

### A.2.8. CONSUMO

En esta opción podrá visualizar el comportamiento de su consumo, además de ver cuantos kWh consumió y cuánto costó.



Figura A22. Pantalla Consumo.

Revisa tu consumo

[INSTRUCCIONES >>](#)

Seleccione Sensor

Select an option

dd/mm/aaaa

--:--:--

dd/mm/aaaa

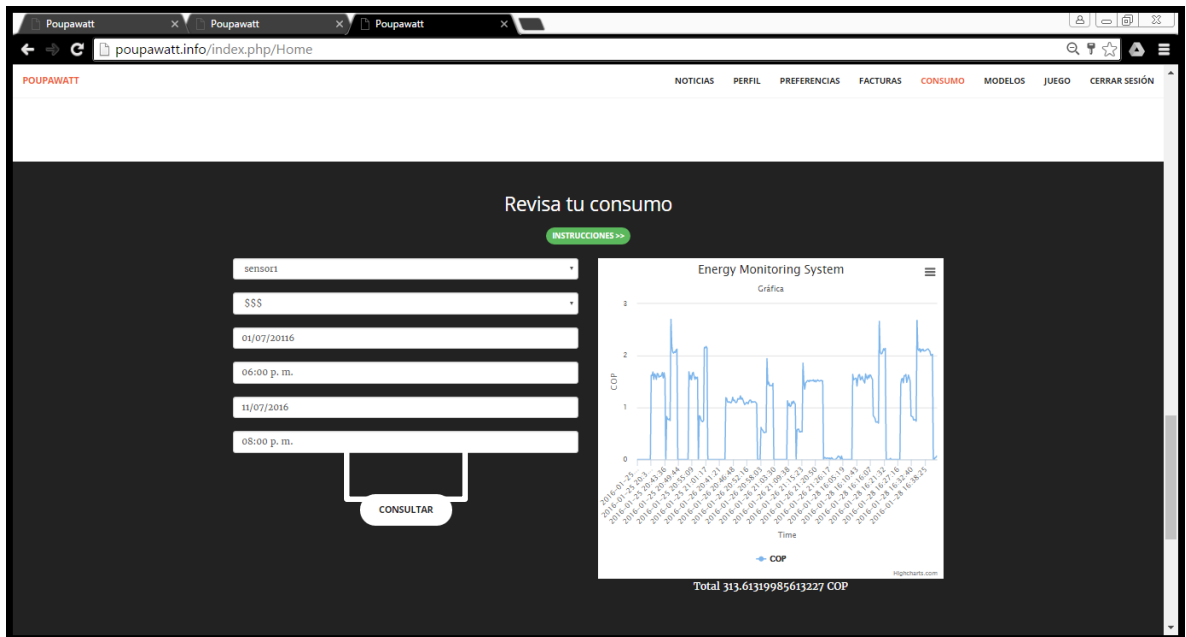
--:--:--

CONSULTAR

Fuente: Los autores.

Antes de visualizar el comportamiento de su consumo, debe primero llenar los datos allí se le solicitan, en los cuales debe seleccionar el sensor a consultar, seleccionar si por costo (\$\$\$) o KWh, seleccionar las fechas y horas en las que se desea ver dicho consumo.

Figura A23. Pantalla consumo 2.



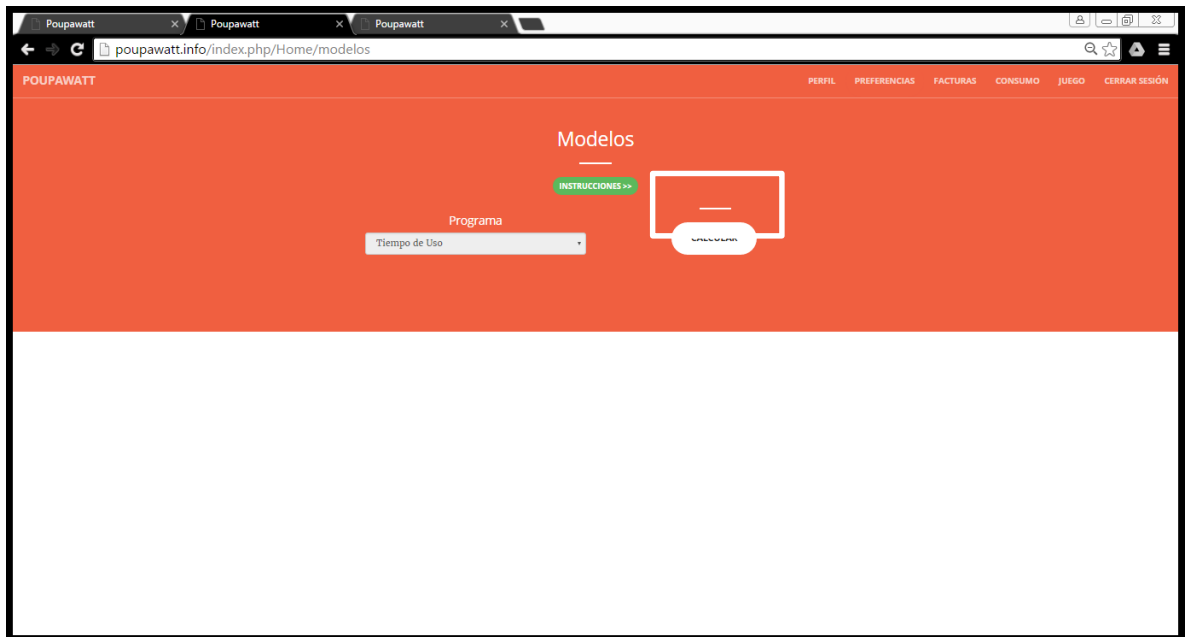
Fuente: Los autores.

Una vez completados los datos, se debe dar clic en “Consultar”, lo cual arrojará como resultado una gráfica que muestra el consumo entre los rangos de fechas seleccionados.

### A.2.9. MODELOS

En esta opción podrá encontrar el programa anteriormente seleccionado en preferencias y como consumir según ese.

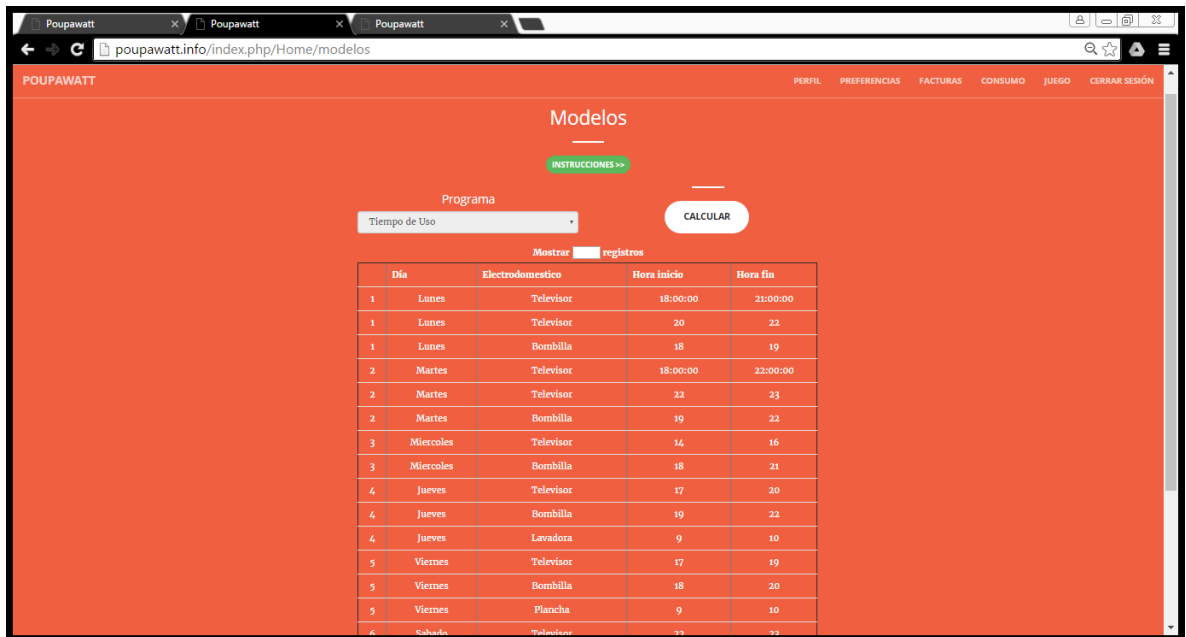
Figura A24. Pantalla modelos.



Fuente: Los autores.

Para conocer como consumir según el programa seleccionado, debe dar clic en el botón "Calcular".

Figura A25. Pantalla modelos 2.



Fuente: Los autores.

Una vez calculado, la aplicación nos arroja una tabla compuesta por: día, electrodoméstico, hora de inicio y hora de fin, la cual le dice al usuario en qué hora del día debe utilizar ese electrodoméstico para reducir el consumo.

### A.2.10. JUEGO

En esta opción encontrará un reto que se planteará al usuario para reducir el consumo, este juego lo puede realizar cuantas veces quiera y de la forma que desee, hasta que logre cumplir su meta, con el consumo deseado.

Figura A26. Pantalla Juego.

POUPAWATT

Juego

INSTRUCCIONES >>

Dia Inicio dd/mm/aaaa Dia Final dd/mm/aaaa

Porcentaje Ahorro

Electrodomestico 1

Seleccione Electrodomestico

Hora Inicio 1 Hora Final 1

Hora Inicio 2 Hora Final 2

Hora Inicio 3 Hora Final 3

Electrodomestico 2

Seleccione Electrodomestico

Hora Inicio 1 Hora Final 1

Hora Inicio 2 Hora Final 2

Hora Inicio 3 Hora Final 3

Electrodomestico 3

Seleccione Electrodomestico

Hora Inicio 1 Hora Final 1

Hora Inicio 2 Hora Final 2

Electrodomestico 4

Seleccione Electrodomestico

Hora Inicio 1 Hora Final 1

Hora Inicio 2 Hora Final 2

Fuente: Los autores.

Para empezar a jugar debe llenar los datos que ahí se le solicitan, seleccionando un día de inicio y un día de final, entre estos días debe haber al menos una semana de diferencia; luego se propone el porcentaje que desea ahorrar, encontrara un listado de 5 electrodomésticos a los cuales debe asignar un horario de horas en el cual los va a utilizar para reducir el consumo.

Figura A27. Pantalla juego 2.

POUPAWATT

PERFIL PREFERENCIAS FACTURAS CONSUMO MODELOS CERRAR SESIÓN

Día Inicio 01/07/2016 Día Final 30/07/2016

Porcentaje Ahorro 20

Electrodoméstico 1: Televisor

Hora Inicio 1 7:00 p. m. Hora Final 1 9:00 p. m.

Hora Inicio 2 Hora Final 2

Hora Inicio 3 Hora Final 3

Electrodoméstico 2: Nevera

Hora Inicio 1 11:00 a. m. Hora Final 1 3:00 p. m.

Hora Inicio 2 Hora Final 2

Hora Inicio 3 Hora Final 3

Electrodoméstico 3: Seleccione Electrodomestico

Electrodoméstico 4: Seleccione Electrodomestico

Hora Inicio 1 Hora Final 1

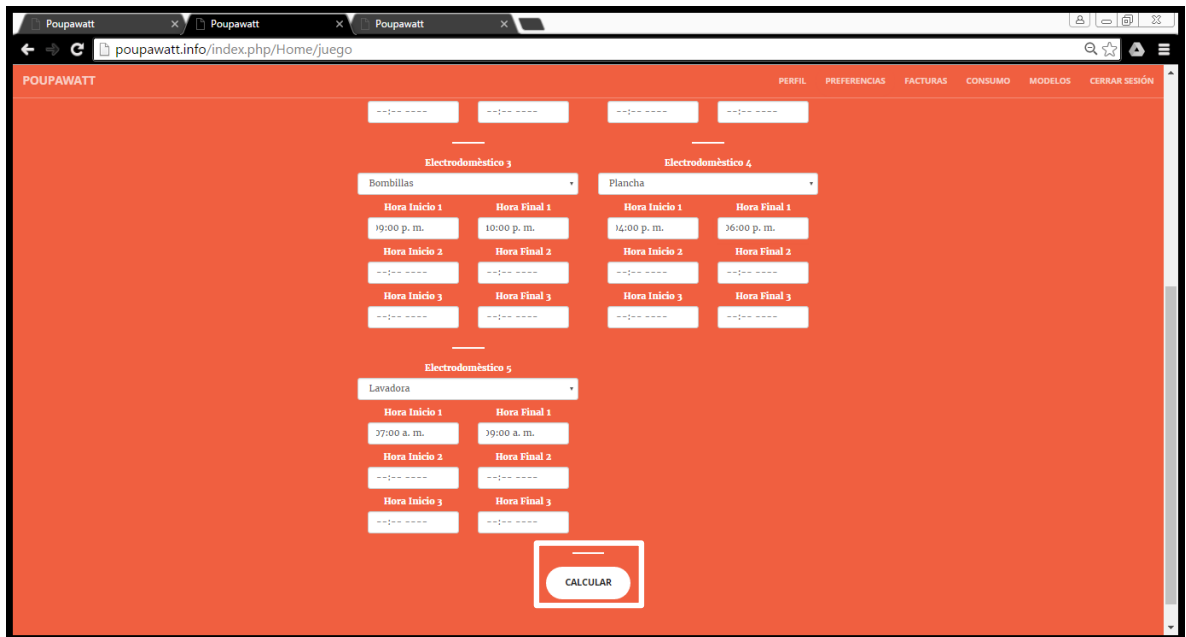
Hora Inicio 2 Hora Final 2

Hora Inicio 3 Hora Final 3

Fuente: Los autores.

Una vez completado los datos requeridos, se debe dar clic en “calcular” para que arroje los resultados y verificar si se realizó bien o mal. Y así poder seleccionar la configuración que le ayude a reducir su consumo.

Figura A28. Pantalla figura 3.



Fuente: Los autores.

## 11. ANEXO B

### Manual del programador aplicación web

#### B1. Introducción

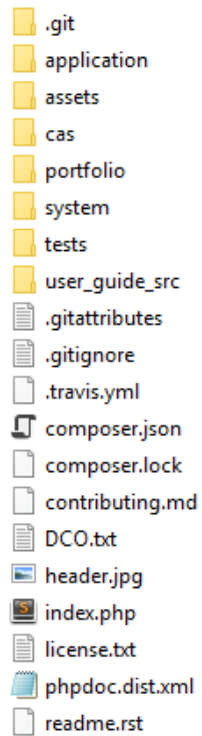
En este anexo se explica la estructura y funcionamiento del código fuente de la aplicación web para futuros cambios en la misma. Para un manejo del código y aplicación más fácil sin importar el programador, se diseñó bajo el parámetro de diseño modelo vista controlador (MVC) este permite separar el funcionamiento de la visualización, en el controlador se encuentra todo el código de funcionamiento, como conexiones a bases de datos, operaciones internas, opciones y cambios en formularios. En la vista como su nombre lo dice es la vista al usuario, esta ya es la parte de HCI, recolección y muestra de información al usuario. El lenguaje de programación usado fue php bajo el framework Codeigniter 3.0.6 y una plantilla de bootstrap.

#### B2. Configuración básica del sitio.

Un sitio diseñado bajo el parámetro de diseño MVC controlador se visualiza como en la figura B1.

Figura B1. Estructura sitio MVC





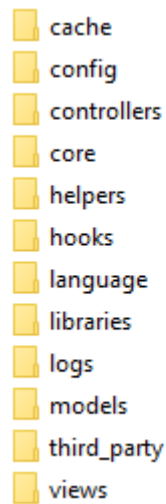
Fuente: Los autores

Dentro de la carpeta application se encuentran todos los archivos que permiten el funcionamiento y visualización del sitio. En la carpeta assets se encuentran los estilos css y las imágenes del sitio.

Las demás carpetas son creadas por el framework y la plantilla utilizada, sin embargo, dentro de application siempre estarán los archivos que interesan al programador.

Una vez dentro de la carpeta application se observan las de la figura B2

Figura B2. Carpeta applications.



Fuente: Los autores

En la carpeta config se encuentran los archivos de configuración para visualización en local o en el servidor, al igual que el archivo para los parámetros de conexión de las bases de datos.

### **B2.3 Configuración del sitio para desarrollo local o en servidor.**

Ir a la ruta `..\html\application\config` y seleccionar el archivo `config.php`

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

/*
|-----
|
| Base Site URL
|-----
|
|
| URL to your CodeIgniter root. Typically this will be your base URL,
| WITH a trailing slash:
|
|   http://example.com/
|
| WARNING: You MUST set this value!
|
| If it is not set, then CodeIgniter will try guess the protocol and path
| your installation, but due to security concerns the hostname will be
| set
| to $_SERVER['SERVER_ADDR'] if available, or localhost otherwise.
| The auto-detection mechanism exists only for convenience during
| development and MUST NOT be used in production!
|
| If you need to allow multiple domains, remember that this file is still
| a PHP script and you can easily do that on your own.
```

```

|
*/
$config['base_url'] = 'http://poupawatt.info';

/*
|-----
--
| Index File
|-----
--
|
| Typically this will be your index.php file, unless you've renamed it to
| something else. If you are using mod_rewrite to remove the page set
this
| variable so that it is blank.
|
*/
$config['index_page'] = 'index.php';

/*
|-----
--
| URI PROTOCOL
|-----
--
|
| This item determines which server global should be used to retrieve the
| URI string. The default setting of 'REQUEST_URI' works for most
servers.
| If your links do not seem to work, try one of the other delicious
flavors:
|
| 'REQUEST_URI'   Uses $_SERVER['REQUEST_URI']
| 'QUERY_STRING'  Uses $_SERVER['QUERY_STRING']
| 'PATH_INFO'     Uses $_SERVER['PATH_INFO']
|
| WARNING: If you set this to 'PATH_INFO', URIs will always be URL-
decoded!
*/
$config['uri_protocol'] = 'REQUEST_URI';

/*
|-----
--
| URL suffix
|-----
--
|
|
*/
|-----
--
| URL suffix
|-----
--

```

```

|
| This option allows you to add a suffix to all URLs generated by
CodeIgniter.
| For more information please see the user guide:
|
| https://codeigniter.com/user\_guide/general/urls.html
|
| Note: This option is ignored for CLI requests.
*/
$config['url_suffix'] = '';

/*
-----
--
| Default Language
|-----
--
|
| This determines which set of language files should be used. Make sure
| there is an available translation if you intend to use something other
| than english.
|
*/
$config['language'] = 'english';

/*
-----
--
| Default Character Set
|-----
--
|
| This determines which character set is used by default in various
methods
| that require a character set to be provided.
|
| See http://php.net/htmlspecialchars for a list of supported charsets.
|
*/
$config['charset'] = 'UTF-8';

/*
-----
--
| Enable/Disable System Hooks
|-----
--
|
| If you would like to use the 'hooks' feature you must enable it by
| setting this variable to TRUE (boolean). See the user guide for
details.
|
*/
$config['enable_hooks'] = FALSE;

```

```

/*
|-----|
--
| Class Extension Prefix
|-----|
--
|
| This item allows you to set the filename/classname prefix when
extending
| native libraries. For more information please see the user guide:
|
| https://codeigniter.com/user_guide/general/core_classes.html
| https://codeigniter.com/user_guide/general/creating_libraries.html
|
*/
$config['subclass_prefix'] = 'MY_';

/*
|-----|
--
| Composer auto-loading
|-----|
--
|
| Enabling this setting will tell CodeIgniter to look for a Composer
| package auto-loader script in application/vendor/autoload.php.
|
|   $config['composer_autoload'] = TRUE;
|
| Or if you have your vendor/ directory located somewhere else, you
| can opt to set a specific path as well:
|
|   $config['composer_autoload'] = '/path/to/vendor/autoload.php';
|
| For more information about Composer, please visit
http://getcomposer.org/
|
| Note: This will NOT disable or override the CodeIgniter-specific
|   autoloading (application/config/autoload.php)
*/
$config['composer_autoload'] = FALSE;

/*
|-----|
--
| Allowed URL Characters
|-----|
--
|
| This lets you specify which characters are permitted within your URLs.
| When someone tries to submit a URL with disallowed characters they will
| get a warning message.
|

```

```

| As a security measure you are STRONGLY encouraged to restrict URLs to
| as few characters as possible.  By default only these are allowed: a-z
0-9~%.:_-
|
| Leave blank to allow all characters -- but only if you are insane.
|
| The configured value is actually a regular expression character group
| and it will be executed as: ! preg_match('/^[<permitted_uri_chars>]+$'/i
|
| DO NOT CHANGE THIS UNLESS YOU FULLY UNDERSTAND THE REPERCUSSIONS!!
|
| Note: This option is ignored for CLI requests.
|
*/
$config['permitted_uri_chars'] = 'a-z 0-9~%.:_-';

/*
|-----
--
| Enable Query Strings
|-----
--
|
| By default CodeIgniter uses search-engine friendly segment based URLs:
| example.com/who/what/where/
|
| By default CodeIgniter enables access to the $_GET array.  If for some
| reason you would like to disable it, set 'allow_get_array' to FALSE.
|
| You can optionally enable standard query string based URLs:
| example.com?who=me&what=something&where=here
|
| Options are: TRUE or FALSE (boolean)
|
| The other items let you set the query string 'words' that will
| invoke your controllers and its functions:
| example.com/index.php?c=controller&m=function
|
| Please note that some of the helpers won't work as expected when
| this feature is enabled, since CodeIgniter is designed primarily to
| use segment based URLs.
|
*/
$config['allow_get_array'] = TRUE;
$config['enable_query_strings'] = FALSE;
$config['controller_trigger'] = 'c';
$config['function_trigger'] = 'm';
$config['directory_trigger'] = 'd';

/*
|-----
--
| Error Logging Threshold

```

```

|-----
--
|
| You can enable error logging by setting a threshold over zero. The
| threshold determines what gets logged. Threshold options are:
|
| 0 = Disables logging, Error logging TURNED OFF
| 1 = Error Messages (including PHP errors)
| 2 = Debug Messages
| 3 = Informational Messages
| 4 = All Messages
|
| You can also pass an array with threshold levels to show individual
error types
|
| array(2) = Debug Messages, without Error Messages
|
| For a live site you'll usually only enable Errors (1) to be logged
otherwise
| your log files will fill up very fast.
|
*/
$config['log_threshold'] = 0;

/*
|-----
--
| Error Logging Directory Path
|-----
--
|
| Leave this BLANK unless you would like to set something other than the
default
| application/logs/ directory. Use a full server path with trailing
slash.
|
*/
$config['log_path'] = '';

/*
|-----
--
| Log File Extension
|-----
--
|
| The default filename extension for log files. The default 'php' allows
for
| protecting the log files via basic scripting, when they are to be
stored
| under a publicly accessible directory.
|
| Note: Leaving it blank will default to 'php'.
|

```

```

*/
$config['log_file_extension'] = '';

/*
|-----
--
| Log File Permissions
|-----
--
|
| The file system permissions to be applied on newly created log files.
|
| IMPORTANT: This MUST be an integer (no quotes) and you MUST use octal
|             integer notation (i.e. 0700, 0644, etc.)
*/
$config['log_file_permissions'] = 0644;

/*
|-----
--
| Date Format for Logs
|-----
--
|
| Each item that is logged has an associated date. You can use PHP date
| codes to set your own date formatting
|
*/
$config['log_date_format'] = 'Y-m-d H:i:s';

/*
|-----
--
| Error Views Directory Path
|-----
--
|
| Leave this BLANK unless you would like to set something other than the
| default
| application/views/errors/ directory. Use a full server path with
| trailing slash.
|
*/
$config['error_views_path'] = '';

/*
|-----
--
| Cache Directory Path
|-----
--
|
| Leave this BLANK unless you would like to set something other than the
| default

```



```

| application/cache/ directory. Use a full server path with trailing
| slash.
|
*/
$config['cache_path'] = '';

/*
|-----
--
| Cache Include Query String
|-----
--
|
| Whether to take the URL query string into consideration when generating
| output cache files. Valid options are:
|
| FALSE      = Disabled
| TRUE       = Enabled, take all query parameters into account.
|              Please be aware that this may result in numerous cache
|              files generated for the same page over and over again.
| array('q') = Enabled, but only take into account the specified list
|              of query parameters.
|
*/
$config['cache_query_string'] = FALSE;

/*
|-----
--
| Encryption Key
|-----
--
|
| If you use the Encryption class, you must set an encryption key.
| See the user guide for more info.
|
| https://codeigniter.com/user\_guide/libraries/encryption.html
|
*/
$config['encryption_key'] = '';

/*
|-----
--
| Session Variables
|-----
--
|
| 'sess_driver'
|
| The storage driver to use: files, database, redis, memcached
|
| 'sess_cookie_name'
|

```

```

|   The session cookie name, must contain only [0-9a-z_-] characters
|
| 'sess_expiration'
|
|   The number of SECONDS you want the session to last.
|   Setting to 0 (zero) means expire when the browser is closed.
|
| 'sess_save_path'
|
|   The location to save sessions to, driver dependent.
|
|   For the 'files' driver, it's a path to a writable directory.
|   WARNING: Only absolute paths are supported!
|
|   For the 'database' driver, it's a table name.
|   Please read up the manual for the format with other session drivers.
|
|   IMPORTANT: You are REQUIRED to set a valid save path!
|
| 'sess_match_ip'
|
|   Whether to match the user's IP address when reading the session data.
|
|   WARNING: If you're using the database driver, don't forget to update
|           your session table's PRIMARY KEY when changing this setting.
|
| 'sess_time_to_update'
|
|   How many seconds between CI regenerating the session ID.
|
| 'sess_regenerate_destroy'
|
|   Whether to destroy session data associated with the old session ID
|   when auto-regenerating the session ID. When set to FALSE, the data
|   will be later deleted by the garbage collector.
|
| Other session cookie settings are shared with the rest of the
application,
| except for 'cookie_prefix' and 'cookie_httponly', which are ignored
here.
|
| */
$config['sess_driver'] = 'files';
$config['sess_cookie_name'] = 'ci_session';
$config['sess_expiration'] = 7200;
$config['sess_save_path'] = NULL;
$config['sess_match_ip'] = FALSE;
$config['sess_time_to_update'] = 300;
$config['sess_regenerate_destroy'] = FALSE;

/*
|-----
--
| Cookie Related Variables

```

```

|-----|
--
|
| 'cookie_prefix' = Set a cookie name prefix if you need to avoid
collisions
| 'cookie_domain' = Set to .your-domain.com for site-wide cookies
| 'cookie_path'   = Typically will be a forward slash
| 'cookie_secure' = Cookie will only be set if a secure HTTPS
connection exists.
| 'cookie_httponly' = Cookie will only be accessible via HTTP(S) (no
javascript)
|
| Note: These settings (with the exception of 'cookie_prefix' and
|       'cookie_httponly') will also affect sessions.
|
*/
$config['cookie_prefix']   = '';
$config['cookie_domain']   = '';
$config['cookie_path']     = '/';
$config['cookie_secure']   = FALSE;
$config['cookie_httponly'] = FALSE;

/*
|-----|
--
| Standardize newlines
|-----|
--
|
| Determines whether to standardize newline characters in input data,
| meaning to replace \r\n, \r, \n occurrences with the PHP_EOL value.
|
| This is particularly useful for portability between UNIX-based OSes,
| (usually \n) and Windows (\r\n).
|
*/
$config['standardize_newlines'] = FALSE;

/*
|-----|
--
| Global XSS Filtering
|-----|
--
|
| Determines whether the XSS filter is always active when GET, POST or
| COOKIE data is encountered
|
| WARNING: This feature is DEPRECATED and currently available only
|         for backwards compatibility purposes!
|
*/
$config['global_xss_filtering'] = FALSE;

```

```

/*
|-----
--
| Cross Site Request Forgery
|-----
--
| Enables a CSRF cookie token to be set. When set to TRUE, token will be
| checked on a submitted form. If you are accepting user data, it is
strongly
| recommended CSRF protection be enabled.
|
| 'csrf_token_name' = The token name
| 'csrf_cookie_name' = The cookie name
| 'csrf_expire' = The number in seconds the token should expire.
| 'csrf_regenerate' = Regenerate token on every submission
| 'csrf_exclude_uris' = Array of URIs which ignore CSRF checks
*/
$config['csrf_protection'] = FALSE;
$config['csrf_token_name'] = 'csrf_test_name';
$config['csrf_cookie_name'] = 'csrf_cookie_name';
$config['csrf_expire'] = 7200;
$config['csrf_regenerate'] = TRUE;
$config['csrf_exclude_uris'] = array();

/*
|-----
--
| Output Compression
|-----
--
|
| Enables Gzip output compression for faster page loads. When enabled,
| the output class will test whether your server supports Gzip.
| Even if it does, however, not all browsers support compression
| so enable only if you are reasonably sure your visitors can handle it.
|
| Only used if zlib.output_compression is turned off in your php.ini.
| Please do not use it together with httpd-level output compression.
|
| VERY IMPORTANT: If you are getting a blank page when compression is
enabled it
| means you are prematurely outputting something to your browser. It
could
| even be a line of whitespace at the end of one of your scripts. For
| compression to work, nothing can be sent before the output buffer is
called
| by the output class. Do not 'echo' any values with compression
enabled.
|
*/
$config['compress_output'] = FALSE;

/*

```

```

|-----
--
| Master Time Reference
|-----
--
|
| Options are 'local' or any PHP supported timezone. This preference
tells
| the system whether to use your server's local time as the master 'now'
| reference, or convert it to the configured one timezone. See the 'date
| helper' page of the user guide for information regarding date handling.
|
| */
$config['time_reference'] = 'local';

/*
|-----
--
| Rewrite PHP Short Tags
|-----
--
|
| If your PHP installation does not have short tag support enabled CI
| can rewrite the tags on-the-fly, enabling you to utilize that syntax
| in your view files. Options are TRUE or FALSE (boolean)
|
| Note: You need to have eval() enabled for this to work.
|
| */
$config['rewrite_short_tags'] = FALSE;

/*
|-----
--
| Reverse Proxy IPs
|-----
--
|
| If your server is behind a reverse proxy, you must whitelist the proxy
| IP addresses from which CodeIgniter should trust headers such as
| HTTP_X_FORWARDED_FOR and HTTP_CLIENT_IP in order to properly identify
| the visitor's IP address.
|
| You can use both an array or a comma-separated list of proxy addresses,
| as well as specifying whole subnets. Here are a few examples:
|
| Comma-separated: '10.0.1.200,192.168.5.0/24'
| Array:          array('10.0.1.200', '192.168.5.0/24')
| */
$config['proxy_ips'] = '';

```

Buscar la línea que hace referencia a la modificación del siguiente parámetro

```
$config['base_url'] = ' ';
```

Dentro de las comillas se indica la base url para el desarrollo y/o publicación.

Cuando se trabaja en local se debe poner 'http://localhost/html'.

Cuando se trabaja en el servidor se debe poner el dominio asociado, en este caso 'http://poupawatt.info'. En caso de no contar con dominio se debe poner la ip del servidor que aloja el sitio, en este caso 'http://198.199.72.168'.

## B2.4 Configuración bases de datos

Ir a la ruta ../html\application\config y seleccionar el archivo database.php

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

/*
| -----
| DATABASE CONNECTIVITY SETTINGS
| -----
| This file will contain the settings needed to access your database.
|
| For complete instructions please consult the 'Database Connection'
| page of the User Guide.
|
| -----
| EXPLANATION OF VARIABLES
| -----
|
| ['dsn']      The full DSN string describe a connection to the
database.
| ['hostname'] The hostname of your database server.
| ['username'] The username used to connect to the database
| ['password'] The password used to connect to the database
| ['database'] The name of the database you want to connect to
| ['dbdriver'] The database driver. e.g.: mysqli.
|
|           Currently supported:
|           cubrid, ibase, mssql, mysql, mysqli, oci8,
|           odbc, pdo, postgre, sqlite, sqlite3, sqlsrv
| ['dbprefix'] You can add an optional prefix, which will be added
|               to the table name when using the Query Builder class
| ['pconnect'] TRUE/FALSE - Whether to use a persistent connection
| ['db_debug'] TRUE/FALSE - Whether database errors should be
displayed.
| ['cache_on'] TRUE/FALSE - Enables/disables query caching
| ['cachedir'] The path to the folder where cache files should be
stored
| ['char_set'] The character set used in communicating with the
database
| ['dbcollat'] The character collation used in communicating with the
database
|
|           NOTE: For MySQL and MySQLi databases, this setting is
only used
|
|           as a backup if your server is running PHP < 5.2.3 or
MySQL < 5.0.7
|
|           (and in table creation queries made with DB Forge).
```

```

|         There is an incompatibility in PHP with
mysql_real_escape_string() which
|         can make your site vulnerable to SQL injection if you
are using a
|         multi-byte character set and are running versions lower
than these.
|         Sites using Latin-1 or UTF-8 database character set and
collation are unaffected.
|     ['swap_pre'] A default table prefix that should be swapped with the
dbprefix
|     ['encrypt'] Whether or not to use an encrypted connection.
|
|         'mysql' (deprecated), 'sqlsrv' and 'pdo/sqlsrv' drivers
accept TRUE/FALSE
|         'mysqli' and 'pdo/mysql' drivers accept an array with the
following options:
|
|         'ssl_key' - Path to the private key file
|         'ssl_cert' - Path to the public key certificate file
|         'ssl_ca' - Path to the certificate authority file
|         'ssl_capath' - Path to a directory containing trusted CA
certificats in PEM format
|         'ssl_cipher' - List of *allowed* ciphers to be used for
the encryption, separated by colons (':')
|         'ssl_verify' - TRUE/FALSE; Whether verify the server
certificate or not ('mysqli' only)
|
|     ['compress'] Whether or not to use client compression (MySQL only)
|     ['stricton'] TRUE/FALSE - forces 'Strict Mode' connections
|         - good for ensuring strict SQL while
developing
|     ['ssl_options'] Used to set various SSL options that can be used when
making SSL connections.
|     ['failover'] array - A array with 0 or more data for connections if
the main should fail.
|     ['save_queries'] TRUE/FALSE - Whether to "save" all executed queries.
|         NOTE: Disabling this will also effectively disable both
|         $this->db->last_query() and profiling of DB queries.
|         When you run a query, with this setting set to TRUE
(default),
|         CodeIgniter will store the SQL statement for debugging
purposes.
|         However, this may cause high memory usage, especially if
you run
|         a lot of SQL queries ... disable this to avoid that
problem.
|
|     The $active_group variable lets you choose which connection group to
make active. By default there is only one group (the 'default' group).
|
|     The $query_builder variables lets you determine whether or not to load
the query builder class.
*/
$active_group = 'default';

```

```

$query_builder = TRUE;

$db['default'] = array(
    'dsn' => '',
    'hostname' => 'localhost',
    'username' => 'root',
    'password' => 'manuelbernal',
    'database' => 'poupawatt',
    'dbdriver' => 'mysqli',
    'dbprefix' => '',
    'pconnect' => FALSE,
    'db_debug' => (ENVIRONMENT !== 'production'),
    'cache_on' => FALSE,
    'cachedir' => '',
    'char_set' => 'utf8',
    'dbcollat' => 'utf8_general_ci',
    'swap_pre' => '',
    'encrypt' => FALSE,
    'compress' => FALSE,
    'stricton' => FALSE,
    'failover' => array(),
    'save_queries' => TRUE
);

$db['ems'] = array(
    'dsn' => '',
    'hostname' => 'localhost',
    'username' => 'root',
    'password' => 'manuelbernal',
    'database' => 'ems',
    'dbdriver' => 'mysqli',
    'dbprefix' => '',
    'pconnect' => FALSE,
    'db_debug' => (ENVIRONMENT !== 'production'),
    'cache_on' => FALSE,
    'cachedir' => '',
    'char_set' => 'utf8',
    'dbcollat' => 'utf8_general_ci',
    'swap_pre' => '',
    'encrypt' => FALSE,
    'compress' => FALSE,
    'stricton' => FALSE,
    'failover' => array(),
    'save_queries' => TRUE
);

```

Los siguientes parámetros son genéricos y deben ser ajustados según la base de datos creada.

```

$db['aquí va el nombre de la base datos'] = array(
    'dsn' => "",
    'hostname' => 'localhost',

```



```

        'username' => 'root',
        'password' => 'en caso de contraseña poner, caso
contrario dejar en blanco',
        'database' => 'nombre base de datos',
        'dbdriver' => 'mysqli',
        'dbprefix' => "",
        'pconnect' => FALSE,
        'db_debug' => (ENVIRONMENT !== 'production'),
        'cache_on' => FALSE,
        'cachedir' => "",
        'char_set' => 'utf8',
        'dbcollat' => 'utf8_general_ci',
        'swap_pre' => "",
        'encrypt' => FALSE,
        'compress' => FALSE,
        'stricton' => FALSE,
        'failover' => array(),
        'save_queries' => TRUE
    );

```

Los parámetros en negrilla son los principales, con la modificación de estos como se indica la base de datos quedará asociada al sitio para su uso.

### **B3. Modelos**

Los modelos son los encargados de tomar la información recogida por los diferentes formularios y llevarla a la base de datos, de igual manera la extrae para ser visualizada.

El primer modelo se encuentra en la ruta `..\html\application\models` y abrir el archivo `Graphic_model.php`. En estos se crea una clase que contendrá las funciones de almacenamiento y extracción de datos según formulario y tabla.

```
<?php
```

```
class Graphic_model extends CI_Model
```

```

{

    public function __construct()
    {
        parent::__construct();
    }

    public function
generateGraphic($selectrodomestic,$stype,$firstDate,$secondDate)
    {
        $db2 = $this->load->database('ems', TRUE);

        if($stype == 1){
            $this->db->select('kwh,time');
        }
        else{
            $this->db->select('price,time');
        }
        $db2->from($selectrodomestic);
        $db2->where('time >=', $firstDate);
        $db2->where('time <=', $secondDate);
        $query = $db2->get();
        $ret = $query->result();

        $arrayx = [];
        $arrayy = [];

        if($stype == 1) {
            foreach ($ret as $item) {
                array_push($arrayx, $item->time);
                array_push($arrayy, $item->kwh);
            }
        }
        else{
            foreach ($ret as $item) {
                array_push($arrayx, $item->time);
                array_push($arrayy, $item->price);
            }
        }

        $graphicData['arrayx'] =$arrayx;
        $graphicData['arrayy'] =$arrayy;
        return $graphicData;
    }

    public function getCities(){
        $query = $this->db->get('tbl_ciudades');
        return $query->result();
    }

    public function getRoles(){

```

```

        $query = $this->db->get('rol');
        return $query->result();
    }

    public function getNeighborhoods($barrio){
        $this->db->from('tbl_barrios');
        $this->db->where('intLocalidad', $barrio);
        $query = $this->db->get();
        return $query->result();
    }

    public function getLocations($localidad){
        $this->db->from('tbl_localidad');
        $this->db->where('intContador', $localidad);
        $query = $this->db->get();
        return $query->result();
    }

    public function sensores($user){
        $this->db->from('tbl_sensores');
        $this->db->where('id_user', $user);
        $query = $this->db->get();
        return $query->result();
    }

    public function login($email,$password){
        $this->db->from('users');
        $this->db->where('varEmail', $email);
        $this->db->where('password', $password);
        $this->db->where('intEstado', 1);
        $query = $this->db->get();
        $ret = $query->result();

        if(count($ret)){
            return array(
                'result' => 1,
                'nombre' => $ret[0]->nombre,
                'id' => $ret[0]->intNivel,
                'rol' => $ret[0]->rol,
            );
        }
        else{
            return 0;
        }
    }

    public function signup($user_new){
        $this->db->insert('users', $user_new);
    }
    public function crearPreferencia($array){
        $this->db->insert('preferences', $array);
    }
}

```

```

public function modificarPreferencia($array,$id){

    $this->db->where('id', $id);
    $this->db->update('preferences', $array);

}

public function
generateGraphicSum($electrodomestic,$type,$firstDate,$secondDate)
{
    $db2 = $this->load->database('ems', TRUE);

    if($type == 1){
        $db2->select_sum('kwh');
    }
    else{
        $db2->select_sum('price');
    }
    $db2->from($electrodomestic);
    $db2->where('time >=', $firstDate);
    $db2->where('time <=', $secondDate);
    $query = $db2->get();
    $ret = $query->result();
    return $ret;
}

public function crearFactura($array){
    $this->db->insert('tbl_factura', $array);
    $insert_id = $this->db->insert_id();
    return $insert_id;
}

public function listFacturas(){
    $this->db->from('tbl_factura');
    $this->db->where('id', $this->session->userdata('id'));
    $query = $this->db->get();
    return $query->result();
}

public function listProgramas(){
    $this->db->from('tbl_programa');
    $query = $this->db->get();
    return $query->result();
}

public function listElectrodomesticos(){
    $this->db->from('tbl_electrodomestico');
    $query = $this->db->get();
    return $query->result();
}

```

```

    }

    public function cargarFactura($factura){
        $this->db->from('tbl_factura');
        $this->db->where('idFactura', $factura);
        $query = $this->db->get();
        return $query->result();
    }

    public function modificarFactura($array,$id){

        $this->db->where('idFactura', $id);
        $this->db->update('tbl_factura', $array);

    }

    public function eliminarFactura($id){

        $this->db->where('idFactura', $id);
        $this->db->delete('tbl_factura');

    }

    public function traerPreferencias($id){
        $this->db->from('preferences');
        $this->db->where('id_user', $id);
        $query = $this->db->get();
        return $query->result();
    }

    public function datosuser($id){
        $this->db->from('users');
        $this->db->where('intNivel', $id);
        $query = $this->db->get();
        return $query->result();
    }

    public function submituser($id, $array){
        $this->db->where('intNivel', $id);
        $this->db->update('users', $array);
    }

    public function docsusers(){
        $this->db->select('documento');
        $query = $this->db->get('users');
        return $query->result();
    }

    public function datosuserdoc($id){
        $this->db->from('users');
        $this->db->where('documento', $id);
        $query = $this->db->get();
        return $query->result();
    }

```

```

    }

    function deleteuser($id) {
        $this->db->where('intNivel', $id);
        $this->db->delete('users');
    }

    public function submitpassword($id, $array){
        $this->db->where('intNivel', $id);
        $this->db->update('users', $array);
    }

}

```

La primera función es

```

public function
generateGraphic($electrodomestic,$type,$firstDate,$secondDate) public
function
generateGraphicSum($electrodomestic,$type,$firstDate,$secondDate)

```

Esta recolecta los datos para la generación de la gráfica en la sección consumo y la sumatoria del valor entre rango seleccionado.

Las funciones

```

public function getCities(){
public function getNeighborhoods($barrio){
public function getLocations($localidad){

```

Son las encargas de extraer la información de las ciudades, localidades y barrios disponibles de la base de datos para el registro y modificación de datos personales.

La función

```

public function login($email,$password){

```

Esta función es la encargada de extraer y comparar los datos para el login.

La función

```

public function signup($user_new){

```

Esta función permite la creación de nuevos usuario

Las funciones

```

    public function crearPreferencia($array){
    public function modificarPreferencia($array,$id){

```

Estas funciones permiten al usuario crear y modificar sus preferencias, de manera que una es la encargada de almacenar y la otra de cargar y re almacenarlas.

Las funciones

```

public function crearFactura($array){
public function listFacturas(){
public function cargarFactura($factura){
public function modificarFactura($array,$id){
public function eliminarFactura($id){

```

Estas funciones son las encargadas de la creación, modificación y eliminación de facturas, y otra está hecha exclusivamente para cargar y mostrar los datos relacionados ya almacenados.

La función

```
public function getRoles() {
```

Se encarga de cargar al nuevo usuario su rol dentro del sistema como usuario estándar.

Las funciones

```
public function listProgramas() {  
public function listElectrodomesticos() {  
public function traerPreferencias($id) {
```

Son las encargadas de traer los programas y electrodomésticos existentes en la selección de preferencias, además de cargar las ya existentes para su modificación.

Por último las funciones

```
public function datosuser($id) {  
public function docsusers() {  
public function datosuserdoc($id) {  
public function deleteuser($id) {  
public function submitpassword($id, $array) {
```

Son las encargadas del manejo de usuarios, permitiendo modificar, adicionar y eliminar.

## B4. Controladores

Los controladores son los encargados del funcionamiento del sitio, están divididos en controlador para generación de gráficas, funcionamiento usuario estándar, funcionamiento usuario administrador, creación nuevo usuario, login y algoritmos modelos.

### B4.1 Generación de graficas

Ir a la ruta ..\html\application\controllers abrir archivo Excel

Este archivo permite la creación de los arrays para ser almacenados y posteriormente graficados.

```
<?php  
defined('BASEPATH') OR exit('No direct script access allowed');  
  
use Ghunti\HighchartsPHP\Highchart;
```

```

use Ghunti\HighchartsPHP\HighchartJsExpr;
use PHPExcel as PHPExcel;

class Excel extends CI_Controller
{

    public function __construct()
    {
        parent::__construct();
        $this->load->model('graphic_model', 'gm');
    }

    public function index()
    {

        $electrodomestic = $this->input->post('electrodomestic');
        $type = $this->input->post('type');
        $firstDate = $this->input->post('firstDate');
        $secondDate = $this->input->post('secondDate');
        $firstHour = $this->input->post('firstHour');
        $secondHour = $this->input->post('secondHour');

        $firstDate = $firstDate . ' ' . $firstHour . ':00:00';
        $secondDate = $secondDate . ' ' . $secondHour . ':00:00';

        $graphic = $this->gm->generateGraphic($electrodomestic, $type,
        $firstDate, $secondDate);

        $array['xvalues'] = $graphic['arrayx'];
        $array['yvalues'] = $graphic['arrayy'];

        $objPHPExcel = new PHPExcel();
        // Set document properties
        $objPHPExcel->getProperties()->setCreator("Maarten Balliauw")
            ->setLastModifiedBy("Maarten Balliauw")
            ->setTitle("Office 2007 XLSX Test Document")
            ->setSubject("Office 2007 XLSX Test Document")
            ->setDescription("Test document for Office 2007 XLSX,
generated using PHP classes.")
            ->setKeywords("office 2007 openxml php")
            ->setCategory("Test result file");
        // Add some data
        $objPHPExcel->setActiveSheetIndex(0)
            ->setCellValue('A1', 'Date')
            ->setCellValue('B1', 'Value')
        ;

        $i = 2;
    }
}

```



```

        foreach ( $array['xvalues'] as $item) {
            $objPHPExcel->setActiveSheetIndex(0)
                ->setCellValue('A'.$i, $item);
            $i = $i + 1;
        }

        $i = 2;
        foreach ( $array['yvalues'] as $item) {
            $objPHPExcel->setActiveSheetIndex(0)
                ->setCellValue('B'.$i, $item);
            $i = $i + 1;
        }
// Rename worksheet
    $objPHPExcel->getActiveSheet()->setTitle('Report');
// Set active sheet index to the first sheet, so Excel opens this as the
first sheet
    $objPHPExcel->setActiveSheetIndex(0);
// Redirect output to a client's web browser (OpenDocument)
    header('Content-Type:
application/vnd.oasis.opendocument.spreadsheet');
    header('Content-Disposition: attachment;filename="Report.ods"');
    header('Cache-Control: max-age=0');
// If you're serving to IE 9, then the following may be needed
    header('Cache-Control: max-age=1');
// If you're serving to IE over SSL, then the following may be needed
    header('Expires: Mon, 26 Jul 1997 05:00:00 GMT'); // Date in the
past
    header('Last-Modified: ' . gmdate('D, d M Y H:i:s') . ' GMT'); //
always modified
    header('Cache-Control: cache, must-revalidate'); // HTTP/1.1
    header('Pragma: public'); // HTTP/1.0
    $objWriter = PHPExcel_IOFactory::createWriter($objPHPExcel,
'OpenDocument');
    $objWriter->save('php://output');
    exit;
}
}

```

## B4.2 Control usuario estándar

Este control es el encargado de recoger la información de las diferentes secciones del sitio para el usuario estándar.

```

?php
defined('BASEPATH') OR exit('No direct script access allowed');

use Ghunti\HighchartsPHP\Highchart;
use Ghunti\HighchartsPHP\HighchartJsExpr;
use PHPExcel as PHPExcel;

```

```

class Home extends CI_Controller
{

    public function __construct()
    {
        //$_SESSION = array();

        parent::__construct();
        $this->load->model('graphic_model', 'gm');
        $this->load->model('graphic_model2', 'gm2');
        $this->load->library('session');

        if($this->session->userdata('id')== FALSE)
            redirect(base_url());

    }

    public function index()
    {

        $array['sensor'] = $this->gm->sensores($this->session->
        >userdata('id'));
        $this->load->view('home', $array);

    }

    public function salir()
    {
        $this->session->sess_destroy();
        echo "<script>>window.location.href = '" . base_url() .
        "';</script>";

    }

    public function generateGraphic()
    {
        $electrodomestic = $this->input->post('electrodomestic');
        $type = $this->input->post('type');
        $firstDate = $this->input->post('firstDate');
        $secondDate = $this->input->post('secondDate');
        $firstHour = $this->input->post('firstHour');
        $secondHour = $this->input->post('secondHour');

        $firstDate = $firstDate . ' ' . $firstHour . ':00:00';
        $secondDate = $secondDate . ' ' . $secondHour . ':00:00';

        if ($type == 1)
            $texto = 'Kwh';
        else
            $texto = 'COP';
    }
}

```

```

        $graphic = $this->gm->generateGraphic($electrodomestic, $type,
        $firstDate, $secondDate);

        $array['xvalues'] = $graphic['arrayx'];
        $array['yvalues'] = $graphic['arrayy'];
        $array['texto'] = $texto;

        $array['nombreGrafica'] = 'Gráfica';
        ($this->load->view('chart', $array));

    }

    public function generateSumGraphic()
    {
        $electrodomestic = $this->input->post('electrodomestic');
        $type = $this->input->post('type');
        $firstDate = $this->input->post('firstDate');
        $secondDate = $this->input->post('secondDate');
        $firstHour = $this->input->post('firstHour');
        $secondHour = $this->input->post('secondHour');

        $firstDate = $firstDate . ' ' . $firstHour . ':00:00';
        $secondDate = $secondDate . ' ' . $secondHour . ':00:00';

        if ($electrodomestic != 'all') {
            $total = $this->gm->generateGraphicSum($electrodomestic,
            $type, $firstDate, $secondDate);

            if ($type == 1) {
                $texto = 'Kwh';
                echo 'Total ' . $total[0]->kwh . ' ' . $texto;
            } else {
                $texto = 'COP';
                echo 'Total ' . $total[0]->price . ' ' . $texto;
            }
        }

        return false;
    }

}

public function facturas()
{
    $array['facturas'] = $this->gm->listFacturas();
    $this->load->view('facturas', $array);
}

public function preferencias()
{
    $array['programas'] = $this->gm->listProgramas();
}

```

```

        $array['electrodomesticos'] = $this->gm-
>listElectrodomesticos();
        $array['preferencias'] = $this->gm->traerPreferencias($this-
>session->userdata('id'));
        $this->load->view('preferencias', $array);

    }

    public function crear_factura()
    {

        $array = array(

            'id' => $this->session->userdata('id'),
            'varNumeroCuenta' => $this->input->post('cuenta'),
            'varClaseServicio' => $this->input->post('servicio'),
            'datePeriodoIni' => $this->input->post('inicio'),
            'datePeriodoFin' => $this->input->post('fin'),
            'intLecturaActual' => $this->input->post('lectura_actual'),
            'intLecturaAnterior' => $this->input-
>post('lectura_anterior'),
            'intTotalConsumo' => $this->input->post('total_consumo'),
            'floCPT' => $this->input->post('CPT'),
            'floValorkWh' => $this->input->post('kwh'),
            'intTotalPago' => $this->input->post('total_pago')
        );

        $id = $this->gm->crearFactura($array);

        echo "<p><a class='pull-left editarFactura' data-id='". $id."'
style='color: #f5f5f5; cursor: pointer;'> Factura Número: " .
$array['varNumeroCuenta'] . "</a></p>";

    }

    public function cargar_factura(){

        $array['factura'] = $this->gm->cargarFactura($this->input-
>post('cuenta'));
        $this->load->view('editar_facturas', $array);

    }

    public function modificar_factura(){

        $array = array(

            'id' => $this->session->userdata('id'),
            'varNumeroCuenta' => $this->input->post('cuenta'),
            'varClaseServicio' => $this->input->post('servicio'),
            'datePeriodoIni' => $this->input->post('inicio'),
            'datePeriodoFin' => $this->input->post('fin'),
            'intLecturaActual' => $this->input->post('lectura_actual'),

```

```

        'intLecturaAnterior' => $this->input->post('lectura_anterior'),
        'intTotalConsumo' => $this->input->post('total_consumo'),
        'floCPT' => $this->input->post('CPT'),
        'floValorkWh' => $this->input->post('kwh'),
        'intTotalPago' => $this->input->post('total_pago')
    );

    $id = $this->input->post('id');

    $this->gm->modificarFactura($array,$id);

}

public function eliminar_factura()
{
    $this->gm->eliminarFactura($this->input->post('id'));
}

public function crear_preferencia(){
    $array = array(
        'id_user' => $this->session->userdata('id'),
        'programa' => $this->input->post('programa'),
        'electrodomesticos' => $this->input->post('hiddenRadios'),
        'dia' => $this->input->post('hiddenDay'),
        'horainicio' => $this->input->post('horainicio'),
        'horafinal' => $this->input->post('horafin'),
        'reduccion' => $this->input->post('reduccion'),
        'valor' => $this->input->post('valor')
    );
    if ($this->input->post('idexist')) {
        $result = $this->gm->modificarPreferencia($array, $this->input->post('idexist'));
    }else{
        $result = $this->gm->crearPreferencia($array);
    }
}

public function juego(){

    $array['preferencias'] = $this->gm->traerPreferencias($this->session->userdata('id'));
    $array['electrodomesticos'] = $this->gm->listElectrodomesticos();
    $this->load->view('juego',$array);

}

public function sumar_numeros(){

```

```

        $numeroActual = $this->input->post('prueba');
        echo $numeroActual + 10;
    }

    public function modelos() {
        $array['preferencias'] = $this->gm->traerPreferencias($this->session->userdata('id'));
        $this->load->view('modelos', $array);
    }

    public function perfil() {
        $array['user'] = $this->gm->datosuser($this->session->userdata('id'));
        $array['ciudades'] = $this->gm->getCities();
        $this->load->view('perfil', $array);
    }

    public function preferenciadia() {
        $data['prefdia'] = $this->gm2->consultaDia($this->input->post('dia'));
        $this->output->set_content_type('application/json')->set_output(json_encode($data));
    }
}

```

### B4.3 Control usuario administrador

Este controlador recolecta y muestra toda la información referente a los parámetros del sistema.

```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

use Ghunti\HighchartsPHP\Highchart;
use Ghunti\HighchartsPHP\HighchartJsExpr;
use PHPExcel as PHPExcel;

class HomeAdmin extends CI_Controller {

    public function __construct()
    {
        //$_SESSION = array();

        parent::__construct();
        $this->load->model('graphic_model', 'gm');
        $this->load->model('graphic_model2', 'gm2');
        $this->load->library('session');

        if($this->session->userdata('id') == FALSE) {
            redirect(base_url());
        }
    }
}

```

```

    }
    if($this->session->userdata('rol') == 2){
        redirect(base_url().'index.php/Home');
    }
}

public function index()
{
    $array['sensor'] = $this->gm->sensores($this->session->
    >userdata('id'));
    $this->load->view('homeAdmin', $array);
}

public function sensores(){
    $array['sensores'] = $this->gm2->listSensores();
    $array['electrodomesticos'] = $this->gm-
    >listElectrodomesticos();
    $array['usuarios'] = $this->gm2->listUsers();
    $this->load->view('sensores', $array);
}

public function crear_sensor(){
    $array = array(
        'Serial' => $this->input->post('serial'),
        'id_electrodomestico' => $this->input-
    >post('electrodomestico'),
        'id_user' => $this->input->post('usuario')
    );
    $id = $this->gm2->crearSensor($array);
    echo "<p><a class='pull-left editarSensor' data-id='".
    $array['Serial'] .' ' style='color: #f5f5f5; cursor: pointer;'> Sensor: "
    . $array['Serial'] . "</a></p>";
}

public function cargar_sensor(){
    $array['sensor'] = $this->gm2->cargarSensor($this->input-
    >post('sensor'));
    $this->output->set_content_type('application/json')-
    >set_output(json_encode($array['sensor']));
}

public function cargar_iduser_doc(){

```

```

        $iduserdoc = $this->gm2->iduser($this->input-
>post('documento'));
        $this->output->set_content_type('application/json')-
>set_output(json_encode($iduserdoc));

    }

    public function cargar_iduser(){

        $userdoc = $this->gm2->docuser($this->input->post('id'));
        $this->output->set_content_type('application/json')-
>set_output(json_encode($userdoc));

    }

    public function modificar_sensor(){

        $array = array(

            'Serial' => $this->input->post('serial'),
            'id_electrodomestico' => $this->input-
>post('electrodomestico'),
            'id_user' => $this->input->post('usuario')
        );

        $id = $this->input->post('serial');
        $this->gm2->modificarSensor($array,$id);

    }

    public function eliminar_sensor(){
        $this->gm2->eliminarSensor($this->input->post('id'));
    }

    public function electrodomesticos(){
        $array['electrodomesticos'] = $this->gm-
>listElectrodomesticos();
        $this->load->view('electrodomesticos', $array);

    }

    public function crear_electrodomestico(){

        $array = array(

            'varElectrodomestico' => $this->input-
>post('electrodomestico'),
            'intModelo' => $this->input->post('modelo'),
            'varClaseEnergetica' => $this->input-
>post('clase_energetica'),
            'varMarca' => $this->input->post('marca'),
            'floConsumo' => $this->input->post('consumo'),
            'intVoltajeOp' => $this->input->post('voltaje')
        );
    }

```



```

        $id = $this->gm2->crearElectrodomestico($array);

        echo "<p><a class='pull-left editarSensor' data-id='". $id ."'"
style='color: #f5f5f5; cursor: pointer;'> Sensor: " .
$array['varElectrodomestico'] . "</a></p>";
    }

    public function cargar_electrodomestico(){

        $array['electrodomestico'] = $this->gm2-
>cargarElectrodomestico($this->input->post('electrodomestico'));
        $this->output->set_content_type('application/json')-
>set_output(json_encode($array['electrodomestico']));
    }

    public function modificar_electrodomestico(){

        $array = array(
            'varElectrodomestico' => $this->input-
>post('electrodomestico'),
            'intModelo' => $this->input->post('modelo'),
            'varClaseEnergetica' => $this->input-
>post('clase_energetica'),
            'varMarca' => $this->input->post('marca'),
            'floConsumo' => $this->input->post('consumo'),
            'intVoltajeOp' => $this->input->post('voltaje')
        );

        $id = $this->input->post('id');
        $this->gm2->modificarElectrodomestico($array,$id);
    }

    public function eliminar_electrodomestico(){
        $this->gm2->eliminarElectrodomestico($this->input->post('id'));
    }

    public function programas(){
        $array['programas'] = $this->gm2->listProgramas();
        $this->load->view('programas', $array);
    }

    public function crear_programa(){

        $array = array(
            'varPrograma' => $this->input->post('programa'),
            'floPrecio' => $this->input->post('precio'),
            'floConsumo' => $this->input->post('consumo'),
            'txtDescripcion' => $this->input->post('descripcion')
        );
    }

```

```

        $sid = $this->gm2->crearPrograma($array);

        echo "<p><a class='pull-left editarPrograma' data-id='". $sid ."'
style='color: #f5f5f5; cursor: pointer;'> Programa: " .
$array['varPrograma'] . "</a></p>";
    }

    public function cargar_programa() {

        $array['programa'] = $this->gm2->cargarPrograma($this->input-
>post('programa'));
        $this->output->set_content_type('application/json')-
>set_output(json_encode($array['programa']));
    }

    public function modificar_programa() {

        $array = array(

            'varPrograma' => $this->input->post('programa'),
            'floPrecio' => $this->input->post('precio'),
            'floConsumo' => $this->input->post('consumo'),
            'txtDescripcion' => $this->input->post('descripcion')
        );

        $sid = $this->input->post('id');
        $this->gm2->modificarPrograma($array,$sid);
    }

    public function eliminar_programa() {
        $this->gm2->eliminarPrograma($this->input->post('id'));
    }

    public function usuarios() {
        $array['ciudades'] = $this->gm->getCities();
        $array['roles'] = $this->gm->getroles();
        $this->load->view('usuarios', $array);
    }

    public function cargar_docs() {

        $array['documentos'] = $this->gm->docsusers();
        $this->output->set_content_type('application/json')-
>set_output(json_encode($array['documentos']));
    }

    public function select_user() {
        $array['user'] = $this->gm->datosuserdoc($this->input-
>post('id'));
    }

```

```

        $this->output->set_content_type('application/json')-
>set_output(json_encode($array['user']));
    }

    public function juego() {
        $array['electrodomesticos'] = $this->gm-
>listElectrodomesticos();
        $this->load->view('juego2', $array);
    }

    public function calculo_juego() {

        $dia = $this->input->post('dia');
        //var_dump($this->input->post());
        $dia2 = $this->input->post('dia2');

//asignar los valores que llegan en variables

        $porcentaje = $this->input->post('porcentaje');

// Se asigna las variables para los Spinner

        $primeraelec = $this->input->post('electrodomestico1');
        $segundaelec = $this->input->post('electrodomestico2');
        $terceraelec = $this->input->post('electrodomestico3');
        $cuartaelec = $this->input->post('electrodomestico4');
        $quintaelec = $this->input->post('electrodomestico5');

// Se asigna las variables de las horas

        $hora1a =$this->input->post('hora1a');
        $hora1b =$this->input->post('hora1b');
        $hora1c =$this->input->post('hora1c');
        $hora1d =$this->input->post('hora1d');
        $hora1e =$this->input->post('hora1e');
        $hora1f =$this->input->post('hora1f');

        $hora2a =$this->input->post('hora2a');
        $hora2b =$this->input->post('hora2b');
        $hora2c =$this->input->post('hora2c');
        $hora2d =$this->input->post('hora2d');
        $hora2e =$this->input->post('hora2e');
        $hora2f =$this->input->post('hora2f');

        $hora3a =$this->input->post('hora3a');
        $hora3b =$this->input->post('hora3b');
        $hora3c =$this->input->post('hora3c');
        $hora3d =$this->input->post('hora3d');
        $hora3e =$this->input->post('hora3e');
        $hora3f =$this->input->post('hora3f');

        $hora4a =$this->input->post('hora4a');
        $hora4b =$this->input->post('hora4b');

```

```
$hora4c =$this->input->post('hora4c');
$hora4d =$this->input->post('hora4d');
$hora4e =$this->input->post('hora4e');
$hora4f =$this->input->post('hora4f');
```

```
$hora5a =$this->input->post('hora5a');
$hora5b =$this->input->post('hora5b');
$hora5c =$this->input->post('hora5c');
$hora5d =$this->input->post('hora5d');
$hora5e =$this->input->post('hora5e');
$hora5f =$this->input->post('hora5f');
//var_dump($this->input->post());
```

```
//Primer sensor
```

```
$result1 = $this->gm2->sumaSensor($dia, $dia2, 'sensor1');
if(!empty($result1)){
    $primero_comsumo = $result1;
    // $db = cerrar();
else{
    $primero_comsumo = 0;
}
```

```
//Segundo sensor
```

```
$result2 = $this->gm2->sumaSensor($dia, $dia2, 'sensor2');
if(!empty($result2)){
    $segundo_comsumo = $result2;
    // $db = cerrar();
else{
    $segundo_comsumo = 0;
}
```

```
//Tercer Sensor
```

```
$result3 = $this->gm2->sumaSensor($dia, $dia2, 'sensor3');
if(!empty($result3)){
    $tercer_comsumo = $result3;
    // $db = cerrar();
else{
    $tercer_comsumo = 0;
}
```

```
//Cuarto Sensor
```

```
$result4 = $this->gm2->sumaSensor($dia, $dia2, 'sensor4');
if(!empty($result4)){
    $cuarto_comsumo = $result4;
    // $db = cerrar();
}
```

```

    }else{
        $cuarto_comsumo = 0;
    }

    //Quineto Sensor

    $result5 = $this->gm2->sumaSensor($dia, $dia2, 'sensor5');
    if(!empty($result5)){

        $quinto_comsumo = $result5;
    }else{
        $quinto_comsumo = 0;
    }

    //var_dump($quinto_comsumo);
    // Declaración del valor default del consumo
    $consumo = "4.7546433";
    $consumo_teorico = ($primero_comsumo + $segundo_comsumo +
$tercer_comsumo + $cuarto_comsumo + $quinto_comsumo);

    // Se calcula la reducción

    $valor_porcentaje = ($porcentaje/100);
    $valor_previo = ($valor_porcentaje*$consumo);
    $valor_final = ($consumo - $valor_previo);

// Primera elección

if($primeraelec == 'Televisor'){

    $sumatorial = (($horalb - $horala) + ($horald - $horalc) +
($horalf - $horale));
    $valor_consumo1= ($sumatorial * 0.063479);

}elseif ($primeraelec == 'Nevera'){

    $sumatorial = (($horalb - $horala) + ($horald - $horalc) +
($horalf - $horale));
    $valor_consumo1= ($sumatorial * 0.1554345);

}elseif ($primeraelec == 'Lavadora'){

    $sumatorial = (($horalb - $horala) + ($horald - $horalc) +
($horalf - $horale));
    $valor_consumo1= ($sumatorial * 0.3053246);

}elseif ($primeraelec == 'Lampara'){

```

```

    $sumatoria1 = (($hora1b - $hora1a) + ($hora1d - $hora1c) +
($hora1f - $hora1e));
    $valor_consumo1= ($sumatoria1 * 0.0246618);

}else{

    $sumatoria1 = (($hora1b - $hora1a) + ($hora1d - $hora1c) +
($hora1f - $hora1e));
    $valor_consumo1= ($sumatoria1 * 0.1649808);

}

// Segunda elección

if($segundaelec == 'Televisor'){

    $sumatoria2 = (($hora2b - $hora2a) + ($hora2d - $hora2c) +
($hora2f - $hora2e));
    $valor_consumo2= ($sumatoria2 * 0.063479);

}elseif ($segundaelec == 'Nevera'){

    $sumatoria2 = (($hora2b - $hora2a) + ($hora2d - $hora2c) +
($hora2f - $hora2e));
    $valor_consumo2 = ($sumatoria2 * 0.1554345);

}elseif ($segundaelec == 'Lavadora'){

    $sumatoria2 = (($hora2b - $hora2a) + ($hora2d - $hora2c) +
($hora2f - $hora2e));
    $valor_consumo2= ($sumatoria2 * 0.3053246);

}elseif ($segundaelec == 'Lampara'){

    $sumatoria2 = (($hora2b - $hora2a) + ($hora2d - $hora2c) +
($hora2f - $hora2e));
    $valor_consumo2= ($sumatoria2 * 0.0246618);

}else{

    $sumatoria2 = (($hora2b - $hora2a) + ($hora2d - $hora2c) +
($hora2f - $hora2e));

```

```

    $valor_consumo2= ($sumatoria2 * 0.1649808);

}

// Tercera elección
if($terceraelec == 'Televisor'){

    $sumatoria3 = (($hora3b - $hora3a) + ($hora3d - $hora3c) +
($hora3f - $hora3e));
    $valor_consumo3= ($sumatoria3 * 0.063479);

}elseif ($terceraelec == 'Nevera'){

    $sumatoria3 = (($hora3b - $hora3a) + ($hora3d - $hora3c) +
($hora3f - $hora3e));
    $valor_consumo3 = ($sumatoria3 * 0.1554345);

}elseif ($terceraelec == 'Lavadora'){

    $sumatoria3 = (($hora3b - $hora3a) + ($hora3d - $hora3c) +
($hora3f - $hora3e));
    $valor_consumo3 = ($sumatoria3 * 0.3053246);

}elseif ($terceraelec == 'Lampara'){

    $sumatoria3 = (($hora3b - $hora3a) + ($hora3d - $hora3c) +
($hora3f - $hora3e));
    $valor_consumo3 = ($sumatoria3 * 0.0246618);

}else{

    $sumatoria3 = (($hora3b - $hora3a) + ($hora3d - $hora3c) +
($hora3f - $hora3e));
    $valor_consumo3= ($sumatoria3 * 0.1649808);

}

// Cuarta elección
if($cuartaelec == 'Televisor'){

```

```

    $sumatoria4 = (($hora4b - $hora4a) + ($hora4d - $hora4c) +
($hora4f - $hora4e));
    $valor_consumo4= ($sumatoria4 * 0.063479);

}elseif ($cuartaelec == 'Nevera'){

    $sumatoria4 = (($hora4b - $hora4a) + ($hora4d - $hora4c) +
($hora4f - $hora4e));
    $valor_consumo4 = ($sumatoria4 * 0.1554345);

}elseif ($cuartaelec == 'Lavadora'){

    $sumatoria4 = (($hora4b - $hora4a) + ($hora4d - $hora4c) +
($hora4f - $hora4e));
    $valor_consumo4= ($sumatoria4 * 0.3053246);

}elseif ($cuartaelec == 'Lampara'){

    $sumatoria4 = (($hora4b - $hora4a) + ($hora4d - $hora4c) +
($hora4f - $hora4e));
    $valor_consumo4= ($sumatoria4 * 0.0246618);

}else{

    $sumatoria4 = (($hora4b - $hora4a) + ($hora4d - $hora4c) +
($hora4f - $hora4e));
    $valor_consumo4= ($sumatoria4 * 0.1649808);

}

// Quinta elección
if($quintaelec == 'Televisor'){

    $sumatoria5 = (($hora5b - $hora5a) + ($hora5d - $hora5c) +
($hora5f - $hora5e));
    $valor_consumo5= ($sumatoria5 * 0.063479);

}elseif ($quintaelec == 'Nevera'){

    $sumatoria5 = (($hora5b - $hora5a) + ($hora5d - $hora5c) +
($hora5f - $hora5e));
    $valor_consumo5 = ($sumatoria5 * 0.1554345);

```



```

}elseif ($quintaelec == 'Lavadora'){

    $sumatoria5 = (($hora5b - $hora5a) + ($hora5d - $hora5c) +
($hora5f - $hora5e));
    $valor_consumo5 = ($sumatoria5 * 0.3053246);

}elseif ($quintaelec == 'Lampara'){

    $sumatoria5 = (($hora5b - $hora5a) + ($hora5d - $hora5c) +
($hora5f - $hora5e));
    $valor_consumo5 = ($sumatoria5 * 0.0246618);

}else{

    $sumatoria5 = (($hora5b - $hora5a) + ($hora5d - $hora5c) +
($hora5f - $hora5e));
    $valor_consumo5 = ($sumatoria5 * 0.1649808);

}

$sumatoria_total = $valor_consumo1 + $valor_consumo2 + $valor_consumo3 +
$valor_consumo4 + $valor_consumo5;

if ($sumatoria_total <= $valor_final){

    $ahorro = ($consumo - $sumatoria_total);
    $price = ($ahorro*400);
    $bit = "uno";

    $response["con"] = 10;
    $response["message"] = "Excelente! Lograste ahorrar";
    $response["success"] = $ahorro;
    $response["check"] = $price;

    //echo json_encode(array('respuesta'=> $response));
    $this->output->set content type('application/json')-
>set_output(json_encode($response));

}elseif ($sumatoria_total > $valor_final){

    $ahorro = ($sumatoria_total - $valor_final);
    $price = ($ahorro*400);
    $bit = "dos";

    $response["con"] = 20;

```

```

        $response["message"] = "No lograste ahorrar el valor
pensado!";
        $response["success"] = $ahorro;
        $response["check"] = $price;

        //$response["loco"] = $consumo_teorico;

        /*$response["primero"] = $valor_consumo1;
        $response["segundo"] = $valor_consumo2;
        $response["tercero"] = $valor_consumo3;
        $response["cuarto"] = $valor_consumo4;
        $response["quinto"] = $valor_consumo5;
        $response["sexto"] = $sumatoria_total;
        $response["septimo"] = $valor_final;
        $response["octavo"] = $consumo;*/

        //echo json_encode(array('respuesta'=> $response));
        $this->output->set_content_type('application/json')-
>set_output(json_encode($response));
    }

}
}

```

#### B4.4 Control nuevo usuario

Este controlador hace uso de las clases creadas en los modelos para permitirse subir la información a la base de datos.

```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Newuser extends CI_Controller {

    public function __construct()
    {
        parent::__construct();
        $this->load->model('graphic_model', 'gm');
        $this->load->library('session');
    }

    public function index()
    {
        $array['ciudades'] = $this->gm->getCities();
        $this->load->view('new_user', $array);
    }
}

```

```

    public function change_city(){

        $localidades = $this->gm->getLocations($this->input-
>post('ciudad'));

        $html = '<option value="">Selecione</option>';
        foreach ($localidades as $localidad) {
            $html .= '<option value="'. $localidad-
>intLocalidad.'">' . $localidad->varLocalidad.'"</option>';
        }
        echo ($html);
    }

    public function change_location(){

        $neighborhoods = $this->gm->getNeighborhoods($this->input-
>post('localidad'));
        $html = '<option value="">Selecione</option>';
        foreach ($neighborhoods as $neighborhood) {
            $html .= '<option value="'. $neighborhood-
>intBarrio.'">' . $neighborhood->varBarrio.'"</option>';
        }
        echo ($html);
    }

    public function create_user()
    {
        $array = array(
            'username' => $this->input->post('username'),
            'password' => $this->input->post('password'),
            'nombre' => $this->input->post('nombre'),
            'apellido' => $this->input->post('apellido'),
            'documento' => $this->input->post('documento'),
            'direccion' => $this->input->post('direccion'),
            'telefono' => $this->input->post('telefono'),
            'estrato' => $this->input->post('estrato'),
            'varEmail' => $this->input->post('email'),
            'varBarrio' => $this->input->post('barrio'),
            'varLocalidad' => $this->input->post('localidad'),
            'varCiudad' => $this->input->post('ciudad'),
            'intEstado' => 1,
            'rol' => 2
        );

        $this->gm->signup($array);
    }

    public function alter_user(){
        $array = array(

```

```

        'username' => $this->input->post('username'),
        'nombre' => $this->input->post('nombre'),
        'apellido' => $this->input->post('apellido'),
        'documento' => $this->input->post('documento'),
        'direccion' => $this->input->post('direccion'),
        'telefono' => $this->input->post('telefono'),
        'estrato' => $this->input->post('estrato'),
        'varEmail' => $this->input->post('email'),
        'varBarrio' => $this->input->post('barrio'),
        'varLocalidad' => $this->input->post('localidad'),
        'varCiudad' => $this->input->post('ciudad'),
        'intEstado' => $this->input->post('estado'),
        'rol' => $this->input->post('rol')
    );
    $id = $this->input->post('id');
    if ($id) {
        $this->gm->submituser($id, $array);
    }else{
        $this->gm->submituser($this->session->userdata('id'),
$array);
    }
}

    public function delete_user() {
        $this->gm->deleteuser($this->input->post('id'));
    }

    public function cambiar_clave(){
        $user = $this->gm->datosuser($this->session->userdata('id'));
        if ($user[0]->password == $this->input->post('contra')) {
            $array = array(
                'password' => $this->input->post('newcontra')
            );
            $this->gm->submitpassword($this->session->userdata('id'),
$array);
            echo json_encode('Password actualizado');
        }else{
            echo json_encode('No coincide el password');
        }
    }
}

```

## B5. Views

Una vez esté el controlador que permite el funcionamiento, se procede a la creación de los campos que recogen y muestran la información en el sitio, mejor conocidos como formularios.

### B5.1 Usuarios

Este formulario cuenta con un campo que se autocompleta para la filtración de los usuarios y su futura modificación y/o eliminación, este es de uso exclusivo del administrador pues cuenta con los campos que habilitan o deshabilitan al usuario y su cambio de rol.

```
<!DOCTYPE html>
<html lang="en">

<head>

    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="">
    <meta name="author" content="">

    <title>Poupawatt</title>

    <!-- Bootstrap Core CSS -->
    <link href="<?php echo base_url()
?>assets/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">

    <!-- Custom Fonts -->
    <link href="<?php echo base_url() ?>assets/vendor/font-
awesome/css/font-awesome.min.css" rel="stylesheet"
        type="text/css">
    <link
href='https://fonts.googleapis.com/css?family=Open+Sans:300italic,400ital
ic,600italic,700italic,800italic,400,300,600,700,800'
        rel='stylesheet' type='text/css'>
    <link
href='https://fonts.googleapis.com/css?family=Merriweather:400,300,300ita
lic,400italic,700,700italic,900,900italic'
        rel='stylesheet' type='text/css'>

    <!-- Plugin CSS -->
    <link href="<?php echo base_url() ?>assets/vendor/magnific-
popup/magnific-popup.css" rel="stylesheet">

    <!-- Theme CSS -->
    <link href="<?php echo base_url() ?>assets/css/creative.min.css"
rel="stylesheet">

    <!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and
media queries -->
    <!-- WARNING: Respond.js doesn't work if you view the page via
file:// -->
    <!--[if lt IE 9]>
    <script
src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
```

```

        <script
src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></scrip
t>
        <![endif]-->

</head>

<body id="page-top">

<nav id="mainNav" class="navbar navbar-default navbar-fixed-top">
    <div class="container-fluid">
        <!-- Brand and toggle get grouped for better mobile display -->
        <div class="navbar-header">
            <button type="button" class="navbar-toggle collapsed" data-
toggle="collapse"
                data-target="#bs-example-navbar-collapse-1">
                <span class="sr-only">Toggle navigation</span> Menu <i
class="fa fa-bars"></i>
            </button>
            <a class="navbar-brand page-scroll" href="{<?php echo
base_url() ?>index.php/HomeAdmin}">Poupawatt</a>
        </div>

        <!-- Collect the nav links, forms, and other content for toggling
-->
        <div class="collapse navbar-collapse" id="bs-example-navbar-
collapse-1">
            <ul class="nav navbar-nav navbar-right">

                <li>
                    <a class="page-scroll" href="{<?php echo base_url()
?>index.php/HomeAdmin/sensores}">Sensores</a>
                </li>
                <li>
                    <a class="page-scroll" href="{<?php echo base_url()
?>index.php/HomeAdmin/electrodomesticos}">Electrodomésticos</a>
                </li>
                <li>
                    <a class="page-scroll" href="{<?php echo base_url()
?>index.php/HomeAdmin/programas}">Programas</a>
                </li>
                <li>
                    <a class="page-scroll" href="{<?php echo base_url()
?>index.php/HomeAdmin/usuarios}">Usuarios</a>
                </li>
                <li>
                    <a class="page-scroll" href="{<?php echo base_url()
?>index.php/Home/salir}">Salir</a>
                </li>

            </ul>
        </div>

```

```

        <!-- /.navbar-collapse -->
    </div>
    <!-- /.container-fluid -->
</nav>

<section class="bg-primary" id="about">
    <div class="container">
        <div class="row">

            <div class=" text-center">
                <h2 class="section-heading">Usuarios</h2>
                <hr class="light" >
                <p><button id="Inst-pref" class="btn btn-success btn-sm
sr-button">Instrucciones >></button></p>
                <div class="col-md-12">
            </div>
            <div id="autocomp" class="">
                <div class="col-md-12">
                    <div class="col-md-2">
                        <input id="userselect" class="form-control
typeahead" type="number" placeholder="Documento" >
                    </div>
                    <div class="col-md-1">
                        <button id="btn-slect" class="btn btn-
success">Selecionar</button>
                    </div>
                    <input type="hidden" id="iddb" value="">
                </div>
                <br><br>
            </div>

            <div class=" text-center">
                <div class="col-md-6">
                    <div class="col-md-12">
                        <label for="" class="pull-left">Username:</label>
                        <input type="text" placeholder="Username"
id="username" class="form-control required" value="" readonly/>
                    </div>

                    <div class="col-md-12">
                        <label for="" class="pull-left">Email:</label>
                        <input type="text" placeholder="Email" id="email"
class="form-control required" value="" readonly/>
                    </div>

                    <div class="col-md-12">
                        <label for="" class="pull-left">Nombre:</label>
                        <input type="text" placeholder="Nombre"
id="nombre" class="form-control required" value="" readonly/>
                    </div>
                    <div class="col-md-12">
                        <label for="" class="pull-left">Apellido:</label>
                        <input type="text" placeholder="Apellido"
id="apellido" class="form-control required" value="" readonly/>
                    </div>
                </div>
            </div>
        </div>
    </div>

```

```

        </div>
        <div class="col-md-12">
            <label for="" class="pull-
left">Documento:</label>
            <input type="text" placeholder="Documento"
id="documento" class="form-control required" value="" readonly/>
        </div>
        <div class="col-md-12">
            <label for="" class="pull-
left">Dirección:</label>
            <input type="text" placeholder="Direccion"
id="direccion" class="form-control required" value="" readonly/>
        </div>
        <div class="col-md-12">
            <br><select name="" id="estado" class="form-
control inputFormulario required">
                <option value="">Seleccione un
Estado</option>
                <option value="1">Habilitado</option>
                <option value="0">Deshabilitado</option>
            </select>
        </div>
    </div>

    <div class="col-md-6">

        <div class="col-md-12">
            <label for="" class="pull-left">Teléfono:</label>
            <input type="text" placeholder="Telefono"
id="telefono" class="form-control required" value="" readonly/>
        </div>
        <div class="col-md-12">
            <label for="" class="pull-left">Estrato:</label>
            <select id="estrato" class="form-control select-
dropdown required">
                <option value="">Seleccione Estrato</option>
                <option value="1">Estrato 1</option>
                <option value="2">Estrato 2</option>
                <option value="3">Estrato 3</option>
                <option value="4">Estrato 4</option>
                <option value="5">Estrato 5</option>
                <option value="6">Estrato 6</option>
            </select>
            <input type="hidden" id="estratodb" value="">
        </div>
        <div class="col-md-12">
            <label for="" class="pull-left">Ciudad:</label>
            <select id="ciudad" class="form-control select-
dropdown required">
                <option value="">Seleccione Ciudad</option>

                <?php foreach ($ciudades as $ciudad) {
                    ?>

```



```

                                <option value="<?php echo $ciudad-
>intContador ?>"><?php echo $ciudad->varCiudad; ?></option>
                                <?php
                                } ?>
                                </select>
                                <input type="hidden" id="ciudaddb" value="">
                                </div>
                                <div class="col-md-12">
                                <label for="" class="pull-
left">Localidad:</label>
                                <select id="localidad" class="form-control
select-dropdown required">

                                </select>
                                <input type="hidden" id="localidaddb" value="">
                                </div>
                                <div class="col-md-12">
                                <label for="" class="pull-left">Barrio:</label>
                                <select id="barrio" class="form-control select-
dropdown required">

                                </select>
                                <input type="hidden" id="barriodb" value="">
                                </div>
                                <div class="col-md-12">
                                <br><select name="" id="rol" class="form-control
inputFormulario required">
                                <option value="">Seleccione un ROL</option>
                                <?php if ($roles !== false) {
                                foreach ($roles as $rol) {
                                ?>
                                <option
value=<?php if ($rol !== null && $rol->id !== null) {echo ($rol->id);}
                                ?><?php if ($rol !== null && $rol-
>descripcion !== null) {echo ($rol->descripcion);}
                                ?></option>

                                <?php }}
                                ?>
                                </select>
                                </div>
                                </div>
                                <div class="col-md-12">
                                <hr class="light"></hr>
                                <button id="btn-actualizar" class="page-scroll
btn btn-default btn-xl sr-button">Actualizar</button>
                                <button id="btn-eliminar" class="page-scroll btn
btn-danger btn-xl sr-button">Eliminar</button>
                                <button id="btn-confirm" class="page-scroll btn
btn-success btn-xl sr-button">Confirmar</button>
                                <button id="btn-cancel" class="page-scroll btn
btn-danger btn-xl sr-button">Cancelar</button>
                                </div>

```

```

        </div>
    </div>
</div>
</section>

</body>

</html>

<!-- Modal -->
<div id="ModalResp" class="modal fade">
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-
dismiss="modal">&times;</button>
                <h4 id="modal-title" class="modal-title"></h4>
            </div>
            <!-- dialog buttons -->
            <div class="modal-footer">
                <button type="button" class="btn btn-danger" data-
dismiss="modal">CERRAR</button>
            </div>
            <input type="hidden" name="id_grav_delete" value="">
        </div>
    </div>
</div>
<!-- /.modal -->

<!-- Modal Instrucciones-->
<div id="ModalIns" class="modal fade">
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-
dismiss="modal">&times;</button>
                <h4 id="modal-title" class="modal-title">Instrucciones</h4>
            </div>
            <div class="modal-Body">
                <p>Instrucciones aqui</p>
            </div>
            <!-- dialog buttons -->
            <div class="modal-footer">
                <button type="button" class="btn btn-success" data-
dismiss="modal">Aceptar</button>
            </div>
        </div>
    </div>
</div>
<!-- /.modal -->

<div id="ModalDeleteUser" class="modal fade">
    <div class="modal-dialog">
        <div class="modal-content">

```

```

        <!-- dialog body -->
        <div class="modal-body">
            <button type="button" class="close" data-
dismiss="modal">&times;</button>
                Está a punto de eliminar un usuario de forma permanente. ¿Está
seguro (a) ?
            </div>
        <!-- dialog buttons -->
        <div class="modal-footer"><button type="button" class="btn btn-
success" id='btn-confirm-delete'>CONFIRMAR</button><button type="button"
class="btn btn-danger" data-dismiss="modal">CANCELAR</button></div>
            <form id="form_delete_inventario">
                <input type="hidden" name="id_user_delete" id="id_user_delete"
value="">
            </form>
        </div>
    </div>
</div>

```

```

<script src="<?php echo base_url()
?>assets/vendor/jquery/jquery.min.js"></script>
<!-- Bootstrap Core JavaScript -->
<script src="<?php echo base_url()
?>assets/vendor/bootstrap/js/bootstrap.min.js"></script>
<script src="<?php echo base_url()
?>assets/js/typeahead.jquery.min.js"></script>

```

```

<script>

```

```

    $('#estrato').attr('disabled', true);
    $('#ciudad').attr('disabled', true);
    $('#localidad').attr('disabled', true);
    $('#barrio').attr('disabled', true);
    $('#estado').attr('disabled', true);
    $('#rol').attr('disabled', true);
    $('#btn-actualizar').hide();
    $('#btn-eliminar').hide();
    $('#btn-cancel').hide();
    $('#btn-confirm').hide();
    $('#ModalIns').modal('show');

    $('#Inst-pref').click(function (e) {
        $('#ModalIns').modal('show');
    });

    $('#btn-slect').click(function (e) {
        $.post("<?php echo base_url() ?>index.php/HomeAdmin/select_user",
{
            id: $('#userselect').val()
        }, function (data) {
            if (data[0] != null) {
                $('#iddb').val(data[0].intNivel);
                $('#username').val(data[0].username);
            }
        });
    });

```

```

        $('#email').val(data[0].varEmail);
        $('#nombre').val(data[0].nombre);
        $('#apellido').val(data[0].apellido);
        $('#documento').val(data[0].documento);
        $('#direccion').val(data[0].direccion);
        $('#telefono').val(data[0].telefono);
        $('#estrato').val(data[0].estrato);
        $('#ciudaddb').val(data[0].varCiudad);
        $('#localidaddb').val(data[0].varLocalidad);
        $('#barriodb').val(data[0].varBarrio);
        $('#estado').val(data[0].intEstado);
        $('#rol').val(data[0].rol);

        if ($('#ciudaddb').val()) {
            $('#ciudad').val($('#ciudaddb').val());
            $.post("<?php echo base_url ()
?>index.php/Newuser/change_city", {
                ciudad: $('#ciudaddb').val()
            }, function (data) {
                $('#localidad').append(data);
                $('#localidad').val( $('#localidaddb').val());

            })
                .fail(function () {
                    alert("Error en la conexión");
                });
        };

        if ($('#localidaddb').val()) {
            $('#barrio').empty();

            $.post("<?php echo base_url ()
?>index.php/Newuser/change_location", {
                localidad: $('#localidaddb').val()
            }, function (data) {
                $('#barrio').append(data);
                $('#barrio').val( $('#barriodb').val());

            })
                .fail(function () {
                    alert("Error en la conexión");
                });
        };

        $('#btn-actualizar').show();
        $('#btn-eliminar').show();
    }else{
        alert("Usuario Invalido");
    };

    })
        .fail(function () {
            alert("Error en la conexion");
        });

```

```
});  
});
```

```
$('#btn-actualizar').click(function (e) {  
    if ($('#iddb').val()) {  
        $('#username').attr("readonly", false);  
        $('#nombre').attr("readonly", false);  
        $('#apellido').attr("readonly", false);  
        $('#documento').attr("readonly", false);  
        $('#direccion').attr("readonly", false);  
        $('#telefono').attr("readonly", false);  
        $('#estrato').attr('disabled', false);  
        $('#ciudad').attr('disabled', false);  
        $('#localidad').attr('disabled', false);  
        $('#barrio').attr('disabled', false);  
        $('#estado').attr('disabled', false);  
        $('#rol').attr('disabled', false);  
        $('#btn-cancel').show();  
        $('#btn-confirm').show();  
        $('#btn-actualizar').hide();  
    }else{  
        alert('Seleccione un usuario');  
    }  
});
```

```
});
```

```
$('#btn-eliminar').click(function (e) {  
    $('#ModalDeleteUser').modal('show');  
    $('#id_user_delete').val($('#iddb').val());  
});
```

```
$('#btn-confirm-delete').click(function (e) {  
    $.post("<?php echo base_url() ?>index.php/Newuser/delete_user", {  
        id: $('#iddb').val()  
    }, function (data) {  
        alert("Usuario Eliminado!");  
        location.reload();  
    })  
    .fail(function () {  
        alert("Error en la conexion");  
    });  
});
```

```
$('#btn-cancel').click(function (e) {  
    location.reload();  
});
```

```
$('#btn-confirm').click(function (e) {  
    e.preventDefault();
```

```
    var validator = 1;
```

```
    $(".required").each(function (index) {  
        if ($(this).val() == '' ) {
```

```

        $(this).addClass('input-required');
        validator = 0;
    }

});

if (validator == 0) {
    alert('Completa los campos en rojo');
    return false;
}

$.post("<?php echo base_url() ?>index.php/Newuser/alter_user", {
    id: $('#iddb').val(),
    username: $('#username').val(),
    email: $('#email').val(),
    nombre: $('#nombre').val(),
    apellido: $('#apellido').val(),
    direccion: $('#direccion').val(),
    documento: $('#documento').val(),
    password: $('#password').val(),
    telefono: $('#telefono').val(),
    estrato: $('#estrato').val(),
    ciudad: $('#ciudad').val(),
    localidad: $('#localidad').val(),
    barrio: $('#barrio').val(),
    estado: $('#estado').val(),
    rol: $('#rol').val()
}, function (data) {
    alert("Datos Modificados!");
    location.reload();
})
.fail(function () {
    alert("Error en la conexion");
});
});

$('#ciudad').change(function () {
    $('#localidad').show().empty();
    $('#barrio').show().empty();

    $.post("<?php echo base_url() ?>index.php/Newuser/change_city", {
        ciudad: $('#ciudad').val()
    }, function (data) {
        $('#localidad').append(data);
    })
    .fail(function () {
        alert("Error en la conexión");
    })
});

$('#localidad').change(function () {
    $('#barrio').empty();

```

```

$.post("<?php echo base_url()
?>index.php/Newuser/change_location", {
    localidad: $('#localidad').val()
}, function (data) {
    $('#barrio').append(data);
})
.fail(function () {
    alert("Error en la conexión");
});
});

var substringMatcher = function(strs) {
    return function findMatches(q, cb) {
        var matches, substrRegex;

        // an array that will be populated with substring matches
        matches = [];

        // regex used to determine if a string contains the substring `q`
        substrRegex = new RegExp(q, 'i');

        // iterate through the pool of strings and for any string that
        // contains the substring `q`, add it to the `matches` array
        $.each(strs, function(i, str) {
            if (substrRegex.test(str)) {
                matches.push(str);
            }
        });

        cb(matches);
    };
};

var documentos = [];
$.post("<?php echo base_url() ?>index.php/HomeAdmin/cargar_docs", {
    localidad: $('#localidad').val()
}, function (data) {
    $.each(data, function( index) {
        documentos.push(data[index].documento);
    });
})
.fail(function () {
    alert("Error en la conexión");
});

$('#autocomp .typeahead').typeahead({
    hint: true,
    highlight: true,
    minLength: 1
},
{
    name: 'documentos',
    source: substringMatcher(documentos)
});

```

```
</script>
```

## B5.2 Perfil

Este formulario parte del anterior, pero en este no se habilitan los campos del administrador pues este es para el usuario estándar.

```
<!DOCTYPE html>  
<html lang="en">
```

```
<head>
```

```
<meta charset="utf-8">  
<meta http-equiv="X-UA-Compatible" content="IE=edge">  
<meta name="viewport" content="width=device-width, initial-scale=1">  
<meta name="description" content="">  
<meta name="author" content="">  
  
<title>Poupawatt</title>  
  
<!-- Bootstrap Core CSS -->  
<link href="<?php echo base_url() ?>assets/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">  
  
<!-- Custom Fonts -->  
<link href="<?php echo base_url() ?>assets/vendor/font-awesome/css/font-awesome.min.css" rel="stylesheet" type="text/css">  
<link  
href='https://fonts.googleapis.com/css?family=Open+Sans:300italic,400italic,600italic,700italic,800italic,400,300,600,700,800'  
rel='stylesheet' type='text/css'>  
<link  
href='https://fonts.googleapis.com/css?family=Merriweather:400,300,300italic,400italic,700,700italic,900,900italic'  
rel='stylesheet' type='text/css'>  
  
<!-- Plugin CSS -->  
<link href="<?php echo base_url() ?>assets/vendor/magnific-popup/magnific-popup.css" rel="stylesheet">  
  
<!-- Theme CSS -->  
<link href="<?php echo base_url() ?>assets/css/creative.min.css" rel="stylesheet">  
  
<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries -->  
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->  
<!--[if lt IE 9]>  
<script  
src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
```



```

        <script
src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></scrip
t>
        <![endif]-->

</head>

<body id="page-top">

<nav id="mainNav" class="navbar navbar-default navbar-fixed-top">
    <div class="container-fluid">
        <!-- Brand and toggle get grouped for better mobile display -->
        <div class="navbar-header">
            <button type="button" class="navbar-toggle collapsed" data-
toggle="collapse"
                data-target="#bs-example-navbar-collapse-1">
                <span class="sr-only">Toggle navigation</span> Menu <i
class="fa fa-bars"></i>
            </button>
            <a class="navbar-brand page-scroll" href="{<?php echo
base_url() ?>index.php/Home}">Poupawatt</a>
        </div>

        <!-- Collect the nav links, forms, and other content for toggling
-->
        <div class="collapse navbar-collapse" id="bs-example-navbar-
collapse-1">
            <ul class="nav navbar-nav navbar-right">

                <li>
                    <a class="page-scroll" href="{<?php echo base_url()
?>index.php/Home#about}">Facturas</a>
                </li>
                <li>
                    <a class="page-scroll" href="{<?php echo base_url()
?>index.php/Home#services}">Preferencias</a>
                </li>
                <li>
                    <a class="page-scroll" href="{<?php echo base_url()
?>index.php/Home#consumo}">Consumo</a>
                </li>
                <li>
                    <a class="page-scroll" href="{<?php echo base_url()
?>index.php/Home#noticias}">Noticias</a>
                </li>
                <li>
                    <a class="page-scroll" href="{<?php echo base_url()
?>index.php/Home/juego}">Juego</a>
                </li>
                <li>
                    <a class="page-scroll" href="{<?php echo base_url()
?>index.php/Home/modelos}">Modelos</a>
                </li>
                <li>

```

```

        <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home/salir">Salir</a>
    </li>

</ul>
</div>
<!-- /.navbar-collapse -->
</div>
<!-- /.container-fluid -->
</nav>

<section class="bg-primary" id="about">
    <div class="container">
        <div class="row">
            <div class="text-center">
                <h2 class="section-heading">Perfil</h2>
                <hr class="light">
                <p><button id="Inst-pref" class="btn btn-success btn-sm
sr-button">Instrucciones >></button></p>
                <div class="col-md-6">
                    <div class="col-md-12">
                        <label for="" class="pull-left">Username:</label>

                        <input type="text" placeholder="Username"
id="username" class="form-control required" value="<?php if
(count($user)) { echo $user[0]->username; }?" readonly/>
                    </div>
                    <div class="col-md-12">
                        <label for="" class="pull-left">Email:</label>

                        <input type="text" placeholder="Email" id="email"
class="form-control required" value="<?php if (count($user)) { echo
$user[0]->varEmail; }?" readonly/>
                    </div>

                    <div class="col-md-12">
                        <label for="" class="pull-left">Nombre:</label>

                        <input type="text" placeholder="Nombre"
id="nombre" class="form-control required" value="<?php if (count($user))
{ echo $user[0]->nombre; }?" readonly/>
                    </div>
                    <div class="col-md-12">
                        <label for="" class="pull-left">Apellido:</label>
                        <input type="text" placeholder="Apellido"
id="apellido" class="form-control required" value="<?php if
(count($user)) { echo $user[0]->apellido; }?" readonly/>
                    </div>
                    <div class="col-md-12">
                        <label for="" class="pull-
left">Documento:</label>

```

```

        <input type="text" placeholder="Documento"
id="documento" class="form-control required" value="<?php if
(count($user)) { echo $user[0]->documento; }?>" readonly/>
    </div>
    <div class="col-md-12">
        <label for="" class="pull-
left">Dirección:</label>
        <input type="text" placeholder="Direccion"
id="direccion" class="form-control required" value="<?php if
(count($user)) { echo $user[0]->direccion; }?>" readonly/>
    </div>
    </div>

    <div class="col-md-6">
        <div class="col-md-12">
            <label for="" class="pull-left">Teléfono:</label>
            <input type="text" placeholder="Telefono"
id="telefono" class="form-control required" value="<?php if
(count($user)) { echo $user[0]->telefono; }?>" readonly/>
        </div>
        <div class="col-md-12">
            <label for="" class="pull-left">Estrato:</label>
            <select id="estrato" class="form-control select-
dropdown required">
                <option value="1">Estrato 1</option>
                <option value="2">Estrato 2</option>
                <option value="3">Estrato 3</option>
                <option value="4">Estrato 4</option>
                <option value="5">Estrato 5</option>
                <option value="6">Estrato 6</option>
            </select>
            <input type="hidden" id="estratodb" value="<?php
if (count($user)) { echo $user[0]->estrato; }?>">
        </div>
        <div class="col-md-12">
            <label for="" class="pull-left">Ciudad:</label>
            <select id="ciudad" class="form-control select-
dropdown required">
                <option value="">Seleccione Ciudad</option>
                <?php foreach ($ciudades as $ciudad) {
                    ?>
                    <option value="<?php echo $ciudad-
>intContador ?>"><?php echo $ciudad->varCiudad; ?></option>
                    <?php
                    } ?>
                </select>
            <input type="hidden" id="ciudaddb" value="<?php
if (count($user)) { echo $user[0]->varCiudad; }?>">
        </div>
        <div class="col-md-12">
            <label for="" class="pull-
left">Localidad:</label>

```

```

                <select id="localidad" class="form-control
select-dropdown required">

                </select>
                <input type="hidden" id="localidaddb"
value="<?php if (count($user)) { echo $user[0]->varLocalidad; }?>">
            </div>
            <div class="col-md-12">
                <label for="" class="pull-left">Barrio:</label>
                <select id="barrio" class="form-control select-
dropdown required">

                </select>
                <input type="hidden" id="barriodb" value="<?php
if (count($user)) { echo $user[0]->varBarrio; }?>">
            </div>
        </div>
        <div class="col-md-12">
            <hr class="light"></hr>
            <button id="btn-actualizar" class="page-scroll
btn btn-default btn-xl sr-button">Actualizar</button>
            <button id="btn-cambio-contraseña" class="page-scroll
btn btn-default btn-xl sr-button">Cambiar Contraseña</button>
            <button id="btn-confirm" class="page-scroll btn
btn-success btn-xl sr-button">Confirmar</button>
            <button id="btn-cancel" class="page-scroll btn
btn-danger btn-xl sr-button">Cancelar</button>
        </div>
    </div>
</div>
</div>
</div>
</section>
</body>
</html>

<div id="Modalcontra" class="modal fade">
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-
dismiss="modal">&times;</button>
                <h4 id="modal-title" class="modal-title">Cambiar
contraseña</h4>
            </div>
            <!-- dialog body -->
            <div class="modal-body">
                <form id="formcontra">
                    <label for="" class="pull-left">Contraseña:</label>
                    <input type="password" placeholder="Contraseña"
id="contra" class="form-control required" value="" />

```

```

        <label for="" class="pull-left">Nueva Contraseña:</label>
        <input type="password" placeholder="Nueva Contraseña"
id="newcontra" class="form-control required" value=""/>

        <label for="" class="pull-left">Confirme Nueva
Contraseña:</label>
        <input type="password" placeholder="Confirme Nueva
Contraseña" id="confnewcontra" class="form-control required" value=""/>
    </form>

</div>
<!-- dialog buttons -->
<div class="modal-footer">
    <button type="button" class="btn btn-success" id='btn-
confirm-contra'>CONFIRMAR</button>
    <button type="button" class="btn btn-danger" data-
dismiss="modal">CANCELAR</button>
</div>
<form id="form_delete_inventario">
    <input type="hidden" name="id_user_delete" id="id_user_delete"
value="">
</form>
</div>
</div>
</div>
<!-- Modal Instrucciones-->
<div id="ModalIns" class="modal fade">
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-
dismiss="modal">&times;</button>
                <h4 id="modal-title" class="modal-title">Instrucciones</h4>
            </div>
            <div class="modal-Body">
                <p>Instrucciones aqui</p>
            </div>
            <!-- dialog buttons -->
            <div class="modal-footer">
                <button type="button" class="btn btn-success" data-
dismiss="modal">Acepta</button>
            </div>
        </div>
    </div>
</div>
<!-- /.modal -->

<script src="<?php echo base_url()
?>assets/vendor/jquery/jquery.min.js"></script>
<script src="<?php echo base_url()
?>assets/vendor/bootstrap/js/bootstrap.min.js"></script>

```

<script>

```
$( window ).load(function() {
    $('#ModalIns').modal('show');
    $('#estrato').attr('disabled', true);
    $('#ciudad').attr('disabled', true);
    $('#localidad').attr('disabled', true);
    $('#barrio').attr('disabled', true);
    $('#btn-cancel').hide();
    $('#btn-confirm').hide();

    if ($('#estratodb').val()) {
        $('#estrato').val( $('#estratodb').val());
    };

    if ($('#ciudaddb').val()) {
        $('#ciudad').val( $('#ciudaddb').val());
        $.post("<?php echo base_url()
?>index.php/Newuser/change_city", {
            ciudad: $('#ciudaddb').val()
        }, function (data) {
            $('#localidad').append(data);
            $('#localidad').val( $('#localidaddb').val());
        });
        .fail(function () {
            alert("Error en la conexión");
        });
    };

    if ($('#localidaddb').val()) {
        $('#barrio').empty();

        $.post("<?php echo base_url()
?>index.php/Newuser/change_location", {
            localidad: $('#localidaddb').val()
        }, function (data) {
            $('#barrio').append(data);
            $('#barrio').val( $('#barriodb').val());
        });
        .fail(function () {
            alert("Error en la conexión");
        });
    };

});

$('#Inst-pref').click(function (e) {
    $('#ModalIns').modal('show');
});
```

```

    $('#btn-actualizar').click(function (e) {
        $('#username').attr("readonly", false);
        $('#nombre').attr("readonly", false);
        $('#apellido').attr("readonly", false);
        $('#documento').attr("readonly", false);
        $('#direccion').attr("readonly", false);
        $('#telefono').attr("readonly", false);
        $('#estrato').attr('disabled', false);
        $('#ciudad').attr('disabled', false);
        $('#localidad').attr('disabled', false);
        $('#barrio').attr('disabled', false);
        $('#btn-cancel').show();
        $('#btn-confirm').show();
        $('#btn-actualizar').hide();
        $('#btn-cambio-contrasena').hide();
    });
    $('#btn-cancel').click(function (e) {
        location.reload();
    });

    $('#btn-cambio-contrasena').click(function (e) {
        $('#formcontrasena').trigger("reset");
        $(".required").each(function (index) {
            $(this).removeClass('input-required');
        });
        $('#Modalcontrasena').modal('show');
    });

    $('#btn-confirm-contrasena').click(function (e) {
        e.preventDefault();

        var validator = 1;

        $(".required").each(function (index) {
            if ($(this).val() == '' ) {
                $(this).addClass('input-required');
                validator = 0;
            }else{
                $(this).removeClass('input-required');
            }
        });

        if (validator == 0) {
            alert('Completa los campos en rojo');
            return false;
        }

        if ($('#newcontrasena').val() != $('#confnewcontrasena').val()) {
            alert('Las contraseñas no coinciden');
            return false;
        }
    });

```

```

    }

    $.post("<?php echo base_url() ?>index.php/Newuser/cambiar_clave",
    {
        contra: $('#contra').val(),
        newcontra: $('#newcontra').val()
    }, function (data) {
        alert(data);
    })
    .fail(function () {
        alert("Error en la conexion");
    })
    });

    $('#btn-confirm').click(function (e) {
        e.preventDefault();

        var validator = 1;

        $(".required").each(function (index) {
            if ($(this).val() == '' ) {
                $(this).addClass('input-required');
                validator = 0;
            }
        });

        if (validator == 0) {
            alert('Completa los campos en rojo');
            return false;
        }

        $.post("<?php echo base_url() ?>index.php/Newuser/alter_user", {
            email: $('#email').val(),
            username: $('#username').val(),
            nombre: $('#nombre').val(),
            apellido: $('#apellido').val(),
            direccion: $('#direccion').val(),
            documento: $('#documento').val(),
            password: $('#password').val(),
            telefono: $('#telefono').val(),
            estrato: $('#estrato').val(),
            ciudad: $('#ciudad').val(),
            localidad: $('#localidad').val(),
            barrio: $('#barrio').val()
        }, function (data) {
            alert("Datos Modificados!");
            location.reload();
        })
        .fail(function () {
            alert("Error en la conexion");
        })
    });

```



```

    $('#ciudad').change(function () {
        $('#localidad').show().empty();
        $('#barrio').show().empty();

        $.post("<?php echo base_url() ?>index.php/Newuser/change_city", {
            ciudad: $('#ciudad').val()
        }, function (data) {
            $('#localidad').append(data);
        })
        .fail(function () {
            alert("Error en la conexión");
        })
    });

    $('#localidad').change(function () {
        $('#barrio').empty();

        $.post("<?php echo base_url()
?>index.php/Newuser/change_location", {
            localidad: $('#localidad').val()
        }, function (data) {
            $('#barrio').append(data);
        })
        .fail(function () {
            alert("Error en la conexión");
        })
    });
</script>

```

### B5.3 Nuevo usuario

Este formulario es el registro de los nuevos usuarios, tiene todos los campos para un registro completo y son compatibles con los de la aplicación móvil.

```

<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>New User</title>

    <link rel='stylesheet prefetch'
href='http://fonts.googleapis.com/css?family=Roboto:400,100,300,500,700,900'>
    <link rel='stylesheet prefetch'
href='http://fonts.googleapis.com/css?family=Montserrat:400,700'>
    <link rel='stylesheet prefetch'
href='https://maxcdn.bootstrapcdn.com/font-awesome/4.3.0/css/font-awesome.min.css'>
    <link rel="stylesheet" href=" ../assets/css/style.css">

```

```

    <link rel="stylesheet" href="../../assets/css/bootstrap.css">

</head>
<body>

<div class="container">
    <div class="info">
    </div>
</div>
<div class="form">
    <div class="thumbnail"></div>
    <form class="register-form">
        <input type="text" placeholder="name"/>
        <input type="password" placeholder="password"/>
        <input type="text" placeholder="email address"/>
        <button>create</button>
    </form>
    <form class="login-form" name="form" novalidate>
        <input type="text" placeholder="Username" id="username"
class="required" value=""/>
        <input type="text" placeholder="Email" id="email"
class="required" value=""/>
        <input type="password" placeholder="Password" id="password"
class="required" value=""/>
        <input type="text" placeholder="Nombre" id="nombre"
class="required" value=""/>
        <input type="text" placeholder="Apellido" id="apellido"
class="required" value=""/>
        <input type="text" placeholder="Documento" id="documento"
class="required" value=""/>
        <input type="text" placeholder="Direccion" id="direccion"
class="required" value=""/>
        <input type="text" placeholder="Telefono" id="telefono"
class="required" value=""/>
        <select id="estrato" class="select-dropdown required">
            <option value="1">Estrato 1</option>
            <option value="2">Estrato 2</option>
            <option value="3">Estrato 3</option>
            <option value="4">Estrato 4</option>
            <option value="5">Estrato 5</option>
            <option value="6">Estrato 6</option>
        </select>
        <select id="ciudad" class="select-dropdown required">
            <option value="">Seleccione Ciudad</option>

            <?php foreach ($ciudades as $ciudad) {
                ?>
                <option value="<?php echo $ciudad->intContador ?>"><?php
echo $ciudad->varCiudad; ?></option>
                <?php
            } ?>
        </select>

```

```

        <select id="localidad" class="select-dropdown required">

        </select>
        <select id="barrio" class="select-dropdown required">

        </select>

        <button id="Signup">Registrarme</button>
        <p class="message">Ya estas registrado? <a href="<?php echo
base_url (); ?>">Ingresar</a></p>
        </form>
    </div>

<script src="http://cdnjs.cloudflare.com/ajax/libs/jquery/2.1.3/jquery.min.js"></script>

<script src="../../assets/js/index.js"></script>

</body>
</html>

<script>
    $('#localidad').hide();
    $('#barrio').hide();

    $('#ciudad').change(function () {
        $('#localidad').show().empty();
        $('#barrio').show().empty();

        $.post("Newuser/change_city", {
            ciudad: $('#ciudad').val()
        }, function (data) {
            $('#localidad').append(data);
        })
        .fail(function () {
            alert("Error en la conexión");
        })
    });

    $('#localidad').change(function () {
        $('#barrio').empty();

        $.post("Newuser/change_location", {
            localidad: $('#localidad').val()
        }, function (data) {
            $('#barrio').append(data);
        })
    });

```

```

        .fail(function () {
            alert("Error en la conexión");
        })
    });

$('#Signup').click(function (e) {

    e.preventDefault();

    var validator = 1;

    $(".required").each(function (index) {
        if ($(this).val() == " ") {
            $(this).addClass('input-required');
            validator = 0;
        }
    });

    if (validator == 0) {
        alert('Completa los campos en rojo');
        return false;
    }

    $.post("Newuser/create_user", {
        username: $('#username').val(),
        email: $('#email').val(),
        nombre: $('#nombre').val(),
        apellido: $('#apellido').val(),
        direccion: $('#direccion').val(),
        documento: $('#documento').val(),
        password: $('#password').val(),
        telefono: $('#telefono').val(),
        estrato: $('#estrato').val(),
        ciudad: $('#ciudad').val(),
        localidad: $('#localidad').val(),
        barrio: $('#barrio').val()
    }, function (data) {
        alert("Usuario Creado!");
        window.location.href = "<?php echo base_url(); ?>";
    })
    .fail(function () {
        alert("Error en la conexión");
    });

```

```
    })  
  });  
</script>
```

## B5.4 Programas

Esta sección esta destinada al manejo de información de los programas a los que los usuarios pueden acceder, el formulario está compuesto por el siguiente código y utiliza los modelos anteriormente mencionados.

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="utf-8">
```

```
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1">
```

```
  <meta name="description" content="">
```

```
  <meta name="author" content="">
```

```
  <title>Poupawatt</title>
```

```
  <!-- Bootstrap Core CSS -->
```

```
  <link href="<?php echo base_url()  
>assets/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
```

```
  <!-- Custom Fonts -->
```

```
  <link href="<?php echo base_url()>assets/vendor/font-  
awesome/css/font-awesome.min.css" rel="stylesheet"
```

```
    type="text/css">
```

```
  <link
```

```
href='https://fonts.googleapis.com/css?family=Open+Sans:300italic,400italic,600italic,700italic,800italic,400,300,600,700,800'
```

```
rel='stylesheet' type='text/css'>
```

```
</link
```

```
href='https://fonts.googleapis.com/css?family=Merriweather:400,300,300italic,400italic,700,700italic,900,900italic'
```

```
rel='stylesheet' type='text/css'>
```

```
<!-- Plugin CSS -->
```

```
<link href="<?php echo base_url() ?>assets/vendor/magnific-popup/magnific-popup.css" rel="stylesheet">
```

```
<!-- Theme CSS -->
```

```
<link href="<?php echo base_url() ?>assets/css/creative.min.css" rel="stylesheet">
```

```
<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries -->
```

```
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
```

```
<!--[if lt IE 9]>
```

```
<script
```

```
src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
```

```
<script
```

```
src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>
```

```
<![endif]-->
```

```
</head>
```

```

<body id="page-top">

<nav id="mainNav" class="navbar navbar-default navbar-fixed-top">
  <div class="container-fluid">
    <!-- Brand and toggle get grouped for better mobile display -->
    <div class="navbar-header">
      <button type="button" class="navbar-toggle collapsed" data-
toggle="collapse"
      data-target="#bs-example-navbar-collapse-1">
        <span class="sr-only">Toggle navigation</span> Menu <i
class="fa fa-bars"></i>
      </button>
      <a class="navbar-brand page-scroll" href="<?php echo base_url()
?>index.php/HomeAdmin">Poupawatt</a>
    </div>

    <!-- Collect the nav links, forms, and other content for toggling -->
    <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-
1">
      <ul class="nav navbar-nav navbar-right">

        <li>
          <a class="page-scroll" href="<?php echo base_url()
?>index.php/HomeAdmin/sensores">Sensores</a>
        </li>
        <li>
          <a class="page-scroll" href="<?php echo base_url()
?>index.php/HomeAdmin/electrodomesticos">Electrodomésticos</a>
        </li>

```

```

        <li>
            <a class="page-scroll" href="<?php echo base_url()
?>index.php/HomeAdmin/programas">Programas</a>
        </li>
        <li>
            <a class="page-scroll" href="<?php echo base_url()
?>index.php/HomeAdmin/usuarios">Usuarios</a>
        </li>
        <li>
            <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home/salir">Salir</a>
        </li>

```

```

    </ul>
</div>
<!-- /.navbar-collapse -->
</div>
<!-- /.container-fluid -->
</nav>

```

```

<section class="bg-primary" id="about">
    <div class="container">
        <div class="row">
            <div class="col-lg-8 col-lg-offset-2 text-center">
                <h2 class="section-heading">Programas</h2>
                <hr class="light">
                <p><button id="Inst-pref" class="btn btn-success btn-sm sr-
button">Instrucciones >></button></p>

```



```

<div class="col-md-6" id="lista-programas">
  <?php if (!count($programas)) { ?>
    <p id="no-programas">No tienes programas agregados!</p>
  <?php } else {
    foreach ($programas as $programa) {
      ?>
      <p class="pull-left"><a class='editarPrograma' data-
id="<?php echo $programa->intPrograma; ?>"
        style='color: #f5f5f5; cursor: pointer;'>
Programa: <?php echo $programa->varPrograma; ?> </a>
      </p>
      <p>&nbsp;</p>

      <?php
    }
  }
  ?>
</div>

```

```

<div class="col-md-6" id="zona-programa">
  <input type="text" id="programa" placeholder="Programa"
    class="form-control inputFormulario required"/>
  <input type="text" id="precio" placeholder="Precio"
    class="form-control inputFormulario required"/>
  <input type="text" id="consumo" placeholder="Consumo"
    class="form-control inputFormulario required"/>
  <textarea type="text" rows="6" id="descripcion"
placeholder="Descripción"
    class="form-control inputFormulario required"/></textarea>

```

```
<hr class="light">
<button id="crearPrograma" class="page-scroll btn btn-default
btn-xl sr-button">Agregar Programa
</button>
<button id="editarPrograma" data-id="" class="page-scroll btn
btn-default btn-xl sr-button">Actualizar
</button>
<button id="eliminarPrograma" data-id="" class="page-scroll btn
btn-default btn-xl sr-button" >Eliminar
</button>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</section>
```

```
</body>
```

```
</html>
```

```
<!-- Modal -->
```

```
<div id="ModalResp" class="modal fade">
```

```
<div class="modal-dialog">
```

```
<div class="modal-content">
```

```
<div class="modal-header">
```

```
<button type="button" class="close" data-
dismiss="modal">&times;</button>
```

```
<h4 id="modal-title" class="modal-title"></h4>
```

```

        </div>
    <!-- dialog buttons -->
    <div class="modal-footer">
        <button type="button" class="btn btn-danger" data-
dismiss="modal">CERRAR</button>
    </div>
    <input type="hidden" name="id_grav_delete" value="">
</div>
</div>
</div>
<!-- /.modal -->

<!-- Modal Instrucciones-->
<div id="Modallns" class="modal fade">
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-
dismiss="modal">&times;</button>
                <h4 id="modal-title" class="modal-title">Instrucciones</h4>
            </div>
            <div class="modal-Body">
                <p>Instrucciones aqui</p>
            </div>
            <!-- dialog buttons -->
            <div class="modal-footer">
                <button type="button" class="btn btn-success" data-
dismiss="modal">Aceptar</button>
            </div>
        </div>
    </div>

```

```

    </div>
</div>
<!-- /.modal -->

<script src="<?php echo base_url()
?>assets/vendor/jquery/jquery.min.js"></script>
<!-- Bootstrap Core JavaScript -->
<script src="<?php echo base_url()
?>assets/vendor/bootstrap/js/bootstrap.min.js"></script>

<script>
    $( document ).ready(function() {
        $('#Modallns').modal('show');
        $('#editarPrograma').hide();
        $('#eliminarPrograma').hide();
    });

    $('#Inst-pref').click(function (e) {
        $('#Modallns').modal('show');
    });

    $('#crearPrograma').click(function (e) {

        e.preventDefault();

        var validator = 1;

        $(".required").each(function (index) {
            if ($(this).val() == "") {
                $(this).addClass('input-required');
            }
        });
    });

```

```

        validator = 0;
    }else{
        $(this).removeClass('input-required');
    }

});

if (validator == 0) {
    $('#ModalResp').find('#modal-title').html('Completa los campos en
rojo');
    $('#ModalResp').modal('show');
    //return false;
}else{
    $.post("crear_programa", {
        programa: $('#programa').val(),
        precio: $('#precio').val(),
        consumo: $('#consumo').val(),
        descripcion: $('#descripcion').val()
    }, function (data) {
        $('#ModalResp').find('#modal-title').html('Programa Creado!');
        $('#ModalResp').modal('show');
        $('#no-programas').empty();
        $('#lista-programas').append(data);
    })
    .fail(function () {
        alert("Error en la conexion");
    })
}

})

```

```

$( "#lista-programas" ).on( "click", ".editarPrograma", function() {
    $.post("cargar_programa", {
        programa: $(this).attr('data-id')
    }, function (data) {
        $('#programa').val(data[0].varPrograma),
        $('#precio').val(data[0].floPrecio),
        $('#consumo').val(data[0].floConsumo),
        $('#descripcion').val(data[0].txtDescripcion)

        $('#editarPrograma').attr('data-id', data[0].intPrograma);
        $('#eliminarPrograma').attr('data-id', data[0].intPrograma);
        $('#editarPrograma').show();
        $('#eliminarPrograma').show();
        $('#crearPrograma').hide();
    })
    .fail(function () {
        alert("Error en la conexion");
    })
});

```

```

$('#editarPrograma').click(function (e) {
    e.preventDefault();
    var validator = 1;

```

```

$(".required").each(function (index) {
    if ($(this).val() == "") {
        $(this).addClass('input-required');
        validator = 0;
    }

```

```

    }
  });

  if (validator == 0) {
    alert('Completa los campos en rojo');
    //return false;
  }else{
    $.post("modificar_programa", {
      id: $(this).attr('data-id'),
      programa: $('#programa').val(),
      precio: $('#precio').val(),
      consumo: $('#consumo').val(),
      descripcion: $('#descripcion').val()
    }, function (data) {
      alert("Programa Modificado!");
      location.reload();
    })
    .fail(function () {
      alert("Error en la conexion");
    })
  }
});

$('#eliminarPrograma').click(function(){
  $.post("eliminar_programa", {
    id: $(this).attr('data-id')
  }, function (data) {
    alert("Programa Eliminado!");
    location.reload();
  })
});

```

```
.fail(function () {  
    alert("Error en la conexion");  
})  
});
```

```
</script>
```

## **B5.5 Electrodomésticos**

Los electrodomésticos tienen el mismo manejo de los programas

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
    <meta charset="utf-8">
```

```
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
```

```
    <meta name="viewport" content="width=device-width, initial-scale=1">
```

```
    <meta name="description" content="">
```

```
    <meta name="author" content="">
```

```
    <title>Poupawatt</title>
```

```
    <!-- Bootstrap Core CSS -->
```

```
    <link href="<?php echo base_url()  
>assets/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
```

```
    <!-- Custom Fonts -->
```

```
    <link href="<?php echo base_url()  
>assets/vendor/font-awesome/css/font-awesome.min.css" rel="stylesheet"
```



```

        type="text/css">
<link

href='https://fonts.googleapis.com/css?family=Open+Sans:300italic,400italic
,600italic,700italic,800italic,400,300,600,700,800'

        rel='stylesheet' type='text/css'>
<link

href='https://fonts.googleapis.com/css?family=Merriweather:400,300,300itali
c,400italic,700,700italic,900,900italic'

        rel='stylesheet' type='text/css'>

<!-- Plugin CSS -->
<link href="<?php echo base_url() ?>assets/vendor/magnific-
popup/magnific-popup.css" rel="stylesheet">

<!-- Theme CSS -->
<link href="<?php echo base_url() ?>assets/css/creative.min.css"
rel="stylesheet">

<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and
media queries -->
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
<!--[if lt IE 9]>
<script
src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
<script
src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>
<![endif]-->

```

```
</head>
```

```
<body id="page-top">
```

```
<nav id="mainNav" class="navbar navbar-default navbar-fixed-top">
```

```
  <div class="container-fluid">
```

```
    <!-- Brand and toggle get grouped for better mobile display -->
```

```
    <div class="navbar-header">
```

```
      <button type="button" class="navbar-toggle collapsed" data-  
toggle="collapse"
```

```
        data-target="#bs-example-navbar-collapse-1">
```

```
        <span class="sr-only">Toggle navigation</span> Menu <i  
class="fa fa-bars"></i>
```

```
      </button>
```

```
      <a class="navbar-brand page-scroll" href="<?php echo base_url()  
>index.php/HomeAdmin">Poupawatt</a>
```

```
    </div>
```

```
    <!-- Collect the nav links, forms, and other content for toggling -->
```

```
    <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-  
1">
```

```
      <ul class="nav navbar-nav navbar-right">
```

```
        <li>
```

```
          <a class="page-scroll" href="<?php echo base_url()  
>index.php/HomeAdmin/sensores">Sensores</a>
```

```
        </li>
```

```
        <li>
```

```

        <a class="page-scroll" href="<?php echo base_url()
?>index.php/HomeAdmin/electrodomesticos">Electrodomésticos</a>
    </li>
    <li>
        <a class="page-scroll" href="<?php echo base_url()
?>index.php/HomeAdmin/programas">Programas</a>
    </li>
    <li>
        <a class="page-scroll" href="<?php echo base_url()
?>index.php/HomeAdmin/usuarios">Usuarios</a>
    </li>
    <li>
        <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home/salir">Salir</a>
    </li>

```

```

    </ul>
</div>
<!-- /.navbar-collapse -->
</div>
<!-- /.container-fluid -->
</nav>

```

```

<section class="bg-primary" id="about">
    <div class="container">
        <div class="row">
            <div class="col-lg-8 col-lg-offset-2 text-center">
                <h2 class="section-heading">Electrodomésticos</h2>
                <hr class="light">
            </div>

```

```
<p><button id="Inst-pref" class="btn btn-success btn-sm sr-  
button">Instrucciones >></button></p>
```

```
<div class="col-md-6" id="lista-electrodomesticos">  
  <?php if (!count($electrodomesticos)) { ?>  
    <p id="no-electrodomesticos">No tienes electrodomésticos  
agregados!</p>  
  <?php } else {  
    foreach ($electrodomesticos as $elec) {  
      ?>  
      <p class="pull-left"><a class='editarElectrodomestico' data-  
id="<?php echo $elec->intContador; ?>"  
        style='color: #f5f5f5; cursor: pointer;'>  
Electrodoméstico: <?php echo $elec->varElectrodomestico; ?> </a>  
      </p>  
      <p>&nbsp;</p>  
  
      <?php  
    }  
  }  
  ?>  
</div>
```

```
<div class="col-md-6" id="zona-sensor">  
  <input type="text" id="electrodomestico"  
placeholder="Electrodoméstico"  
class="form-control inputFormulario required"/>  
  <input type="text" id="modelo" placeholder="Modelo"  
class="form-control inputFormulario required"/>
```

```

        <input type="text" id="clase_energetica" placeholder="Clase
Energética"
        class="form-control inputFormulario required"/>
        <input type="text" id="marca" placeholder="marca"
        class="form-control inputFormulario required"/>
        <input type="text" id="consumo" placeholder="Consumo"
        class="form-control inputFormulario required"/>
        <input type="text" id="voltaje" placeholder="Voltaje Op"
        class="form-control inputFormulario required"/>

        <hr class="light">
        <button id="crearElectrodomestico" class="page-scroll btn btn-
default btn-xl sr-button">Agregar Electrodomestico
        </button>
        <button id="editarElectrodomestico" data-id="" class="page-
scroll btn btn-default btn-xl sr-button">Actualizar
        </button>
        <button id="eliminarElectrodomestico" data-id="" class="page-
scroll btn btn-default btn-xl sr-button" >Eliminar
        </button>

    </div>

</div>
</div>
</div>
</section>

</body>

```

```
</html>
```

```
<!-- Modal -->
```

```
<div id="ModalResp" class="modal fade">
```

```
<div class="modal-dialog">
```

```
<div class="modal-content">
```

```
<div class="modal-header">
```

```
<button type="button" class="close" data-  
dismiss="modal">&times;</button>
```

```
<h4 id="modal-title" class="modal-title"></h4>
```

```
</div>
```

```
<!-- dialog buttons -->
```

```
<div class="modal-footer">
```

```
<button type="button" class="btn btn-danger" data-  
dismiss="modal">CERRAR</button>
```

```
</div>
```

```
<input type="hidden" name="id_grav_delete" value="">
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<!-- /.modal -->
```

```
<!-- Modal Instrucciones-->
```

```
<div id="Modallns" class="modal fade">
```

```
<div class="modal-dialog">
```

```
<div class="modal-content">
```

```
<div class="modal-header">
```

```
<button type="button" class="close" data-  
dismiss="modal">&times;</button>
```

```
<h4 id="modal-title" class="modal-title">Instrucciones</h4>
```

```

    </div>
    <div class="modal-Body">
        <p>Instrucciones aqui</p>
    </div>
<!-- dialog buttons -->
<div class="modal-footer">
    <button type="button" class="btn btn-success" data-
dismiss="modal">Aceptar</button>
</div>
</div>
</div>
</div>
<!-- /.modal -->

<script src="<?php echo base_url()
?>assets/vendor/jquery/jquery.min.js"></script>
<!-- Bootstrap Core JavaScript -->
<script src="<?php echo base_url()
?>assets/vendor/bootstrap/js/bootstrap.min.js"></script>

<script>
$( document ).ready(function() {
    $('#Modallns').modal('show');
    $('#editarElectrodomestico').hide();
    $('#eliminarElectrodomestico').hide();
});

$('#Inst-pref').click(function (e) {
    $('#Modallns').modal('show');
});

```

```

$('#crearElectrodomestico').click(function (e) {

    e.preventDefault();

    var validator = 1;

    $(".required").each(function (index) {
        if ($(this).val() == "") {
            $(this).addClass('input-required');
            validator = 0;
        }else{
            $(this).removeClass('input-required');
        }
    });

    if (validator == 0) {
        $('#ModalResp').find('#modal-title').html('Completa los campos en rojo');
        $('#ModalResp').modal('show');
        //return false;
    }else{
        $.post("crear_electrodomestico", {
            electrodomestico: $('#electrodomestico').val(),
            modelo: $('#modelo').val(),
            clase_energetica: $('#clase_energetica').val(),
            marca: $('#marca').val(),
            consumo: $('#consumo').val(),
            voltaje: $('#voltaje').val()
        }
    )
    }
});

```



```

    }, function (data) {
        $('#ModalResp').find('#modal-title').html('Electrodoméstico
Creado!');
        $('#ModalResp').modal('show');
        $('#no-electrodomesticos').empty();
        $('#lista-electrodomesticos').append(data);
    })
    .fail(function () {
        alert("Error en la conexion");
    })
}

})

$( "#lista-electrodomesticos" ).on( "click", ".editarElectrodomestico",
function() {
    $.post("cargar_electrodomestico", {
        electrodomestico: $(this).attr('data-id')
    }, function (data) {
        $('#electrodomestico').val(data[0].varElectrodomestico);
        $('#modelo').val(data[0].intModelo);
        $('#clase_energetica').val(data[0].varClaseEnergetica);
        $('#marca').val(data[0].varMarca);
        $('#consumo').val(data[0].floConsumo);
        $('#voltaje').val(data[0].intVoltajeOp);

        $('#editarElectrodomestico').attr('data-id', data[0].intContador);
        $('#eliminarElectrodomestico').attr('data-id', data[0].intContador);
        $('#editarElectrodomestico').show();
    });
});

```

```

    $('#eliminarElectrodomestico').show();
    $('#crearElectrodomestico').hide();
  })
  .fail(function () {
    alert("Error en la conexion");
  })
});

```

```

$('#editarElectrodomestico').click(function (e) {
  e.preventDefault();
  var validator = 1;

```

```

$(".required").each(function (index) {
  if ($(this).val() == "") {
    $(this).addClass('input-required');
    validator = 0;
  }
});

```

```

if (validator == 0) {
  alert('Completa los campos en rojo');
  //return false;
}else{
  $.post("modificar_electrodomestico", {
    id: $(this).attr('data-id'),
    electrodomestico: $('#electrodomestico').val(),
    modelo: $('#modelo').val(),
    clase_energetica: $('#clase_energetica').val(),
    marca: $('#marca').val(),
    consumo: $('#consumo').val(),

```

```

        voltaje: $('#voltaje').val()
    }, function (data) {
        alert("Electrodoméstico Modificado!");
    })
    .fail(function () {
        alert("Error en la conexion");
    })
    }
});

$('#eliminarElectrodomestico').click(function(){
    $.post("eliminar_electrodomestico", {
        id: $(this).attr('data-id')
    }, function (data) {
        alert("Electrodoméstico Eliminado!");
        location.reload();
    })
    .fail(function () {
        alert("Error en la conexion");
    })
});

```

</script>

## **B5.6 Preferencias**

Las preferencias por el contrario al anterior utilizan diferentes partes del modelo ya nombrado para hacer carga y modificación de datos.

<!DOCTYPE html>

<html lang="en">

<head>

```
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">
<meta name="description" content="">
<meta name="author" content="">
```

```
<title>Poupawatt</title>
```

```
<!-- Bootstrap Core CSS -->
```

```
<link href="<?php echo base_url()
?>assets/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
```

```
<!-- Custom Fonts -->
```

```
<link href="<?php echo base_url() ?>assets/vendor/font-
awesome/css/font-awesome.min.css" rel="stylesheet"
type="text/css">
```

```
<link
```

```
href='https://fonts.googleapis.com/css?family=Open+Sans:300italic,400italic
,600italic,700italic,800italic,400,300,600,700,800'
```

```
rel='stylesheet' type='text/css'>
```

```
<link
```

```
href='https://fonts.googleapis.com/css?family=Merriweather:400,300,300itali
c,400italic,700,700italic,900,900italic'
```

```
rel='stylesheet' type='text/css'>
```

```
<!-- Plugin CSS -->
```

```
<link href="<?php echo base_url() ?>assets/vendor/magnific-  
popup/magnific-popup.css" rel="stylesheet">
```

```
<!-- Theme CSS -->  
<link href="<?php echo base_url() ?>assets/css/creative.min.css"  
rel="stylesheet">
```

```
<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and  
media queries -->
```

```
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->  
<!--[if lt IE 9]>  
<script  
src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>  
<script  
src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>  
<![endif]-->
```

```
</head>
```

```
<body id="page-top">
```

```
<nav id="mainNav" class="navbar navbar-default navbar-fixed-top">
```

```
<div class="container-fluid">  
<!-- Brand and toggle get grouped for better mobile display -->  
<div class="navbar-header">  
<button type="button" class="navbar-toggle collapsed" data-  
toggle="collapse"  
data-target="#bs-example-navbar-collapse-1">  
<span class="sr-only">Toggle navigation</span> Menu <i  
class="fa fa-bars"></i>
```

```

        </button>
        <a class="navbar-brand page-scroll" href="<?php echo base_url()
?>index.php/Home">Poupawatt</a>
    </div>

    <!-- Collect the nav links, forms, and other content for toggling -->
    <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-
1">
        <ul class="nav navbar-nav navbar-right">

            <li>
                <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home#about">Facturas</a>
            </li>
            <li>
                <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home#consumo">Consumo</a>
            </li>
            <li>
                <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home#noticias">Noticias</a>
            </li>
            <li>
                <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home/juego">Juego</a>
            </li>
            <li>
                <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home/modelos">Modelos</a>
            </li>

```

```

        <li>
            <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home/perfil">Ajustes</a>
        </li>
        <li>
            <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home/salir">Salir</a>
        </li>

```

```

    </ul>
</div>
<!-- /.navbar-collapse -->
</div>
<!-- /.container-fluid -->
</nav>

```

```

<section class="bg-primary" id="about">
    <div class="container">
        <div class="row">
            <div class=" text-center">
                <h2 class="section-heading">Tus Preferencias</h2>
                <hr class="light">
                <p><button id="Inst-pref" class="btn btn-success btn-sm sr-
button">Instrucciones >></button></p>

                <div class="col-md-12">
                    <p>PARA DARTE UN MEJOR SERVICIO ESTABLECE TUS
PREFERENCIAS</p>

```

```

        <label for="">Selecciona el tipo de programa que se ajusta a
ti</label>
        <div class="col-md-11">
            <select name="" id="programa" class="form-control
inputFormulario required" >
                <option value="">Selecione</option>
                <?php
                foreach ($programas as $programa) {
                    ?>
                    <option
                        value="<?php echo $programa->intPrograma
?>"><?php echo $programa->varPrograma ?></option>
                    <?php
                    }
                    ?>
                </select>
            </div>
            <div class="col-md-1">
                <button id="Inst-modelos" class="btn btn-danger btn-sm sr-
button">?</button>
            </div>
            <input type="hidden" id="programadb" value="<?php if
(count($preferencias)) { echo $preferencias[0]->programa; }?>">
            <div class="col-md-6">
                <h4>Selecciona un día</h4>

                <div class="radio pull-left">
                    <input class="radiodia" type="radio" name="optradiodia"
value="1"/>Lunes
                </div>

```



```
<br></br>
<div class="radio pull-left">
  <input class="radiodia" type="radio" name="optradiodia"
value="2"/>Martes
</div>
<br></br>
<div class="radio pull-left">
  <input class="radiodia" type="radio" name="optradiodia"
value="3"/>Miercoles
</div>
<br></br>
<div class="radio pull-left">
  <input class="radiodia" type="radio" name="optradiodia"
value="4"/>Jueves
</div>
<br></br>
<div class="radio pull-left">
  <input class="radiodia" type="radio" name="optradiodia"
value="5"/>Viernes
</div>
<br></br>
<div class="radio pull-left">
  <input class="radiodia" type="radio" name="optradiodia"
value="6"/>Sabado
</div>
<br></br>
<div class="radio pull-left">
  <input class="radiodia" type="radio" name="optradiodia"
value="7"/>Domingo
</div>
```

```

</div>
<div class="col-md-6">
  <h4>Que electrodoméstico no puedes dejar de usar?</h4>
  <?php
  foreach ($electrodomesticos as $electrodomestico) {

    ?>

    <div class="radio pull-left">
      <input class="radioelec" type="radio" name="optradio"
value="<?php echo $electrodomestico->intContador; ?>"/><?php echo " " .
$electrodomestico->varElectrodomestico; ?>
    </div>

    <p>&nbsp;</p>
    <?php
    }
    ?>
  </div>

  <p>&nbsp;</p>
  <div class="col-md-12">
    <div class="col-md-6">
      <p>Hora de Inicio
      <input type="time" id="horainicio" class="form-control
required" style="color: black" /></p>
    </div>

    <div class="col-md-6">

```

```

        <p>Hora de Finalización
        <input type="time" id="horafin" class="form-control
required" style="color: black;"/></p>
    </div>

    <div class="col-md-6">
        <p>Reducción
        <input type="text" id="reduccion" class="form-control
required" style="color: black;"/></p>
        <input type="hidden" id="reducciondb" value="<?php if
(count($preferencias)) { echo $preferencias[0]->reduccion; }?>">
    </div>

    <div class="col-md-6">
        <p>Valor
        <input type="text" id="valor" class="form-control required"
style="color: black;"/></p>
        <input type="hidden" id="valordb" value="<?php if
(count($preferencias)) { echo $preferencias[0]->valor; }?>">
    </div>
</div>
<input type="hidden" id="hiddenDay" />
<input type="hidden" id="hiddenRadios" />
<input type="hidden" id="hiddenDayexist" />

<div class="col-md-12">
    <p>&nbsp;</p>
    <button id="crearPreferencia" class="btn-default btn"
>Guardar</button>
</div>

```

```

        </div>

        </div>
    </div>
</div>
</section>

</body>

</html>

<!-- Modal Instrucciones Modelos-->
<div id="ModallnsMod" class="modal fade">
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-
dismiss="modal">&times;</button>
                <h4 id="modal-title" class="modal-title">Instrucciones</h4>
            </div>
            <div class="modal-Body">
                <h3>TIPOS DE PROGRAMAS</h3>
                <?php
                foreach ($programas as $programa) {
                    ?>
                    <h4><?php echo $programa->varPrograma; ?></h4>

```

```

        <p id="" style="text-align: justify;"><?php echo $programa-
>txtDescripcion; ?></p>
        <?php
        }

        ?>
    </div>
    <!-- dialog buttons -->
    <div class="modal-footer">
        <button type="button" class="btn btn-success" data-
dismiss="modal">Aceptar</button>
    </div>
</div>
</div>
</div>
<!-- /.modal -->

<!-- Modal Instrucciones-->
<div id="Modallns" class="modal fade">
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-
dismiss="modal">&times;</button>
                <h4 id="modal-title" class="modal-title">Instrucciones</h4>
            </div>
            <div class="modal-Body">
                <p>Instrucciones aqui</p>
            </div>
        </div>
    </div>
    <!-- dialog buttons -->

```

```

        <div class="modal-footer">
            <button type="button" class="btn btn-success" data-
dismiss="modal">Aceptar</button>
        </div>
    </div>
</div>
</div>
<!-- /.modal -->

```

```

<script src="<?php echo base_url()
?>assets/vendor/jquery/jquery.min.js"></script>
<script src="<?php echo base_url()
?>assets/vendor/bootstrap/js/bootstrap.min.js"></script>

```

```

<script>
    $( window ).load(function() {
        $('#Modallns').modal('show');
    });

    $('#Inst-pref').click(function (e) {
        $('#Modallns').modal('show');
    });
    $('#Inst-modelos').click(function (e) {
        $('#ModallnsMod').modal('show');
    });

    var arrayDays = [];

```

```

var arrayradios = [];

if ($('#programadb').val()) {
    $('#programa').val( $('#programadb').val());
    $('#programa').attr('disabled',true);
};

if ($('#valordb').val()) {
    $('#valor').val( $('#valordb').val());
    $('#valor').attr('readonly',true);
};

if ($('#reducciondb').val()) {
    $('#reduccion').val( $('#reducciondb').val());
    $('#reduccion').attr('readonly',true);
};

$('.radiodia').click(function () {
    var arrayDays = [];
    if($(this).is(':checked')) {
        arrayDays.push($(this).val());
    }
    else{

        var elec = $(this).val();
        arrayDays = jQuery.grep(arrayDays, function(value) {
            return value != elec;
        });
    }
    $('#hiddenDay').val(arrayDays);
};

```

```

console.log(arrayDays);

$.post("preferenciadia", {
  dia: $('#hiddenDay').val()

}, function (data) {
  if (data.prefdia) {
    $('.radioelec').each(function (index) {
      if (data.prefdia.electrodomesticos == $(this).val()) {
        $(this).prop('checked', true);
      }else{
        $(this).prop('checked', false);
      }
    });
    $('#horainicio').val(data.prefdia.horainicio);
    $('#horafin').val(data.prefdia.horafinal);
    $('#hiddenDayexist').val(data.prefdia.id);
    $('#hiddenRadios').val(data.prefdia.electrodomesticos);

  }else{
    $('.radioelec').each(function (index) {
      if ($(this).is(':checked')) {
        $(this).prop('checked', false);
      }
    });
    $('#horainicio').val("");
    $('#horafin').val("");
    $('#hiddenDayexist').val("");
    $('#hiddenRadios').val("");
  }
}

```



```

    })
    .fail(function () {
        alert("Error en la conexión");
    })

})

$('.radioelec').click(function () {
    arrayradios = [];
    if($(this).is(':checked')) {
        arrayradios.push($(this).val());
    }
    else{

        var elec = $(this).val();

        arrayradios = jQuery.grep(arrayradios, function(value) {
            return value != elec;
        });
    }
    $('#hiddenRadios').val(arrayradios);
    console.log(arrayradios);

})

$('#crearPreferencia').click(function (e) {

```

```

e.preventDefault();

var validator = 1;

$(".required").each(function (index) {
    if ($(this).val() == " ") {
        $(this).addClass('input-required');
        validator = 0;
    }else{
        $(this).removeClass('input-required');
    }
});

if (validator == 0) {
    alert('Completa los campos en rojo');
    return false;
}

$.post("crear_preferencia", {
    idexist: $('#hiddenDayexist').val(),
    programa: $('#programa').val(),
    hiddenDay: $('#hiddenDay').val(),
    hiddenRadios: $('#hiddenRadios').val(),
    horafin: $('#horafin').val(),
    horainicio: $('#horainicio').val(),
    reduccion: $('#reduccion').val(),
    valor: $('#valor').val()

```

```

    }, function (data) {
        alert("Hemos guardado tus preferencias, gracias!");
        location.reload();
    })
    .fail(function () {
        alert("Error en la conexión");
    })
})

```

```
</script>
```

### **B5.7 Modelos**

La implementación de estos algoritmos utiliza el siguiente código para el view

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
    <meta charset="utf-8">
```

```
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
```

```
    <meta name="viewport" content="width=device-width, initial-scale=1">
```

```
    <meta name="description" content="">
```

```
    <meta name="author" content="">
```

```
    <title>Poupawatt</title>
```

```
    <!-- Bootstrap Core CSS -->
```

```
    <link          href="<?php          echo          base_url()
?>assets/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
```

```

<!-- Custom Fonts -->
<link href="<?php echo base_url() ?>assets/vendor/font-
awesome/css/font-awesome.min.css" rel="stylesheet"
      type="text/css">
<link

href='https://fonts.googleapis.com/css?family=Open+Sans:300italic,400italic
,600italic,700italic,800italic,400,300,600,700,800'
      rel='stylesheet' type='text/css'>
<link

href='https://fonts.googleapis.com/css?family=Merriweather:400,300,300itali
c,400italic,700,700italic,900,900italic'
      rel='stylesheet' type='text/css'>

<!-- Plugin CSS -->
<link href="<?php echo base_url() ?>assets/vendor/magnific-
popup/magnific-popup.css" rel="stylesheet">

<!-- Theme CSS -->
<link href="<?php echo base_url() ?>assets/css/creative.min.css"
rel="stylesheet">

<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and
media queries -->
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
<!--[if lt IE 9]>

```

```

    <script
src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
    <script
src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>
    <![endif]-->

</head>

<body id="page-top">

<nav id="mainNav" class="navbar navbar-default navbar-fixed-top">
    <div class="container-fluid">
        <!-- Brand and toggle get grouped for better mobile display -->
        <div class="navbar-header">
            <button type="button" class="navbar-toggle collapsed" data-
toggle="collapse"
                data-target="#bs-example-navbar-collapse-1">
                <span class="sr-only">Toggle navigation</span> Menu <i
class="fa fa-bars"></i>
            </button>
            <a class="navbar-brand page-scroll" href="<?php echo base_url()
?>index.php/Home">Poupawatt</a>
        </div>

        <!-- Collect the nav links, forms, and other content for toggling -->
        <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-
1">
            <ul class="nav navbar-nav navbar-right">

                <li>

```

```

        <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home#about">Facturas</a>
    </li>
    <li>
        <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home#services">Preferencias</a>
    </li>
    <li>
        <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home#consumo">Consumo</a>
    </li>
    <li>
        <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home#noticias">Noticias</a>
    </li>
    <li>
        <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home/juego">Juego</a>
    </li>
    <li>
        <a class="page-scroll"
        href="<?php echo base_url()
?>index.php/Home/perfil">Ajustes</a>
    </li>
    <li>
        <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home/salir">Salir</a>
    </li>

```

```

        </ul>
    </div>
    <!-- /.navbar-collapse -->
</div>
<!-- /.container-fluid -->
</nav>

<section class="bg-primary" id="about">
    <div class="container">
        <div class="row">
            <div class="col-lg-8 col-lg-offset-2 text-center">
                <h2 class="section-heading">Modelos</h2>
                <hr class="light">
                <p><button id="Inst-Consumo" class="btn btn-success btn-sm sr-
button">Instrucciones >></button></p>
                <div class="col-md-6" id="div-algoritmos">
                    <h4 for="inicio">Programa</h4>
                    <select name="" id="programa" class="form-control
inputFormulario required">
                        <option value="">Seleccione Programa</option>
                        <option value="1">Oferta a la Demanda</option>
                        <option value="2">Tiempo de Uso</option>
                    </select>
                </div>
                <input type="hidden" name="programa-selec" id="programa-selec"
value="<?php if ($preferencias) {if ($preferencias[0]->programa !== null)
{echo ($preferencias[0]->programa);}}
?>">
                <div id="div-check">

```

```
        <input type="checkbox" id="check-oferta" value="">Ingresar
horas<br>
```

```
        <input type="hidden" name="valcheck" id="valcheck" value="0">
</div>
```

```
<div class="col-md-6" id="div-oferta">
```

```
    <h4>Televisor</h4>
```

```
    <div class="col-md-6" >
```

```
        <label for="inicio">Hora operación inicial</label>
```

```
        <input type="number" min="0" max="23" id="horaintv"
class="form-control inputFormulario required"/>
```

```
    </div>
```

```
    <div class="col-md-6" >
```

```
        <label for="fin">Hora operación Final</label>
```

```
        <input type="number" min="0" max="23" id="horafintv"
class="form-control inputFormulario required"/>
```

```
    </div>
```

```
    <h4>Lavadora</h4>
```

```
    <div class="col-md-6" >
```

```
        <label for="inicio">Hora operación inicial</label>
```

```
        <input type="number" min="0" max="23" id="horainlav"
class="form-control inputFormulario required"/>
```

```
    </div>
```

```
    <div class="col-md-6" >
```

```
        <label for="fin">Hora operación Final</label>
```

```
        <input type="number" min="0" max="23" id="horafinlav"
class="form-control inputFormulario required"/>
```

```
    </div>
```



```

<h4>Bombilla</h4>
<div class="col-md-6" >
  <label for="inicio">Hora operación inicial</label>
  <input type="number" min="0" max="23" id="horainbom"
class="form-control inputFormulario required"/>
</div>
<div class="col-md-6" >
  <label for="fin">Hora operación Final</label>
  <input type="number" min="0" max="23" id="horafinbom"
class="form-control inputFormulario required"/>
</div>

<h4>Plancha</h4>
<div class="col-md-6" >
  <label for="inicio">Hora operación inicial</label>
  <input type="number" min="0" max="23" id="horainpl"
class="form-control inputFormulario required"/>
</div>
<div class="col-md-6" >
  <label for="fin">Hora operación Final</label>
  <input type="number" min="0" max="23" id="horafinpl"
class="form-control inputFormulario required"/>
</div>

</div>

<hr class="light">
<button id="btn-calcular" class="page-scroll btn btn-default btn-
xl sr-button">Calcular</button>

```

```

        <br></br>
    <div id="div-tabla" class="col-md-12">
        <table id="tablerespuesta" class="table" border="1">
            <thead>
                <tr>
                    <th></th>
                    <th>Día</th>
                    <th>Electrodomestico</th>
                    <th>Hora inicio</th>
                    <th>Hora fin</th>
                </tr>
            </thead>
            <tbody>

            </tbody>
        </table>
    </div>

</div>
</div>
</div>
</section>

</body>

</html>

<!-- Modal -->
<div id="ModalResp" class="modal fade">
    <div class="modal-dialog">

```

```

<div class="modal-content">
  <div class="modal-header">
    <button type="button" class="close" data-
dismiss="modal">&times;</button>
    <h4 id="modal-title" class="modal-title"></h4>
  </div>
  <!-- dialog buttons -->
  <div class="modal-footer">
    <button type="button" class="btn btn-danger" data-
dismiss="modal">CERRAR</button>
  </div>
</div>
</div>
<!-- /.modal -->

```

```

<!-- Modal Instrucciones-->
<div id="Modallns" class="modal fade">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <button type="button" class="close" data-
dismiss="modal">&times;</button>
        <h4 id="modal-title" class="modal-title">Instrucciones</h4>
      </div>
      <div class="modal-Body">
        <p>Escriba aqui las instrucciones</p>
      </div>
      <!-- dialog buttons -->
      <div class="modal-footer">

```

```

        <button type="button" class="btn btn-success" data-
dismiss="modal">Aceptar</button>
    </div>
</div>
</div>
</div>
<!-- /.modal -->

```

```

<!-- Modal Preferencias-->
<div id="ModalPref" class="modal fade" data-keyboard="false" data-
backdrop="static">
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-
dismiss="modal">&times;</button>
                <h4 id="modal-title" class="modal-title">No se encontraron
preferencias!</h4>
            </div>
            <div class="modal-Body">
                <p>Guarde preferencias</p>
            </div>
            <!-- dialog buttons -->
            <div class="modal-footer">
                <form action="<?php echo base_url() ?>index.php/Home#services"
method="get">
                    <button type="submit" class="btn btn-success" >Aceptar</button>
                </form>
            </div>
        </div>
    </div>
</div>

```

```

</div>
</div>
<!-- /.modal -->

<script src="<?php echo base_url()
?>assets/vendor/jquery/jquery.min.js"></script>
<!-- Bootstrap Core JavaScript -->
<script src="<?php echo base_url()
?>assets/vendor/bootstrap/js/bootstrap.min.js"></script>
<script src="<?php echo base_url() ?>assets/js/dataTables.min.js"></script>

<script>
    $( window ).load(function() {
        putdatatable();
        $("#div-tabla").hide();
        if ($('#programa-selec').val()) {
            $('#Modallns').modal('show');
        }else{
            $('#ModalPref').modal('show');
        }
    };

    $('#programa').val($('#programa-selec').val());
    $('#programa').attr('disabled',true);
    if ($('#programa-selec').val() == 1) {
        $('#div-check').show();
    }else{
        $('#div-check').hide();
    }
    $('#div-oferta').hide();

```

```

});

$('#Inst-Consumo').click(function (e) {
    $('#Modallns').modal('show');
});

$("##check-oferta").click(function() {
    // this function will get executed every time the #home element is clicked
    (or tab-spacebar changed)
    if($(this).is(":checked")) // "this" refers to the element that fired the event
    {
        $('#div-oferta').show();
        $('#valcheck').val(1);
    }else{
        $('#div-oferta').hide();
        $('#valcheck').val(0);
    }
});

$('##btn-calcular').click(function (e) {

    e.preventDefault();

    var validator = 1;
    if ($('#programa-selec').val() == 1 && $("##check-oferta").is(":checked") )
    {
        $(".required").each(function (index) {
            if ($(this).val() == "") {
                $(this).addClass('input-required');
            }
        });
    }
});

```

```

        validator = 0;
    }else{
        $(this).removeClass('input-required');
    }

});

};

if (validator == 0) {
    $('#ModalResp').find('#modal-title').html('Completa los campos en
rojo');
    $('#ModalResp').modal('show');
    //return false;
}else{
    $.post("<?php                echo                base_url()
?>index.php/Algoritmos/algoritmoOferta", {
        hi1: $('#horaintv').val(),
        hf1: $('#horafintv').val(),
        hi2: $('#horainlav').val(),
        hf2: $('#horafinlav').val(),
        hi3: $('#horainbom').val(),
        hf3: $('#horafinbom').val(),
        hi4: $('#horainpl').val(),
        hf4: $('#horafinpl').val(),
        programa: $('#programa').val(),
        check: $('#valcheck').val()
    }, function (data) {
        var r = [];
        var tabla = $("#tablerespuesta");

```

```

        tabla.find('td.dataTables_empty').parent('tr').remove();
        table.destroy();
        $.each(data, function(key, value) {
            agregarFila(r,value);
        });
        tabla.find('tbody').html(r.join(""));
        putdatatable();
        $("#div-tabla").show();

    })

    .fail(function () {
        alert("Error en la conexion");
    })
}

});

```

```

function agregarFila(r,datos){
    r.push('<tr><td>');
    r.push(datos.dia);
    r.push('</td><td>');
    if (datos.dia == 1) {
        r.push('Lunes');
    }else if (datos.dia == 2) {
        r.push('Martes');
    }else if (datos.dia == 3) {
        r.push('Miercoles');
    }else if (datos.dia == 4) {
        r.push('Jueves');
    }else if (datos.dia == 5) {

```



```

        r.push('Viernes');
    }else if (datos.dia == 6) {
        r.push('Sabado');
    }else {
        r.push('Domingo');
    }
    r.push('</td><td>');
    if (datos.electrodomestico == 1) {
        r.push('Televisor');
    }else if (datos.electrodomestico == 2) {
        r.push('Nevera');
    }else if (datos.electrodomestico == 3) {
        r.push('Lavadora');
    }else if (datos.electrodomestico == 4) {
        r.push('Bombilla');
    }else {
        r.push('Plancha');
    }
    r.push('</td><td>');
    r.push(datos.hora1);
    r.push('</td><td>');
    r.push(datos.hora2);
    r.push('</td></tr>');
}

```

```

function putdatatable(){
    table = $('#tablerespuesta').DataTable({
        "language": {
            "decimal":      ",",
            "emptyTable":   "No se encontraron resultados",

```

```

"info":      "Mostrando de _START_ a _END_ de _TOTAL_ registros",
"infoEmpty": "Mostrando de 0 a 0 de 0 registros",
"infoFiltered": "(Filtrados de _MAX_ registros)",
"infoPostFix": "",
"thousands": ".",
"lengthMenu": "Mostrar _MENU_ registros",
"loadingRecords": "Cargando...",
"processing": "Procesando...",
"search":    "Buscar:",
"zeroRecords": "No se encontraron registros coincidentes",
"paginate": {
  "first":    "Primero",
  "last":     "Último",
  "next":     "Siguiente",
  "previous": "Anterior"
},
"aria": {
  "sortAscending": ": activate to sort column ascending",
  "sortDescending": ": activate to sort column descending"
}
},
"pagingType": "full_numbers",
"lengthMenu": [[-1], ["All"]],
"bFilter": false,
});
}

```

</script>

## B5.8 Home usuario estándar

El home del usuario estándar difiere en las secciones que presenta respecto al del administrador, a continuación su código.

```
<!DOCTYPE html>
<html lang="en">

<head>

    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="">
    <meta name="author" content="">

    <title>Poupawatt</title>

    <!-- Bootstrap Core CSS -->
    <link          href="<?php          echo          base_url()
?>assets/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">

    <!-- Custom Fonts -->
    <link href="<?php echo base_url() ?>assets/vendor/font-
awesome/css/font-awesome.min.css" rel="stylesheet"
        type="text/css">
    <link

href='https://fonts.googleapis.com/css?family=Open+Sans:300italic,400italic
,600italic,700italic,800italic,400,300,600,700,800'
        rel='stylesheet' type='text/css'>
    <link
```

```
href='https://fonts.googleapis.com/css?family=Merriweather:400,300,300itali  
c,400italic,700,700italic,900,900italic'
```

```
rel='stylesheet' type='text/css'>
```

```
<!-- Plugin CSS -->
```

```
<link href="<?php echo base_url() ?>assets/vendor/magnific-  
popup/magnific-popup.css" rel="stylesheet">
```

```
<!-- Theme CSS -->
```

```
<link href="<?php echo base_url() ?>assets/css/creative.min.css"  
rel="stylesheet">
```

```
<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and  
media queries -->
```

```
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
```

```
<!--[if lt IE 9]>
```

```
<script
```

```
src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
```

```
<script
```

```
src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>
```

```
<![endif]-->
```

```
</head>
```

```
<body id="page-top">
```

```
<nav id="mainNav" class="navbar navbar-default navbar-fixed-top">
```

```
<div class="container-fluid">
```

```
<!-- Brand and toggle get grouped for better mobile display -->
```

```

<div class="navbar-header">
  <button type="button" class="navbar-toggle collapsed" data-
toggle="collapse"
      data-target="#bs-example-navbar-collapse-1">
    <span class="sr-only">Toggle navigation</span> Menu <i
class="fa fa-bars"></i>
  </button>
  <a class="navbar-brand page-scroll" href="#page-
top">Poupawatt</a>
</div>

```

```

<!-- Collect the nav links, forms, and other content for toggling -->
<div class="collapse navbar-collapse" id="bs-example-navbar-collapse-
1">
  <ul class="nav navbar-nav navbar-right">
    <li>
      <a class="page-scroll" href="<?php echo base_url()
?>index.php/HomeAdmin/sensores">Sensores</a>
    </li>
    <li>
      <a class="page-scroll" href="<?php echo base_url()
?>index.php/HomeAdmin/electrodomesticos">Electrodomésticos</a>
    </li>
    <li>
      <a class="page-scroll" href="<?php echo base_url()
?>index.php/HomeAdmin/programas">Programas</a>
    </li>
    <li>
      <a class="page-scroll" href="<?php echo base_url()
?>index.php/HomeAdmin/usuarios">Usuarios</a>

```

```

        </li>
        <li>
            <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home/salir">Salir</a>
        </li>

    </ul>
</div>
<!-- /.navbar-collapse -->
</div>
<!-- /.container-fluid -->
</nav>

<header>
    <div class="header-content">
        <div class="header-content-inner">
            <h1 id="homeHeading">BIENVENIDO <?php echo $this->session-
>userdata('nombre'); ?></h1>
            <hr>
            <p>Vista Administrador </p>
            <a href="#about" class="btn btn-primary btn-xl page-
scroll">Empezar</a>
        </div>
    </div>
</header>

<section class="bg-primary" id="about">
    <div class="container">
        <div class="row">
            <div class="col-lg-8 col-lg-offset-2 text-center">

```

```
<h2 class="section-heading">Tus Facturas</h2>
<hr class="light">
<p class="text-faded">Lorem ipsum dolor sit amet, consectetur
adipisicing elit. Asperiores error illum
    ipsum nostrum numquam velit. Dolorem doloremque eaque est,
facere modi obcaecati possimus quae quas
    quisquam tempora totam vel, veniam.</p>
<a href="<?php echo base_url() ?>index.php/Home/facturas"
class="page-scroll btn btn-default btn-xl sr-button">Ir a mis Facturas!</a>
</div>
</div>
</div>
</section>
```

```
<section id="services">
<div class="container">
<div class="row">
<div class="col-lg-12 text-center">
<h2 class="section-heading">Tus Preferencias</h2>
<hr class="primary">
</div>
</div>
</div>
<div class="container">
<div class="row">
<div class="col-lg-8 col-lg-offset-2 text-center">
<p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. A
consectetur deserunt, est fuga nesciunt
    pariatur perferendis quisquam sunt? Corporis doloribus eius
eligendi esse laboriosam laborum magnam
```

```
sed! Excepturi quam, repudiandae?</p>
<a href="<?php echo base_url() ?>index.php/Home/preferencias"
class="btn btn-primary btn-xl page-scroll">Ir a mis Preferencias!</a>
</div>
</div>
```

```
</div>
</section>
```

```
<section id="noticias">
<div class="container">
<div class="row">
<div class="col-lg-12 text-center">
<h2 class="section-heading">Apagar Paga</h2>
<hr class="primary">
</div>
</div>
```

```
</div>
<div class="container">
<div class="row">
<div class="col-md-12">
<div class="col-md-6" style="text-align: justify;">
<p>El Presidente Juan Manuel Santos descartó el
racionamiento de energía eléctrica gracias al ahorro
de los colombianos y anunció que el Fenómeno del Niño
concluirá en mayo y no en junio como
estaba previsto.
</p>
```

```
<p> "Hoy podemos decir que agracias al horro de los
colombianos ya no existe el fantasma de un
```



racionamiento de energía", aseveró el Mandatario. Indicó que "ahorramos en cuatro semanas más de lo que nos habíamos propuesto ahorrar en seis semanas" y agregó que el ahorro de energía este viernes fue del 5.76 por ciento.

El Ministerio de Minas y Energía destacó los esfuerzos de las entidades públicas y privadas que han adoptado medidas para el buen uso de la energía y han aportado al ahorro nacional, logrando alejar el fantasma del apagón.

"Las empresas y las familias del país demostramos que unidos podemos lograr lo que nos proponemos. Agradecemos todo el apoyo que recibió la campaña Apagar Paga e invitamos a continuar con la cultura del ahorro", dijo el Presidente de la República.

```
</div>
<div class="col-md-6">
  
</div>
</div>
</div>
</div>
</section>
```

```

<section id="consumo">
  <aside class="bg-dark">
    <div class="container text-center">
      <div class="call-to-action">
        <h2>Revisa tu consumo</h2>

        <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. </p>

        <?php if (!count($sensor)) { ?>
          <p style="color: orangered;">No tienes sensores activados!
Solicitalos al administrador.</p>
        <?php } else { ?>

          <div class="col-md-6">
            <select name="" id="electrodomestic" class="form-control"
style="margin-bottom: 20px;">
              <option style="" class="form-control" value="">Selecione
Sensor</option>
              <?php foreach ($sensor as $val) {
                ?>
                <option value="<?php echo $val->Serial ?>"><?php
echo $val->Serial ?></option>
              <?php
              }
              ?>
            </select>

```

```
        <select name="" id="type" class="form-control inputSmall"
style="margin-bottom: 20px;" value="Select type">
            <option style="color: black !important;" value="0">Select an
option</option>
            <option style="color: black !important;"
value="1">kWh</option>
            <option style="color: black !important;"
value="2">$$$</option>
        </select>
```

```
        <input type="date" id="firstDate" class="form-control
inputSmall" style="margin-bottom: 20px;">
```

```
        <input type="number" id="firstHour" placeholder="Initial Hour"
class="form-control inputSmall" style="margin-bottom: 20px;">
```

```
        <input type="date" id="secondDate" class="form-control
inputSmall" style="margin-bottom: 20px;">
```

```
        <input type="number" id="SecondHour" placeholder="Final
Hour" class="form-control inputSmall" style="margin-bottom: 20px;">
```

```
        <p>&nbsp;</p>
```

```
        <button id="generate" class="btn btn-default btn-xl sr-
button">Consultar</button>
```

```
</div>
```

```

<div class="col-md-6">

    <div id="graphic"></div>
    <p id="totalSuma" style="color: white;"></p>
</div>

<?php } ?>

    <!--<a href="http://startbootstrap.com/template-
overviews/creative/" class="btn btn-default btn-xl sr-button">Download
    Now!</a-->
</div>
</div>
</aside>
</section>

<section id="contact">
    <div class="container">
        <div class="row">
            <div class="col-lg-8 col-lg-offset-2 text-center">
                <h2 class="section-heading">Poupawatt</h2>
                <hr class="primary">
                <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Architecto atque beatae corporis enim error eveniet excepturi facere, ipsa,
iure iusto, libero modi mollitia nostrum numquam officia provident quas vel
vero.</p>
            </div>
            <div class="col-lg-4 col-lg-offset-2 text-center">
                <i class="fa fa-phone fa-3x sr-contact"></i>

```

```

        <p>123-456-6789</p>
    </div>
    <div class="col-lg-4 text-center">
        <i class="fa fa-envelope-o fa-3x sr-contact"></i>

        <p><a                                href="mailto:your-email@your-
domain.com">administrador@poupawatt.com</a></p>
    </div>
</div>
</div>
</section>

<!-- jQuery -->
<script          src="<?php          echo          base_url()
?>assets/vendor/jquery/jquery.min.js"></script>

<!-- Bootstrap Core JavaScript -->
<script          src="<?php          echo          base_url()
?>assets/vendor/bootstrap/js/bootstrap.min.js"></script>

<!-- Plugin JavaScript -->
<script          src="http://cdnjs.cloudflare.com/ajax/libs/jquery-
easing/1.3/jquery.easing.min.js"></script>
<script          src="<?php          echo          base_url()
?>assets/vendor/scrollreveal/scrollreveal.min.js"></script>
<script  src="<?php  echo  base_url()  ?>assets/vendor/magnific-
popup/jquery.magnific-popup.min.js"></script>

<!-- Theme JavaScript -->

```

```
<script src="<?php echo base_url() ?>assets/js/creative.min.js"></script>
```

```
</body>
```

```
</html>
```

```
<script src="../../assets/js/remodal.js"></script>
```

```
<script src="<?php echo base_url() ?>assets/js/highcharts.js"></script>
```

```
<script src="https://code.highcharts.com/modules/exporting.js"></script>
```

```
<script>
```

```
$('#generate').click(function (e) {
```

```
    e.preventDefault();
```

```
    if ($('#electrodomestic').val() == "" || $('#type').val() == "" ||  
    $('#firstDate').val() == "" || $('#secondDate').val() == "") {  
        alert('Completa todos los campos');  
        return false;  
    }
```

```
    $.post("Home/generateSumGraphic", {
```

```
        electrodomestic: $('#electrodomestic').val(),
```

```
        type: $('#type').val(),
```

```
        firstDate: $('#firstDate').val(),
```

```
        secondDate: $('#secondDate').val(),
```

```
        firstHour: $('#firstHour').val(),
```

```

        secondHour: $('#SecondHour').val()
    }, function (data) {
        if ($('#electrodomestic').val() != 'all') {
            $('#totalSuma').html(data);
        }
    })
    .fail(function () {
        alert("error");
    })

$.post("Home/generateGraphic", {

    electrodomestic: $('#electrodomestic').val(),
    type: $('#type').val(),
    firstDate: $('#firstDate').val(),
    secondDate: $('#secondDate').val(),
    firstHour: $('#firstHour').val(),
    secondHour: $('#SecondHour').val()
}, function (data) {
    if ($('#electrodomestic').val() == 'all') {
        $('#totalSuma').html(data);

    } else {
        $('#graphic').html(data);
    }

})
    .fail(function () {
        alert("error");
    })

```

```
    })
  })
</script>
```

### **B5.9 Home administrador**

El home administrador cuenta con menos secciones que el estándar sin embargo su código es más extenso por la información que maneja

```
<!DOCTYPE html>
<html lang="en">
```

```
<head>
```

```
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <meta name="description" content="">
  <meta name="author" content="">
```

```
  <title>Poupawatt</title>
```

```
  <!-- Bootstrap Core CSS -->
```

```
  <link href="<?php echo base_url()
?>assets/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
```

```
  <!-- Custom Fonts -->
```

```
  <link href="<?php echo base_url() ?>assets/vendor/font-
awesome/css/font-awesome.min.css" rel="stylesheet"
  type="text/css">
  <link
```



```
href='https://fonts.googleapis.com/css?family=Open+Sans:300italic,400italic,600italic,700italic,800italic,400,300,600,700,800'
```

```
rel='stylesheet' type='text/css'>
```

```
</link
```

```
href='https://fonts.googleapis.com/css?family=Merriweather:400,300,300italic,400italic,700,700italic,900,900italic'
```

```
rel='stylesheet' type='text/css'>
```

```
<!-- Plugin CSS -->
```

```
<link href="<?php echo base_url() ?>assets/vendor/magnific-popup/magnific-popup.css" rel="stylesheet">
```

```
<!-- Theme CSS -->
```

```
<link href="<?php echo base_url() ?>assets/css/creative.min.css" rel="stylesheet">
```

```
<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries -->
```

```
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
```

```
<!--[if lt IE 9]>
```

```
<script
```

```
src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
```

```
<script
```

```
src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>
```

```
<![endif]-->
```

```
</head>
```

```

<body id="page-top">

<nav id="mainNav" class="navbar navbar-default navbar-fixed-top">
  <div class="container-fluid">
    <!-- Brand and toggle get grouped for better mobile display -->
    <div class="navbar-header">
      <button type="button" class="navbar-toggle collapsed" data-
toggle="collapse"
      data-target="#bs-example-navbar-collapse-1">
        <span class="sr-only">Toggle navigation</span> Menu <i
class="fa fa-bars"></i>
      </button>
      <a class="navbar-brand page-scroll" href="#page-
top">Poupawatt</a>
    </div>

    <!-- Collect the nav links, forms, and other content for toggling -->
    <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-
1">
      <ul class="nav navbar-nav navbar-right">
        <li>
          <a class="page-scroll" href="#about">Facturas</a>
        </li>
        <li>
          <a class="page-scroll" href="#services">Preferencias</a>
        </li>
        <li>
          <a class="page-scroll" href="#consumo">Consumo</a>
        </li>
        <li>

```

```

        <a class="page-scroll" href="#noticias">Noticias</a>
    </li>
    <li>
        <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home/modelos">Modelos</a>
    </li>
    <li>
        <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home/juego">Juego</a>
    </li>
    <li>
        <a class="page-scroll"
        href="<?php echo base_url()
?>index.php/Home/perfil">Ajustes</a>
    </li>
    <li>
        <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home/salir">Salir</a>
    </li>

</ul>
</div>
<!-- /.navbar-collapse -->
</div>
<!-- /.container-fluid -->
</nav>

<header>
    <div class="header-content">

```

```

<div class="header-content-inner">
  <h1 id="homeHeading">BIENVENIDO <?php echo $this->session-
>userdata('nombre'); ?></h1>
  <hr>
  <p>Esta aplicativo es de prueba </p>
  <a href="#about" class="btn btn-primary btn-xl page-
scroll">Empezar</a>
</div>
</div>
</header>

```

```

<section class="bg-primary" id="about">
  <div class="container">
    <div class="row">
      <div class="col-lg-8 col-lg-offset-2 text-center">
        <h2 class="section-heading">Tus Facturas</h2>
        <hr class="light">
        <p class="text-faded">Lorem ipsum dolor sit amet, consectetur
adipisicing elit. Asperiores error illum
        ipsum nostrum numquam velit. Dolorem doloremque eaque est,
facere modi obcaecati possimus quae quas
        quisquam tempora totam vel, veniam.</p>
        <a href="<?php echo base_url() ?>index.php/Home/facturas"
class="page-scroll btn btn-default btn-xl sr-button">Ir a mis Facturas!</a>
      </div>
    </div>
  </div>
</section>

```

```

<section id="services">

```

```

<div class="container">
  <div class="row">
    <div class="col-lg-12 text-center">
      <h2 class="section-heading">Tus Preferencias</h2>
      <hr class="primary">
    </div>
  </div>
</div>
<div class="container">
  <div class="row">
    <div class="col-lg-8 col-lg-offset-2 text-center">
      <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. A
consectetur deserunt, est fuga nesciunt
      <p>pariatur preferendis quisquam sunt? Corporis doloribus eius
eligendi esse laboriosam laborum magnam
      <p>sed! Excepturi quam, repudiandae?</p>
      <a href="<?php echo base_url() ?>index.php/Home/preferencias"
class="btn btn-primary btn-xl page-scroll">Ir a mis Preferencias!</a>
    </div>
  </div>
</div>
</div>
</section>
<section id="noticias">
  <div class="container">
    <div class="row">
      <div class="col-lg-12 text-center">
        <h2 class="section-heading">Apagar Paga</h2>
        <hr class="primary">
      </div>
    </div>
  </div>
</section>

```

</div>

</div>

<div class="container">

<div class="row">

<div class="col-md-12">

<div class="col-md-6" style="text-align: justify;">

<p>El Presidente Juan Manuel Santos descartó el racionamiento de energía eléctrica gracias al ahorro de los colombianos y anunció que el Fenómeno del Niño concluirá en mayo y no en junio como estaba previsto.

</p>

<p> "Hoy podemos decir que agracias al horro de los colombianos ya no existe el fantasma de un racionamiento de energía", aseveró el Mandatario. Indicó que "ahorramos en cuatro semanas más de lo que nos habíamos propuesto ahorrar en seis semanas" y agregó que el ahorro de energía este viernes fue del 5.76 por ciento.

</p>

<p> El Ministerio de Minas y Energía destacó los esfuerzos de las entidades públicas y privadas que han adoptado medidas para el buen uso de la energía y han aportado al ahorro nacional, logrando alejar el fantasma del apagón.</p>

<p>"Las empresas y las familias del país demostramos que unidos podemos lograr lo que nos

proponemos. Agradecemos todo el apoyo que recibió la campaña Apagar Paga e invitamos a continuar con la cultura del ahorro", dijo el Presidente de la República.</p>

```
</div>
<div class="col-md-6">
  
</div>
</div>
</div>
</div>
</section>
```

```
<section id="consumo">
  <aside class="bg-dark">
    <div class="container text-center">
      <div class="call-to-action">
        <h2>Revisa tu consumo</h2>

        <p><button id="Inst-Consumo" class="btn btn-success btn-sm sr-
button">Instrucciones >></button></p>
```

```
<?php if (!count($sensor)) { ?>
  <p style="color: orangered;">No tienes sensores activados!
Solicitalos al administrador.</p>
<?php } else { ?>
```

```

<div class="col-md-6">
    <select name="" id="electrodomestic" class="form-control"
style="margin-bottom: 20px;">
        <option style="" class="form-control" value="">Selezione
Sensor</option>
        <?php foreach ($sensor as $val) {
            ?>
            <option value="<?php echo $val->Serial ?>"><?php
echo $val->Serial ?></option>
        <?php
        }
        ?>
    </select>

```

```

<select name="" id="type" class="form-control inputSmall"
style="margin-bottom: 20px;" value="Select type">
    <option style="color: black !important;" value="0">Select an
option</option>
    <option style="color: black !important;"
value="1">kWh</option>
    <option style="color: black !important;"
value="2">$$$</option>
</select>

```

```

<input type="date" id="firstDate" class="form-control
inputSmall" style="margin-bottom: 20px;">

```



```
<input type="time" id="firstHour" placeholder="Initial Hour"
class="form-control inputSmall" style="margin-bottom: 20px;">
```

```
<input type="date" id="secondDate" class="form-control
inputSmall" style="margin-bottom: 20px;">
```

```
<input type="time" id="SecondHour" placeholder="Final Hour"
class="form-control inputSmall" style="margin-bottom: 20px;">
```

```
<p>&nbsp;</p>
```

```
<button id="generate" class="btn btn-default btn-xl sr-
button">Consultar</button>
```

```
</div>
```

```
<div class="col-md-6">
```

```
<div id="graphic"></div>
```

```
<p id="totalSuma" style="color: white;"></p>
```

```
</div>
```

```
<?php } ?>
```

```
<!--<a href="http://startbootstrap.com/template-
overviews/creative/" class="btn btn-default btn-xl sr-button">Download
Now!</a-->
```

```
</div>
```

```

    </div>
  </aside>
</section>

<section id="contact">
  <div class="container">
    <div class="row">
      <div class="col-lg-8 col-lg-offset-2 text-center">
        <h2 class="section-heading">Poupawatt</h2>
        <hr class="primary">
        <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Architecto atque beatae corporis enim error eveniet excepturi facere, ipsa,
iure iusto, libero modi mollitia nostrum numquam officia provident quas vel
vero.</p>
      </div>
      <div class="col-lg-4 col-lg-offset-2 text-center">
        <i class="fa fa-phone fa-3x sr-contact"></i>

        <p>123-456-6789</p>
      </div>
      <div class="col-lg-4 text-center">
        <i class="fa fa-envelope-o fa-3x sr-contact"></i>

        <p><a href="mailto:your-email@your-
domain.com">administrador@poupawatt.com</a></p>
      </div>
    </div>
  </div>
</section>

```

```
<!-- jQuery -->
<script src="<?php echo base_url()
?>assets/vendor/jquery/jquery.min.js"></script>
```

```
<!-- Bootstrap Core JavaScript -->
<script src="<?php echo base_url()
?>assets/vendor/bootstrap/js/bootstrap.min.js"></script>
```

```
<!-- Plugin JavaScript -->
<script src="http://cdnjs.cloudflare.com/ajax/libs/jquery-
easing/1.3/jquery.easing.min.js"></script>
<script src="<?php echo base_url()
?>assets/vendor/scrollreveal/scrollreveal.min.js"></script>
<script src="<?php echo base_url() ?>assets/vendor/magnific-
popup/jquery.magnific-popup.min.js"></script>
```

```
<!-- Theme JavaScript -->
<script src="<?php echo base_url() ?>assets/js/creative.min.js"></script>
```

```
</body>
```

```
<!-- Modal Instrucciones-->
<div id="Modallns" class="modal fade">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <button type="button" class="close" data-
dismiss="modal">&times;</button>
        <h4 id="modal-title" class="modal-title">Instrucciones</h4>
      </div>
```

```

    <div class="modal-Body">
        <p>Escriba aqui las instrucciones</p>
    </div>
    <!-- dialog buttons -->
    <div class="modal-footer">
        <button type="button" class="btn btn-success" data-
dismiss="modal">Aceptar</button>
    </div>
</div>
</div>
</div>
<!-- /.modal -->

</html>
<script src="../../assets/js/remodal.js"></script>
<script src="<?php echo base_url() ?>assets/js/highcharts.js"></script>
<script src="https://code.highcharts.com/modules/exporting.js"></script>

<script>
    $('#Inst-Consumo').click(function (e) {
        $('#Modallns').modal('show');
    });

    $('#generate').click(function (e) {

        e.preventDefault();

```

```

    if ($('#electrodomestic').val() == "" || $('#type').val() == "" ||
    $('#firstDate').val() == "" || $('#secondDate').val() == "") {
        alert('Completa todos los campos');
        return false;
    }

```

```

$.post("Home/generateSumGraphic", {

```

```

    electrodomestic: $('#electrodomestic').val(),
    type: $('#type').val(),
    firstDate: $('#firstDate').val(),
    secondDate: $('#secondDate').val(),
    firstHour: $('#firstHour').val(),
    secondHour: $('#SecondHour').val()
}, function (data) {
    if ($('#electrodomestic').val() != 'all') {
        $('#totalSuma').html(data);
    }
})
.fail(function () {
    alert("error");
});

```

```

$.post("Home/generateGraphic", {

```

```

    electrodomestic: $('#electrodomestic').val(),
    type: $('#type').val(),
    firstDate: $('#firstDate').val(),
    secondDate: $('#secondDate').val(),
    firstHour: $('#firstHour').val(),

```

```

        secondHour: $('#SecondHour').val()
    }, function (data) {
        if ($('#electrodomestic').val() == 'all') {
            $('#totalSuma').html(data);

        } else {
            $('#graphic').html(data);
        }

    })

    .fail(function () {
        alert("error");
    })
})
</script>

```

### **B5.10 Juego**

El juego cuenta con un formulario simple, en el que se recoge diferente información y el controlador del home estándar realiza las operaciones.

```

<!DOCTYPE html>
<html lang="en">

```

```

<head>

```

```

    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="">
    <meta name="author" content="">

```

```
<title>Poupawatt</title>
```

```
<!-- Bootstrap Core CSS -->
```

```
<link href="<?php echo base_url() ?>assets/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
```

```
<!-- Custom Fonts -->
```

```
<link href="<?php echo base_url() ?>assets/vendor/font-awesome/css/font-awesome.min.css" rel="stylesheet"
```

```
type="text/css">
```

```
<link
```

```
href='https://fonts.googleapis.com/css?family=Open+Sans:300italic,400italic,600italic,700italic,800italic,400,300,600,700,800'
```

```
rel='stylesheet' type='text/css'>
```

```
<link
```

```
href='https://fonts.googleapis.com/css?family=Merriweather:400,300,300italic,400italic,700,700italic,900,900italic'
```

```
rel='stylesheet' type='text/css'>
```

```
<!-- Plugin CSS -->
```

```
<link href="<?php echo base_url() ?>assets/vendor/magnific-popup/magnific-popup.css" rel="stylesheet">
```

```
<!-- Theme CSS -->
```

```
<link href="<?php echo base_url() ?>assets/css/creative.min.css" rel="stylesheet">
```

```

    <!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and
media queries -->
    <!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
    <!--[if lt IE 9]>
    <script
src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
    <script
src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>
    <![endif]-->

</head>

<body id="page-top">

<nav id="mainNav" class="navbar navbar-default navbar-fixed-top">
    <div class="container-fluid">
        <!-- Brand and toggle get grouped for better mobile display -->
        <div class="navbar-header">
            <button type="button" class="navbar-toggle collapsed" data-
toggle="collapse"
                data-target="#bs-example-navbar-collapse-1">
                <span class="sr-only">Toggle navigation</span> Menu <i
class="fa fa-bars"></i>
            </button>
            <a class="navbar-brand page-scroll" href="<?php echo base_url()
?>index.php/Home">Poupawatt</a>
        </div>

        <!-- Collect the nav links, forms, and other content for toggling -->

```



```

<div class="collapse navbar-collapse" id="bs-example-navbar-collapse-
1">
    <ul class="nav navbar-nav navbar-right">

        <li>
            <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home#about">Facturas</a>
        </li>
        <li>
            <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home#services">Preferencias</a>
        </li>
        <li>
            <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home#consumo">Consumo</a>
        </li>
        <li>
            <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home#noticias">Noticias</a>
        </li>
        <li>
            <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home/modelos">Modelos</a>
        </li>
        <li>
            <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home/perfil">Ajustes</a>
        </li>
        <li>

```

```

        <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home/salir">Salir</a>
    </li>

</ul>
</div>
<!-- /.navbar-collapse -->
</div>
<!-- /.container-fluid -->
</nav>

<section class="bg-primary" id="about">
    <div class="container">
        <div class="row">
            <div class="col-lg-8 col-lg-offset-2 text-center">
                <h2 class="section-heading">Juego</h2>
                <hr class="light">
                <p><button id="Inst-Consumo" class="btn btn-success btn-sm
sr-button">Instrucciones >></button></p>
            <div class="col-md-12" >
                <div class="col-md-6" >
                    <label for="inicio">Día Inicio</label>
                    <input type="date" id="diainicio" name="diainicio"
class="form-control inputFormulario required"/>
                </div>
                <div class="col-md-6" >
                    <label for="fin">Día Final</label>
                    <input type="date" id="diafin" name="diafin" class="form-
control inputFormulario required"/>

```

```

</div>
<label for="porcentaje">Porcentaje Ahorro</label>
<input type="number" min="0" max="100" name="porcentaje"
class="form-control inputFormulario required"/>
</div>

```

```

<div class="col-md-6">
  <hr class="light"></hr>
  <label for="inicio">Electrodomèstico 1</label>
  <select          name=""          id="electrodomestico1"
name="electrodomestico1" class="form-control inputFormulario required">
    <option value="">Seleccione Electrodomestico</option>
    <?php if ($electrodomesticos !== false) {
      foreach ($electrodomesticos as $selec) {
        ?>
          <option value=<?php if ($selec !== null
&& $selec->varElectrodomestico !== null) {echo ($selec-
>varElectrodomestico);}
            ?><?php if ($selec !== null && $selec-
>varElectrodomestico !== null) {echo ($selec->varElectrodomestico);}
            ?></option>
          <?php }}
        ?>
      }
    }
  </select>
  <div class="col-md-6" >
    <label for="inicio">Hora Inicio 1</label>
    <input type="time" id="hora1a" name="hora1a"
class="form-control inputFormulario required"/>

```

```

</div>
<div class="col-md-6" id="zona-programa">
  <label for="fin">Hora Final 1</label>
  <input type="time" id="hora1b" name="hora1b"
class="form-control inputFormulario required"/>
</div>

<div class="col-md-6" id="zona-programa">
  <label for="inicio">Hora Inicio 2</label>
  <input type="time" id="hora1c" name="hora1c" class="form-
control inputFormulario"/>
</div>

<div class="col-md-6" id="zona-programa">
  <label for="fin">Hora Final 2</label>
  <input type="time" id="hora1d"
name="hora1d" class="form-control inputFormulario"/>
</div>

<div class="col-md-6" id="zona-programa">
  <label for="inicio">Hora Inicio 3</label>
  <input type="time" id="hora1e"
name="hora1e" class="form-control inputFormulario"/>
</div>

<div class="col-md-6" id="zona-programa">
  <label for="fin">Hora Final 3</label>
  <input type="time" id="hora1f" name="hora1f" class="form-
control inputFormulario"/>
</div>
</div>

```

```

<div class="col-md-6">
    <hr class="light"></hr>
    <label for="inicio">Electrodomèstico 2</label>
    <select          name=""          id="electrodomestico2"
name="electrodomestico2" class="form-control inputFormulario required">
        <option value="">Seleccione Electrodomestico</option>
        <?php if ($electrodomesticos !== false) {
            foreach ($electrodomesticos as $selec) {
                ?>
                    <option value=<?php if ($selec !== null
&& $selec->varElectrodomestico !== null) {echo ($selec-
>varElectrodomestico);}
                ?>><?php if ($selec !== null && $selec-
>varElectrodomestico !== null) {echo ($selec->varElectrodomestico);}
                ?></option>
            }
        }
    ?>
</select>
<div class="col-md-6" >
    <label for="inicio">Hora Inicio 1</label>
    <input type="time" id="hora2a" name="hora2a"
class="form-control inputFormulario required"/>
</div>
<div class="col-md-6" id="zona-programa">
    <label for="fin">Hora Final 1</label>
    <input type="time" id="hora2b" name="hora2b"
class="form-control inputFormulario required"/>
</div>

```

```

        <div class="col-md-6" id="zona-programa">
            <label for="inicio">Hora Inicio 2</label>
            <input type="time" id="hora2c" name="hora2c"
class="form-control inputFormulario"/>
        </div>
        <div class="col-md-6" id="zona-programa">
            <label for="fin">Hora Final 2</label>
            <input type="time" id="hora2d" name="hora2d"
class="form-control inputFormulario"/>
        </div>
        <div class="col-md-6" id="zona-programa">
            <label for="inicio">Hora Inicio 3</label>
            <input type="time" id="hora2e" name="hora2e"
class="form-control inputFormulario"/>
        </div>
        <div class="col-md-6" id="zona-programa">
            <label for="fin">Hora Final 3</label>
            <input type="time" id="hora2f" name="hora2f" class="form-
control inputFormulario"/>
        </div>
    </div>
    <div class="col-md-6">
        <hr class="light"></hr>
        <label for="inicio">Electrodomèstico 3</label>
        <select name="" id="electrodomestico3"
name="electrodomestico3" class="form-control inputFormulario required">
            <option value="">Seleccione Electrodomestico</option>
            <?php if ($electrodomesticos !== false) {

```

```

        foreach ($electrodomesticos as $selec) {
            ?>
                <option value=<?php if ($selec !== null
&& $selec->varElectrodomestico !== null) {echo ($selec-
>varElectrodomestico);}
                    ?>><?php if ($selec !== null && $selec-
>varElectrodomestico !== null) {echo ($selec->varElectrodomestico);}
                    ?></option>
                <?php }}
            ?>
        </select>
        <div class="col-md-6" >
            <label for="inicio">Hora Inicio 1</label>
            <input type="time" id="hora3a" name="hora3a"
class="form-control inputFormulario required"/>
        </div>
        <div class="col-md-6" id="zona-programa">
            <label for="fin">Hora Final 1</label>
            <input type="time" id="hora3b" name="hora3b"
class="form-control inputFormulario required"/>
        </div>
        <div class="col-md-6" id="zona-programa">
            <label for="inicio">Hora Inicio 2</label>
            <input type="time" id="hora3c" name="hora3c"
class="form-control inputFormulario"/>
        </div>
        <div class="col-md-6" id="zona-programa">
            <label for="fin">Hora Final 2</label>

```

```

        <input type="time" id="hora3d" name="hora3d"
class="form-control inputFormulario"/>
    </div>

```

```

    <div class="col-md-6" id="zona-programa">
        <label for="inicio">Hora Inicio 3</label>
        <input type="time" id="hora3e" name="hora3e"
class="form-control inputFormulario"/>
    </div>

```

```

    <div class="col-md-6" id="zona-programa">
        <label for="fin">Hora Final 3</label>
        <input type="time" id="hora3f" name="hora3f" class="form-
control inputFormulario"/>
    </div>

```

```

</div>

```

```

<div class="col-md-6">
    <hr class="light"></hr>
    <label for="inicio">Electrodomèstico 4</label>
    <select name="" id="electrodomestico4"
name="electrodomestico4" class="form-control inputFormulario required">
        <option value="">Seleccione Electrodomestico</option>
        <?php if ($electrodomesticos !== false) {
            foreach ($electrodomesticos as $selec) {
                ?>
                    <option value=<?php if ($selec !== null
&& $selec->varElectrodomestico !== null) {echo ($selec-
>varElectrodomestico);}
                ?><?php if ($selec !== null && $selec-
>varElectrodomestico !== null) {echo ($selec->varElectrodomestico);}
            }
        }
    </select>

```



```
?></option>
```

```
<?php }}
```

```
?>
```

```
</select>
```

```
<div class="col-md-6" id="zona-programa">
```

```
<label for="inicio">Hora Inicio 1</label>
```

```
<input type="time" id="hora4a" name="hora4a"
```

```
class="form-control inputFormulario required"/>
```

```
</div>
```

```
<div class="col-md-6" id="zona-programa">
```

```
<label for="fin">Hora Final 1</label>
```

```
<input type="time" id="hora4b" name="hora4b"
```

```
class="form-control inputFormulario required"/>
```

```
</div>
```

```
<div class="col-md-6" id="zona-programa">
```

```
<label for="inicio">Hora Inicio 2</label>
```

```
<input type="time" id="hora4c" name="hora4c"
```

```
class="form-control inputFormulario"/>
```

```
</div>
```

```
<div class="col-md-6" id="zona-programa">
```

```
<label for="fin">Hora Final 2</label>
```

```
<input type="time" id="hora4d" name="hora4d"
```

```
class="form-control inputFormulario"/>
```

```
</div>
```

```
<div class="col-md-6" id="zona-programa">
```

```
<label for="inicio">Hora Inicio 3</label>
```

```

        <input type="time" id="hora4e" name="hora4e"
class="form-control inputFormulario"/>
    </div>
    <div class="col-md-6" id="zona-programa">
        <label for="fin">Hora Final 3</label>
        <input type="time" id="hora4f" name="hora4f" class="form-
control inputFormulario"/>
    </div>
</div>

<div class="col-md-6">
    <hr class="light"></hr>
    <label for="inicio">Electrodomèstico 5</label>
    <select name="" id="electrodomestico5"
name="electrodomestico5" class="form-control inputFormulario required">
        <option value="">Seleccione Electrodomestico</option>
        <?php if ($Electrodomesticos !== false) {
            foreach ($Electrodomesticos as $Selec) {
                ?>
                    <option value=<?php if ($Selec !== null
&& $Selec->varElectrodomestico !== null) {echo ($Selec-
>varElectrodomestico);}
                    ?>><?php if ($Selec !== null && $Selec-
>varElectrodomestico !== null) {echo ($Selec->varElectrodomestico);}
                    ?></option>
                <?php }}
            ?>
        </select>
    <div class="col-md-6" id="zona-programa">

```

```

        <label for="inicio">Hora Inicio 1</label>
        <input type="time" id="hora5a" name="hora5a"
class="form-control inputFormulario required"/>
    </div>
    <div class="col-md-6" id="zona-programa">
        <label for="fin">Hora Final 1</label>
        <input type="time" id="hora5b" name="hora5b"
class="form-control inputFormulario required"/>
    </div>

    <div class="col-md-6" id="zona-programa">
        <label for="inicio">Hora Inicio 2</label>
        <input type="time" id="hora5c" name="hora5c"
class="form-control inputFormulario"/>
    </div>
    <div class="col-md-6" id="zona-programa">
        <label for="fin">Hora Final 2</label>
        <input type="time" id="hora5d" name="hora5d"
class="form-control inputFormulario"/>
    </div>

    <div class="col-md-6" id="zona-programa">
        <label for="inicio">Hora Inicio 3</label>
        <input type="time" id="hora5e" name="hora5e"
class="form-control inputFormulario"/>
    </div>
    <div class="col-md-6" id="zona-programa">
        <label for="fin">Hora Final 3</label>
        <input type="time" id="hora5f" name="hora5f" class="form-
control inputFormulario"/>

```

```

        </div>
    </div>
    <div class="col-md-12">
        <hr class="light"></hr>
        <button id="calcular" class="page-scroll btn btn-default btn-xl
sr-button">Calcular
        </button>
    </div>
</div>
</div>
</div>
</section>

</body>

</html>

<!-- Modal -->
<div id="ModalResp" class="modal fade">
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-
dismiss="modal">&times;</button>
                <h4 id="modal-title" class="modal-title"></h4>
            </div>
            <!-- dialog buttons -->
            <div class="modal-footer">
                <button type="button" class="btn btn-danger" data-
dismiss="modal">CERRAR</button>

```

```

    </div>
    <input type="hidden" name="id_grav_delete" value="">
  </div>
</div>
</div>
</div>
<!-- /.modal -->

<!-- Modal Instrucciones-->
<div id="Modallns" class="modal fade">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <button type="button" class="close" data-
dismiss="modal">&times;</button>
        <h4 id="modal-title" class="modal-title">Instrucciones</h4>
      </div>
      <div class="modal-Body">
        <p>Escriba aqui las instrucciones</p>
      </div>
      <!-- dialog buttons -->
      <div class="modal-footer">
        <button type="button" class="btn btn-success" data-
dismiss="modal">Aceptar</button>
      </div>
      <input type="hidden" name="id_grav_delete" value="">
    </div>
  </div>
</div>
</div>
<!-- /.modal -->

```

```

<script src="<?php echo base_url()
?>assets/vendor/jquery/jquery.min.js"></script>
<!-- Bootstrap Core JavaScript -->
<script src="<?php echo base_url()
?>assets/vendor/bootstrap/js/bootstrap.min.js"></script>

```

```

<script>
    $( document ).ready(function() {
        $('#Modallns').modal('show');
    });

    $('#Inst-Consumo').click(function (e) {
        $('#Modallns').modal('show');
    });

    $('#calcular').click(function (e) {
        e.preventDefault();
        var validator = 1;

        $(".required").each(function (index) {
            if ($(this).val() == " ") {
                $(this).addClass('input-required');
                validator = 0;
            }else{
                $(this).removeClass('input-required');
            }
        });

        if (validator == 0) {

```

```

    $('#ModalResp').find('#modal-title').html('Completa los campos en
rojo');
    $('#ModalResp').modal('show');
    //return false;
} else {
    $.post("<?php          echo          base_url()
?>index.php/HomeAdmin/calculo_juego", {
    dia : $('#diainicio').val(),
    dia2 : $('#diafin').val(),
    porcentaje : $('#porcentaje').val(),
    primeraelec : $('#electrodomestico1').val(),
    segundaelec : $('#electrodomestico2').val(),
    terceraelec : $('#electrodomestico3').val(),
    cuartaelec : $('#electrodomestico4').val(),
    quintaelec : $('#electrodomestico5').val(),
    hora1a : $('#hora1a').val(),
    hora1b : $('#hora1b').val(),
    hora1c : $('#hora1c').val(),
    hora1d : $('#hora1d').val(),
    hora1e : $('#hora1e').val(),
    hora1f : $('#hora1f').val(),
    hora2a : $('#hora2a').val(),
    hora2b : $('#hora2b').val(),
    hora2c : $('#hora2c').val(),
    hora2d : $('#hora2d').val(),
    hora2e : $('#hora2e').val(),
    hora2f : $('#hora2f').val(),
    hora3a : $('#hora3a').val(),
    hora3b : $('#hora3b').val(),
    hora3c : $('#hora3c').val(),

```

```
hora3d :$('#hora3d').val(),
hora3e :$('#hora3e').val(),
hora3f :$('#hora3f').val(),
hora4a :$('#hora4a').val(),
hora4b :$('#hora4b').val(),
hora4c :$('#hora4c').val(),
hora4d :$('#hora4d').val(),
hora4e :$('#hora4e').val(),
hora4f :$('#hora4f').val(),
hora5a :$('#hora5a').val(),
hora5b :$('#hora5b').val(),
hora5c :$('#hora5c').val(),
hora5d :$('#hora5d').val(),
hora5e :$('#hora5e').val(),
hora5f :$('#hora5f').val(),
```

```
}, function (data) {
    $('#ModalResp').find('#modal-title').html(data.message);
    $('#ModalResp').modal('show');
```

```
})
.fail(function () {
    alert("Error en la conexion");
})
```

```
}
```

```
});
```

```
</script>
```



## B5.11 Facturas

Al igual que los otros views este es un formulario que utiliza el controlador descrito arriba para guardar y mostrar información, el formulario es sencillo ya que tiene campos que están en las facturas y solo hay que copiar.

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="utf-8">
```

```
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1">
```

```
  <meta name="description" content="">
```

```
  <meta name="author" content="">
```

```
  <title>Poupawatt</title>
```

```
  <!-- Bootstrap Core CSS -->
```

```
  <link href="<?php echo base_url()>assets/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
```

```
  <!-- Custom Fonts -->
```

```
  <link href="<?php echo base_url()>assets/vendor/font-awesome/css/font-awesome.min.css" rel="stylesheet"
```

```
    type="text/css">
```

```
  <link
```

```
    href='https://fonts.googleapis.com/css?family=Open+Sans:300italic,400italic,600italic,700italic,800italic,400,300,600,700,800'
```

```
    rel='stylesheet' type='text/css'>
```

```

<link
href='https://fonts.googleapis.com/css?family=Merriweather:400,300,300itali
c,400italic,700,700italic,900,900italic'
rel='stylesheet' type='text/css'>

<!-- Plugin CSS -->
<link href="<?php echo base_url() ?>assets/vendor/magnific-
popup/magnific-popup.css" rel="stylesheet">

<!-- Theme CSS -->
<link href="<?php echo base_url() ?>assets/css/creative.min.css"
rel="stylesheet">

<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and
media queries -->
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
<!--[if lt IE 9]>
<script
src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
<script
src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>
<![endif]-->

</head>

<body id="page-top">

<nav id="mainNav" class="navbar navbar-default navbar-fixed-top">

```

```

<div class="container-fluid">
  <!-- Brand and toggle get grouped for better mobile display -->
  <div class="navbar-header">
    <button type="button" class="navbar-toggle collapsed" data-
toggle="collapse"
      data-target="#bs-example-navbar-collapse-1">
      <span class="sr-only">Toggle navigation</span> Menu <i
class="fa fa-bars"></i>
    </button>
    <a class="navbar-brand page-scroll" href="<?php echo base_url()
?>index.php/Home">Poupawatt</a>
  </div>

  <!-- Collect the nav links, forms, and other content for toggling -->
  <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-
1">
    <ul class="nav navbar-nav navbar-right">

      <li>
        <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home#services">Preferencias</a>
      </li>
      <li>
        <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home#consumo">Consumo</a>
      </li>
      <li>
        <a class="page-scroll" href="#noticias">Noticias</a>
      </li>
      <li>

```

```

        <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home/modelos">Modelos</a>
    </li>
    <li>
        <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home/juego">Juego</a>
    </li>
    <li>
        <a class="page-scroll"
            href="<?php echo base_url() ?>.index.php/perfil">Ajustes</a>
    </li>
    <li>
        <a class="page-scroll" href="<?php echo base_url()
?>index.php/Home/salir">Salir</a>
    </li>

```

```

    </ul>
</div>
<!-- /.navbar-collapse -->
</div>
<!-- /.container-fluid -->
</nav>

```

```

<section class="bg-primary" id="about">
    <div class="container">
        <div class="row">
            <div class="col-lg-8 col-lg-offset-2 text-center">
                <h2 class="section-heading">Tus Facturas</h2>
                <hr class="light">
            </div>
        </div>
    </div>

```

```

<div class="col-md-6" id="lista-facturas">
  <?php if (!count($facturas)) { ?>
    <p id="no-facturas">No tienes facturas agregadas!</p>
  <?php } else {
    foreach ($facturas as $factura) {
      ?>
      <p class="pull-left"><a class='editarFactura' data-
id="<?php echo $factura->idFactura; ?>"
style='color: #f5f5f5; cursor: pointer;'>
Factura
      Número: <?php echo $factura->datePeriodoIni; ?>
</a>
      </p>
      <p>&nbsp;</p>
      <?php
    }
  }
  ?>
</div>

```

```

<div class="col-md-6" id="zona-factura">
  <input type="text" id="cuenta" placeholder="Numero Cuenta"
class="form-control inputFormulario required"/>
  <select name="" id="servicio" class="form-control
inputFormulario required">
    <option value="">Seleccione Clase de Servicio</option>
    <option value="Residencial">Residencial</option>

```

```

        <option value="Comercial">Comercial</option>
        <option value="Industrial">Industrial</option>
    </select>
    <label for="inicio">Periodo Facturado Inicio</label>
    <input type="date" id="inicioDate" class="form-control
inputFormulario required"/>
    <label for="fin">Periodo Facturado Final</label>
    <input type="date" id="fin" class="form-control inputFormulario
required"/>
    <input type="text" id="lectura_actual" placeholder="Lectura
Actual"
        class="form-control inputFormulario required"/>
    <input type="text" id="lectura_anterior" placeholder="Lectura
Anterior"
        class="form-control inputFormulario required"/>
    <input type="text" id="total_consumo" placeholder="Total
Consumo"
        class="form-control inputFormulario required"/>
    <input type="text" id="CPT" placeholder="CPT" class="form-
control inputFormulario required"/>
    <input type="text" id="kwh" placeholder="Valor kWh"
class="form-control inputFormulario required"/>
    <input type="text" id="total_pago" placeholder="Total Pago"
        class="form-control inputFormulario required"/>
    <hr class="light">
    <button id="crearFactura" class="page-scroll btn btn-default btn-
xl sr-button">Agregar Factura
    </button>

</div>

```

```

        </div>
    </div>
</div>
</section>

</body>

</html>

<script src="<?php echo base_url()
?>assets/vendor/jquery/jquery.min.js"></script>

<script>
    $('#crearFactura').click(function (e) {

        e.preventDefault();

        var validator = 1;

        $(".required").each(function (index) {
            if ($(this).val() == "") {
                $(this).addClass('input-required');
                validator = 0;
            }

        });

        if (validator == 0) {
            alert('Completa los campos en rojo');
        }
    });

```

```

        //return false;
    }

$.post("crear_factura", {
    cuenta: $('#cuenta').val(),
    servicio: $('#servicio').val(),
    inicio: $('#inicioDate').val(),
    fin: $('#fin').val(),
    lectura_actual: $('#lectura_actual').val(),
    lectura_anterior: $('#lectura_anterior').val(),
    total_consumo: $('#total_consumo').val(),
    CPT: $('#CPT').val(),
    kwh: $('#kwh').val(),
    total_pago: $('#total_pago').val()
}, function (data) {
    alert("Factura Creada!");
    $('#no-facturas').empty();
    $('#lista-facturas').append(data);
})
.fail(function () {
    alert("Error en la conexion");
})
})

$( "#lista-facturas" ).on( "click", ".editarFactura", function() {
$.post("cargar_factura", {
    cuenta: $(this).attr('data-id')
}, function (data) {

```



```
$('#zona-factura').html(data);  
})  
.fail(function () {  
    alert("Error en la conexion");  
})  
});
```

```
</script>
```

## 12. ANEXO C

### Manual de Usuario aplicación móvil

Figura C1. Logo aplicación móvil.



Fuente: Los autores

#### **C.1 Descripción de la aplicación**

Poupa Watt es una aplicación para smartphones con versión 4.3 o superior del sistema Android. Permite al usuario consultar su consumo de energía eléctrica y revisar los históricos del mismo, de igual manera posee un juego que brinda la oportunidad al usuario de interactuar con los diferentes electrodomésticos que el posee. Con el fin de generar un nuevo hábito de consumo y lograr que el usuario encuentre la mejor forma de ahorrar el valor que este desee.

#### **C.2 Instalación de la Aplicación**

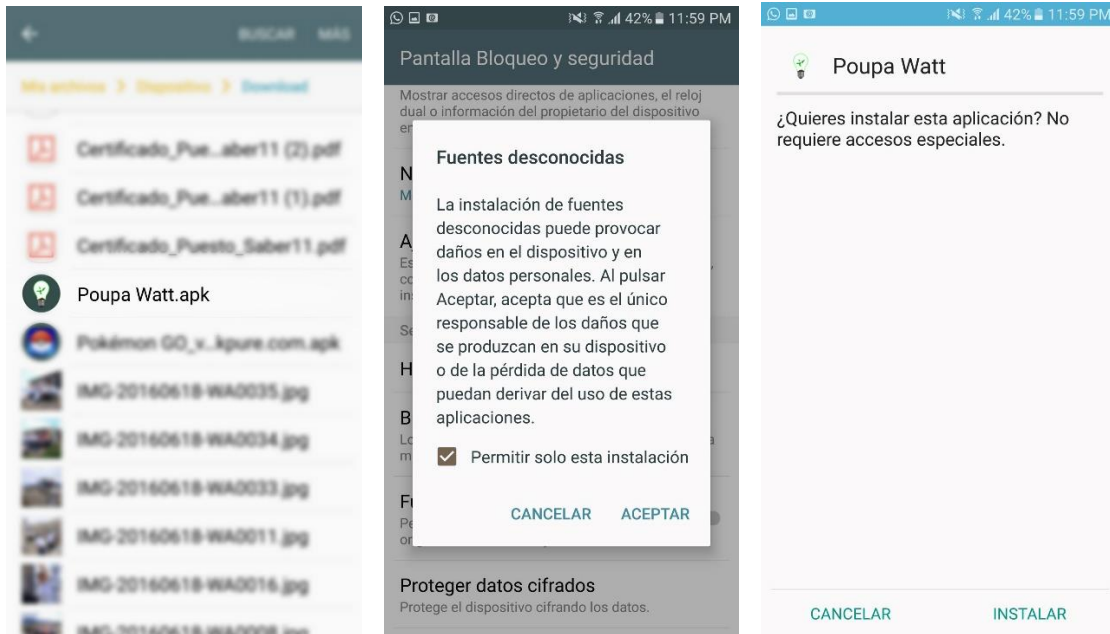
Para la instalación basta con obtener el APK<sup>1</sup> de la aplicación. Una vez descargada en el celular procedemos la instalación. Normalmente Android trae configurado bloquear la instalación de aplicaciones fuera de la playstore, por ello sencillamente lo que procedemos hacer; es permitir la instalación de fuentes desconocidas sin

---

<sup>1</sup> *Android Application Package* significado en español: Aplicación empaquetada de Android // Formato de extensión para sistema Android

ningún tipo de riesgo. Una vez aceptemos este permiso, la aplicación comenzará a instalarse.

Figura C2. Instalación APK.



Fuente: Los autores

### C.3 Navegación por la Aplicación

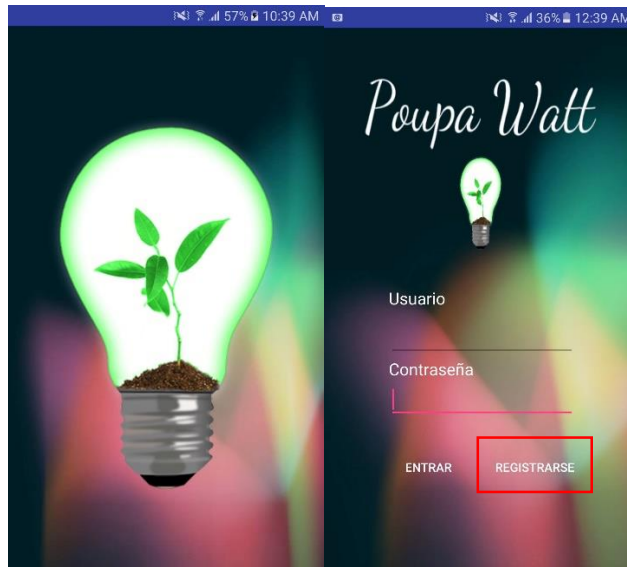
#### C.3.1 PANTALLA DE INICIO

Al momento de ingresar a la aplicación se despliega un screen splash<sup>2</sup> que muestra el logotipo de nuestra aplicación. Seguido a esto encontramos el espacio de Login; donde se es necesario introducir los datos correspondientes para poder acceder. Si el usuario no posee una cuenta, este cuenta con la posibilidad de loguearse dirigiéndose al botón de registrar que lo llevará a una nueva ventana.

Figura C3. Pantalla de inicio.

---

<sup>2</sup> Pantalla de presentación de la aplicación



Fuente: Los autores

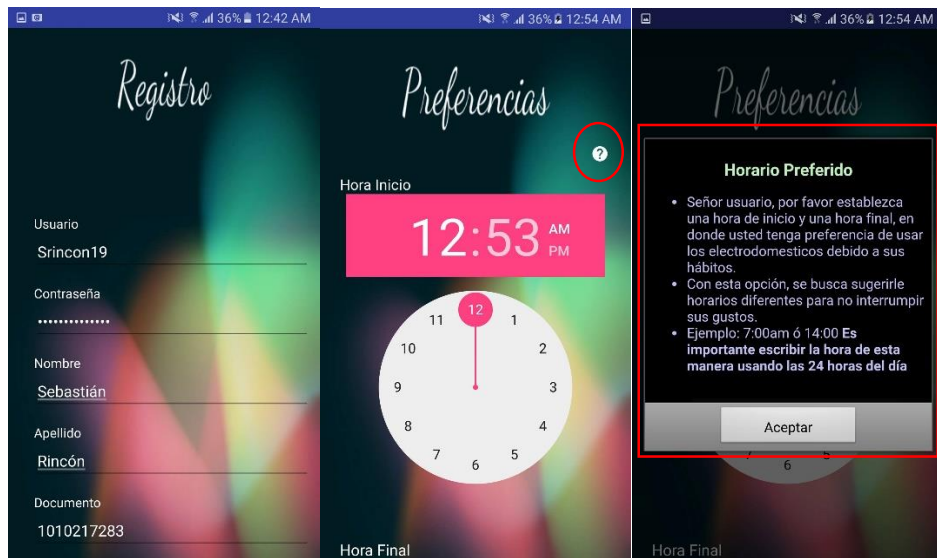
### **C.3.2 REGISTRO**

Para el registro de un usuario nuevo se requieren sus principales datos (nombre, apellido, documento, etc...) para que estos sean registrados en la base de datos de la aplicación. Una vez completado con los datos que se solicitan, presionamos el botón de registrar. Ahí pasamos al layout<sup>3</sup> de preferencias, donde se pide al usuario registrar su preferencia principal, es decir; el electrodoméstico que el usuario frecuente más o desee que no se modifique dentro de los algoritmos.

Figura C4. Registro.

---

<sup>3</sup> Croquis, esquema, o bosquejo de distribución de las piezas o elementos que se encuentran dentro de un diseño en particular

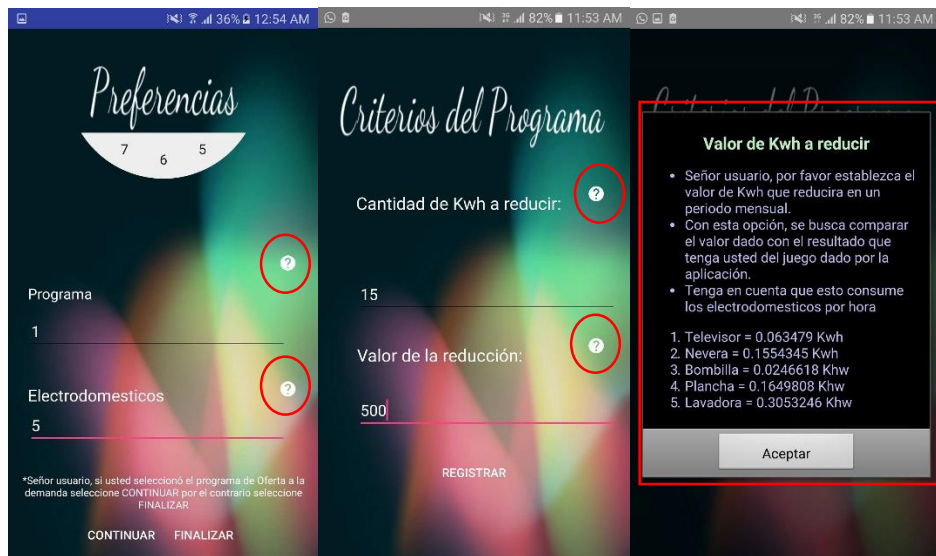


Fuente: Los autores

Es por eso que se hace necesario introducir la hora inicial y la hora final de uso de dicho electrodoméstico. En la parte derecha de cada campo aparece un botón de ayuda con un signo de interrogación (?), donde se despliega un diálogo que explica el formato y como llenar los datos solicitados en el campo correspondiente.

En la parte inferior del registro de preferencias encontramos dos campos más; programa y electrodomésticos.

Figura C5. Registro preferencias



Fuente: Los autores

En la aplicación contamos dos tipos de programas a la respuesta de la demanda de energía; Oferta a la demanda, donde el usuario oferta una cantidad de Kwh (Vatio-hora) por un precio que este considere correcto y con el cual tenga como meta a lograr, por otro lado contamos con el programa Tiempo de uso, este maneja dos precios diferentes de Kwh dependiendo de la hora, ya sea hora pico que va de 6:00 pm hasta las 12:00 am y la hora valle que va de las 12:00 am hasta las 5:00 pm correspondientemente. Es por eso que al final de la layout donde se encuentran los botones existe un indicativo que dice “Señor usuario si usted seleccionó el programa de Oferta a la demanda seleccione CONTINUAR por el contrario seleccione FINALIZAR. En lo anterior descrito esta una breve explicación de estos dos tipos de programas, pero el usuario cuenta con la posibilidad de una opción de ayuda en la aplicación donde encontrará una explicación más amplia de estos.

Al seleccionar FINALIZAR se retorna de nuevo a la página de inicio, es decir al Login inicial mostrado anteriormente. Pero por el contrario si accede al registro de los criterios del programa de oferta a la demanda, se hace necesario introducir la cantidad de Kwh que el usuario reducirá en un periodo de una semana. Para esta

decisión se debe tener en cuenta que este es el consumo de los electrodomésticos en una hora:

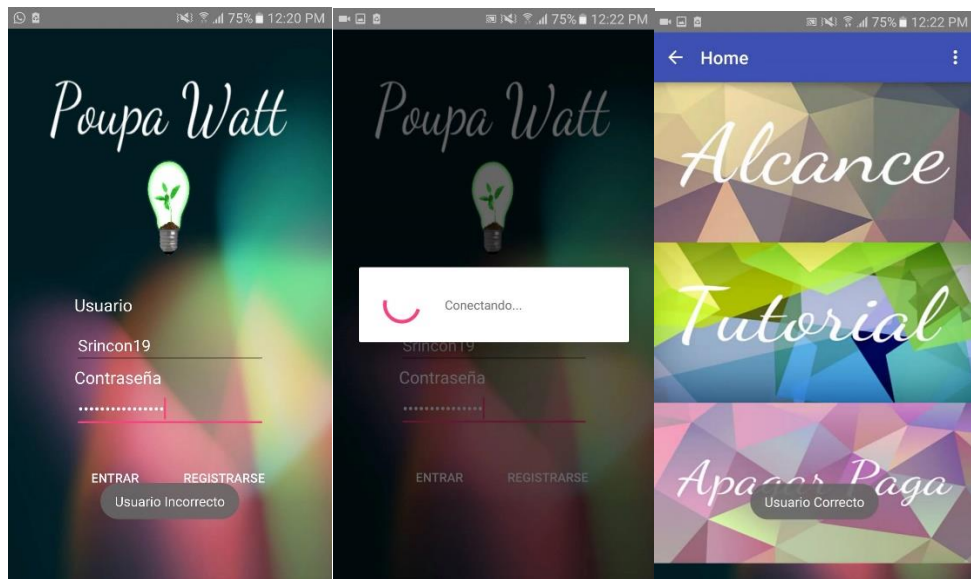
Televisor = 0.063479; || Nevera = 0.1554345; || Lavadora = 0.3053246; || Bombilla = 0.0246618; || Plancha = 0.1649808;

El valor en pesos (\$) es decidido por el usuario puesto que será lo que le paguen si logra reducir el valor de Kwh introducido en la otra casilla. De igual manera posee el botón de ayuda a la derecha (?) en donde se explica lo que debe de introducir en las casillas.

### ***C3.3. LOGIN***

Después de terminado el registro, la aplicación nos redirige a la ventana principal donde se realiza el login correspondiente con los datos que se acaban de registrar, puesto que estos ya se encuentran almacenados en el sistema. Si los datos son incorrectos aparecerá un mensaje informando que los datos que acabamos de escribir son erróneos. Una vez se introduzcan los datos correctos la aplicación nos accede al menú principal, donde el usuario podrá interactuar con las diferentes opciones que esta cuenta.

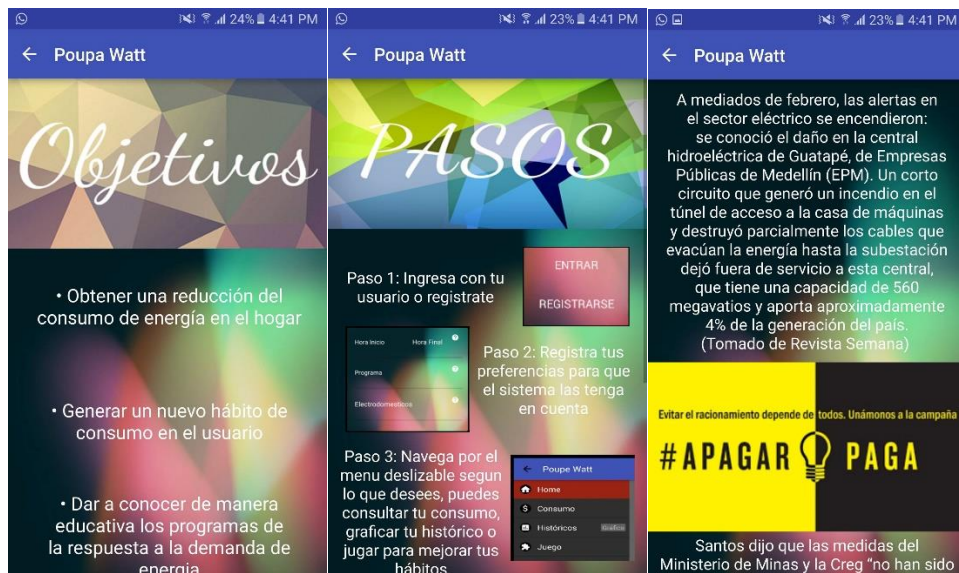
Figura C6. Login



Fuente: Los autores

Una vez dentro, el usuario tiene la posibilidad de escoger entre las 3 opciones que allí se encuentran planteadas como lo son; Alcance, Tutorial y Apagar Paga. Cada una de estas opciones contiene información para el usuario, como se muestra a continuación.

Figura C7. Home

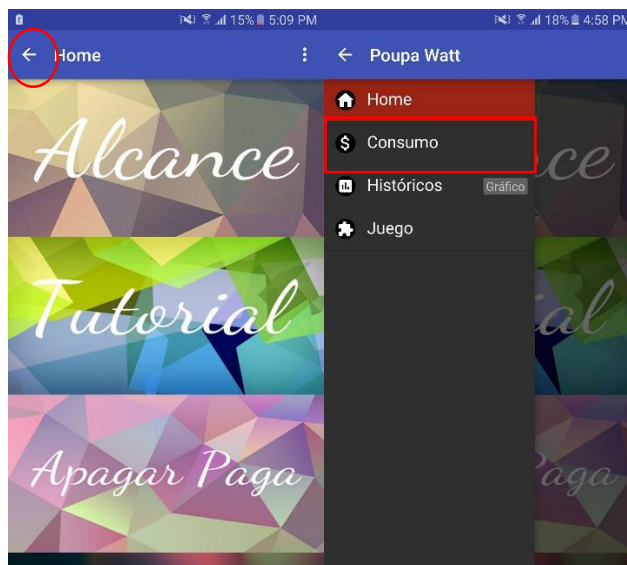




Fuente: Los autores

En la opción de “Alcance” se muestran los objetivos de la aplicación y del sistema, los cuales el usuario ha de cumplir al seguir los pasos dados por la aplicación y la plataforma web. Por otro lado en “Tutorial” el usuario tiene algunos pasos que puede seguir para que así se ayude y logre utilizar la aplicación, es decir, un corto manual como este. Por último en la opción de “Apagar paga” es una pequeña sección de noticias donde se muestra el proyecto que implanto el estado para generar una reducción en el consumo de energía eléctrica en los hogares colombianos, el cual se encuentra muy cercano a los objetivos de la aplicación.

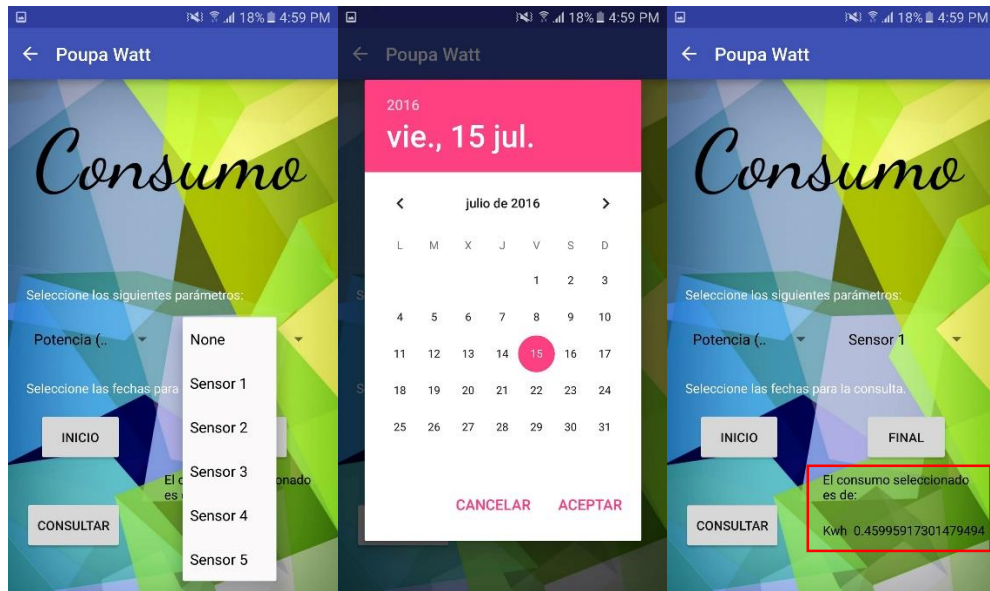
Figura C8. Selección consumo.



Fuente: Los autores

Al presionar la flecha (←) en la parte superior izquierda como lo muestra la imagen, se despliega el menú deslizante que le permite al usuario realizar la navegación entre las diferentes opciones que presenta la aplicación, ya sea para consultar el consumo, graficar los históricos o bien intentar generar la mejor manera de ahorrar energía en su hogar con el juego.

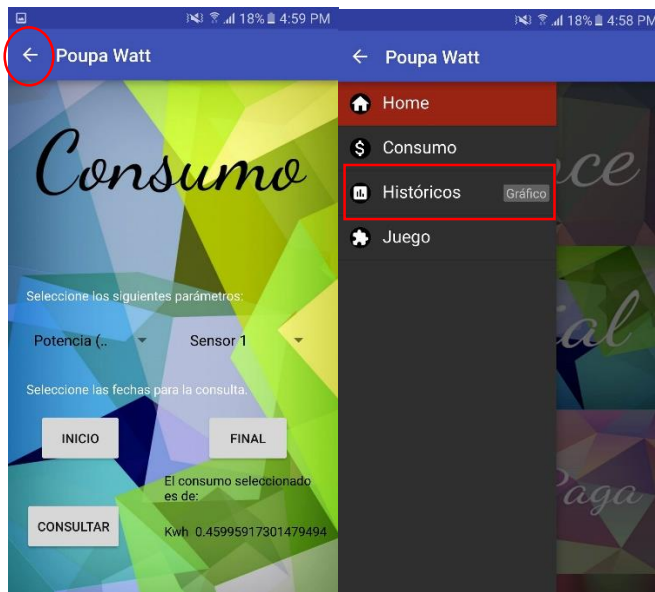
Figura C9. Consumo



Fuente: Los autores

Una vez estando en la sección de consumo, lo primero a realizar es seleccionar los parámetros que se solicitan como lo son; saber si desea ver el consumo en potencia (kwh) o precio (\$) como también posteriormente seleccionar el rango de fechas en las que se desea observar el consumo, colocando la fecha de inicio y la fecha final, para que así la aplicación pueda generar la sumatoria, y finalmente en la parte inferior derecha se muestra el consumo que ha tenido el usuario dentro del rango establecido.

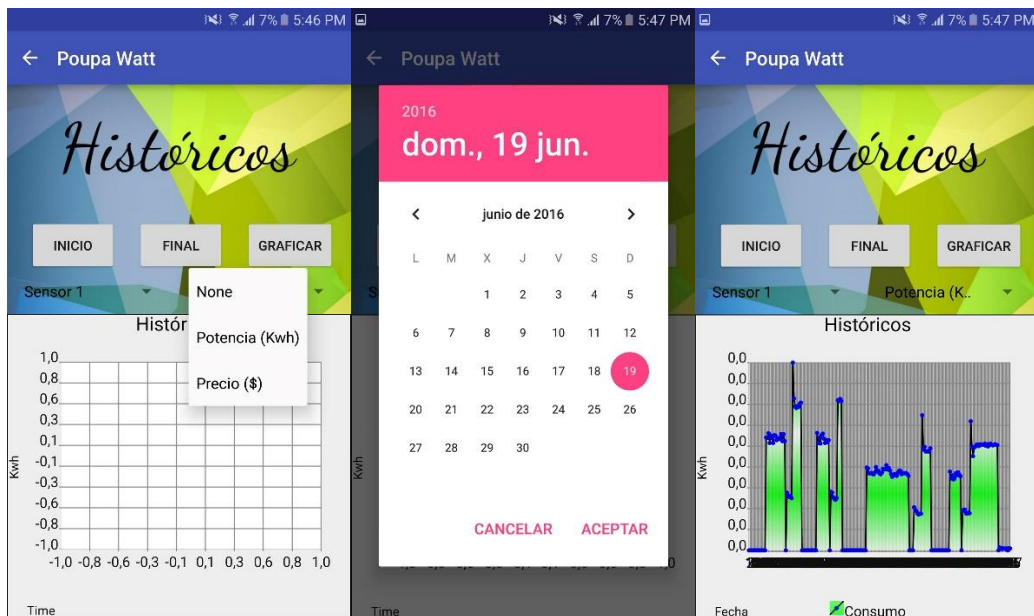
Figura C10. Selección históricos.



Fuente: Los autores.

Otra opción que encontramos al presionar la flecha (←) en la parte superior izquierda es graficar los históricos como se muestra a continuación.

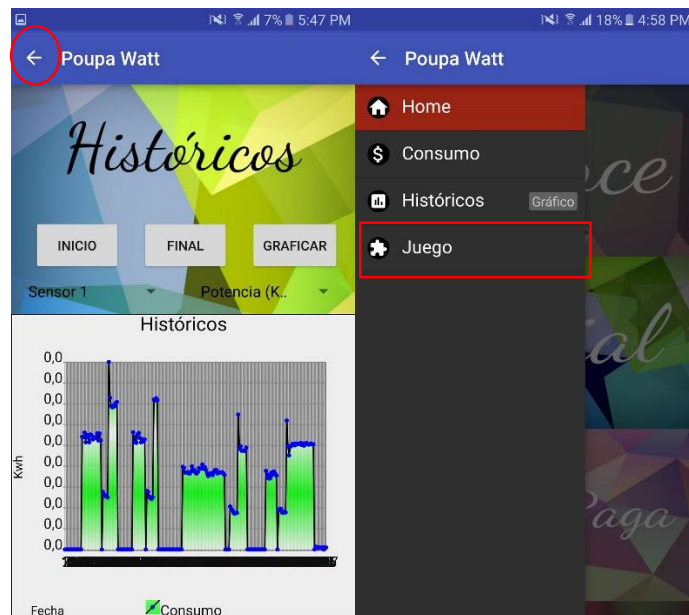
Figura C11. Históricos



Fuente: Los autores

Esta parte es muy parecida a la sección anterior donde se consultaba el consumo, puesto que es necesario seleccionar los parámetros como el consumo en potencia (kwh) o precio (\$) y de igual forma tanto el día de inicio como el día final para que se puedan dar los intervalos de la gráfica. Una vez dados podemos seleccionar la opción de graficar y se obtiene el consumo del usuario en ese tiempo.

Figura C12. Selección Juego.



Fuente: Los autores

Por ultimo al presionar la flecha (←) en la parte superior izquierda podemos encontrar la última opción en nuestro menú deslizante, “El juego” donde el usuario podrá interactuar con las horas y los consumos de los electrodomésticos con el fin de buscar un ahorro que pueda ejecutar en su hogar.

Figura C13. Juego



Fuente: Los autores

El juego tiene la capacidad de que el usuario interactúe con 5 electrodomésticos; la nevera, la plancha, la lavadora, el televisor y la lámpara. Al entrar en esta sección lo primero que encontramos son las instrucciones en las cuales está paso a paso como se llenan los espacios para obtener el resultado que se desea obtener en nuestro ahorro. Lo primero a realizar es seleccionar una fecha de inicio y una fecha final que calculará el consumo, ya sea de un día, de una semana o de un mes dependiendo de lo que escoja el usuario.

Luego de eso debe introducir que porcentaje de ese consumo desea reducir, es decir, si quiere reducir el 5% el 10% o el 30% de lo que consume en un día, una semana o un mes dependiendo del intervalo que haya deja entre las fechas registradas. Luego de esto puede escoger cualquiera de los 5 electrodomésticos en donde tiene la posibilidad de poner 3 horarios diferentes, puede ser uno en la mañana, uno en la tarde y uno en noche o como el usuario prefiera. Por otro lado puede solo dar un intervalo de horario y no usar los otros dos. Es muy importante que se ingrese la hora en formato de hora militar (24 horas), una vez complete los 5 electrodomésticos en las horas que se desee se selecciona la opción de jugar.

Lo que la aplicación hace es validar las horas introducidas junto con los electrodomésticos y realizar la sumatoria respectiva para saber si cumple con la condición de reducir el 5% el 10% o el porcentaje que haya ingresado el usuario, cuando sea inferior se mostrará un mensaje como el que vemos en la imagen diciendo que se cumplió el objetivo con la cantidad de Kwh y el valor en precios que se ahorraría si se cumplen las horas pactadas en el juego, por el contrario si el objetivo no se logra el juego genera un texto que nos refleja que no logramos ahorrar lo que deseábamos y la cantidad de Kwh que se consumieron de más.

Figura C14. Resultados juego



Fuente: Los autores.

## 13. ANEXO D

### Manual del programador aplicación móvil

#### 1. Introducción

Dentro de este anexo, se explicará la estructura total de la aplicación móvil de forma que el usuario o programador que decida hacerle algún cambio pueda orientarse. De manera tal que tenga un entendimiento fácil, con el fin de que tenga la capacidad de abortar la aplicación desde todos los puntos de vista posible. Se estructurará el proyecto para que pueda entender en que consiste y el porqué de cada uno de los archivos utilizados en este proyecto.

#### 2. Estructura de Android Studio

Android Studio es un software libre enfocado en el entorno del desarrollo Android que se divide en tres partes importantes para el entendimiento de su funcionamiento partiendo de la carpeta **MANIFEST**, allí es donde se almacena uno de los archivos .xml más importantes como lo es el *AndroidManifest*, el cual es el encargado del control y los valores de inicio de la aplicación móvil. Por otro lado está la carpeta **JAVA** en donde se contienen todo los archivos .java que son los encargados de la lógica programable de cada una de las actividades y layouts que componen la app. Es decir esta es la columna vertebral que se encarga de conectar todo y unir la aplicación. Por ultimo encontramos la carpeta **RES** que es donde se encuentran todos los recursos que podremos usar para el desarrollo, es decir, allí se encontrarán cargadas todas las imágenes necesarias para la interfaz como cada uno de los layouts con toda la parte gráfica correspondiente. De la misma manera esta posee los value donde se guardan todos los strings, colores, styles y demás para la personalización del sistema.

##### 2.1 Estructura de los archivos carpeta Manifest

Como se mencionó anteriormente esta es la carpeta que contiene un único archivo pero es el más importante, porque todo archivo .java que no se encuentre declarado en este generará error dentro de la compilación puesto que genera conflicto de enlaces. Esta parte es la que centra en dar las reglas por las cuales se registrará la app. Su lenguaje de programación es en **JAVA SCRIPT** por aquello que bajo este es el que se tiende a desarrollar todo tipo de aplicativos. El archivo que está dentro de esta carpeta se llama:

- ***AndroidManifest.xml***: Dentro de este archivo encontramos gran parte de la configuración inicial de la aplicación, puesto que aquí es donde se declara el nombre de la aplicación, el icono que debe estar previamente cargado en la carpeta de **RES** y el estilo que tendrá. Parte de lo más importante es la declaración de los .java que son utilizados dentro de la carpeta **JAVA** porque como se mencionaba, si no se declaran se genera un error en la compilación.

Si se desea agregar una nueva clase sencillamente dentro del AndroidManifest en la parte donde se declaran las actividades se escribe de esta manera <activity android:name=".Login"></activity>.

El código fuente es el siguiente:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="v1.tesis.app.sebastianrincon.com.poupewatt">

    <uses-permission android:name="android.permission.INTERNET"/>

    <application
        android:name=".Myaplicacion"
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity

            android:name=".SplashScreenActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <activity android:name=".Inicio"></activity>
        <activity android:name=".Login"></activity>
        <activity android:name=".Register"></activity>
        <activity android:name=".Preferencias"></activity>
        <activity android:name=".Preferencias2"></activity>
        <activity
            android:name=".Juego"
            android:parentActivityName=".Inicio" >
            <meta-data
                android:name="android.support.PARENT_ACTIVITY"
                android:value=".Inicio" />
```



```

</activity>
  <activity
    android:name=".Grafica"
    android:parentActivityName=".Inicio" >
    <meta-data
      android:name="android.support.PARENT_ACTIVITY"
      android:value=".Inicio" />
  </activity>

  <activity
    android:name=".Descrip1"
    android:parentActivityName=".Inicio" >
    <meta-data
      android:name="android.support.PARENT_ACTIVITY"
      android:value=".Inicio" />
  </activity>

  <activity
    android:name=".Consumo"
    android:parentActivityName=".Inicio" >
    <meta-data
      android:name="android.support.PARENT_ACTIVITY"
      android:value=".Inicio" />
  </activity>

  <activity
    android:name=".Descrip2"
    android:parentActivityName=".Inicio" >
    <meta-data
      android:name="android.support.PARENT_ACTIVITY"
      android:value=".Inicio" />
  </activity>

  <activity
    android:name=".Descrip3"
    android:parentActivityName=".Inicio" >
    <meta-data
      android:name="android.support.PARENT_ACTIVITY"
      android:value=".Inicio" />
  </activity>

```

```
</application>
```

```
</manifest>
```

## 2.2 Estructura de los archivos carpeta Java

Dentro de esta encontramos toda la estructura de la aplicación puesto que son los archivos que contienen la programación principal, es decir donde se declara cada objeto puesto en la interfaz y se conectan entre sí. Es por esto que es donde más archivos se encuentran como lo veremos a continuación:

- ***SplashScreenActivity.java***: Esta activity, es la encargada de generar una imagen por 4 segundos recién inicia la aplicación. Es por eso que se llama Screen Splash, para modificar ese tiempo de duración sencillamente es modificar el valor cuando se está declarando la variable, es decir, en esta línea de código:

```
private static final long SPLASH_SCREEN_DELAY = 4000;
```

El código Fuente es:

```
package v1.tesis.app.sebastianrincon.com.poupewatt;
```

```
import android.app.Activity;  
import android.content.Intent;  
import android.content.pm.ActivityInfo;  
import android.os.Bundle;  
import android.view.Window;
```

```
import java.util.Timer;  
import java.util.TimerTask;
```

```
public class SplashScreenActivity extends Activity {
```

```
    // Set the duration of the splash screen  
    private static final long SPLASH_SCREEN_DELAY = 4000;
```

```
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);
```

```

// Set portrait orientation
setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);
// Hide title bar
requestWindowFeature(Window.FEATURE_NO_TITLE);

setContentView(R.layout.splash_screen);

TimerTask task = new TimerTask() {
    @Override
    public void run() {

        // Start the next activity
        Intent mainIntent = new Intent().setClass(
            SplashScreenActivity.this, Login.class);
        startActivity(mainIntent);

        // Close the activity so the user won't able to go back this
        // activity pressing Back button
        finish();
    }
};

// Simulate a long loading process on application startup.
Timer timer = new Timer();
timer.schedule(task, SPLASH_SCREEN_DELAY);
}
}

```

- **Login.java:** Como su nombre lo indica es donde se genera el logeo de la aplicación, entonces es donde se utilizan los archivos **PHP** para generar la conexión y la consulta de la base de datos. Para nuestro código los llamamos de esta manera:

```
private static final String LOGIN_URL = "http://198.199.72.168/cas/login1.php";
```

De igual manera se anida el layout correspondiente que en este caso sería:

```
super.onCreate(savedInstanceState);
setContentView(R.layout.login);
```

Después se tiene toda la estructura donde se maneja `onPreExecute` que es lo que se ejecuta cuando se activa la clase, el `doInBackground` que es lo que se ejecuta en segundo plano mientras se realiza la operación o todo lo que se programe en ella y finalmente el `onPostExecute` que es lo que finalmente se ejecutara después del procesamiento en segundo plano.

El código fuente es:

```
package v1.tesis.app.sebastianrincon.com.poupewatt;

import java.util.ArrayList;
import java.util.List;
import org.apache.http.NameValuePair;
import org.apache.http.message.BasicNameValuePair;
import org.json.JSONException;
import org.json.JSONObject;

import android.app.Activity;
import android.app.AlertDialog;
import android.app.ProgressDialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.SharedPreferences;
import android.content.SharedPreferences.Editor;
import android.os.AsyncTask;
import android.os.Bundle;
import android.preference.PreferenceManager;
import android.support.v7.app.ActionBarActivity;
import android.util.Log;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class Login extends Activity implements OnClickListener {

    private EditText user, pass;
    private Button mSubmit, mRegister;
    private int loco;
    private int resul;
```

```

private String mensg;

private ProgressDialog pDialog;

// Clase JSONParser
JSONParser jsonParser = new JSONParser();

//private static final String LOGIN_URL = "http://172.18.7.127/cas/login1.php";
private static final String LOGIN_URL = "http://198.199.72.168/cas/login1.php";
// private static final String LOGIN_URL = "http://172.18.7.226/cas/login.php";

// La respuesta del JSON es
private static final String TAG_SUCCESS = "success";
private static final String TAG_MESSAGE = "message";

@Override
public void onBackPressed() {

    AlertDialog.Builder builder = new AlertDialog.Builder(this);
    builder.setMessage("¿Desea salir de la aplicación?");
    builder.setTitle("Salir");
    builder.setPositiveButton("Si", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {
            finish();
        }
    });
    builder.setNegativeButton("No", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {
            dialog.cancel();
        }
    });

    AlertDialog dialog = builder.create();
    dialog.show();
}

@Override
protected void onCreate(Bundle savedInstanceState) {
    // TODO Auto-generated method stub

```

```

super.onCreate(savedInstanceState);
setContentView(R.layout.login);

// setup input fields
user = (EditText) findViewById(R.id.username);
pass = (EditText) findViewById(R.id.password);

// setup buttons
mSubmit = (Button) findViewById(R.id.login);
mRegister = (Button) findViewById(R.id.register);

// register listeners
mSubmit.setOnClickListener(this);
mRegister.setOnClickListener(this);

}

@Override
public void onClick(View v) {
    // TODO Auto-generated method stub
    switch (v.getId()) {
        case R.id.login:
            new AttemptLogin().execute();
            break;
        case R.id.register:
            Intent i = new Intent(this, Register.class);
            startActivity(i);
            break;

        default:
            break;
    }
}

class AttemptLogin extends AsyncTask<String, String, String> {

    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        progressDialog = new ProgressDialog(Login.this);
        progressDialog.setMessage("Conectando...");
    }
}

```

```

        pDialog.setIndeterminate(false);
        pDialog.setCancelable(true);
        pDialog.show();
    }

    @Override
    protected String doInBackground(String... args) {
        int success;
        String username = user.getText().toString();
        String password = pass.getText().toString();

        // Building Parameters
        List params = new ArrayList();
        params.add(new BasicNameValuePair("username", username));
        params.add(new BasicNameValuePair("password", password));

        Log.d("request!", "starting");
        // getting product details by making HTTP request
        JSONObject json = jsonParser.makeHttpRequest(LOGIN_URL, "POST",
            params);

        // check your log for json response
        Log.d("Login attempt", json.toString());

        try {

            resul = json.getInt("success");
            mensg = json.getString("message");

        } catch (JSONException e) {
            e.printStackTrace();
        }

        return null;
    }

    protected void onPostExecute(String file_url) {
        // dismiss the dialog once product deleted
        pDialog.dismiss();
    }

```

```

        if (resul < 2){

            Toast to = Toast.makeText(getApplicationContext(), msg,
Toast.LENGTH_LONG);
            to.show();

            Intent lol = new Intent().setClass(
                Login.this, Inicio.class);
            startActivity(lol);

            finish();

        }else {

            Toast to2 = Toast.makeText(getApplicationContext(), msg,
Toast.LENGTH_LONG);
            to2.show();
        }
    }
}
}
}

```

- **Register.java:** Si no se tiene usuario se debe registrar en donde igualmente se llama un archivo **PHP** el cual hará el registro en la base de datos de todos los datos ingresados por el usuario, el llamado se realiza de la siguiente manera:

```

private static String url_create_Device =
"http://198.199.72.168/cas/agregarUsers.php";

```

Para anidar el layout de esta clase se conecta de esta manera:

```

super.onCreate(savedInstanceState);
setContentView(R.layout.registro);

```

Se maneja la estructura de tres fases como se mencionó en el login utilizando el `onPreExecute`, `doInBackground` y el `onPostExecute`. De igual manera se utilizó el `JSONParser` para generar la conexión entre **Android** y **PHP**.



El código fuente es:

```
package v1.tesis.app.sebastianrincon.com.poupewatt;
```

```
import android.app.Activity;
import android.app.Notification;
import android.app.ProgressDialog;
import android.content.Intent;
import android.os.AsyncTask;
import android.os.Bundle;
import android.support.v7.app.ActionBarActivity;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
```

```
import org.apache.http.NameValuePair;
import org.apache.http.message.BasicNameValuePair;
import org.json.JSONException;
import org.json.JSONObject;
import java.util.ArrayList;
import java.util.List;
```

```
public class Register extends Activity {
```

```
    private ProgressDialog pDialog ; // para mostrar el dialogo
    private
        EditText
        txtusername,
    txtpassword,txtnombre,txtapellido,txtdocumento,txtdireccion,txttelefono,txtestrato,tx
    temail;
```

```
    private int valor;
    private String mens;
```

```
    JSONParser jsonParser = new JSONParser();//clase para realizar las
    transferencia
```

```
    //cuadros de texto
```

```
    //private
    static
    String
    url_create_Device
    =
    "http://172.18.7.226/cas/agregarUsers.php";//direccion url del archivo php
    private
    static
    String
    url_create_Device
    =
    "http://198.199.72.168/cas/agregarUsers.php";//direccion url del archivo php
```

```

//      private      static      String      url_create_Device      =
"http://172.18.47.127/cas/agregarUsers.php";//direccion url del archivo php

private static final String TAG_SUCCESS = "success"; //para verificar si se realizo
bien transferencia
private static final String TAG_MESSAGE = "message"; //para conocer el tipo de
error o los mensajes

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.registro);

    // declaracion de los Edit Text
    txtusername = (EditText) findViewById(R.id.username);
    txtpassword = (EditText) findViewById(R.id.password);
    txtnombre = (EditText) findViewById(R.id.nombre);
    txtapellido = (EditText) findViewById(R.id.apellido);
    txtdocumento = (EditText) findViewById(R.id.documento);
    txtdireccion = (EditText) findViewById(R.id.direccion);
    txttelefono = (EditText) findViewById(R.id.telefono);
    txtestrato = (EditText) findViewById(R.id.estrato);
    txtemail = (EditText) findViewById(R.id.email);

    //declaracion del boton Create button

    Button btnCreateDevice = (Button) findViewById(R.id.btnCreate);

    // declaracion del evento del boton
    btnCreateDevice.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(View view) {
            // creating new Empleado in background thread
            new CreateNewDevice().execute();
            //cuando se hace click al boton se empieza el procesos de agregar
        }
    });
}

/**

```

```

* Background Async Task to Create new Device
* */
class CreateNewDevice extends AsyncTask<String, String, String> {

    /**
     * Antes de empezar Asynk Task se muestra un mensaje de dialgo.
     */
    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        pDialog = new ProgressDialog(Register.this);
        pDialog.setMessage("Registrando Usuario...");
        pDialog.setIndeterminate(false);
        pDialog.setCancelable(true);
        pDialog.show();

    }

    /**
     * Proceso de agregar
     */
    protected String doInBackground(String... args) {

        //asignacion de las variables
        String username = txtusername.getText().toString();
        String password = txtpassword.getText().toString();
        String nombre = txtnombre.getText().toString();
        String apellido = txtapellido.getText().toString();
        String documento = txtdocumento.getText().toString();
        String direccion = txtdireccion.getText().toString();
        String telefono = txttelefono.getText().toString();
        String estrato = txtestrato.getText().toString();
        String email = txtemail.getText().toString();

        Log.d("enviar",      username.toString()      +      password.toString()+
nombre.toString()      +      apellido.toString()      +      documento.toString()      +
direccion.toString()+ telefono.toString() + estrato.toString()); // solo es un mensaje
en log para ver si se gurdan las variables.
    }
}

```

//aqui se lista los elementos con la etiqueta respectiva para enviarla al php y las procese correctamente

```
List<NameValuePair> params = new ArrayList<NameValuePair>();
params.add(new BasicNameValuePair("username", username));
params.add(new BasicNameValuePair("password", password));
params.add(new BasicNameValuePair("nombre", nombre));
params.add(new BasicNameValuePair("apellido", apellido));
params.add(new BasicNameValuePair("documento", documento));
params.add(new BasicNameValuePair("direccion", direccion));
params.add(new BasicNameValuePair("telefono", telefono));
params.add(new BasicNameValuePair("estrato", estrato));
params.add(new BasicNameValuePair("email", email));
```

```
// obteniendo JSON Object
```

```
// aqui se envia utilizando la url antes declarada, usando el metodo POST y
enviando los parametros
```

```
JSONObject json = jsonParser.makeHttpRequest(url_create_Device,
"POST", params);
```

```
// checkar en log cat la respuesta
```

```
Log.d("Create Response", json.toString());
```

```
// revisar si el tag resulto exitoso
```

```
try{
```

```
    valor = json.getInt("success");
    mens = json.getString("message");
```

```
    Log.d("Funciono", json.toString());
```

```
}catch (JSONException e){
    e.printStackTrace();
}
```

```
return null;
```

```

    }

    /**
     * antes de completar el background se cierra el dialogo
     * **/
    protected void onPostExecute(String file_url) {
        // dismiss the dialog once done
        pDialog.dismiss();
        if (valor < 2){

            Toast to = Toast.makeText(getApplicationContext(), "Usuario
Registrado", Toast.LENGTH_LONG);
            to.show();

            Intent lol = new Intent().setClass(
                Register.this, Preferencias.class);
            startActivity(lol);
            finish();

        }else {

            if (valor <4){

                Toast to2 = Toast.makeText(getApplicationContext(), "El Usuario ya se
encuentra registrado", Toast.LENGTH_LONG);
                to2.show();
                pDialog.dismiss();

            } else {

                Toast to2 = Toast.makeText(getApplicationContext(), "Error en la base
de datos", Toast.LENGTH_LONG);
                to2.show();
                pDialog.dismiss();
            }

        }

    }
}

```

```
}  
}
```

- **Preferencias.java:** Luego de registrar los datos más importantes se registran las preferencias principales del usuario. En donde entran a valer timepickers que son los atributos de tiempo usado dentro del lenguaje de **Android** como todos los anteriores se utiliza el JSONParser para hacer la conexión de los archivos **PHP** que más adelante se explicarán uno a uno. Para llamar a el correspondiente de esta clase se realiza de la siguiente manera:

```
private static String url_create_Device =  
"http://198.199.72.168/cas/agregarPreferences.php";
```

De igual forma tienen su layout correspondiente anidado de esta manera:

```
super.onCreate(savedInstanceState);  
setContentView(R.layout.preferencias);
```

Se maneja la estructura de tres fases como se mencionó en el login utilizando el onPreExecute, doInBackground y el onPostExecute.

El código fuente es:

```
package v1.tesis.app.sebastianrincon.com.poupewatt;  
  
import android.app.Dialog;  
import android.app.DialogFragment;  
import android.app.TimePickerDialog;  
import android.os.Bundle;  
import android.support.v7.app.ActionBarActivity;  
import android.app.Activity;  
import android.app.ProgressDialog;  
import android.content.Intent;  
import android.os.AsyncTask;  
import android.os.Bundle;  
import android.text.format.DateFormat;  
import android.util.Log;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.ImageButton;
```

```

import android.widget.TimePicker;

import org.apache.http.NameValuePair;
import org.apache.http.message.BasicNameValuePair;
import org.json.JSONException;
import org.json.JSONObject;
import java.util.ArrayList;
import java.util.Calendar;
import java.util.List;

public class Preferencias extends Activity{

    private ProgressDialog pDialog2 ; // para mostrar el dialogo
    private EditText txtprograma, txtelectrodomesticos, txthorainico, txthorafinal;
    private TimePicker timePicker1;
    private TimePicker timePicker2;
    private Calendar calendar;

    JSONParser jsonParser = new JSONParser();//clase para realizar las
    transferencia
    //cuadros de texto

    //      private      static      String      url_create_Device      =
"http://172.18.7.226/cas/agregarPreferencias.php";//direccion url del archivo php
    private      static      String      url_create_Device      =
"http://198.199.72.168/cas/agregarPreferencias.php";//direccion url del archivo php
    //private      static      String      url_create_Device      =
"http://172.18.47.127/cas/agregarPreferencias.php";//direccion url del archivo php

    private static final String TAG_SUCCESS = "success"; //para verificar si se realizo
bien transferencia
    private static final String TAG_MESSAGE = "message"; //para conocer el tipo de
error o los mensajes

    WebView clog;
    private String horainicio, horafinal;

    protected void onCreate(Bundle savedInstanceState) {

```

```
super.onCreate(savedInstanceState);
setContentView(R.layout.preferencias);
```

```
clog = new WebView(Preferencias.this);
```

```
txtprograma = (EditText) findViewById(R.id.programa);
txtelectrodomesticos = (EditText) findViewById(R.id.electrodomesticos);
//txthorainico = (EditText) findViewById(R.id.horainico);
//txthorafinal = (EditText) findViewById(R.id.horafinal);
```

```
timePicker1 = (TimePicker) findViewById(R.id.timePicker1);
calendar = Calendar.getInstance();
timePicker2 = (TimePicker) findViewById(R.id.timePicker2);
calendar = Calendar.getInstance();
```

```
final ImageButton btninfohora = (ImageButton) findViewById(R.id.infohora);
btninfohora.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        metodo1();
    }
});
```

```
ImageButton btninfoprogramas = (ImageButton)
findViewById(R.id.infoprogramas);
btninfoprogramas.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        metodo2();
    }
});
```



```

        ImageButton btninfoelectrodomesticos = (ImageButton)
        findViewById(R.id.infoelectrodomesticos);
        btninfoelectrodomesticos.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                metodo3();
            }
        });

```

```

Button btnCreateDevice = (Button) findViewById(R.id.registrar);
Button btnsiguiente = (Button) findViewById(R.id.registrar2);

```

```

btnsiguiente.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

```

```

        Intent lolo2 = new Intent().setClass(
            Preferencias.this,Preferencias2.class);
        horainicio = "0"+timePicker1.getCurrentHour() + ":" +
timePicker1.getCurrentMinute();
        horafinal = "0"+timePicker2.getCurrentHour() + ":" +
timePicker2.getCurrentMinute();
        lolo2.putExtra("programa", txtprograma.getText().toString());
        lolo2.putExtra("electrodomesticos",
txtelectrodomesticos.getText().toString());
        lolo2.putExtra("horainicio",horainicio);
        lolo2.putExtra("horafinal", horafinal);
        startActivity(lolo2);

```

```

    }
}); // hago clic en el botón

```

```

btnCreateDevice.setOnClickListener(new View.OnClickListener() {

```

```

    @Override
    public void onClick(View view) {
        // creating new Empleado in background thread

```

```

        new CreateNewDevice().execute();

        //cuando se hace click al boton se empieza el procesos de agregar
    }
});
}

public void showTimePickerDialog(View v) {
    DialogFragment newFragment = new TimePickerFragment();
    newFragment.show(getFragmentManager(), "timePicker");
}

public void metodo1(){

    clog.getFullLogDialog(1).show();

}

public void metodo2(){

    clog.getFullLogDialog(2).show();
}

public void metodo3(){

    clog.getFullLogDialog(3).show();
}

class CreateNewDevice extends AsyncTask<String, String, String> {

    /**
     * Antes de empezar Asynk Task se muestra un mensaje de dialgo.
     */
    @Override
    protected void onPreExecute() {
        super.onPreExecute();
    }
}

```

```

        pDialog2 = new ProgressDialog(Preferencias.this);
        pDialog2.setMessage("Registrando Preferencias...");
        pDialog2.setIndeterminate(false);
        pDialog2.setCancelable(true);
        pDialog2.show();

    }

    /**
     * Proceso de agregar
     */
    protected void doInBackground(String... args) {

        //asignacion de las variables
        String programa = txtprograma.getText().toString();
        String electrodomesticos = txtelectrodomesticos.getText().toString();
        //String horainicio = txthorainico.getText().toString();
        //String horafinal = txthorafinal.getText().toString();
        // hora = "0"+timePicker1.getCurrentHour() + ":" +
timePicker1.getCurrentMinute();
        horainicio = "0"+timePicker1.getCurrentHour() + ":" +
timePicker1.getCurrentMinute();
        horafinal = "0"+timePicker2.getCurrentHour() + ":" +
timePicker2.getCurrentMinute();

        Log.d("enviar", programa.toString() +
electrodomesticos.toString()+horainicio.toString()+horafinal.toString()); // solo es un
mensaje en log para ver si se guardan las variables.

//aqui se lista los elementos con la etiqueta respectiva para enviarla al php y las
procese correctamente
        List<NameValuePair> params = new ArrayList<NameValuePair>();
        params.add(new BasicNameValuePair("programa", programa));
        params.add(new BasicNameValuePair("electrodomesticos",
electrodomesticos));
        params.add(new BasicNameValuePair("horainicio", horainicio));
        params.add(new BasicNameValuePair("horafinal", horafinal));
    }
}

```

```

        // obteniendo JSON Object
        // aqui se envia utilizando la url antes declarada, usando el metodo POST y
enviando los parametros
        JSONObject json = jsonParser.makeHttpRequest(url_create_Device,
            "POST", params);

        // checkar en log cat la respuesta
        Log.d("Create Response", json.toString());

        // revisar si el tag resulto exitoso
        try{
            int success = json.getInt(TAG_SUCCESS);

            if (success == 1) { // si retorna 1
                Log.d("User Created!", json.toString());
                finish();

                return json.getString(TAG_MESSAGE); // mensaje que retorna para
mostrar mas adelante
            } else {
                Log.d("Registering Failure!", json.getString(TAG_MESSAGE));
                return json.getString(TAG_MESSAGE); //mensaje de error
            }
        } catch (JSONException e){
            e.printStackTrace();
        }
        return null;
    }

    /**
     * antes de completar el background se cierra el dialogo
     * **/
    protected void onPostExecute(String file_url) {
        // dismiss the dialog once done
        pDialog2.dismiss();
        Intent lolo = new Intent().setClass(
            Preferencias.this, Login.class);
        startActivity(lolo);
    }

```

```
    }  
  }  
}
```

- **Preferencias2.java:** Es un complemento de las preferencias anteriores para seguir completando datos dentro del registro. En donde entran a valer timepickers que son los atributos de tiempo usado dentro del lenguaje de **Android** como todos los anteriores se utiliza el JSONParser para hacer la conexión de los archivos **PHP** que más adelante se explicarán uno a uno. Para llamar a el correspondiente de esta clase se realiza de la siguiente manera:

```
private static String url_create_Device =  
"http://198.199.72.168/cas/agregarPreferencias2.php";
```

De igual forma tienen su layout correspondiente anidado de esta manera:

```
super.onCreate(savedInstanceState);  
setContentView(R.layout.preferencias2);
```

El código Fuente es:

```
package v1.tesis.app.sebastianrincon.com.poupewatt;
```

```
import android.app.Activity;  
import android.app.ProgressDialog;  
import android.content.Intent;  
import android.os.AsyncTask;  
import android.os.Bundle;  
import android.support.v7.app.ActionBarActivity;  
import android.util.Log;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.ImageButton;  
  
import org.apache.http.NameValuePair;  
import org.apache.http.message.BasicNameValuePair;  
import org.json.JSONException;
```

```

import org.json.JSONObject;

import java.util.ArrayList;
import java.util.List;

/**
 * Created by ASUS on 16/4/2016.
 */
public class Preferencias2 extends Activity {

    private ProgressDialog pDialog2 ; // para mostrar el dialogo
    private EditText txtvalor, txtreduccion;

    JSONParser jsonParser = new JSONParser();//clase para realizar las
    transferencia
    //cuadros de texto

    // private static String url_create_Device =
    "http://172.18.7.226/cas/agregarPreferencias2.php";//direccion url del archivo php
    private static String url_create_Device =
    "http://198.199.72.168/cas/agregarPreferencias2.php";//direccion url del archivo
    php
    // private static String url_create_Device =
    "http://172.18.47.127/cas/agregarPreferencias2.php";//direccion url del archivo php

    private static final String TAG_SUCCESS = "success"; //para verificar si se
    realizo bien transferencia
    private static final String TAG_MESSAGE = "message"; //para conocer el tipo de
    error o los mensajes

    WebView clog;
    private String programa, electrodomesticos, horainicio, horafinal;

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        setContentView(R.layout.preferencias2);

        clog = new WebView(Preferencias2.this);

```

```
programa = getIntent().getStringExtra("programa");
electrodomesticos = getIntent().getStringExtra("electrodomesticos");
horainicio = getIntent().getStringExtra("horainicio");
horafinal = getIntent().getStringExtra("horafinal");
txtreduccion = (EditText) findViewById(R.id.reduccion);
txtvalor = (EditText) findViewById(R.id.valor);
```

```
Button btnCreateDevice = (Button) findViewById(R.id.registrar2);
```

```
ImageButton btninfohora = (ImageButton) findViewById(R.id.info reduccion);
```

```
btninfohora.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        metodo4();
    }
});
```

```
ImageButton btninfo programas = (ImageButton) findViewById(R.id.info valor);
btninfo programas.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        metodo5();
    }
});
```

```
btnCreateDevice.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View view) {
        // creating new Empleado in background thread
        new CreateNewDevice().execute();
        //cuando se hace click al boton se empieza el procesos de agregar
```

```

    }
  });
}

public void metodo4(){

    clog.getFullLogDialog(4).show();

}

public void metodo5(){

    clog.getFullLogDialog(5).show();

}

class CreateNewDevice extends AsyncTask<String, String, String> {

    /**
     * Antes de empezar Asynk Task se muestra un mensaje de dialgo.
     */
    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        pDialog2 = new ProgressDialog(Preferencias2.this);
        pDialog2.setMessage("Registrando los criterios...");
        pDialog2.setIndeterminate(false);
        pDialog2.setCancelable(true);
        pDialog2.show();

    }

    /**
     * Proceso de agregar
     */
    protected String doInBackground(String... args) {

        //asignacion de las variables

```



```
String reduccion = txtreduccion.getText().toString();
String valor = txtvalor.getText().toString();
```

```
Log.d("enviar", reduccion.toString() + valor.toString()); // solo es un
mensaje en log para ver si se guardan las variables.
```

```
//aqui se lista los elementos con la etiqueta respectiva para enviarla al php y las
procese correctamenten
```

```
List<NameValuePair> params = new ArrayList<NameValuePair>();
params.add(new BasicNameValuePair("programa", programa));
params.add(new BasicNameValuePair("electrodomesticos",
electrodomesticos));
params.add(new BasicNameValuePair("horainicio",horainicio));
params.add(new BasicNameValuePair("horafinal",horafinal));
params.add(new BasicNameValuePair("reduccion", reduccion));
params.add(new BasicNameValuePair("valor", valor));
```

```
// obteniendo JSON Object
```

```
// aqui se envia utilizando la url antes declarada, usando el metodo POST y
enviando los parametros
```

```
JSONObject json = jsonParser.makeHttpRequest(url_create_Device,
"POST", params);
```

```
// checkar en log cat la respuesta
```

```
Log.d("Create Response", json.toString());
```

```
// revisar si el tag resulto exitoso
```

```
try{
```

```
int success = json.getInt(TAG_SUCCESS);
```

```
if (success == 1) { // si retorna 1
```

```
Log.d("User Created!", json.toString());
```

```
finish();
```

```
return json.getString(TAG_MESSAGE); // mensaje que retorna para
mostrar mas adelante
```

```
} else {
```

```
Log.d("Registering Failure!", json.getString(TAG_MESSAGE));
```

```

        return json.getString(TAG_MESSAGE); //mensaje de error
    }
} catch (JSONException e){
    e.printStackTrace();
}
return null;
}

/**
 * antes de completar el background se cierra el dialogo
 * **/
protected void onPostExecute(String file_url) {
    // dismiss the dialog once done
    pDialog2.dismiss();
    Intent prefer = new Intent().setClass(
        Preferencias2.this, Login.class);
    startActivity(prefer);

}

}

}

```

- **JSONParse.java:** Esta clase es muy importante para el funcionamiento de las demás, es la encargada de la conexión con **PHP**. Utilizando el protocolo de httpClient en donde se genera un HttpPost encargado de enviar los datos y un HttpResponse encargado de recibirlos directamente desde un archivo externo.

El código fuente es:

```
package v1.tesis.app.sebastianrincon.com.poupewatt;
```

```

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.UnsupportedEncodingException;
import java.util.List;

```

```

import org.apache.http.HttpEntity;
import org.apache.http.HttpResponse;
import org.apache.http.NameValuePair;
import org.apache.http.client.ClientProtocolException;
import org.apache.http.client.entity.UrlEncodedFormEntity;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.client.utils.URLEncodedUtils;
import org.apache.http.impl.client.DefaultHttpClient;
import org.json.JSONException;
import org.json.JSONObject;

import android.util.Log;

public class JSONParser {

    static InputStream is = null;
    static JSONObject jObj = null;
    static String json = "";

    // constructor
    public JSONParser() {

    }

    public JSONObject getJSONFromUrl(final String url) {

        // Making HTTP request
        try {
            // Construct the client and the HTTP request.
            DefaultHttpClient httpClient = new DefaultHttpClient();
            HttpPost httpPost = new HttpPost(url);

            // Execute the POST request and store the response locally.
            HttpResponse httpResponse = httpClient.execute(httpPost);
            // Extract data from the response.
            HttpEntity httpEntity = httpResponse.getEntity();
            // Open an inputStream with the data content.
            is = httpEntity.getContent();

```

```

} catch (UnsupportedEncodingException e) {
    e.printStackTrace();
} catch (ClientProtocolException e) {
    e.printStackTrace();
} catch (IOException e) {
    e.printStackTrace();
}

try {
    // Create a BufferedReader to parse through the inputStream.
    BufferedReader reader = new BufferedReader(new InputStreamReader(
        is, "iso-8859-1"), 8);
    // Declare a string builder to help with the parsing.
    StringBuilder sb = new StringBuilder();
    // Declare a string to store the JSON object data in string form.
    String line = null;

    // Build the string until null.
    while ((line = reader.readLine()) != null) {
        sb.append(line + "\n");
    }

    // Close the input stream.
    is.close();
    // Convert the string builder data to an actual string.
    json = sb.toString();
} catch (Exception e) {
    Log.e("Buffer Error", "Error converting result " + e.toString());
}

// Try to parse the string to a JSON object
try {
    jsonObj = new JSONObject(json);
} catch (JSONException e) {
    Log.e("JSON Parser", "Error parsing data " + e.toString());
}

// Return the JSON Object.
return jsonObj;

```

```

}

public JSONObject makeHttpRequest(String url, String method,
                                List params) {

    // Haciendo la Petición HTTP
    try {

        // check for request method
        if(method == "POST"){
            // request method is POST
            // defaultHttpClient
            DefaultHttpClient httpClient = new DefaultHttpClient();
            HttpPost httpPost = new HttpPost(url);
            httpPost.setEntity(new UrlEncodedFormEntity(params));

            HttpResponse httpResponse = httpClient.execute(httpPost);
            HttpEntity httpEntity = httpResponse.getEntity();
            is = httpEntity.getContent();

        }else if(method == "GET"){
            // request method is GET
            DefaultHttpClient httpClient = new DefaultHttpClient();
            String paramString = URLEncodedUtils.format(params, "utf-8");
            url += "?" + paramString;
           HttpGet httpGet = new HttpGet(url);

            HttpResponse httpResponse = httpClient.execute(httpGet);
            HttpEntity httpEntity = httpResponse.getEntity();
            is = httpEntity.getContent();
        }

    } catch (UnsupportedEncodingException e) {
        e.printStackTrace();
    } catch (ClientProtocolException e) {
        e.printStackTrace();
    } catch (IOException e) {
        e.printStackTrace();
    }
}

```

```

try {
    BufferedReader reader = new BufferedReader(new InputStreamReader(
        is, "iso-8859-1"), 8);
    StringBuilder sb = new StringBuilder();
    String line = null;
    while ((line = reader.readLine()) != null) {
        sb.append(line + "\n");
    }
    is.close();
    json = sb.toString();
} catch (Exception e) {
    Log.e("Buffer Error", "Error converting result " + e.toString());
}

// try parse the string to a JSON object
try {
    jsonObj = new JSONObject(json);
} catch (JSONException e) {
    Log.e("JSON Parser", "Error parsing data " + e.toString());
}

// return JSON String
return jsonObj;
}
}

```

- **Webview.java:** Esta también es una clase que se encarga de generar unos pequeños diálogos partiendo de una imagen puesta en cualquiera de los layout de la aplicación. Es por eso que se genera una función getFullDialog la cual contiene todo los parámetros para generar esa pestaña con la información que deseemos, es allí donde hacemos parte de los recursos con los que contamos para nuestra aplicación ubicados en la carpeta **RAW** dentro de la carpeta **RES**. Allí encontramos archivos .txt como lo son contenido, otrocontenido, nuevocontenido, criterios, criterios2, malo.

El código fuente es:

```

package v1.tesis.app.sebastianrincon.com.poupewatt;
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;

```

```

import android.content.SharedPreferences;
import android.content.pm.PackageManager;
import android.graphics.Color;
import android.preference.PreferenceManager;
import android.util.Log;
import android.view.ContextThemeWrapper;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;

/**
 * Created by user on 24/09/2014.
 */
public class WebView {
    private final Context context;
    private int txt;
    public WebView(Context context) {
        this.context = context;
    }

    public AlertDialog getFullLogDialog(int txt3) {
        txt = txt3;
        android.webkit.WebView wv = new android.webkit.WebView(this.context);

        wv.setBackgroundColor(Color.BLACK);
        wv.loadDataWithBaseURL(null, this.getLog(), "text/html", "UTF-8",
            null);

        AlertDialog.Builder builder = new AlertDialog.Builder(
            new ContextThemeWrapper(
                this.context, android.R.style.Theme_Dialog));
        builder
            // setTitle(context.getResources().getString(R.string.changelog_full_title))
            .setView(wv)
            .setCancelable(false)
            // OK button
            .setPositiveButton(
                // context.getResources().getString(R.string.changelog_ok_button),
                context.getResources().getString(android.R.string.ok),

```

```

        new DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog,
                int which) {

                }
        });

    return builder.create();
}

/** modes for HTML-Lists (bullet, numbered) */
private enum Listmode {
    NONE, ORDERED, UNORDERED,
};

private Listmode listMode = Listmode.NONE;
private StringBuffer sb = null;

private String getLog() {
    // read changelog.txt file
    sb = new StringBuffer();
    try {
        InputStream ins;
        switch (txt) {
            case 1: //contenido
                ins = context.getResources().openRawResource(
                    R.raw.contenido);
                break;
            case 2: //otrocontenido
                ins = context.getResources().openRawResource(
                    R.raw.otrocontenido);
                break;
            case 3: //nuevocontenido
                ins = context.getResources().openRawResource(
                    R.raw.nuevocontenido);
                break;
            case 4: //otrocontenido
                ins = context.getResources().openRawResource(
                    R.raw.criterios);
                break;
        }
    }
}

```



```

        case 5: //nuevocontenido
            ins = context.getResources().openRawResource(
                R.raw.criterios2);
            break;
        default:
            ins = context.getResources().openRawResource(
                R.raw.malo);
    }
    BufferedReader br = new BufferedReader(new InputStreamReader(ins));

    String line = null;
    boolean advanceToEOVS = false; // if true: ignore further version
    // sections
    while ((line = br.readLine()) != null) {
        line = line.trim();
        char marker = line.length() > 0 ? line.charAt(0) : 0;
        if (marker == '$') {
            // begin of a version section
            this.closeList();
            String version = line.substring(1).trim();
            // stop output?
        } else if (!advanceToEOVS) {
            switch (marker) {
                case '%':
                    // line contains version title
                    this.closeList();
                    sb.append("<div class='title'>"
                        + line.substring(1).trim() + "</div>\n");
                    break;
                case '_':
                    // line contains version title
                    this.closeList();
                    sb.append("<div class='subtitle'>"
                        + line.substring(1).trim() + "</div>\n");
                    break;
                case '!':
                    // line contains free text
                    this.closeList();
                    sb.append("<div class='freetext'>"
                        + line.substring(1).trim() + "</div>\n");
            }
        }
    }

```

```

        break;
    case '?':
        this.closeList();
        sb.append("<div class='freetexttitle'>"
            + line.substring(1).trim() + "</div>\n");
        break;
    case '#':
        // line contains numbered list item
        this.openList(Listmode.ORDERED);
        sb.append("<li>" + line.substring(1).trim() + "</li>\n");
        break;
    case '*':
        // line contains bullet list item
        this.openList(Listmode.UNORDERED);
        sb.append("<li>" + line.substring(1).trim() + "</li>\n");
        break;
    default:
        // no special character: just use line as is
        this.closeList();
        sb.append(line + "\n");
    }
}
}
}
this.closeList();
br.close();
} catch (IOException e) {
    e.printStackTrace();
}

return sb.toString();
}

```

```

private void openList(Listmode listMode) {
    if (this.listMode != listMode) {
        closeList();
        if (listMode == Listmode.ORDERED) {
            sb.append("<div class='list'><ol>\n");
        } else if (listMode == Listmode.UNORDERED) {
            sb.append("<div class='list'><ul>\n");
        }
        this.listMode = listMode;
    }
}

```

```

    }
}

private void closeList() {
    if (this.listMode == Listmode.ORDERED) {
        sb.append("</ol></div>\n");
    } else if (this.listMode == Listmode.UNORDERED) {
        sb.append("</ul></div>\n");
    }
    this.listMode = Listmode.NONE;
}
}
}

```

- **Inicio.java:** Esta clase es prácticamente la que se utiliza de Home en la aplicación puesto que se genera un menú deslizable de izquierda a derecha donde contiene la navegación principal. Esta clase acarrea y enlaza gran cantidad de clases para su funcionamiento, como se muestra a continuación, el primer lugar se trata de la carpeta **MENU** en donde en interior se encuentran los archivos:
  - *NavDrawerItem.java:* El cual se encarga de la configuración total de la Action Bar que es la barra superior donde se ocultará el menú y de donde saldrá respectivamente. Entonces se configura los estilos, el color y las imágenes correspondientes a la barra.

El código fuente es:

```
package v1.tesis.app.sebastianrincon.com.poupewatt.menu;
```

```

public class NavDrawerItem {

    private String title;
    private int icon;
    private String count = "0";
    // boolean to set visibility of the counter
    private boolean isCounterVisible = false;

    public NavDrawerItem(){}

    public NavDrawerItem(String title, int icon){
        this.title = title;
        this.icon = icon;
    }
}

```

```

    }

    public NavDrawerItem(String title, int icon, boolean isCounterVisible, String
count){
        this.title = title;
        this.icon = icon;
        this.isCounterVisible = isCounterVisible;
        this.count = count;
    }

    public String getTitle(){
        return this.title;
    }

    public int getIcon(){
        return this.icon;
    }

    public String getCount(){
        return this.count;
    }

    public boolean getCounterVisibility(){
        return this.isCounterVisible;
    }

    public void setTitle(String title){
        this.title = title;
    }

    public void setIcon(int icon){
        this.icon = icon;
    }

    public void setCount(String count){
        this.count = count;
    }

    public void setCounterVisibility(boolean isCounterVisible){
        this.isCounterVisible = isCounterVisible;
    }
}

```

```
}
```

- *NavDrawerListAdapter.java*: Por este lado, se genera es la configuración el menú deslizable. Los nombres de cada uno de los ítems, las imágenes que lo acompañan y la conexión de ellas con cada uno de los fragment.

El código fuente es:

```
package v1.tesis.app.sebastianrincon.com.poupewatt.menu;
```

```
import android.app.Activity;
```

```
import android.content.Context;
```

```
import android.view.LayoutInflater;
```

```
import android.view.View;
```

```
import android.view.ViewGroup;
```

```
import android.widget.BaseAdapter;
```

```
import android.widget.ImageView;
```

```
import android.widget.TextView;
```

```
import java.util.ArrayList;
```

```
import v1.tesis.app.sebastianrincon.com.poupewatt.R;
```

```
public class NavDrawerListAdapter extends BaseAdapter {
```

```
    private Context context;
```

```
    private ArrayList<NavDrawerItem> navDrawerItems;
```

```
    public NavDrawerListAdapter(Context context, ArrayList<NavDrawerItem>  
navDrawerItems){
```

```
        this.context = context;
```

```
        this.navDrawerItems = navDrawerItems;
```

```
    }
```

```
    @Override
```

```
    public int getCount() {
```

```
        return navDrawerItems.size();
```

```
    }
```

```
    @Override
```

```
    public Object getItem(int position) {
```

```
        return navDrawerItems.get(position);
```

```

    }

    @Override
    public long getItemId(int position) {
        return position;
    }

    @Override
    public View getView(int position, View convertView, ViewGroup parent) {
        if (convertView == null) {
            LayoutInflater mInflater = (LayoutInflater)
                context.getSystemService(Activity.LAYOUT_INFLATER_SERVICE);
            convertView = mInflater.inflate(R.layout.drawer_list_item, null);
        }

        ImageView imgIcon = (ImageView) convertView.findViewById(R.id.icon);
        TextView txtTitle = (TextView) convertView.findViewById(R.id.title);
        TextView txtCount = (TextView) convertView.findViewById(R.id.counter);

        imgIcon.setImageResource(navDrawerItems.get(position).getIcon());
        txtTitle.setText(navDrawerItems.get(position).getTitle());

        // displaying count
        // check whether it set visible or not
        if(navDrawerItems.get(position).getCounterVisibility()){
            txtCount.setText(navDrawerItems.get(position).getCount());
        }else{
            // hide the counter view
            txtCount.setVisibility(View.GONE);
        }

        return convertView;
    }
}

```

De igual manera se tiene la carpeta **SECCIONMENU** la cual contiene los fragments que responden a cada uno de los nombres puestos dentro del menú, es decir será el layout de la interfaz que verá el usuario una vez oprima alguna de las opciones que posee. Esta carpeta posee los siguientes archivos:

- *FMConsumo.java*: Este es el fragment que se conecta cuando el usuario oprime la opción de consumo en el menú deslizable. Posee la declaración de su layout como se muestra:

```
View rootView = inflater.inflate(R.layout.fmconsumo, container, false);
```

Contiene un temporizador para que solo muestre un screen splash en pantalla y luego genere la conexión con otra clase que tendrá la labor de realizar las búsquedas para el consumo. Si se desea cambiar el tiempo de duración sencillamente se modifica en la declaración de la siguiente variable:

```
private static final long SPLASH_SCREEN_DELAY = 3000;
```

El código Fuente es:

```
package v1.tesis.app.sebastianrincon.com.poupewatt.seccionmenu;
```

```
import android.app.Fragment;  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.AdapterView;  
import android.widget.AdapterView.Adapter;  
import android.widget.AdapterView.OnItemClickListener;  
import android.widget.AdapterView.OnItemSelectedListener;  
import android.widget.AdapterView.OnItemSelectedListener;
```

```
import java.util.Timer;  
import java.util.TimerTask;
```

```
import v1.tesis.app.sebastianrincon.com.poupewatt.Consumo;  
import v1.tesis.app.sebastianrincon.com.poupewatt.Descrip1;  
import v1.tesis.app.sebastianrincon.com.poupewatt.Descrip2;  
import v1.tesis.app.sebastianrincon.com.poupewatt.R;
```

```
/**
```

```
 * Created by ASUS on 3/4/2016.
```

```
 */
```

```
public class FMConsumo extends Fragment {
```

```

private static final long SPLASH_SCREEN_DELAY = 3000;

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle
savedInstanceState) {
    View rootView = inflater.inflate(R.layout.fmconsumo, container, false);

    TimerTask task = new TimerTask() {
        @Override
        public void run() {

            // Start the next activity
            Intent lo2 = new Intent(getActivity(), Consumo.class);
            startActivity(lo2);

        }
    };

    // Simulate a long loading process on application startup.
    Timer timer = new Timer();
    timer.schedule(task, SPLASH_SCREEN_DELAY);

    return rootView;
}
}

```

- *FMHistoricos.java*: Este es el fragment que se conecta cuando el usuario oprime la opción de históricos en el menú deslizable. Posee la declaración de su layout como se muestra:

```
View rootView = inflater.inflate(R.layout.fmhistoricos, container, false);
```

Y también tiene un temporizador para que solo muestre un screen splash en pantalla y luego genere la conexión con otra clase que tendrá la labor de realizar las búsquedas para el consumo. Si se desea cambiar el tiempo de duración sencillamente se modifica en la declaración de la siguiente variable:

```
private static final long SPLASH_SCREEN_DELAY = 3000;
```



El código Fuente es:

```
package v1.tesis.app.sebastianrincon.com.poupe watt.seccionmenu;
```

```
import android.app.Fragment;
import android.content.Intent;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import java.util.Timer;
import java.util.TimerTask;
import lecho.lib.hellocharts.view.LineChartView;
import v1.tesis.app.sebastianrincon.com.poupe watt.Grafica;
import v1.tesis.app.sebastianrincon.com.poupe watt.R;
```

```
public class FMHistoricos extends Fragment {
```

```
    private static final long SPLASH_SCREEN_DELAY = 3000;
    private LineChartView chart;
```

```
    @Override
```

```
    public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle
savedInstanceState) {
```

```
        TimerTask task = new TimerTask() {
```

```
            @Override
```

```
            public void run() {
```

```
                // Start the next activity
```

```
                Intent lo2 = new Intent(getActivity(), Grafica.class);
```

```
                startActivity(lo2);
```

```
            }
```

```
        };
```

```
        // Simulate a long loading process on application startup.
```

```
        Timer timer = new Timer();
```

```
        timer.schedule(task, SPLASH_SCREEN_DELAY);
```

```

View rootView = inflater.inflate(R.layout.fmhistoricos, container, false);

return rootView;
}
}

```

- *FMHome.java*: Este es el fragment que se conecta cuando el usuario oprime la opción de home en el menú deslizable. Posee la declaración de su layout como se muestra:

```
View rootView = inflater.inflate(R.layout.fmhome, container, false);
```

Dentro del layout fmhome se encuentran 3 imagebutton que re direccionan a 3 diferentes clases dependiendo de donde se busque llegar. Es por esto que acá solo se hace el envío a la clase correspondiente de la siguiente forma:

```

ImageButton btnimagen2 = (ImageButton) rootView.findViewById(R.id.imagen2);
    btnimagen2.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(View v) {
            Intent lo2 = new Intent(getActivity(), Descrip2.class);
            startActivity(lo2);
        }
    });

```

El código Fuente es:

```
package v1.tesis.app.sebastianrincon.com.poupewatt.seccionmenu;
```

```

import android.annotation.TargetApi;
import android.app.Fragment;
import android.content.Intent;
import android.os.Build;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageButton;

```

```

import v1.tesis.app.sebastianrincon.com.poupewatt.Descrip1;
import v1.tesis.app.sebastianrincon.com.poupewatt.Descrip2;
import v1.tesis.app.sebastianrincon.com.poupewatt.Descrip3;
import v1.tesis.app.sebastianrincon.com.poupewatt.R;

public class FMHome extends Fragment {
    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle
savedInstanceState) {
        View rootView = inflater.inflate(R.layout.fmhome, container, false);

        ImageButton          btnimagen1          =          (ImageButton)
rootView.findViewById(R.id.imagen1);
        btnimagen1.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View v) {
                Intent lo = new Intent(getActivity(), Descrip1.class);
                startActivity(lo);
            }

        });

        ImageButton          btnimagen2          =          (ImageButton)
rootView.findViewById(R.id.imagen2);
        btnimagen2.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View v) {
                Intent lo2 = new Intent(getActivity(), Descrip2.class);
                startActivity(lo2);
            }

        });

        ImageButton          btnimagen3          =          (ImageButton)
rootView.findViewById(R.id.imagen3);
        btnimagen3.setOnClickListener(new View.OnClickListener() {

```

```

@Override
public void onClick(View v) {
    Intent lo3 = new Intent(getActivity(), Descrip3.class);
    startActivity(lo3);
}

});

return rootView;

}
}

```

- *FMJuego.java*: Este es el fragment que se conecta cuando el usuario oprime la opción de juego en el menú deslizable. Posee la declaración de su layout como se muestra:

```
View rootView = inflater.inflate(R.layout.fmjuego, container, false);
```

Dentro de ese layout se encuentran las instrucciones del juego entonces hasta que el usuario no oprima el botón de siguiente permanecerá allí.

El código fuente es:

```
package v1.tesis.app.sebastianrincon.com.poupewatt.seccionmenu;
```

```
import android.app.Fragment;
import android.content.Intent;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import v1.tesis.app.sebastianrincon.com.poupewatt.Juego;
import v1.tesis.app.sebastianrincon.com.poupewatt.R;
```

```
/**
```

```
* Created by ASUS on 3/4/2016.
```

```
*/
```

```
public class FMJuego extends Fragment {
```

```

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle
savedInstanceState) {
    View rootView = inflater.inflate(R.layout.fmjuego, container, false);

    Button btnSig = (Button) rootView.findViewById(R.id.sig);

    btnSig.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(View view) {

            Intent lo2 = new Intent(getActivity(), Juego.class);
            startActivity(lo2);
        }
    });

    return rootView;
}
}

```

- **Descrip1.java/Descrip2.java/Descrip3.java:** Estas clases sencillamente contienen información ya sea de objetivos de tutorial o de algunas noticias en general, por eso sencillamente tienen la declaración del layout correspondiente y no más.

El código fuente es:

```
package v1.tesis.app.sebastianrincon.com.poupewatt;
```

```
import android.os.Bundle;
import android.support.v7.app.ActionBar;
import android.support.v7.app.ActionBarActivity;
import android.support.v7.app.AppCompatActivity;
```

```
/**
```

```
* Created by ASUS on 6/4/2016.
```

```

*/
public class Descrip1 extends ActionBarActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        // note that use read_comments.xml instead of our single_post.xml
        setContentView(R.layout.descrip1);
        ActionBar actionBar = getSupportActionBar();
        actionBar.setDisplayHomeAsUpEnabled(true);
    }
}

```

- **Consumo.java:** Esta clase es la encargada de mostrar el consumo de Kwh que el usuario selecciona. En donde entran a valer los datepickers que son los atributos de fechas usadas dentro del lenguaje de **Android** como todos los anteriores se utiliza el JSONParser para hacer la conexión de los archivos **PHP** que más adelante se explicarán uno a uno. Para llamar a el correspondiente de esta clase se realiza de la siguiente manera:

```

private static String url_create_Device =
"http://198.199.72.168//cas/generarconsumo.php";

```

De igual forma tienen su layout correspondiente anidado de esta manera:

```

super.onCreate(savedInstanceState);
setContentView(R.layout.consumo);

```

Posee spinner para la selección de búsqueda y demás cosas. Se maneja la estructura de tres fases como se mencionó en el login utilizando el onPreExecute, doInBackground y el onPostExecute.

El código fuente es:

```

package v1.tesis.app.sebastianrincon.com.poupewatt;

```

```

import android.app.Activity;
import android.app.DatePickerDialog;
import android.app.Dialog;
import android.app.ProgressDialog;
import android.content.Intent;
import android.os.AsyncTask;
import android.os.Bundle;

```

```

import android.support.v4.app.DialogFragment;
import android.support.v7.app.ActionBar;
import android.support.v7.app.ActionBarActivity;
import android.support.v7.app.AppCompatActivity;
import android.util.Log;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.TimePicker;
import android.widget.Toast;

import org.apache.http.NameValuePair;
import org.apache.http.message.BasicNameValuePair;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.util.ArrayList;
import java.util.Calendar;
import java.util.List;

/**
 * Created by ASUS on 28/4/2016.
 */
public class Consumo extends ActionBarActivity {

    Spinner lista,lista2;
    String[] datos = {"None","Potencia (Kwh)","Precio ($)"};
    String[] datos2 = {"None","Sensor 1","Sensor 2","Sensor 3","Sensor 4","Sensor
5"};
    private DatePicker datePicker1, datePicker2;
    private Calendar calendar;
    private TextView dateView;
    private ProgressDialog pDialog2 ;

```

```

private String dia;
private String dia2;
private String A, B;
private String nuevodate, nuevodate2, suma, signo;
private int year, month, day, year2, month2, day2;
TextView Texto;

```

```

JSONParser jsonParser = new JSONParser();//clase para realizar las
transferencia

```

```

//cuadros de texto

```

```

//      private      static      String      url_create_Device      =
"http://172.18.7.127/cas/generarconsumo.php";//direccion url del archivo php
//      private      static      String      url_create_Device      =
"http://192.168.0.103/cas/generarconsumo.php";//direccion url del archivo php
private      static      String      url_create_Device      =
"http://198.199.72.168//cas/generarconsumo.php";//direccion url del archivo php

```

```

private static final String TAG_SUCCESS = "success"; //para verificar si se realizo
bien transferencia

```

```

private static final String TAG_MESSAGE = "message"; //para conocer el tipo de
error o los mensajes

```

```

@Override

```

```

protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.consumo);

```

```

ActionBar actionBar = getSupportActionBar();
actionBar.setDisplayHomeAsUpEnabled(true);

```

```

Texto = (TextView) findViewById(R.id.textView3);
calendar = Calendar.getInstance();
year = calendar.get(Calendar.YEAR);

```



```

month = calendar.get(Calendar.MONTH);
day = calendar.get(Calendar.DAY_OF_MONTH);
showDate(year, month+1, day);

year2 = calendar.get(Calendar.YEAR);
month2 = calendar.get(Calendar.MONTH);
day2 = calendar.get(Calendar.DAY_OF_MONTH);
showDate2(year2, month2+1, day2);

lista = (Spinner)findViewById(R.id.lista1);
ArrayAdapter<String> adaptador = new ArrayAdapter<String> (this,
android.R.layout.simple_spinner_dropdown_item, datos);
lista.setAdapter(adaptador);

lista2 = (Spinner)findViewById(R.id.lista2);
ArrayAdapter<String> adaptador2 = new
ArrayAdapter<String>(this,android.R.layout.simple_spinner_dropdown_item,datos2
);
lista2.setAdapter(adaptador2);

Button btnCreateDevice = (Button) findViewById(R.id.button1);
Texto = (TextView) findViewById(R.id.prueba);

lista.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
    @Override
    public void onItemSelected(AdapterView<?> parent, View view, int position,
long id) {

        switch (position) {

            case 1:

                A="Kilo";
                //dia = datePicker1.getYear() + "-" + datePicker1.getMonth()+"-
"+datePicker1.getDayOfMonth();
                //Toast to = Toast.makeText(getApplicationContext(), dia,
Toast.LENGTH_LONG);
                //to.show();

```

```

        break;

    case 2:

        A="Price";
        //dia2 = datePicker2.getYear() + "-" + datePicker2.getMonth()+"-
"+datePicker2.getDayOfMonth();
        //Toast to1 = Toast.makeText(getApplicationContext(), dia2,
Toast.LENGTH_LONG);
        //to1.show();

        break;

    }

}

@Override
public void onNothingSelected(AdapterView<?> parent) {

}

});

    lista2.setOnItemSelectedListener(new
AdapterView.OnItemSelectedListener(){

        @Override
        public void onItemSelected(AdapterView<?> parent, View view, int position,
long id) {

            switch (position) {

                case 1:

                    B="Sensor1";

```

```
        break;

    case 2:

        B="Sensor2";

        break;

    case 3:

        B="Sensor3";

        break;

    case 4:

        B="Sensor4";

        break;

    case 5:

        B="Sensor5";

        break;

    }

}

@Override
public void onNothingSelected(AdapterView<?> parent) {

}

});
```

```

btnCreateDevice.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View view) {
        // creating new Empleado in background thread
        new CreateNewDevice().execute();

        //cuando se hace click al boton se empieza el procesos de agregar
    }
});

}

@SuppressWarnings("deprecation")
public void setDate(View view) {
    showDialog(999);
    Toast.makeText(getApplicationContext(), "Primer Fecha",
Toast.LENGTH_SHORT)
        .show();
}

@SuppressWarnings("deprecation")
public void setDate2(View view) {
    showDialog(998);
    Toast.makeText(getApplicationContext(), "Segunda Fecha",
Toast.LENGTH_SHORT)
        .show();
}

@Override
protected Dialog onCreateDialog(int id) {
    // TODO Auto-generated method stub
    if (id == 999) {
        return new DatePickerDialog(this, myDateListener, year, month, day);
    }else{

```

```

        return new DatePickerDialog(this, myDateListener2, year2, month2, day2);
    }
}

private DatePickerDialog.OnDateSetListener myDateListener = new
DatePickerDialog.OnDateSetListener() {
    @Override
    public void onDateSet(DatePicker arg0, int arg1, int arg2, int arg3) {
        // TODO Auto-generated method stub
        // arg1 = year
        // arg2 = month
        // arg3 = day
        showDate(arg1, arg2+1, arg3);
    }
};

private void showDate(int year, int month, int day) {
    nuevodate = year + "-" + month + "-" + day;
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.

    return true;
}

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

private DatePickerDialog.OnDateSetListener myDateListener2 = new
DatePickerDialog.OnDateSetListener() {
    @Override
    public void onDateSet(DatePicker arg4, int arg5, int arg6, int arg7) {
        // TODO Auto-generated method stub

        showDate2(arg5, arg6+1, arg7);
    }
};

```

```
private void showDate2(int year2, int month2, int day2) {  
    nuevodate2 = year2 + "-" + month2 + "-" + day2;  
}
```

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

```
class CreateNewDevice extends AsyncTask<String, String, String> {
```

```
    /**  
    * Antes de empezar Asynk Task se muestra un mensaje de dialgo.  
    */
```

```
    @Override
```

```
protected void onPreExecute() {  
    super.onPreExecute();  
    pDialog2 = new ProgressDialog(Consumo.this);  
    pDialog2.setMessage("Consultando...");  
    pDialog2.setIndeterminate(false);  
    pDialog2.setCancelable(true);  
    pDialog2.show();
```

```
}
```

```
/**  
* Proceso de agregar  
*/
```

```
protected String doInBackground(String... args) {
```

```
    //asignacion de las variables  
    String tipoconsumo = A;  
    String tiposensor = B;
```

```
Log.d("enviar", tipoconsumo.toString() + tiposensor.toString()+
nuevodate.toString()+ nuevodate2.toString()); // solo es un mensaje en log para ver
si se guardan las variables.
```

```
//aqui se lista los elementos con la etiqueta respectiva para enviarla al php y las
procese correctamenten
```

```
List<NameValuePair> params = new ArrayList<NameValuePair>();
params.add(new BasicNameValuePair("tipoconsumo", tipoconsumo));
params.add(new BasicNameValuePair("tiposensor", tiposensor));
params.add(new BasicNameValuePair("dia", nuevodate));
params.add(new BasicNameValuePair("dia2", nuevodate2));
```

```
// obteniendo JSON Object
```

```
// aqui se envia utilizando la url antes declarada, usando el metodo POST y
enviando los parametros
```

```
JSONObject json = jsonParser.makeHttpRequest(url_create_Device,
"POST", params);
```

```
// checkar en log cat la respuesta
```

```
Log.d("Create Response Consumo", json.toString());
```

```
// revisar si el tag resulto exitoso
```

```
try {
```

```
    suma = json.getString("success");
    signo = json.getString("message");
```

```
Log.d("Funciono", json.toString());
```

```
} catch (JSONException e) {
    e.printStackTrace();
```

```
}
```

```
return null;
```

```
}
```

```
/**
```

```
* antes de completar el background se cierra el dialogo
```

```

    **/
protected void onPostExecute(String file_url) {
    // dismiss the dialog

    if (suma == "null"){

        Texto.setText("No hay datos en las fechas seleccionadas");

    } else {

        Texto.setText(signo + " " + suma);

    }

    pDialog2.dismiss();

}
}
}

```

- **Juego.java:** Esta clase es la encargada de estructurar todo el juego para que el usuario pueda interactuar dentro de la interfaz correspondiente. En donde entran a valer los datepickers que son los atributos de fechas usadas dentro del lenguaje de **Android** como todos los anteriores se utiliza el JSONParser para hacer la conexión de los archivos **PHP** que más adelante se explicarán uno a uno. Para llamar a el correspondiente de esta clase se realiza de la siguiente manera:

```
private static String url_create_Device = "http://198.199.72.168//cas/juego.php";
```

De igual forma tienen su layout correspondiente anidado de esta manera:

```
super.onCreate(savedInstanceState);
setContentView(R.layout.juego);
```

Posee spinner para la selección de los electrodomésticos y algunas otras cosas. Se maneja la estructura de tres fases como se mencionó en el login utilizando el onPreExecute, doInBackground y el onPostExecute.

El código fuente es:



```
package v1.tesis.app.sebastianrincon.com.poupewatt;
```

```
import android.app.Activity;
import android.app.DatePickerDialog;
import android.app.Dialog;
import android.app.ProgressDialog;
import android.os.AsyncTask;
import android.os.Bundle;
import android.support.v7.app.ActionBar;
import android.support.v7.app.ActionBarActivity;
import android.support.v7.app.AppCompatActivity;
import android.util.Log;
import android.util.SparseArray;
import android.view.Menu;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.EditText;
import android.widget.ExpandableListView;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;
```

```
import org.apache.http.NameValuePair;
import org.apache.http.message.BasicNameValuePair;
import org.json.JSONException;
import org.json.JSONObject;
```

```
import java.util.ArrayList;
import java.util.Calendar;
import java.util.List;
```

```
/**
```

```
 * Created by ASUS on 21/5/2016.
```

```
 */
```

```
public class Juego extends ActionBarActivity {
```

```
    Spinner lista,lista2,lista3,lista4,lista5;
```

```
    String[] datos = {"None","Televisor","Nevera","Lavadora","Lámpara","Plancha"};
```

```

String[] datos2 = {"None","Televisor","Nevera","Lavadora","Lámpara","Plancha"};
String[] datos3 = {"None","Televisor","Nevera","Lavadora","Lámpara","Plancha"};
String[] datos4 = {"None","Televisor","Nevera","Lavadora","Lámpara","Plancha"};
String[] datos5 = {"None","Televisor","Nevera","Lavadora","Lámpara","Plancha"};
private EditText txthora1a, txthora1b,txthora1c,txthora1d,txthora1e,txthora1f;
private EditText txthora2a, txthora2b,txthora2c,txthora2d,txthora2e,txthora2f;
private EditText txthora3a, txthora3b,txthora3c,txthora3d,txthora3e,txthora3f;
private EditText txthora4a, txthora4b,txthora4c,txthora4d,txthora4e,txthora4f;
private EditText txthora5a, txthora5b,txthora5c,txthora5d,txthora5e,txthora5f;
private EditText txtporcentaje;
private String respuesta;
private String precio;
private int confirmacion;
private String nuevodate, nuevodate2;
private Calendar calendar;
private String ahorro;
private ProgressDialog pDialog2 ;
private int year, month, day, year2, month2, day2;
private String A, B, C, D, E;
TextView Texto2;

```

```

JSONParser jsonParser = new JSONParser();//clase para realizar las
transferencia

```

```

//cuadros de texto

```

```

//      private      static      String      url_create_Device      =
"http://172.18.7.226/cas/generarconsumo.php";//direccion url del archivo php
private      static      String      url_create_Device      =
"http://198.199.72.168/cas/juego.php";//direccion url del archivo php
//      private      static      String      url_create_Device      =
"http://172.18.7.127/cas/juego.php";//direccion url del archivo php
private static final String TAG_SUCCESS = "success"; //para verificar si se realizo
bien transferencia
private static final String TAG_MESSAGE = "message"; //para conocer el tipo de
error o los mensajes

```

```

protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.juego);
}

```

```
ActionBar actionBar = getSupportActionBar();
actionBar.setDisplayHomeAsUpEnabled(true);
```

```
calendar = Calendar.getInstance();
year = calendar.get(Calendar.YEAR);
month = calendar.get(Calendar.MONTH);
day = calendar.get(Calendar.DAY_OF_MONTH);
showDate(year, month+1, day);
```

```
year2 = calendar.get(Calendar.YEAR);
month2 = calendar.get(Calendar.MONTH);
day2 = calendar.get(Calendar.DAY_OF_MONTH);
showDate2(year2, month2 + 1, day2);
```

```
txtporcentaje = (EditText) findViewById(R.id.porcentaje);
```

```
txthora1a = (EditText) findViewById(R.id.hora1a);
txthora1b = (EditText) findViewById(R.id.hora1b);
txthora1c = (EditText) findViewById(R.id.hora1c);
txthora1d = (EditText) findViewById(R.id.hora1d);
txthora1e = (EditText) findViewById(R.id.hora1e);
txthora1f = (EditText) findViewById(R.id.hora1f);
```

```
txthora2a = (EditText) findViewById(R.id.hora2a);
txthora2b = (EditText) findViewById(R.id.hora2b);
txthora2c = (EditText) findViewById(R.id.hora2c);
txthora2d = (EditText) findViewById(R.id.hora2d);
txthora2e = (EditText) findViewById(R.id.hora2e);
txthora2f = (EditText) findViewById(R.id.hora2f);
```

```
txthora3a = (EditText) findViewById(R.id.hora3a);
txthora3b = (EditText) findViewById(R.id.hora3b);
txthora3c = (EditText) findViewById(R.id.hora3c);
txthora3d = (EditText) findViewById(R.id.hora3d);
txthora3e = (EditText) findViewById(R.id.hora3e);
txthora3f = (EditText) findViewById(R.id.hora3f);
```

```
txthora4a = (EditText) findViewById(R.id.hora4a);
txthora4b = (EditText) findViewById(R.id.hora4b);
txthora4c = (EditText) findViewById(R.id.hora4c);
txthora4d = (EditText) findViewById(R.id.hora4d);
```

```

txthora4e = (EditText) findViewById(R.id.hora4e);
txthora4f = (EditText) findViewById(R.id.hora4f);

txthora5a = (EditText) findViewById(R.id.hora5a);
txthora5b = (EditText) findViewById(R.id.hora5b);
txthora5c = (EditText) findViewById(R.id.hora5c);
txthora5d = (EditText) findViewById(R.id.hora5d);
txthora5e = (EditText) findViewById(R.id.hora5e);
txthora5f = (EditText) findViewById(R.id.hora5f);

lista = (Spinner)findViewById(R.id.lista1);
ArrayAdapter<String> adaptador = new ArrayAdapter<String> (this,
android.R.layout.simple_spinner_dropdown_item, datos);
lista.setAdapter(adaptador);

lista2 = (Spinner)findViewById(R.id.lista2);
ArrayAdapter<String> adaptador2 = new
ArrayAdapter<String>(this,android.R.layout.simple_spinner_dropdown_item,datos2
);
lista2.setAdapter(adaptador2);

lista3 = (Spinner)findViewById(R.id.lista3);
ArrayAdapter<String> adaptador3 = new
ArrayAdapter<String>(this,android.R.layout.simple_spinner_dropdown_item,datos3
);
lista3.setAdapter(adaptador3);

lista4 = (Spinner)findViewById(R.id.lista4);
ArrayAdapter<String> adaptador4 = new
ArrayAdapter<String>(this,android.R.layout.simple_spinner_dropdown_item,datos4
);
lista4.setAdapter(adaptador4);

lista5 = (Spinner)findViewById(R.id.lista5);
ArrayAdapter<String> adaptador5 = new
ArrayAdapter<String>(this,android.R.layout.simple_spinner_dropdown_item,datos5
);
lista5.setAdapter(adaptador5);

Button btnCreateDevice = (Button) findViewById(R.id.button2);
Texto2 = (TextView) findViewById(R.id.result);

```

```

lista.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
    @Override
    public void onItemSelected(AdapterView<?> parent, View view, int position,
long id) {

        switch (position) {

            case 1:

                //dia = datePicker1.getYear() + "-" + datePicker1.getMonth()+"-
"+datePicker1.getDayOfMonth();
                /*Toast to = Toast.makeText(getApplicationContext(), "Televisor",
Toast.LENGTH_LONG);
                to.show();*/
                A="Televisor";

                break;

            case 2:

                //dia2 = datePicker2.getYear() + "-" + datePicker2.getMonth()+"-
"+datePicker2.getDayOfMonth();
                /*Toast to1 = Toast.makeText(getApplicationContext(), "Nevera",
Toast.LENGTH_LONG);
                to1.show();*/
                A="Nevera";

                break;

            case 3:

                /*Toast to2 = Toast.makeText(getApplicationContext(), "Lavadora",
Toast.LENGTH_LONG);
                to2.show();*/
                A="Lavadora";

                break;

```

case 4:

```
        /*Toast to3 = Toast.makeText(getApplicationContext(), "Lámpara",
Toast.LENGTH_LONG);
        to3.show();*/
        A="Lampara";

        break;
```

case 5:

```
        /*Toast to4 = Toast.makeText(getApplicationContext(), "Plancha",
Toast.LENGTH_LONG);
        to4.show();*/
        A="Plancha";

        break;
```

```
    }
```

```
 }
```

```
 @Override
```

```
 public void onNothingSelected(AdapterView<?> parent) {
```

```
 }
```

```
 });
```

```
 lista2.setOnItemSelectedListener(new
 AdapterView.OnItemSelectedListener){
```

```
 @Override
```

```
 public void onItemSelected(AdapterView<?> parent, View view, int position,
 long id) {
```

```
     switch (position) {
```

```
case 1:
    B="Televisor";
    break;

case 2:

    B="Nevera";
    break;

case 3:
    B="Lavadora";
    break;

case 4:
    B="Lampara";
    break;

case 5:
    B="Plancha";
    break;
}
}

@Override
public void onNothingSelected(AdapterView<?> parent) {
}
});
```

```
lista3.setOnItemSelectedListener(new  
AdapterView.OnItemSelectedListener){
```

```
    @Override  
    public void onItemSelected(AdapterView<?> parent, View view, int position,  
long id) {
```

```
        switch (position) {
```

```
            case 1:
```

```
                C="Televisor";
```

```
                break;
```

```
            case 2:
```

```
                C="Nevera";
```

```
                break;
```

```
            case 3:
```

```
                C="Lavadora";
```

```
                break;
```

```
            case 4:
```

```
                C="Lampara";
```

```
                break;
```

```
            case 5:
```



```
        C="Plancha";  
        break;  
    }  
}  
  
@Override  
public void onNothingSelected(AdapterView<?> parent) {  
  
}  
});
```

```
lista4.setOnItemSelectedListener(new  
AdapterView.OnItemSelectedListener){
```

```
    @Override  
    public void onItemSelected(AdapterView<?> parent, View view, int position,  
long id) {
```

```
        switch (position) {
```

```
            case 1:
```

```
                D="Televisor";
```

```
                break;
```

```
            case 2:
```

```
                D="Nevera";
```

```
                break;
```

```
            case 3:
```

```

        D="Lavadora";

        break;

    case 4:

        D="Lampara";

        break;

    case 5:

        D="Plancha";

        break;

    }

}

@Override
public void onNothingSelected(AdapterView<?> parent) {

}

});

lista5.setOnItemSelectedListener(new
AdapterView.OnItemSelectedListener(){

    @Override
    public void onItemSelected(AdapterView<?> parent, View view, int position,
long id) {

        switch (position) {

            case 1:

```

```
        E="Televisor";  
        break;  
  
    case 2:  
  
        E="Nevera";  
        break;  
  
    case 3:  
  
        E="Lavadora";  
        break;  
  
    case 4:  
  
        E="Lampara";  
        break;  
  
    case 5:  
  
        E="Plancha";  
        break;  
    }  
}  
  
@Override  
public void onNothingSelected(AdapterView<?> parent) {  
  
}  
});
```

```

btnCreateDevice.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View view) {
        // creating new Empleado in background thread
        new CreateNewDevice().execute();

        //cuando se hace click al boton se empieza el procesos de agregar
    }
});

}

@SuppressWarnings("deprecation")
public void setDate(View view) {
    showDialog(999);
    Toast.makeText(getApplicationContext(), "Primer Fecha",
Toast.LENGTH_SHORT)
        .show();
}

@SuppressWarnings("deprecation")
public void setDate2(View view) {
    showDialog(998);
    Toast.makeText(getApplicationContext(), "Segunda Fecha",
Toast.LENGTH_SHORT)
        .show();
}

@Override
protected Dialog onCreateDialog(int id) {
    // TODO Auto-generated method stub
    if (id == 999) {
        return new DatePickerDialog(this, myDateListener, year, month, day);
    }else{

        return new DatePickerDialog(this, myDateListener2, year2, month2, day2);
    }
}

```

```

    }
}

private DatePickerDialog.OnDateSetListener myDateListener = new
DatePickerDialog.OnDateSetListener() {
    @Override
    public void onDateSet(DatePicker arg0, int arg1, int arg2, int arg3) {
        // TODO Auto-generated method stub
        // arg1 = year
        // arg2 = month
        // arg3 = day
        showDate(arg1, arg2+1, arg3);
    }
};

private void showDate(int year, int month, int day) {
    nuevodate = year + "-" + month + "-" + day;
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.

    return true;
}

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

private DatePickerDialog.OnDateSetListener myDateListener2 = new
DatePickerDialog.OnDateSetListener() {
    @Override
    public void onDateSet(DatePicker arg4, int arg5, int arg6, int arg7) {
        // TODO Auto-generated method stub
        // arg1 = year
        // arg2 = month
        // arg3 = day
        showDate2(arg5, arg6+1, arg7);
    }
};

```

```
private void showDate2(int year2, int month2, int day2) {
    nuevodate2 = year2 + "-" + month2 + "-" + day2;
}
```

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

```
class CreateNewDevice extends AsyncTask<String, String, String> {
```

```
    /**
     * Antes de empezar Asynk Task se muestra un mensaje de dialgo.
     */
```

```
    @Override
```

```
    protected void onPreExecute() {
        super.onPreExecute();
        pDialog2 = new ProgressDialog(Juego.this);
        pDialog2.setMessage("Calculando...");
        pDialog2.setIndeterminate(false);
        pDialog2.setCancelable(true);
        pDialog2.show();
    }
```

```
    /**
     * Proceso de agregar
     */
    protected String doInBackground(String... args) {
```

```
        //asignacion de las variables
```

```
        String primeraelec = A;
        String segundaelec = B;
        String terceraelec = C;
        String cuartaelec = D;
        String quintaelec = E;
```

```
        String porcentaje = txtporcentaje.getText().toString();
```

```
String hora1a = txthora1a.getText().toString();
String hora1b = txthora1b.getText().toString();
String hora1c = txthora1c.getText().toString();
String hora1d = txthora1d.getText().toString();
String hora1e = txthora1e.getText().toString();
String hora1f = txthora1f.getText().toString();
```

```
String hora2a = txthora2a.getText().toString();
String hora2b = txthora2b.getText().toString();
String hora2c = txthora2c.getText().toString();
String hora2d = txthora2d.getText().toString();
String hora2e = txthora2e.getText().toString();
String hora2f = txthora2f.getText().toString();
```

```
String hora3a = txthora3a.getText().toString();
String hora3b = txthora3b.getText().toString();
String hora3c = txthora3c.getText().toString();
String hora3d = txthora3d.getText().toString();
String hora3e = txthora3e.getText().toString();
String hora3f = txthora3f.getText().toString();
```

```
String hora4a = txthora4a.getText().toString();
String hora4b = txthora4b.getText().toString();
String hora4c = txthora4c.getText().toString();
String hora4d = txthora4d.getText().toString();
String hora4e = txthora4e.getText().toString();
String hora4f = txthora4f.getText().toString();
```

```
String hora5a = txthora5a.getText().toString();
String hora5b = txthora5b.getText().toString();
String hora5c = txthora5c.getText().toString();
String hora5d = txthora5d.getText().toString();
String hora5e = txthora5e.getText().toString();
String hora5f = txthora5f.getText().toString();
```

```
// dia = datePicker1.getYear() + "-" + (datePicker1.getMonth()+1) + "-" +
datePicker1.getDayOfMonth();
// dia2 = datePicker2.getYear() + "-" + (datePicker2.getMonth()+1) + "-" +
datePicker2.getDayOfMonth();
```

Log.d("enviar", A.toString() + hora1a.toString() + hora1b.toString() + hora1c.toString() + hora1d.toString() + hora1e.toString() + hora1f.toString()); // solo es un mensaje en log para ver si se guardan las variables.

//aqui se lista los elementos con la etiqueta respectiva para enviarla al php y las procese correctamente

```
List<NameValuePair> params = new ArrayList<NameValuePair>();
```

```
params.add(new BasicNameValuePair("dia", nuevodate));  
params.add(new BasicNameValuePair("dia2", nuevodate2));
```

```
params.add(new BasicNameValuePair("porcentaje", porcentaje));
```

```
params.add(new BasicNameValuePair("primeraelec", primeraelec));  
params.add(new BasicNameValuePair("segundaelec", segundaelec));  
params.add(new BasicNameValuePair("terceraelec", terceraelec));  
params.add(new BasicNameValuePair("cuartaelec", cuartaelec));  
params.add(new BasicNameValuePair("quintaelec", quintaelec));
```

```
params.add(new BasicNameValuePair("hora1a", hora1a));  
params.add(new BasicNameValuePair("hora1b", hora1b));  
params.add(new BasicNameValuePair("hora1c", hora1c));  
params.add(new BasicNameValuePair("hora1d", hora1d));  
params.add(new BasicNameValuePair("hora1e", hora1e));  
params.add(new BasicNameValuePair("hora1f", hora1f));
```

```
params.add(new BasicNameValuePair("hora2a", hora2a));  
params.add(new BasicNameValuePair("hora2b", hora2b));  
params.add(new BasicNameValuePair("hora2c", hora2c));  
params.add(new BasicNameValuePair("hora2d", hora2d));  
params.add(new BasicNameValuePair("hora2e", hora2e));  
params.add(new BasicNameValuePair("hora2f", hora2f));
```

```
params.add(new BasicNameValuePair("hora3a", hora3a));  
params.add(new BasicNameValuePair("hora3b", hora3b));  
params.add(new BasicNameValuePair("hora3c", hora3c));  
params.add(new BasicNameValuePair("hora3d", hora3d));  
params.add(new BasicNameValuePair("hora3e", hora3e));  
params.add(new BasicNameValuePair("hora3f", hora3f));
```



```

params.add(new BasicNameValuePair("hora4a", hora4a));
params.add(new BasicNameValuePair("hora4b", hora4b));
params.add(new BasicNameValuePair("hora4c", hora4c));
params.add(new BasicNameValuePair("hora4d", hora4d));
params.add(new BasicNameValuePair("hora4e", hora4e));
params.add(new BasicNameValuePair("hora4f", hora4f));

params.add(new BasicNameValuePair("hora5a", hora5a));
params.add(new BasicNameValuePair("hora5b", hora5b));
params.add(new BasicNameValuePair("hora5c", hora5c));
params.add(new BasicNameValuePair("hora5d", hora5d));
params.add(new BasicNameValuePair("hora5e", hora5e));
params.add(new BasicNameValuePair("hora5f", hora5f));

// obteniendo JSON Object
// aqui se envia utilizando la url antes declarada, usando el metodo POST y
enviando los parametros
JSONObject json = jsonParser.makeHttpRequest(url_create_Device,
        "POST", params);

// checkar en log cat la respuesta
Log.d("Create Response Consumo", json.toString());

// revisar si el tag resulto exitoso
try {

    confirmacion = json.getInt("con");
    ahorro = json.getString("success");
    respuesta = json.getString("message");
    precio = json.getString("check");

    Log.d("Funciono", json.toString());

} catch (JSONException e) {
    e.printStackTrace();
}
return null;
}

```

```

/**
 * antes de completar el background se cierra el dialogo
 **/
protected void onPostExecute(String file_url) {
    // dismiss the dialog

    if (confirmacion < 15){

        Texto2.setText( respuesta + " " + ahorro + " " + "Kwh" + " " + "y por ello"+
" " + "$" + " " + precio + " " + "pesos.");

    } else if (confirmacion > 15) {

        Texto2.setText(respuesta + " " + "consumiste" + " " + ahorro + " " + "Kwh"
+ " " + "más de lo esperado" + " " + "¡Vuelve a intentarlo!");
    } else {

        Texto2.setText("Muy mal perrito"+confirmacion);

    }

    pDialog2.dismiss();

}
}}

```

- **Graficas.java:** Esta clase es la encargada de graficar los datos que el usuario seleccione. En donde entran a valer los datepickers que son los atributos de fechas usadas dentro del lenguaje de **Android** como todos los anteriores se utiliza el JSONParser para hacer la conexión de los archivos **PHP** que más adelante se explicarán uno a uno. Para llamar a el correspondiente de esta clase se realiza de la siguiente manera:

```
private static String url_create_Device = "http://198.199.72.168//cas/grafica.php";
```

De igual forma tienen su layout correspondiente anidado de esta manera:

```
super.onCreate(savedInstanceState);
setContentView(R.layout.grafica);
```

Posee spinner para la selección de los electrodomésticos y algunas otras cosas. Se maneja la estructura de tres fases como se mencionó en el login utilizando el `onPreExecute`, `doInBackground` y el `onPostExecute`.

El código fuente es:

```
package v1.tesis.app.sebastianrincon.com.poupewatt;
```

```
/**
```

```
 * Created by ASUS on 25/5/2016.
```

```
 */import android.app.Activity;  
import android.app.DatePickerDialog;  
import android.app.Dialog;  
import android.app.ProgressDialog;  
import android.graphics.Color;  
import android.graphics.LinearGradient;  
import android.graphics.Paint;  
import android.graphics.Shader;  
import android.os.AsyncTask;  
import android.os.Bundle;  
import android.os.Message;  
import android.support.v7.app.ActionBar;  
import android.support.v7.app.ActionBarActivity;  
import android.support.v7.app.AppCompatActivity;  
import android.util.Log;  
import android.view.Menu;  
import android.view.View;  
import android.widget.AdapterView;  
import android.widget.AdapterView.Adapter;  
import android.widget.Button;  
import android.widget.DatePicker;  
import android.widget.Spinner;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import com.androidplot.Plot;  
import com.androidplot.xy.SimpleXYSeries;  
import com.androidplot.xy.*;  
  
import org.apache.http.NameValuePair;  
import org.apache.http.message.BasicNameValuePair;
```

```

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.text.DecimalFormat;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.Calendar;
import java.util.List;

public class Grafica extends ActionBarActivity {

    Spinner lista, lista2;
    String[] datos = {"None", "Potencia (Kwh)", "Precio ($)"};
    String[] datos2 = {"None", "Sensor 1", "Sensor 2", "Sensor 3", "Sensor 4", "Sensor
5"};
    private DatePicker datePicker1, datePicker2;
    private Calendar calendar;
    private TextView dateView;
    private ProgressDialog pDialog2 ;
    private String dia;
    private String dia2;
    private String A, B;
    private String nuevodate;
    private String nuevodate2;
    private JSONArray suma;
    private JSONArray signo;
    private int year, month, day, year2, month2, day2;
    TextView Texto;

    JSONParser jsonParser = new JSONParser();//clase para realizar las
transferencia
    //cuadros de texto

    //      private      static      String      url_create_Device      =
"http://172.18.7.226/cas/generarconsumo.php";//direccion url del archivo php
    private      static      String      url_create_Device      =
"http://198.199.72.168/cas/graficas.php";//direccion url del archivo php

```

```
// private static String url_create_Device =  
"http://172.18.7.127/cas/graficas.php";//direccion url del archivo php
```

```
private XYPlot mySimpleXYPlot;
```

```
@Override
```

```
public void onCreate(Bundle savedInstanceState)
```

```
{
```

```
    super.onCreate(savedInstanceState);
```

```
    setContentView(R.layout.grafica);
```

```
    ActionBar actionBar = getSupportActionBar();
```

```
    actionBar.setDisplayHomeAsUpEnabled(true);
```

```
    calendar = Calendar.getInstance();
```

```
    year = calendar.get(Calendar.YEAR);
```

```
    month = calendar.get(Calendar.MONTH);
```

```
    day = calendar.get(Calendar.DAY_OF_MONTH);
```

```
    showDate(year, month+1, day);
```

```
    year2 = calendar.get(Calendar.YEAR);
```

```
    month2 = calendar.get(Calendar.MONTH);
```

```
    day2 = calendar.get(Calendar.DAY_OF_MONTH);
```

```
    showDate2(year2, month2+1, day2);
```

```
    lista = (Spinner)findViewById(R.id.lista1);
```

```
    ArrayAdapter<String> adaptador = new ArrayAdapter<String> (this,  
android.R.layout.simple_spinner_dropdown_item, datos);
```

```
    lista.setAdapter(adaptador);
```

```
    lista2 = (Spinner)findViewById(R.id.lista2);
```

```
    ArrayAdapter<String> adaptador2 = new  
ArrayAdapter<String>(this,android.R.layout.simple_spinner_dropdown_item,datos2  
);
```

```
    lista2.setAdapter(adaptador2);
```

```
Button btnCreateDevice = (Button) findViewById(R.id.mimo);
Texto = (TextView) findViewById(R.id.prueba);
```

```
lista.setOnItemClickListener(new AdapterView.OnItemClickListener() {
    @Override
    public void onItemClick(AdapterView<?> parent, View view, int position,
long id) {

        switch (position) {

            case 1:

                A="kwh";
                //dia = datePicker1.getYear() + "-" + datePicker1.getMonth()+"-
"+datePicker1.getDayOfMonth();
                //Toast to = Toast.makeText(getApplicationContext(), dia,
Toast.LENGTH_LONG);
                //to.show();

                break;

            case 2:

                A="price";
                //dia2 = datePicker2.getYear() + "-" + datePicker2.getMonth()+"-
"+datePicker2.getDayOfMonth();
                //Toast to1 = Toast.makeText(getApplicationContext(), dia2,
Toast.LENGTH_LONG);
                //to1.show();

                break;

        }

    }

    @Override
```

```

        public void onNothingSelected(AdapterView<?> parent) {

        }
    });

    lista2.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener()
    {

        @Override
        public void onItemSelected(AdapterView<?> parent, View view, int position,
        long id) {

            switch (position) {

                case 1:

                    B = "sensor1";

                    break;

                case 2:

                    B = "sensor2";

                    break;

                case 3:

                    B = "sensor3";

                    break;

                case 4:

                    B = "sensor4";

```

```

        break;

    case 5:

        B = "sensor5";

        break;

    }

}

@Override
public void onNothingSelected(AdapterView<?> parent) {

}

});

btnCreateDevice.setOnClickListener(new View.OnClickListener() {

@Override
public void onClick(View view) {
    // creating new Empleado in background thread
    new CreateNewDevice().execute();

    //cuando se hace click al boton se empieza el procesos de agregar
}

});

}

@SuppressWarnings("deprecation")
public void setDate(View view) {
    showDialog(999);
}

```



```

        Toast.makeText(getApplicationContext(), "Primer Fecha",
Toast.LENGTH_SHORT)
            .show();
    }

    @SuppressWarnings("deprecation")
    public void setDate2(View view) {
        showDialog(998);
        Toast.makeText(getApplicationContext(), "Segunda Fecha",
Toast.LENGTH_SHORT)
            .show();
    }

    @Override
    protected Dialog onCreateDialog(int id) {
        // TODO Auto-generated method stub
        if (id == 999) {
            return new DatePickerDialog(this, myDateListener, year, month, day);
        }else{

            return new DatePickerDialog(this, myDateListener2, year2, month2, day2);
        }
    }

    private DatePickerDialog.OnDateSetListener myDateListener = new
    DatePickerDialog.OnDateSetListener() {
        @Override
        public void onDateSet(DatePicker arg0, int arg1, int arg2, int arg3) {
            // TODO Auto-generated method stub
            // arg1 = year
            // arg2 = month
            // arg3 = day
            showDate(arg1, arg2+1, arg3);
        }
    };

    private void showDate(int year, int month, int day) {
        nuevodate = year + "-" + month + "-" + day;
    }

    @Override

```

```

public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.

    return true;
}

```

```

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

```

```

private DatePickerDialog.OnDateSetListener myDateListener2 = new
DatePickerDialog.OnDateSetListener() {
    @Override
    public void onDateSet(DatePicker arg4, int arg5, int arg6, int arg7) {
        // TODO Auto-generated method stub
        // arg1 = year
        // arg2 = month
        // arg3 = day
        showDate2(arg5, arg6+1, arg7);
    }
};

private void showDate2(int year2, int month2, int day2) {
    nuevodate2 = year2 + "-" + month2 + "-" + day2;
}

```

```

class CreateNewDevice extends AsyncTask<String, String, String> {

    /**
     * Antes de empezar Asynk Task se muestra un mensaje de dialgo.
     */
    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        pDialog2 = new ProgressDialog(Grafica.this);
        pDialog2.setMessage("Generando...");
        pDialog2.setIndeterminate(false);
        pDialog2.setCancelable(true);
    }
}

```

```

        pDialog2.show();

    }

    /**
     * Proceso de agregar
     */
    protected String doInBackground(String... args) {

        //asignacion de las variables
        String tipoconsumo = A;
        String tiposensor = B;
        // dia = datePicker1.getYear() + "-" + (datePicker1.getMonth()+1) + "-" +
datePicker1.getDayOfMonth();
        // dia2 = datePicker2.getYear() + "-" + (datePicker2.getMonth()+1) + "-" +
datePicker2.getDayOfMonth();

        Log.d("enviar", tipoconsumo.toString() + tiposensor.toString() +
nuevodate.toString() + nuevodate2.toString()); // solo es un mensaje en log para ver
si se gurdan las variables.

//aqui se lista los elementos con la etiqueta respectiva para enviarla al php y las
procese correctamenten
        List<NameValuePair> params = new ArrayList<NameValuePair>();
        params.add(new BasicNameValuePair("tipoconsumo", tipoconsumo));
        params.add(new BasicNameValuePair("tiposensor", tiposensor));
        params.add(new BasicNameValuePair("dia", nuevodate));
        params.add(new BasicNameValuePair("dia2", nuevodate2));

        // obteniendo JSON Object
        // aqui se envia utilizando la url antes declarada, usando el metodo POST y
enviando los parametros
        JSONObject json = jsonParser.makeHttpRequest(url_create_Device,
            "POST", params);

        // verificar en log cat la respuesta
        Log.d("Create Response Consumo", json.toString());
    }
}

```

```

// revisar si el tag resulto exitoso
try {

    suma = json.getJSONArray("kwh");
    signo = json.getJSONArray("time");

    Log.d("Funciono", json.toString());

} catch (JSONException e) {
    e.printStackTrace();
}
return null;
}

/**
 * antes de completar el background se cierra el dialogo
 */
protected void onPostExecute(String file_url) {
    // dismiss the dialog

    mySimpleXYPlot = (XYPlot) findViewById(R.id.mySimpleXYPlot);
    Number[] numSightings = new Number[suma.length()];

    for(int j = 0; j < suma.length(); j++){
        try {
            numSightings[j] = suma.getDouble(j);
        } catch (JSONException e) {
            e.printStackTrace();
        }
    }

    Number[] años = new Number[suma.length()];
    for(int j = 0; j < suma.length(); j++){

        años[j] = 1.0 + j;

    }

    Number[] numSightings2 = {5.2, 8.4, 9.5, 2.6, 5};

```

```

    Number[] años2 = {
        2016, // 2001
        2017, // 2002
        2018, // 2003
        2019, // 2004
        2020 // 2005
    };

    Log.d("Funciono", años.toString() );
    // create our series from our array of nums:

    XYSeries series2 = new SimpleXYSeries(
        Arrays.asList(años),
        Arrays.asList(numSightings),
        "Consumo");

mySimpleXYPlot.getGraphWidget().getGridBackgroundPaint().setColor(Color.WHI
TE);

mySimpleXYPlot.getGraphWidget().getDomainOriginLinePaint().setColor(Color.BL
ACK);

mySimpleXYPlot.getGraphWidget().getRangeOriginLinePaint().setColor(Color.BLA
CK);

    mySimpleXYPlot.setBorderStyle(Plot.BorderStyle.SQUARE, null, null);
    mySimpleXYPlot.getBorderPaint().setStrokeWidth(1);
    mySimpleXYPlot.getBorderPaint().setAntiAlias(false);
    mySimpleXYPlot.getBorderPaint().setColor(Color.WHITE);

    // Create a formatter to use for drawing a series using
LineAndPointRenderer:
    LineAndPointFormatter series1Format = new LineAndPointFormatter(
        Color.rgb(0, 100, 0),           // line color
        Color.rgb(0, 100, 0),           // point color
        Color.rgb(100, 200, 0),null);   // fill color

    // setup our line fill paint to be a slightly transparent gradient:
    Paint lineFill = new Paint();

```

```

        lineFill.setAlpha(200);
        lineFill.setShader(new LinearGradient(0, 0, 0, 250, Color.WHITE,
Color.GREEN, Shader.TileMode.MIRROR));

        LineAndPointFormatter formatter = new
LineAndPointFormatter(Color.rgb(0, 0,0), Color.BLUE, Color.RED,null);
        formatter.setFillPaint(lineFill);
        mySimpleXYPlot.getGraphWidget().setPaddingRight(2);
        mySimpleXYPlot.addSeries(series2, formatter);

// draw a domain tick for each year:
mySimpleXYPlot.setDomainStep(XYStepMode.SUBDIVIDE, años.length);

// customize our domain/range labels
mySimpleXYPlot.setDomainLabel("Fecha");
mySimpleXYPlot.setRangeLabel("Kwh");

// get rid of decimal points in our range labels:
//mySimpleXYPlot.setRangeValueFormat(new DecimalFormat("0"));
mySimpleXYPlot.setDomainValueFormat(new DecimalFormat("0"));

//mySimpleXYPlot.updateDomainMinMaxForOriginModel();
mySimpleXYPlot.redraw();
//mySimpleXYPlot.updateDomainMinMaxForOriginModel();
// mySimpleXYPlot.updateRangeMinMaxForOriginModel();
//mySimpleXYPlot.clear();

pDialog2.dismiss();

    }

}}

```

### 2.3 Estructura de datos carpeta RES

Dentro de esta carpeta encontramos todos los recursos necesarios para la aplicación, ya sean las imágenes como los strings o los layouts que son complementos para las clases que ya se explicación con anterioridad.

- **Carpeta drawable:** Aquí se encuentra todas las imágenes que se usaron dentro de la aplicación. Si se desea agregar alguna mas sencillamente es copiarla dentro de esta carpeta, pueden usarse imágenes en formato .jpg o también .png para este caso, todas entran en formato .png.
- **Carpeta layout:** Dentro de esta carpeta están todas las interfaces que verá el usuario dentro de la aplicación a continuación se muestra una tabla donde se hace la conexión con su clase correspondiente:

Clase	Layout
Consumo	Consumo
Descrip1	Descrip1
Descrip2	Descrip2
Descrip3	Descrip3
Nav_DrawerListAdapter	Drawer_list_item
Fmconsumo	Fmconsumo
Fmhistoricos	Fmhistoricos
Fmhome	Fmhome
Fmjuego	Fmjuego
Grafica	Grafica
Inicio	Inicio
Juego	Juego
Login	Login
Preferencias	Preferencias
Preferencias2	Preferencias2
Registro	Registro
SplashScreenActivity	Splash_screen

- **Carpeta Menú:** Sencillamente posee archivos .xml para complementar la Action Bar utilizada dentro del menú deslizable que está explicado en la parte de arriba.
- **Carpeta Mipmap:** Dentro de ella se encuentran todos los logos, las imágenes en diferentes tamaños que se usan dentro de la app. Como por ejemplo las imágenes del menú, la imagen de la Webview, el logo de la aplicación y demás.
- **Carpeta Raw:** Dentro de ella se encuentran los archivos .txt utilizados dentro de la Webview para visualizar la información que se escriba dentro de ellos.
- **Carpeta Values:** Esta carpeta es la configuración visual de la aplicación como tal, es decir, aquí se encuentran los colores de la aplicación, los estilos visuales de la misma. Por otro lado están los strings que son los fragmentos escritos que declare el programador para llamarlos dentro de cualquier parte de la app.