

**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO**

CARRERA: INGENIERÍA ELÉCTRICA

**Tesis previa a la obtención del título de:
INGENIERO ELÉCTRICO**

**TEMA:
APLICACIÓN ANDROID PARA MEDICIÓN INTELIGENTE DE ENERGÍA
ELÉCTRICA EN USUARIOS RESIDENCIALES URBANOS**

**AUTOR:
LUIS GUSTAVO DE LA CRUZ PUEBLA**

**DIRECTOR:
ESTUARDO J. CORREA ZAPATA**

Quito, Marzo de 2015

DECLARATORIA DE AUTORÍA

Yo, Luis Gustavo De la Cruz Puebla autorizo a la Universidad Politécnica Salesiana la publicación total o parcial de este trabajo de grado y su reproducción sin fines de lucro.

Además declaro que los conceptos y análisis desarrollados y las conclusiones del presente trabajo son de exclusiva responsabilidad del autor.

Quito, 27 de Febrero del 2015

Luis Gustavo De la Cruz Puebla
CC: 171915232-2

AUTOR

CERTIFICA

Haber dirigido y revisado prolijamente cada uno de los capítulos del trabajo de monografía intitulado **“APLICACION ANDROID PARA MEDICIÓN INTELIGENTE DE ENERGÍA ELÉCTRICA EN USUARIOS RESIDENCIALES URBANOS.”** realizada por el Sr. Luis De la Cruz Puebla, previa a la obtención del título de Ingeniero Eléctrico en la Carrera de Ingeniería Eléctrica.

Por cumplir los requisitos autoriza su presentación.

Quito, 17 de Marzo del 2015

Ing. Estuardo J. Correa Zapata
DIRECTOR

ÍNDICE DE CONTENIDO

CAPÍTULO I	1
Android y sus aplicaciones en medición inteligente.....	1
1.1 Sistema Operativo Android	1
1.1.1 Evolución de Android	2
1.1.2 Arquitectura de Android	5
1.2 Plataforma de desarrollo para Android.....	7
1.3 Herramientas de Programación para Android	11
1.4 Aplicaciones y servicios en dispositivos móviles	14
1.4.1 Análisis.....	16
1.4.2 Diseño	17
1.4.3 Desarrollo.....	18
1.4.4 Pruebas de funcionamiento	18
1.4.5 Entrega	18
CAPÍTULO II.....	20
Prototipo de la aplicación	20
2.1 Conectividad inalámbrica WiFi.....	21
2.1.1 Wireless Personal Area Network (WPAN).....	22
2.1.2 Wireless Local Area Network (WLAN)	23
2.1.3 Wireless Metropolitan Area Network (WMAN)	25
2.1.4 Wireless Wide Area Network (WWAN)	26
2.2 Internet y servidor web.....	28
2.2.1 Casos de uso.....	28
2.2.2 Servidor web	31
2.2.3 Servidor Apache.....	32
2.2.4 PHP	32

2.3 Recuperación de datos.....	32
2.4 Organización y presentación de datos.....	34
2.4.1 Base de datos.....	34
2.4.2 Tabla Cliente:.....	35
2.4.3 Tabla Consumo.....	36
2.4.4 Tabla factura.....	36
2.4.5 Flujograma de la aplicación:.....	37
CAPÍTULO III.....	38
Medición inteligente de energía eléctrica.....	38
3.1 Hogares y ciudades inteligentes.....	38
3.2 Desarrollo de la aplicación de medición inteligente.....	40
3.3 Implementación de sistemas de medición inteligente.....	48
3.3.1 Herramientas utilizadas para el desarrollo de la aplicación.....	48
3.3.2 Servicios web desarrollados.....	50
3.3.3 Clases desarrolladas en java.....	51
3.4 Evolución de consumos sin acciones correctivas.....	52
CAPÍTULO IV.....	59
Ingeniería de Proyectos.....	59
4.1 Análisis Costo-Beneficio.....	60
4.4.1 Cálculo de costos-beneficios.....	61
4.2 Servicios a ser implementados sobre el sistema Android.....	66
4.2.1 Aplicaciones para ahorrar el consumo energético en el hogar en el sistema operativo Android.....	66
4.3 Impacto Social de las aplicaciones Android en medición inteligente.....	68
4.4 Plan de negocios para implementar proyectos Android para medición inteligente.....	70
4.4.1 Plan de negocio para la implementación de la aplicación Monitor EE del sistema Android para medición inteligente.....	73

4.4.2 Pruebas a la aplicación	75
CONCLUSIONES	82
RECOMENDACIONES.....	83
REFERENCIAS.....	84
ANEXOS	89
PROGRAMACION DE APLICACIÓN ANDROID.....	90

ÍNDICE DE FIGURAS

FIGURA 1.1 PORCENTAJE DE SISTEMAS OPERATIVOS PARA DISPOSITIVOS MÓVILES.....	4
FIGURA 1.2 ARQUITECTURA DE ANDROID	5
FIGURA 1.3 ARQUITECTURA DE PLATAFORMA JAVA	7
FIGURA 1.4 ARQUITECTURA DE ECLIPSE.....	9
FIGURA 1.5 EMULADOR DEL DISPOSITIVO MÓVIL VIRTUAL	13
FIGURA 1.6 INSTALACIÓN AUTOMÁTICA DE SOFTWARE DEVELOPMENT KIT EN ANDROID STUDIO.....	14
FIGURA 1.7 METODOLOGÍA DE DESARROLLO	16
FIGURA 1.8 SERVICIOS IMPLEMENTADOS EN APLICACIONES DE MEDICIÓN ELÉCTRICA INTELIGENTE	19
FIGURA 2.1 MODELO BÁSICO DE FUNCIONAMIENTO DE LA APLICACIÓN.....	20
FIGURA 2.2 ALCANCE DE REDES INALÁMBRICAS.....	21
FIGURA 2.3 TOPOLOGÍA DE COMUNICACIÓN ADMITIDA EN ZIGBEE.....	22
FIGURA 2.4 ARQUITECTURA RED WiMAX.....	26
FIGURA 2.5 CASOS DE USO DE LA APLICACIÓN	29
FIGURA 2.6 COMANDOS NECESARIOS PARA CONSULTA DE BASES DE DATOS	34
FIGURA 2.7 FLUJOGRAMA DE LA APLICACION	37
FIGURA 3.1 DIAGRAMA DE CASO DE USO ADQUIRIR APLICACIÓN MONITOR EE.....	41
FIGURA 3.2 DIAGRAMA DE CASO DE USO CONSULTAR APLICACIÓN MONITOR EE.....	42
FIGURA 3.2 POTENCIA DE EQUIPOS EN UN HOGAR TIPO.....	53

ÍNDICE DE TABLAS

TABLA 1.1 VERSIONES Y CARACTERÍSTICAS DE ANDROID	4
TABLA 2.1 RESUMEN DE EVOLUCIÓN DE LAS REDES INALÁMBRICAS MÓVILES	28
TABLA 2.2 CONSULTAS GENERALES SOBRE LAS BASES DE DATOS	33
TABLA 3.1 DESCRIPCIÓN DE CASO DE USO: ADQUIRIR APLICACIÓN MONITOR EE.....	43
TABLA 3.2 DESCRIPCIÓN DE CASO DE USO: CONSULTAR APLICACIÓN MONITOR EE.	43
TABLA 3.3 DESCRIPCIÓN DE CASO DE USO: REGISTRAR USUARIO	44
TABLA 3.4 DESCRIPCIÓN DE CASO DE USO: CONSULTAR CONSUMO ELÉCTRICO	45
TABLA 3.5 DESCRIPCIÓN DE CASO DE USO: REALIZAR PAGO-BANCO.....	46
TABLA 3.6 DESCRIPCIÓN DE CASO DE USO: REGISTRAR DATOS DE USUARIO	46
TABLA 3.7 DESCRIPCIÓN DE CASO DE USO: COMPARAR CONSUMO ELÉCTRICO.....	47
TABLA 3.7 DESCRIPCIÓN DE CASO DE USO: ENVIAR NOTIFICACIONES	48
TABLA 3.7 DESCRIPCIÓN DE CASO DE USO: ENVIAR NOTIFICACIONES	57
TABLA 4.1 DEFINICIÓN DE CASOS DE PRUEBAS.....	78
TABLA 4.2 RESULTADOS REALES DE LOS CASOS DE PRUEBAS.....	81

DEDICATORIA.

Este proyecto lo dedico a mis padres, ya que juntos me han brindado el mejor ejemplo de sabiduría y disciplina, y que en todo momento me han dado la fuerza y los ánimos para no decaer pese a las adversidades.

A mis hermanas que son una parte fundamenta en mi vida y Aidee Mantilla que en todo momento me dio palabras de ánimo para seguir con el desarrollo de este trabajo.

Luis Gustavo De la Cruz Puebla

AGRADECIMIENTO.

Agradezco en gran manera a cada uno de mis profesores que compartieron sus conocimientos durante mi carrera universitaria, especialmente al Ing. Estuardo J. Correa ya que sin su guía y consejos este trabajo no hubiese sido posible realizarlo.

Luis Gustavo De la Cruz Puebla

RESUMEN

APLICACIONES ANDROID PARA MEDICIÓN INTELIGENTE DE ENERGÍA ELÉCTRICA EN USUARIOS RESIDENCIALES URBANAS

Luis, Gustavo, De la Cruz, Puebla

lugu_e@hotmail.es

Universidad Politécnica Salesiana

Resumen— La presente investigación refleja el estudio realizado para la creación de una aplicación Android encaminada al sector eléctrico y específicamente relacionado a la comercialización de energía eléctrica, la aplicación realizada tiene la finalidad de acercar la empresa distribuidora hacia los usuarios por medio de información oportuna del consumo de energía. Los datos que requiere la aplicación para mostrar al usuario deben ser tomados de un medidor inteligente el cual alimenta una base de datos, los datos son recuperados por medio de un web server y mostrados en la aplicación.

Índice de Términos—*Smart Grid; Smart Metering; Servidor Web; SGDBD; Conectividad; Android; TIC's.*

Abstract- This research reflects the study to create an Android app aimed at electricity sector and specifically related to the sale of electricity, the application performed aims to bring the distributor to users through timely information power consumption. The data required by the application to show the user must be taken from a smart meter which feeds a database, the data is retrieved by a web server and displayed in the application.

Keywords— *Smart Grid; Smart Metering; Web server; SGDBD; connectivity; Android; ICT.*

INTRODUCCIÓN

En los últimos años la evolución y crecimiento de la carga dentro del sistema eléctrico ecuatoriano ha supuesto un aumento de pérdidas energéticas debido a la obsolescencia de muchos de sus elementos[1], es por esta razón que es imprescindible replantear dicho sistema eléctrico con la finalidad de disminuir las pérdidas energéticas. Dentro de este contexto, una de las opciones que se encuentra en auge en varias latitudes del mundo es la implementación de redes inteligentes, las mismas que se conocen como SMART GRID.[2]

Una red inteligente contempla la modernización de los sistemas actuales de generación, transmisión, distribución y comercialización de la energía. Con la finalidad de mejorar la calidad de servicio, reducir las pérdidas y optimizar recursos[3].

La generación centralizada como es el caso del sistema eléctrico actual, requiere grandes centros de generación y la infraestructura adecuada para el transporte de la potencia generada, esto conlleva algunos inconvenientes, entre los que se puede mencionar el costo de los proyectos, las pérdidas técnicas asociadas al transporte de la energía y el gran número de poblaciones que dependen del sistema[4]. Por otro lado en la implementación de las redes eléctricas inteligentes se plantea la necesidad de acoplar al sistema actual la generación distribuida, en la que la generación de la energía este lo más cerca posible a la carga con lo que se minimizan los impactos ambientales y las pérdidas por transporte, la generación distribuida contempla el uso de energías renovables (Eólica, solar, geotérmica, fotovoltaica, etc.) las cuales permiten dotar de servicio a poblaciones excluidas[3].

Un Smart Grid no es una fórmula que se la pueda aplicar de la noche a la mañana dentro de un sistema, sino más bien es una serie de acciones en las que el principal componente es la implementación de sistemas de comunicación bidireccional entre los centros de consumo de energía (Usuarios) y los centros de distribución (Empresas de distribución) por medio de las tecnologías de la información y telecomunicaciones (TIC)[5], esto sumado a los grandes avances de las tecnologías de la comunicación han permitido que

este tipo de redes estén en la capacidad de transferir y recibir datos en tiempo real.

Algunas de las ventajas de implementar esta nueva concepción de redes eléctricas es que se reducen los tiempos de toma de lectura, cortes y reconexiones, facturación, etc. Todos estos servicios como ya se mencionó deben ser de comunicación bidireccional, además se debe dotar a los usuarios de servicios adicionales tales como disponibilidad de información de los consumos, control inteligente sobre los aparatos de consumo que complementen la medición remota[6].

Dentro de este contexto y debido a gran auge de los dispositivos móviles, se hace necesario el desarrollo de una aplicación móvil bajo la plataforma Android, la misma que este en capacidad de presentar al usuario de forma ágil y sencilla los datos de los consumos eléctricos de su hogar, estas aplicaciones móviles se pueden constituir en los servicios adicionales a ofertar a los clientes los cuales podrán disponer de los datos en cualquier sitio y en cualquier hora de forma directa[7].

Para el desarrollo y ejecución de esta investigación se hace necesario conocer más a fondo a este sistema para lo cual se determina una metodología de investigación en esta primera parte, la metodología escogida es aquella que nos ayudará a conocer la historia y evolución de Android desde su aparición oficial pasando por el principio de funcionamiento y arquitectura interna.

Es así que el marco metodológico idóneo para éste capítulo será el denominado descriptivo con el fin de ir estructurando de forma lógica la información necesaria para el desarrollo de la investigación.

CAPÍTULO I

ANDROID Y SUS APLICACIONES EN MEDICIÓN INTELIGENTE

En el desarrollo de aplicaciones móviles es importante conocer el funcionamiento y la estructura interna del sistema operativo, con el fin de que al momento de usar los recursos físicos y de memoria del dispositivo no existan inconvenientes y la aplicación se ejecute sin ningún problema, también se debe conocer las características técnicas del terminal para tratar de sacarle el mayor provecho ya que dependiendo de este factor se podrán desarrollar aplicaciones de mayor calidad y desempeño[8].

1.1 Sistema Operativo Android

Desde sus inicios el sistema operativo para dispositivos móviles que hoy en día se conoce como Android se mostró con la convicción de revolucionar la industria de los dispositivos móviles, es por esta razón que se creó una alianza entre compañías que lideradas por Google se empezó a dar forma al gran contendor en el campo de los sistemas operativos para dispositivos móviles[9].

En el momento en el que se lanzó al mercado éste sistema operativo llamó la atención el hecho de basar su funcionamiento en Linux lo cual permite que desarrolladores independientes tengan la posibilidad de crear aplicaciones por cuenta propia, cabe destacar que la programación de dichas aplicaciones usa lenguaje C[10] y otros, entre los cuales se destaca el lenguaje Java el cual será usado en el proyecto de investigación.

La fuerza con la que Android se ha cimentado en el mercado se atribuye a la continua reinención que ha tenido, así lo refleja la cantidad de versiones que se han presentado. Éste es un punto fundamental, ya que en función de la capacidad física de los dispositivos (Hardware) las actualizaciones están a la mano disponibles en cualquier momento[9].

Adicional a esto, llama la atención el hecho de que a cada una de las versiones lanzadas al mercado se les ha dado un seudónimo relacionado a postres o dulces, una estrategia de mercado en la cual se da a entender el gusto que provoca en los usuarios la interacción

con este sistema operativo. Cada una de las versiones posee una continuidad alfabética en la primera letra de su nombre[11].

1.1.1 Evolución de Android

Nombre / Versión	Características Técnicas
Cupcake / 1.5	<p>Transiciones animadas entre ventanas. Menor tiempo de búsqueda de los satélites GPS, gracias a la posibilidad de usar A-GPS. Interprete JavaScript. Posibilidad de personalizar los widgets en la pantalla de inicio. Mejoras en la velocidad de navegación web gracias a la inclusión de la última versión de Webkit. Añadida la posibilidad de copiar y pegar texto y buscar texto dentro de una página web. Añadida la posibilidad de grabar y reproducir videos. Inclusión de teclado en la pantalla con soporte para orientación, autocorrección y soporte de diccionarios del usuario</p>
Donut / 1.6	<p>Está basada en el kernel 2.6.29 de Linux Una caja de búsqueda en la pantalla de inicio Mejora de velocidad de la cámara Posibilidad de conectarse a redes VPN, 802.1x Nueva pantalla para controlar la batería y monitorear que aplicaciones más consumen Desde la pantalla se puede parar o desinstalar aplicaciones Las aplicaciones de AndroidMarket aparecen ordenadas por categorías.</p>
Éclair / 2.0 / 2.1	<p>V 2.0 Rediseño de la interfaz del navegador Soporte nativo de flash para la cámara Zoom digital, modo scene, balance de blanco, efectos de color y modo marco Soporte para nuevos tamaños y resoluciones de pantalla Contactos rápidos. Bluetooth 2.1 Soporte nativo de Facebook. Mejoras en Google Maps V 2.1 Reconocimiento de voz. Mejoras en el teclado virtual Galería 3D Uso del gesto de pellizcar para hacer zoom en el navegador y la galería. Nuevas aplicaciones de reloj, tiempo y noticias. Google Goggles</p>

	Mejoras en la duración de la batería.
Froyo / 2.2	Actualizaciones automáticas para aplicaciones. Soporte WiFi IEEE 802.11n Soporte para radio FM Soporte flash 10.1 Soporte API gráfica OpenGL 2.0 Mejora un 450% frente a Éclair. Tethering por USB y hotspotWiFi. Nuevo Market
Gingerbread / 2.3	Soporte NFC. Administración de la energía mejorada y control de aplicaciones mediante administrador de tareas. Soporte para pantallas extra grandes WXGA (Tablets). Soporte nativo para VoIP SIP. Un administrador para descargas para descargas de archivos grandes. Soporte nativo para múltiples cámaras. Mejor aprovechamiento de la batería.
Honeycomb / 3.0 / 3.1 / 3.2	Escritorio 3D con widgets rediseñados. Sistema multitarea mejorado Mejoras en el navegador web Añade soporte para una gran variedad de periféricos y accesorios con conexión USB Mejor soporte para conexión WiFi
Ice CreamSandwich / 4.0	Versión que unifica el uso en cualquier dispositivo. Añadido un gestor de tráfico de datos de internet. Corrector de texto rediseñado Desplegar barra de notificaciones con el dispositivo bloqueado Compartir contenido entre teléfonos vía NFC Reconocimiento de voz del usuario Reconocimiento facial
Jelly Bean / 4.1 / 4.2	Mejora la fluidez de la interface La función de dictado por voz se la puede realizar si que sea necesario una conexión a internet Permite la actualización de aplicaciones parciales de aplicaciones Se pueden crear varias cuentas de usuario en un mismo dispositivo Posibilidad de realizar fotografías en 360°
KitKat /4.4	Mejora el rendimiento de la ram, capaz de ajustarse a dispositivos de 512 MB Incorpora Google Now, utilidad que realiza búsquedas por medio de comandos de voz. Añade nuevos perfiles bluetooth con los que permite envío de mensajes a dispositivos cercanos, uso del móvil como mouse o teclado y control de volumen. Rediseño de QuickOffice que permite crear documentos, hojas de cálculo y realizar presentaciones. Posibilidad de imprimir archivos directamente desde el dispositivo

Lolipop / 5.0	<p>Incorpora SELinux, con lo cual las aplicaciones maliciosas tiene menos oportunidades de atacar los dispositivos</p> <p>Trae activado por defecto el cifrado de datos del teléfono.</p> <p>La actualización a Android L mejora significativamente el uso del dispositivo con una mejor gestión de la batería.</p> <p>Añade la posibilidad de tener múltiples usuarios en el dispositivo y la opción invitado, con la que se restringe el acceso a datos personales.</p> <p>Mejoras en la gestión de las conexiones inalámbricas,</p>
---------------	--

Tabla 1.1 Versiones y características de Android

Fuente: <http://www.mundoandroides.com/los-nombres-postres-de-las-diferentes-versiones-de-android-y-sus-caracteristicas-tecnicas>

La evolución y avances de Android, han contribuido con nuevas experiencias y mejores prestaciones para los usuarios que día a día buscan sacarle el mayor de los provechos a sus o Tablets y Smartphone

La gran mayoría de usuarios han tenido grandes satisfacciones con éste sistema operativo especialmente por sus constantes actualizaciones y sobre todo porque el usuario puede descargar varias aplicaciones de forma gratuita. A comparación con otros sistemas operativos móviles, Android ha ganado mucho terreno puesto que el 80% de sus aplicaciones son gratuitas, tal como se lo demuestra en la Figura 1.1[9].

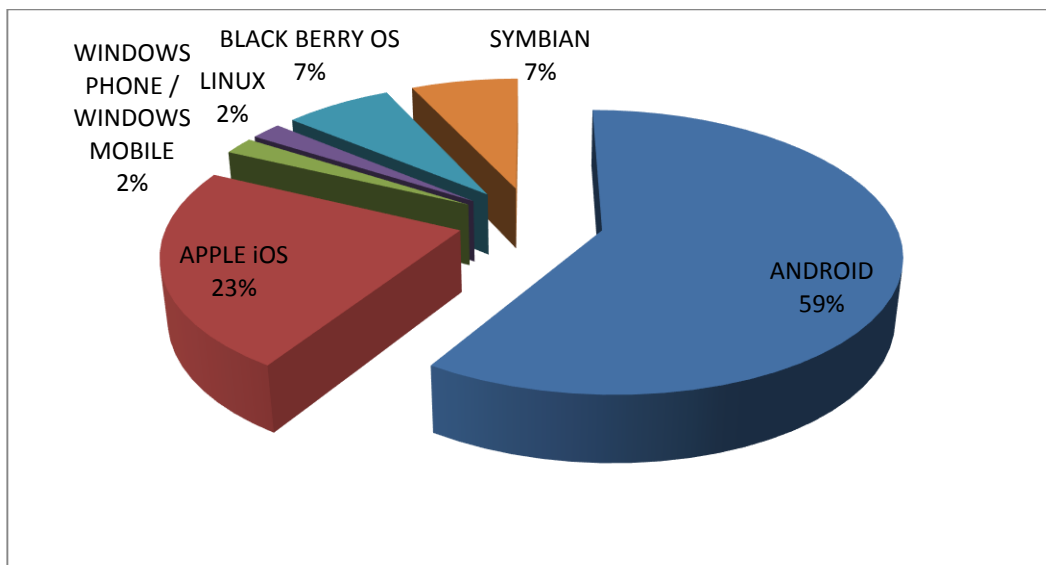


Figura 1.1 Porcentaje de sistemas operativos para dispositivos móviles

Fuente: <http://hablaconatico.wordpress.com/>

La mayor de las garantías que tiene Android es el hecho presentarse como un sistema de código abierto que permite una reparación de errores de forma ágil y no únicamente por la propia compañía sino que en el caso de encontrarse un error de sistema los desarrolladores independientes pueden corregirlos y mejorar los terminales según las necesidades, sin que estas acciones tengan que requerir el pago de licencias por la modificación del código[10].

Salvo por algunas consideraciones que de a poco se han ido corrigiendo con el pasar del tiempo es un sistema muy rápido y fluido, tal vez en su momento se criticó mucho la gestión de la batería, pues ésta daba mucho que desear, se puede decir que fue una consecuencia directa de mantener ciertas las aplicaciones en funcionamiento pero de una forma oculta con el fin de que estén siempre disponibles, obviamente esa estrategia supone un consumo tanto de memoria como de energía lo cual mermaba la batería, problema que en los nuevos terminales se han ido mejorando[12].

1.1.2 Arquitectura de Android

La forma en la que está concebido este sistema operativo o como es estructurado se describe por en la Figura 1.2 en la cual se aprecian cuatro capas a las cuales les corresponden actividades específicas [9].

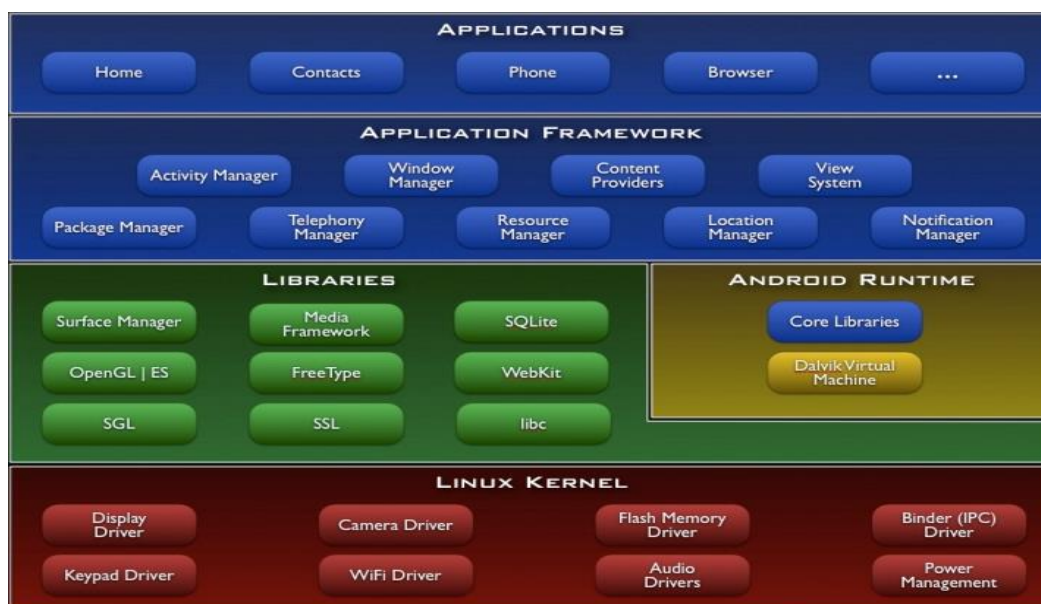


Figura 1.2 Arquitectura de ANDROID

Fuente:<http://androideity.com/2011/07/04/arquitectura-de-android/>

Si se empieza por la parte inferior del esquema se encuentra con la base misma de Android en la que está representado el Kernel o núcleo, que como se dijo anteriormente opera bajo Linux 2.6, esta capa se encarga de la seguridad del dispositivo, la gestión de memoria,[9] enlaza el software con los diferentes recursos de hardware del terminal[12]. “Y gestiona las funciones más básicas del sistema”[13].

En esta capa se encuentran los controladores de las diversas funciones del Smart Phone como son audio, conexiones inalámbricas, pantalla, teclado, cámara, gestión de batería etc.

Para algunos de los casos se tiene que recurrir a la ejecución de máquinas virtuales que en este caso específico se llama Goldfishy el mismo que se encarga de abrir recursos de entrada y salida de información tales como salida de video o lectura de la presión de las teclas[14].

El administrador de energía tiene un espacio separado y en el que Google en vez de modificar el que está por default en Linux incorporó uno totalmente nuevo que ha ido mejorando en cada una de las versiones presentadas al público[14].

En el siguiente nivel pero muy bien diferenciados están tanto las Librerías, como el Runtime de Android. Por su lado las bibliotecas son usadas por varios componentes del sistema y están compiladas en código nativo[9], las mismas que son presentadas a los desarrolladores con lo que tiene acceso completo, por ejemplo a la base de datos, gráficos, etc.

El Runtime de Android permite que cada aplicación se ejecute de forma independiente por medio de una máquina virtual (VM) para mejorar su eficiencia[12]. Esta máquina virtual toma el nombre de Dalvik y fue diseñada específicamente por Google ya que las limitaciones de los dispositivos no permitían que se ejecute una VM estándar de Java[9].

Aprovechando la estructura pre concebida en el Framework los desarrolladores pueden tener acceso a los recursos del dispositivo como por ejemplo administradores de ventanas, localización, notificaciones, acceder a datos de otras aplicaciones etc.[10].

El nivel de las aplicaciones se puede decir que es el que está más cercano al usuario ya que prácticamente es la interface entre el dispositivo y la persona, es claro que no es una regla general ya que habrán aplicaciones propias del sistema que a pesar de estarse ejecutando no tendrán una interacción directa con el usuario[14].

1.2 Plataforma de desarrollo para Android

El lenguaje que se usa para la creación de las aplicaciones para android es el lenguaje Java que vio sus inicios allá por los años 90's[15]. y que gano terreno en muchos ámbitos, la versión que será estudiada en este momento es Java 2 ME que no es más que la edición reducida, ideal para dispositivos que tiene ciertas limitaciones en su procesamiento de imágenes o de memoria como por ejemplo palm's o teléfonos celulares.

La plataforma reducida de java para dispositivos con capacidades restringidas tiene ciertas características propias que la diferencia de otras versiones más completas, por ejemplo posee una máquina virtual la cual consume apenas unos cuantos Kilobytes cuando se ésta ejecutando (KVM) (Kernel-based Virtual Machine)[16].

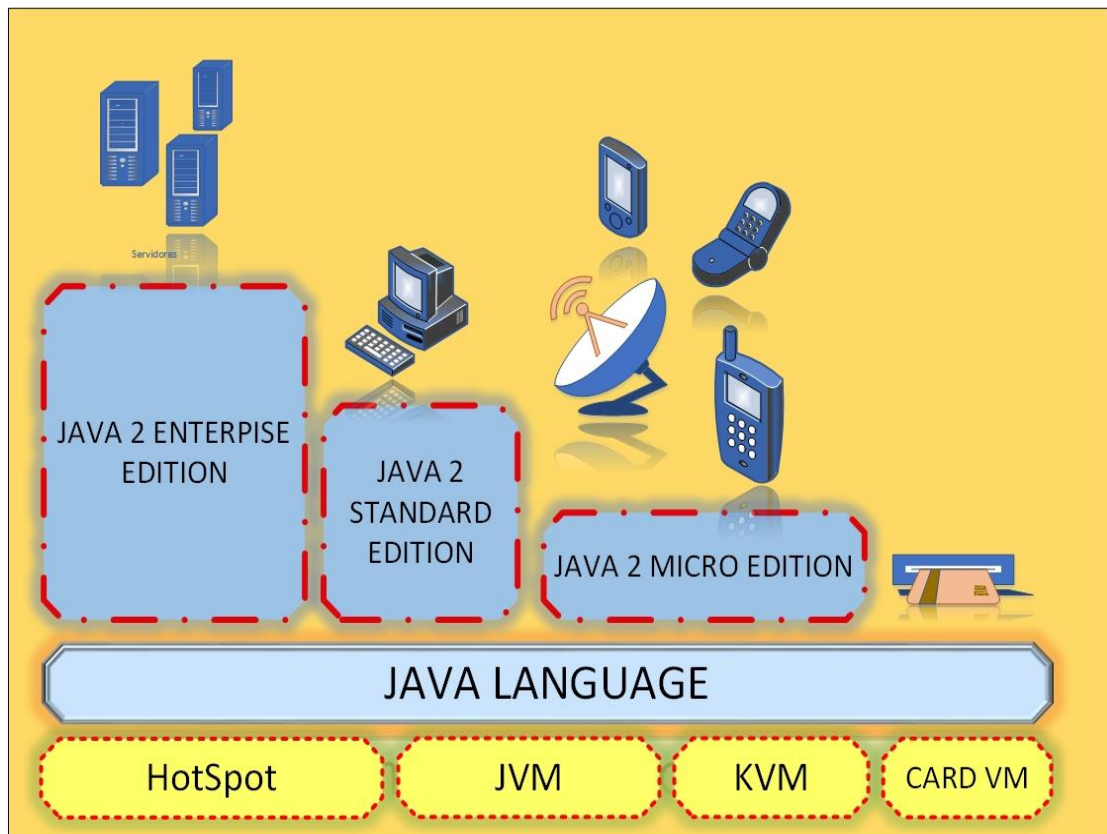


Figura 1.3 Arquitectura de plataforma Java

Fuente: <http://psoler.8m.net/J2ME/indexJ2ME.html>

En la Figura 1.3 se esquematiza la amplia gama de aparatos en los que se requiere una programación en lenguaje java, esto da a entender lo versátil y potente que ha llegado a ser, así puede ir desde chips de tarjetas hasta servidores pasando por ámbitos como la telefonía, los ordenadores de bolsillo, decodificadores de TV y computadoras, prácticamente java ha sido el motor de muchos avances tecnológicos [17].

Hoy en día se puede encontrar varios entornos de desarrollo que permiten la escritura y programación en lenguaje Java, este es el caso de NetBeans o Eclipse. Los cuales al igual que Android poseen las características del código abierto, en el caso de Eclipse en sus inicios fue desarrollada por IBM pero en la actualidad la Fundación Eclipse es la que tiene a su cargo el desarrollo de la misma, que sin ánimo de lucro fomenta en la comunidad de desarrolladores un ambiente propicio para la creación de programas y aplicaciones [18].

El conocimiento de estos entornos de desarrollo en lenguaje Java será de gran ayuda al momento de empezar la programación de las aplicaciones para Android ya que solo basta con añadir ciertos Plug-ins (módulos o accesorios) a Netbeans o Eclipse para tener acceso al ambiente de desarrollo de Android, en tal caso se hablara de Eclipse como una base introductoria hacia la programación [19].

Las ventajas que presenta Eclipse son varias, entre las cuales se mencionan el hecho de funcionar sobre varias plataformas o sistemas operativos, es decir que Eclipse está disponible en versiones para Windows, Linux, Mac OSX, etc.

Adicional a la programación en java que es aquella para la cual está diseñado, tiene la capacidad de integrar C/C++, COBOL, y a más de éstos se pretende crear soporte para lenguajes como Python, Perl, PHP [20].

Mediante un bloque de funciones se puede ilustrar de una forma visual cómo se estructura Eclipse, verificar que posee una estructura modular en la que cada bloque se construye sobre las bases del o de los anteriores, tal vez es por esta razón que se ha desarrollado con tanta amplitud, las necesidades de terceros ha hecho que se aproveche su concepción de código abierto para que integren cada vez más funcionalidades [20].

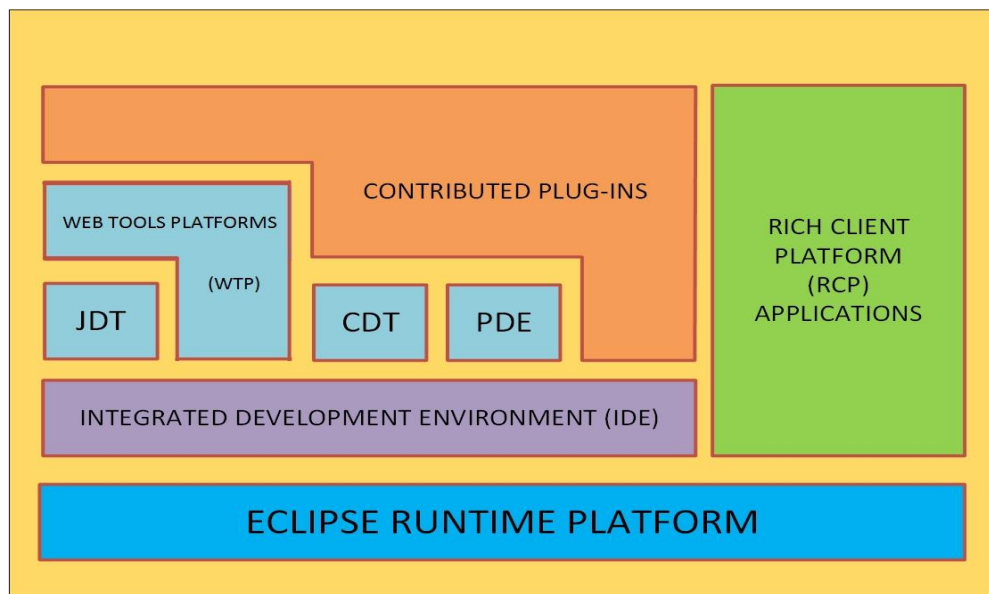


Figura 1.4 Arquitectura de Eclipse

Fuente: <http://repositorio.bib.upct.es:8080/dspace/bitstream/10317/2694/1/pfc4249.pdf>

Los servicios más básicos que ejecuta la plataforma de Eclipse se los menciona de forma breve, posee un registro de Plug-ins el cual es manejado y cargado por el Runtime, se encarga de gestionar los archivos y carpetas del sistema, verifica si existen disponibles actualizaciones importantes por medio direcciones URL para una posterior instalación, brinda también una ayuda compartida a través de los Plug-ins que se han instalado [20].

En el siguiente nivel y por sobre el Runtime se encuentra el IDE (Intergrated Development Environment) las características más importantes se las encuentra en la pantalla del programa las cuales se explican a continuación.

Eclipse provee la posibilidad de visualizar en una lista los proyectos que se han realizado y la opción de verificar el contenido de dichos proyectos, si se encasilla a un grupo de vistas comunes en función de un fin común toma el nombre de Perspectiva [20].

Posee una barra de herramientas, una barra de menú principal las cuales básicamente permiten el acceso a opciones comunes del proyecto, como ejemplos de dichas opciones se puede mencionar pestañas para abrir o crear un nuevo proyecto etc.

Las herramientas de desarrollo en java (Java Development Tools) son las únicas “dirigidas a un lenguaje de programación incluidos en el SDK de Eclipse”[20]. Ya que

los Plug-ins destinados a otros lenguajes son aportes de las comunidades de desarrolladores o a su vez únicamente sub-proyectos del mismo Eclipse.

Para proyectos en los que los archivos están en código fuente java se proporcionan algunas características generales como son: Ayuda, Editor, Plantillas, etc. Para una navegación más fluida y ordenada, en el programa se dispone de varias vistas en las que provee la posibilidad de saltar entre proyectos y tener acceso al contenido de los mismos.

Brinda las facilidades para configurar los proyectos desde las librerías que se van a usar, opciones de compilador, clases java del proyecto. También supone la ayuda de una herramienta de depuración con la que básicamente se puede determinar una ejecución paso a paso para analizar cómo se están ejecutando las sentencias dentro del programa, “establecer puntos de interrupción“ [20].

Eclipse tiene incorporado un IDE de C/C++ al que se denomina C/C++ Development Tools el cual incluye en su interior un editor especializado para C/C++ en el que se colorea el texto según corresponda a la sintaxis del lenguaje, cuenta también con un depurador de errores por defecto que permite la ejecución del programa de forma sistemática con el fin de encontrar errores (Bugs), relacionado al debugger está un C/C++ Program launcher que se asemeja al lanzador de Java [18].

Una parte esencial para el funcionamiento de otros Plug-ins es C/C++ Parser and Syntax API (Application Programming Interface), tiene un apartado especial para ejecutar búsquedas códigos y referencias según corresponda por medio de un SearchEngine y por último algo muy importante que ayuda a que los archivos de C/C++ a que sean compilados, el MakefileGenerator [20].

El Plug-in Development Environment permite hacer uso de herramientas que facilitan la creación, despliegue y manipulación de otros Plug-ins, el mismo que funciona como parte del Software Development Kit (SDK) de Eclipse y es una ayuda para el desarrollador en cada etapa de la creación. Dentro de Eclipse se puede decir que un Plug-in es un programa en Java que extiende las funcionalidades de sí mismo y en vista de que es una plataforma de código abierto es fácilmente expansible por terceras partes, es decir que un Plug-in puede fácilmente ocupar los recursos que ya han sido desarrollados en

otro Plug-in [20].

RichClienPlatform está diseñada para funcionar como plataforma de herramientas abiertas ya que es de código libre pero tranquilamente estas herramientas pueden servir para la construcción de cualquier aplicación.

1.3 Herramientas de Programación para Android

En función de las atribuciones que se da a los desarrolladores se ha pensado en la distribución gratuita de las herramientas necesarias que hacen posible la participación continua de personas ajenas al grupo “Eclipse” en la creación de soluciones efectivas a las diferentes necesidades que van apareciendo, la programación de las aplicaciones para Android se las hace en base a lenguaje Java, por lo cual se han creado plug-ins que se integran a Eclipse o Netbeans con el fin de facilitar el desarrollo, se basa en algunas funcionalidades como por ejemplo un emulador que permite escoger el modelo del terminal así como la versión del sistema operativo para el cual se creara la aplicación, posee asistentes para la creación rápida de aplicaciones, editores de código e “interfaces gráficas que permiten el desarrollo de componentes visualmente”[13].

El entorno que permite desarrollar aplicaciones para la plataforma Android es Eclipse, NetBeans, entre otros, y que para usarlo se debe instalar varios complementos que son de descarga gratuita, con el fin de brindar la funcionalidad necesaria para trabajar sobre la plataforma que ha de ser elegida en base a las necesidades de cada desarrollador.

El entorno de desarrollo Eclipse ofrece grandes ventajas, ya que es un software libre multiplataforma con soporte para numerosos lenguajes de programación, entre ellos Android. El (SDK) de Android también está disponible en otros sistemas operativos no tan populares como Mac OS X o Linux [9].

Android es una plataforma de código libre, por lo que cuenta con un (SDK) disponible para todo aquel desarrollador que lo requiera; incluye entre otros elementos, el conjunto completo de API que este sistema soporta. Para descargarlo, basta con visitar la web de Android:[21].

Antes de empezar con la creación de aplicaciones es necesario adecuar y configurar varias de las funciones que permiten este proceso para lo cual se inicia con Java Runtime Environment (JRE) y Java Virtual Machine (JVM) que permitirán que se ejecuten los códigos de programación de Java en el computador.

En el caso de que no se disponga de Eclipse instalado debe proceder a la descarga y posterior instalación la cual no requiere seguir pasos específicos sino más bien es muy intuitiva y sin complicaciones [9].

Una vez realizada la instalación es hora de integrar el SDK a Eclipse para esto, es necesario descomprimirlo. La ubicación de los ficheros resultantes no es de importancia, pero hay que tener presente el lugar exacto ya que en pasos posteriores se tendrá que recurrir nuevamente a ellas.

Posteriormente se ejecuta el SDK Manager en cual permite escoger los paquetes que se desea instalar, estos paquetes tienen información sobre las distintas versiones de Android librerías, emuladores, si se desea se puede instalar todo el paquete que incluye desde la primera hasta la más reciente versión pero es obvio que tardara mucho, lo más recomendable es escoger una versión no tan baja ya que algunos recursos podrían no estar habilitados como son el uso de sensores o periféricos ni una tan reciente ya que se corre el riesgo que la aplicación no se pueda ejecutar en dispositivos con versiones de Android inferiores lo cual conlleva una pérdida de equipos a los cuales se puede distribuir la aplicación [11].

Por ejemplo si la aplicación creada está pensada para ejecutarse en celulares como en tablets la versión de Android correcta para realizar la programación sería a partir de Android 3.0 (Honeycomb) ya que en ésta se implementó el soporte para dispositivos con pantallas grandes.

El último paso es integrar el ADT (Android Development Tools) a Eclipse, para que el entorno de desarrollo sea el propicio para la creación de aplicaciones para Android, la instalación de ADT se la realiza desde Eclipse, es decir que estando la ventana de Eclipse abierta se ubica la pestaña de Ayuda > Instalar Nuevo Software, y en la ventana que aparece ingresar la dirección <http://dl-ssl.google.com/android/eclipse/> [9]. Después de

seguir los pasos que indica la ventana se requiere reiniciar Elipse con lo cual la próxima vez que se lo ejecute se verán incluidos iconos en la barra de tareas referentes al desarrollo Android.

Android incluye un emulador de un dispositivo móvil virtual (AVD) Android Virtual Device que se ejecuta en una computadora, y que puede ser utilizado para el prototipo, desarrollo y pruebas de aplicaciones para dispositivos Android, previamente a la ejecución en un dispositivo real [11].



Figura 1.5 Emulador del dispositivo móvil virtual

El emulador imita todas las características de un dispositivo real, tanto de hardware, como de software, siendo capaz además, de utilizar diferentes versiones del sistema operativo, al igual que diferentes configuraciones de hardware. La única excepción, es que el emulador carece de la capacidad de realizar llamadas reales (aunque es capaz de emular los eventos que estas producen) [13].

El Emulador de Android está basado en QEMU (Emulador de Máquina), que ayuda a correr programas creados para un tipo de maquina en otra diferente, cuenta también con capacidades de virtualización [22].

Adicional a los programas descritos anteriormente, en éste último tiempo Google empieza a desarrollar un entorno que facilite la programación de aplicaciones llamado Android Studio que reúne muchas de las características que presentan las plataformas Eclipse y NetBeans, y tras pasar por las versiones “Beta” en la actualidad ya se cuenta con la versión completa que elimina algunos problemas de estabilidad. Debido a que el entorno que promociona Google está pensado en maximizar la experiencia de programación por parte de los desarrolladores presenta una interface más limpia y fácil de instalar, es así que por ejemplo que tras la descarga gratuita y posterior instalación no es necesario instalar el SDK manager por separado sino que automáticamente se inicia la descarga de los paquetes necesarios para iniciar con el desarrollo de las aplicaciones.

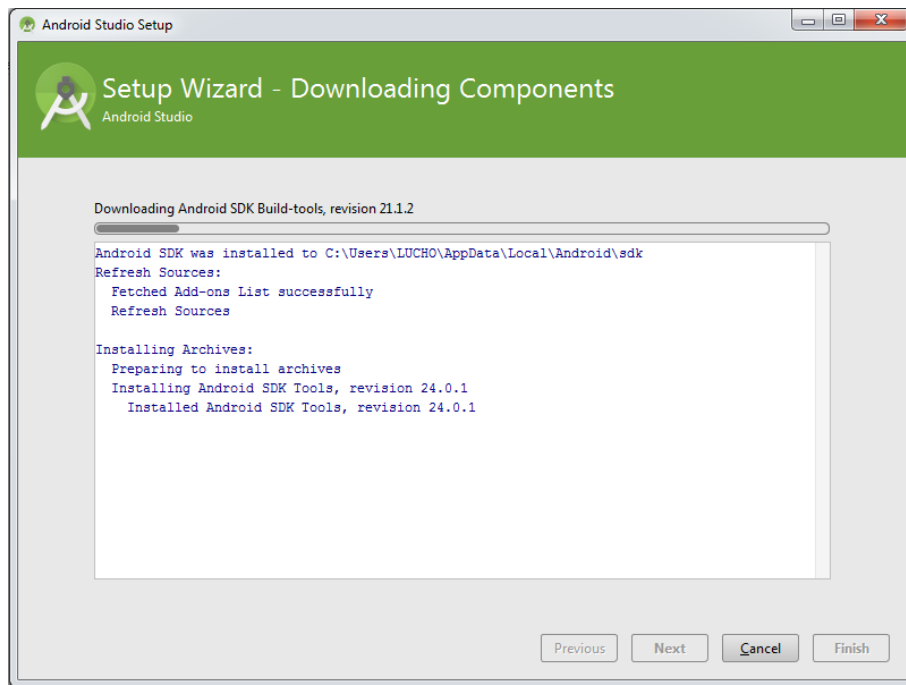


Figura 1.6 Instalación automática de Software Development Kit en Android Studio

1.4 Aplicaciones y servicios en dispositivos móviles

El desarrollo de aplicaciones para dispositivos móviles es la apuesta que muchas instituciones han tomado como un complemento a los servicios que prestan regularmente con la cual llegar a los usuarios de forma más directa y oportuna. Las aplicaciones móviles aprovechan los recursos físicos de los dispositivos (Portabilidad, conectividad, disponibilidad) informando de forma continua a los clientes e interactuando con sus

plataformas de servicios[23], la satisfacción de los clientes en muchos de los casos es indiscutible ya que de una u otra forma saben que las empresas se preocupan de mejorar continuamente y de poner en sus manos herramientas que faciliten su desenvolvimiento diario.

Las aplicaciones para dispositivos móviles sin lugar a duda llaman la atención de los usuarios que continuamente tratan de maximizar la experiencia de uso de sus dispositivos móviles, pero detrás de la presentación de una nueva aplicación al público existen diversos factores a tomar en cuenta por parte de los desarrolladores ya que exige una investigación previa. Esta investigación debe abarcar varios campos que empiezan con una necesidad común a la que se le quiere dar solución, plantear de forma concisa las variables que se requieren para la creación de la aplicación, la interface presentada al usuario de tal forma que sea atractiva y de fácil uso y comprensión [24].

Es importante que el desarrollador cuente con información clara de lo se quiere presentar en la aplicación ya que así se puede seguir un esquema o lineamiento ordenado para la creación de la aplicación, es por esta razón que se establecen pasos para la iniciar un proyecto para dispositivo móvil [13].

La figura 1.6 describe el lapso de tiempo para que el proyecto salga a la luz y pueda ser ofertado como un producto terminado, durante este tiempo se deben verificar cinco pasos ordenados secuencialmente que ayudan a la concepción, desarrollo, verificación de errores y corrección de los mismos [24].

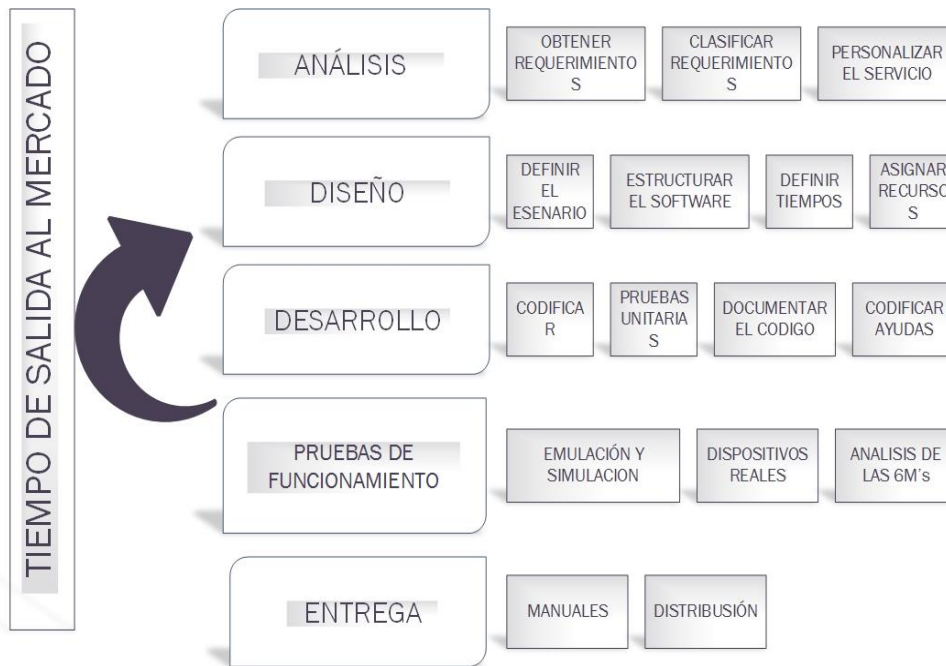


Figura 1.7 Metodología de desarrollo

Fuente: <http://tecnura.udistrital.edu.co/ojs/index.php/revista/article/view/767/752>

Cada uno de los niveles que se representan en la figura cumple una función específica y requiere un análisis individual de cada una de las partes que la conforman.

1.4.1 Análisis

Recopilar información concerniente a los atributos que debe tener la aplicación tiene gran repercusión ya que es el primer paso para tener una idea global de las características fundamentales que debe tener, esta labor se la realiza entre la persona encargada de la programación conjuntamente con el cliente con la finalidad de definir el entorno de la aplicación [25].

El entorno engloba todo el servicio y plataformas disponibles para que sea una realidad; es decir que necesariamente se debe conocer las particularidades de los dispositivos a los que va encaminado, el tratamiento de datos enviados y recibidos, gestión de bases de datos de ser el caso, etc.[26]. En fin es imperativo empaparse de información referente al terreno y campo de acción en el que se va a incursionar con el lanzamiento de la aplicación.

La interface entre el usuario y el dispositivo juega un rol primordial, el dará o no la aceptación y visto bueno a la aplicación en base a su experiencia de uso, una interface bien lograda y simplificada en su usabilidad será claramente mejor recibida que una pobre compleja. Por tal razón se debe clasificar los requerimientos y establecer prioridades, aquellos que sean más representativos obviamente deben tratarse con mucho pulso procurando eliminar dificultades de uso e interpretación superfluas.

La manera más eficaz de personalizar el servicio es incorporar acciones distintivas de aplicaciones populares con lo que el usuario notara cierta familiaridad y enriquecerá la experiencia de uso.

1.4.2 Diseño

Su nombre lo dice claramente, es transportar la idea obtenida tras el análisis a un esquema o diagrama que contemple la mejor alternativa para sacarle provecho a las características del dispositivo, redes a las que se conecta, el entorno, la sociedad, etc.[27].

Definir el escenario requiere establecer la interacción entre el móvil y servidores externos para actualizar información, y en función de las atribuciones que se le quiere dar a la aplicación será necesario mantener una conexión constante o en tal caso se pueden almacenar datos para ser enviados y compartidos el momento que se establezca una conexión [24].

Cuando se habla de la asignación de recursos en la fase de diseño se refiere al tiempo que invertido para el desarrollo, las herramientas utilizadas para la programación (software y hardware) que está ligada a la parte económica. Pero si se toma en cuenta las características técnicas de los dispositivos, la asignación de recursos ligada a la programación y el uso de periféricos[24], así como la prioridad para gestionar la rom y ram del móvil.

Y una parte muy importante para una aplicación es el tiempo en el que se pasa desde el planteamiento de la idea inicial hasta su presentación y posterior comercialización.

1.4.3 Desarrollo

Se refiere a la acción misma de la programación que requiere ser documentada, es recomendable colocar frente a cada línea de código una descripción básica de la instrucción contenida en dicha línea. Esta técnica será de gran ayuda en los casos en los que se requiera realizar corrección de errores[27]. Crear la documentación necesaria de cada acción servirá también para la redacción de ayudas y guías de usuario en el caso de ser necesarias y que además deja constancia de los procedimientos y secuencia en el desarrollo.

1.4.4 Pruebas de funcionamiento

Una vez culminada la etapa de desarrollo y programación es necesario comprobar que la aplicación se ejecuta de acuerdo a lo planteado en la idea original, es válido hacer pruebas de la experiencia de uso con personas que no están ligadas a la creación de la aplicación ya que así se puede tener idea de cambios o mejoramiento de aspecto, presentación y accesos rápidos a utilidades [27].

Como se muestra en la Figura 1.7 durante la etapa de pruebas se pueden presentar necesidades de cambio de programación por múltiples factores, y requiere regresar a la fase de diseño y repetir los pasos posteriores hasta que se el funcionamiento y desempeño sea el esperado y concuerde con la idea original.

El ciclo encerrado entre las pruebas, diseño y programación o desarrollo se lo puede hacer las veces que sea necesario pero hay que tener en cuenta que no es recomendable tardar mucho tiempo en esta fase ya que influye en el tiempo de puesta en el mercado.

1.4.5 Entrega

Es el punto final del proyecto y en el que está listo para ver la luz y ser distribuido en el mercado, la documentación correspondiente a manuales de usuario y ayudas son una parte primordial.

Como se ha visto en el desarrollo de aplicaciones móviles existen pasos específicos que van desde la concepción de la idea inicial y la culminación de ella pasando por las

respectivas pruebas de acuerdo a la función primordial de la aplicación. Para el caso de este proyecto se habla de una aplicación encaminada a la medición inteligente de energía eléctrica como un servicio adicional para los usuarios por parte de las empresas distribuidoras. Una aplicación desarrollada independientemente llamada CURB se acerca a la concepción de medición inteligente con servicios como medición en tiempo real, perfiles de ahorro, alarmas y mensajes referentes al consumo diario [28].



Figura 1.8 Servicios implementados en aplicaciones de medición eléctrica inteligente

Fuente: <http://energycurb.com/index>

La aplicación identifica un problema y muestra de forma comprensible los consumos dentro del hogar, al momento que el usuario tiene la opción de conocer de forma detallada los costos que representan algunos de sus hábitos de consumo puede tomar acciones correctivas para ajustar o prever gastos mensuales innecesarios.

CAPÍTULO II

PROTOTIPO DE LA APLICACIÓN

En este capítulo se trata de ahondar más en la construcción de la aplicación Android para medición inteligente, es por ello que tras obtener la información referente al sistema operativo para la que se está desarrollando la aplicación y las herramientas necesarias que permiten este cometido, es imperante definir algunos de los recursos necesarios para lograr el objetivo.

Cuando se habla de recursos se hace referencia en muchos casos a tecnologías existentes como son las conexiones inalámbricas disponibles en los dispositivos, equipos de recopilación de datos, servidores etc.

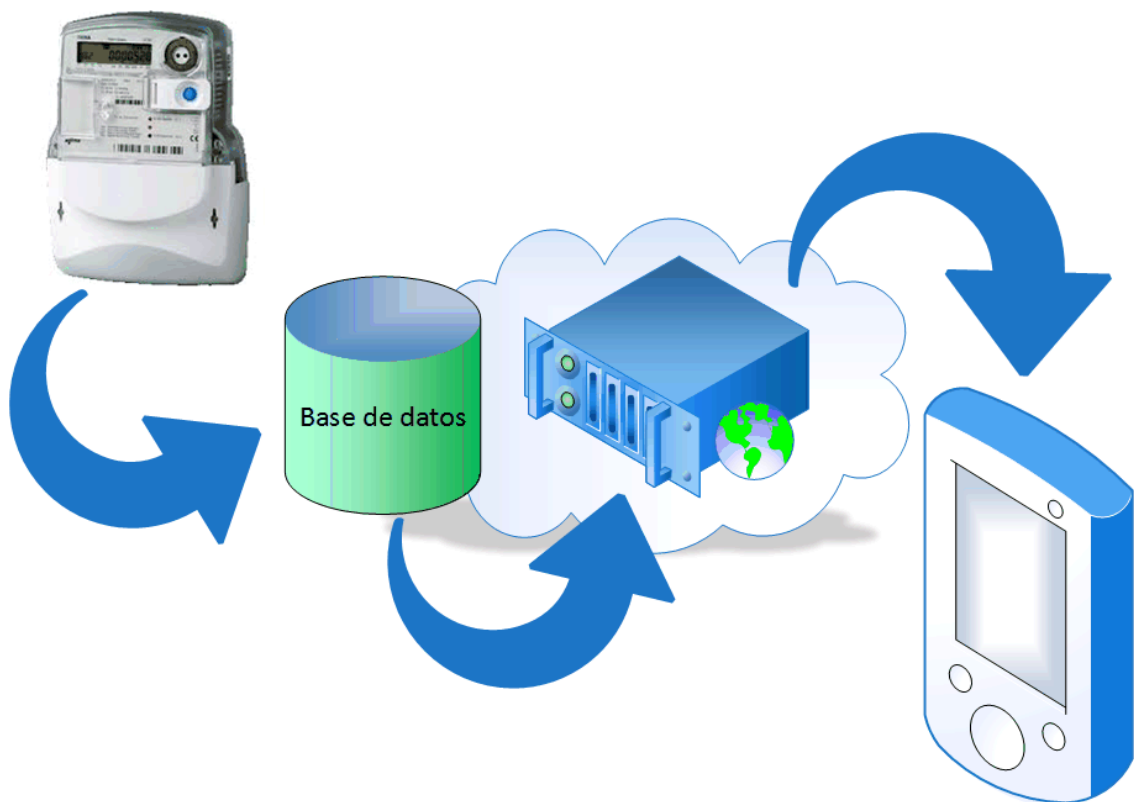


Figura 2.1 Modelo básico de funcionamiento de la aplicación

Fuente: Grupo de investigación

La Figura 2.1 muestra un esquema de lo que se pretende alcanzar con la creación y ejecución de este trabajo en la que las comunicaciones están siempre presentes, esto

debido a que los datos necesariamente deben ser subidos a la nube y por medio de canales de comunicación recobrarlos e interpretarlos de forma eficaz y concisa.

2.1 Conectividad inalámbrica WiFi

Se hace imperioso en este capítulo ahondar en el estudio de dichos canales conocer características y protocolos usados para establecer las comunicaciones y garantizar la entrega completa de los paquetes de datos, dentro de lo que compete a la aplicación móvil las tecnologías inalámbricas idóneas para las conexiones a la nube son la red telefónica por medio del internet proporcionado por las diferentes operadoras móviles y una conexión vía WiFi muy común hoy en día en muchos de los hogares del sector urbano o a su vez conexiones públicas y como por ejemplo centros comerciales [29].

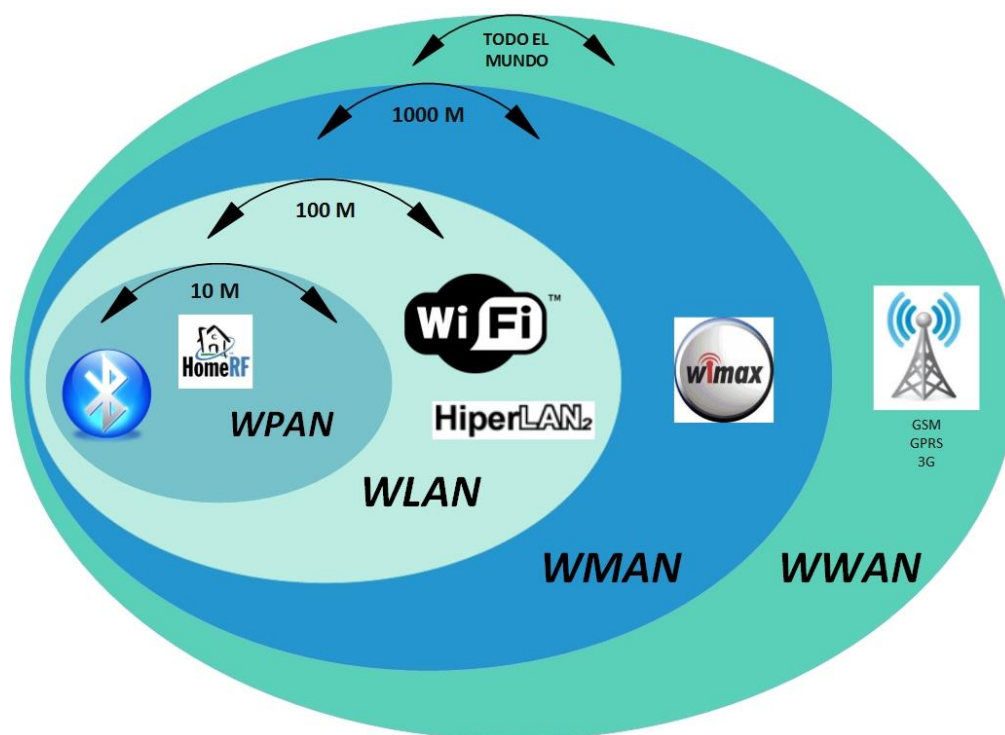


Figura 2.2 Alcance de redes inalámbricas

Fuente: <http://www.taringa.net/posts/info/4178992/Un-poco-sobre-Wi-Fi.html>

Fuente: <http://sistemas-distribuidos.wikispaces.com/2.4.+Plataformas+de+hardware>.

En la figura 2.2 se esquematiza el alcance y tecnologías inalámbricas más conocidas, en virtud de las necesidades de aplicación. A breves rasgos y por lo que se dijo anteriormente queda claro que tanto WLAN y WWAN[30] son las opciones recomendables para el desarrollo de la aplicación móvil, no obstante es conveniente

incluir información pertinente de las tecnologías representadas en la figura.

2.1.1 Wireless Personal Area Network (WPAN)

Las redes de comunicación personales son de bajo alcance es decir que para lograr un envío y recepción de datos, los equipos conectados tienen un limitante en la distancia existente entre ellos, generalmente es usada en conexiones de periféricos como manos libres, impresoras, teclados de computadora etc[30]. Dentro de este tipo de redes la que más destaca o más usada es Bluetooth aunque existen varias tecnologías destinadas a funciones específicas como pueden ser la domótica o control industrial.

Éste tipo de redes tienen la característica de presentar un bajo consumo de energía y velocidades de transmisión de datos llega como en el caso de bluetooth a un máximo de 1Mbps [31]. Se puede nombrar algunas redes (WPAN) que tienen gran potencial y otras que son muy comunes y cotidianas.

Zigbee es un protocolo de comunicación que trabaja bajo el estándar IEEE 802.15.5 que se categoriza dentro de las redes locales para en el ámbito profesional (WPAN). Este protocolo es usado mayoritariamente en aplicaciones de domótica en las que se requiere una red robusta que no corra riesgos de perder datos en su labor de emisión y recepción de datos, y que no consuman muchos recursos energéticos de la batería del dispositivo, la topología de la red a la que se puede implementar es de gran flexibilidad y permite las siguientes configuraciones: estrella, malla y árbol [32].

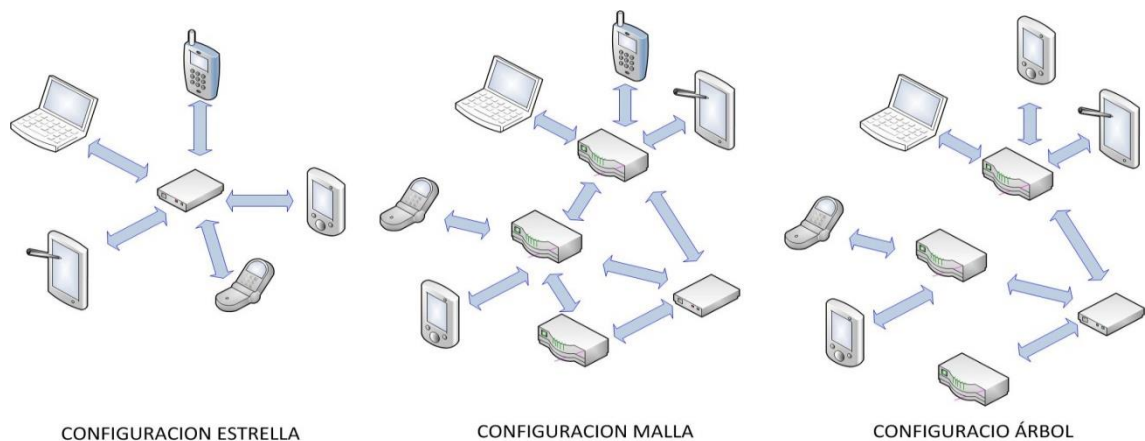


Figura 2.3 Topología de comunicación admitida en zigbee

Las configuraciones mostradas en la Figura 2.3 poseen ventajas y desventajas dependiendo del tipo de red a implementarse, y hace referencia a la configuración de los equipos y como se transfieren los datos por medio de ellos. En el caso de la configuración estrella los equipos o nodos únicamente tienen comunicación con la puerta de enlace pero no se pueden comunicar entre ellos, a diferencia de la configuración en malla en la que los nodos pueden establecer comunicación tanto entre nodos y con la puerta de enlace lo cual permite que la información tenga varios caminos para llegar de un punto a otro [31].

Aunque en la actualidad los sistemas IR (Infra Red) se limitan a controles remotos de electrodomésticos, algunos fabricantes de teléfonos celulares han optado por retomar esta tecnología e incluir emisores y receptores IR pero no con el afán de intercambiar información entre terminales ya que la transmisión de datos es lenta y requiere posicionar los equipos a una distancia corta para evitar interferencia debido a la luz del ambiente [30], sino para que por medio de una aplicación el Smartphone tenga la posibilidad de funcionar como un control remoto universal [33].

2.1.2 Wireless Local Area Network (WLAN)

Las redes (WLAN) permiten puntos de conexión más rápidos y abarcan mayor rango de conexión comprendido en las decenas de metros [34]. Sin que sea necesario amplificadores de la señal. El estándar inicial de esta tecnología fue IEEE 802.11 más conocido como WiFi (Wireless Fidelity) impulsado por Wireless Ethernet Compatibility Alliance (WECA) la cual promueve la estandarización de las comunicaciones inalámbricas (WLAN) con la finalidad garantizar la conectividad de dispositivos de distintos fabricantes y de la cual existen varias actualizaciones o variantes[35].

Estándar IEEE 802.11

Es la base misma con la que se empezó la estandarización y aunque hoy en día está obsoleta sentó los precedentes para las versiones posteriores, éste estándar operaba en la banda de 2.4 GHz transmitiendo los datos mediante la señal emitida por infrarrojo. La mayor debilidad que presentaba era que existía poca interoperabilidad entre equipos de

diferentes marcas lo cual hizo que sea relegado [34].

Estándar IEEE 802.11a

Este estándar al igual que los se hablará posteriormente han logrado una mayor acogida debido a la interoperabilidad que presenta, funciona a una frecuencia de 5 Ghz con velocidades de transmisión entre 6 y 54 Mbps y la cual opera en banda libre y una interferencia relativamente baja [36]. una de las desventajas que presenta es la necesidad de contar con línea de vista [37] lo cual conlleva la instalación de más puntos de acceso

Estándar IEEE 802.11b

Proporciona una mayor velocidad, que bordea los 5.5 Ghz y un ancho de banda de 1 Mbps a una distancia 90 metros y de hasta 11Mbps a una distancia de 32 metros, es usada en Acces Points (AP) en configuraciones punto y multipunto. Para maximizar la distancia de alcance este protocolo permite la implementación de antenas de alta ganancia logrando así que se conecten equipos distantes a unos 8 Km siempre y cuando exista línea de vista, cabe recalcar que la operación de las antenas opera en configuración punto punto [30].

Estándar IEEE 802.11g

Trabaja en la banda de 2.4Ghz y a una velocidad cercana a los 27Mbps y es compatible con el estándar IEE 802.11b aunque se ve reducida la tasa de transmisión de datos, además debido a la saturación que existe en la banda de 2.4 Ghz por parte de dispositivos Bluetooth y teléfonos inalámbricos es propensa a interferencias [30].

Estándar IEEE 802.11n

La llegada de este estándar inicia con el anuncio de la IEEE de crear un grupo con el fin de desarrollar una nueva versión de IEEE 802.11 original la cual promete alcanzar una velocidad de transición de hasta 500 Mbps real logrando así superar en 10 veces a los estándares a y g y en 40 veces al estándar b, la estructura de IEEE 802.11n se basa en

IEEE 802.11 e incorpora MIMO (Multiple-Input Multiple-Output) el cual usa para tal efecto multiples antenas y transmisiones para incrementar la transmisión de datos [30].

2.1.3 Wireless Metropolitan Area Network (WMAN)

El referente de esta red inalámbrica es WiMax aunque existen proyectos como el europeo denominado HiperMAN y coreano denominado WiBRO los cuales pretenden otorgar servicio banda ancha a zonas en las que las condiciones topológicas limitan la llegada de infraestructuras como xDSL y cable-modem. WMAN no se ha concretado ni es tan difundida ya que en algunos casos los problemas relacionados con la topología del área se ha solucionado con antenas amplificadoras basadas en el estándares de la familia IEEE 802.11 [38] aunque esta última parte es discutible ya que WiFi fue creada para (WLAN) y presenta algunas limitaciones.

Ahora bien surge la duda, ¿no es mejor usar tecnologías existentes como las que poseen las operadoras celulares? Y la respuesta es que lamentablemente el servicio de internet móvil en el país aún se maneja por monopolios y encarecen los precios a conveniencia.

Si bien existe existen muchas expectativas referentes a la implementación de WiMax debido a las promesas de proveer velocidades de conexión superiores a WiFi y celulares con bandas de frecuencia que oscilan entre los 2 y 66 Ghz y proveer conexiones con un ancho de banda de hasta 75 Mbps a través de largas distancias [39]. Sin embargo se requiere un estudio que arroje resultados favorables para la implementación de esta tecnología en una área metropolitana, pero uno de los posibles usos para este tipo de redes dentro del campo eléctrico es la tele medición como una parte de los servicios implementados es un SMART GRID. Es necesario recalcar que WiMAX está basado en el estándar IEEE 802.16 [39].

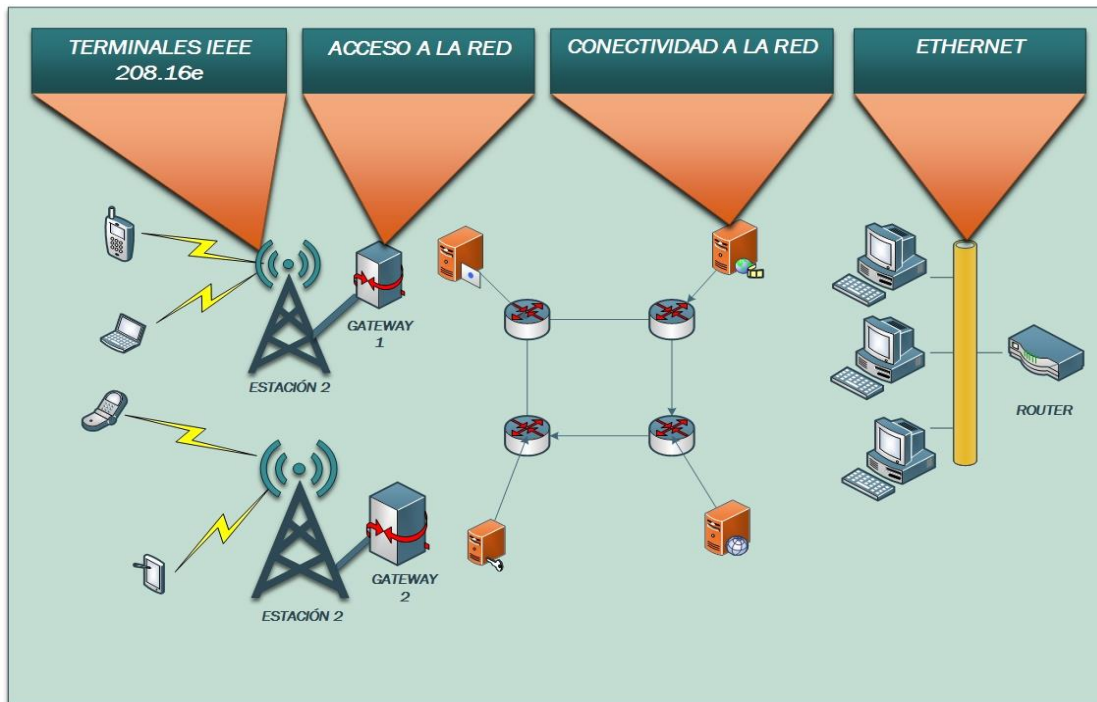


Figura 2.4 Arquitectura red WiMAX

Fuente: <http://inpressco.com/wp-content/uploads/2014/04/Paper84909-912.pdf>

2.1.4 Wireless Wide Area Network (WWAN)

Esta red abarca varias tecnologías que han aparecido con el pasar del tiempo pero que tienen una característica común la cual es permitir las comunicaciones de dispositivos a largas distancias, las mismas que pueden ser entre ciudades o países mediante el uso de satélites y antenas locales ubicadas estratégicamente para garantizar la cobertura del servicio [30].

Tecnología GSM (Sistema Global para las comunicaciones Moviles)

Es considerada como un estándar de segunda generación con la que es posible tener acceso a internet por medio de la red telefónica móvil así como la realización de llamadas telefónicas y envío de mensajes de texto, esta tecnología desplazo a su antecesora (CDPD) que a diferencia de ésta realizaba sus transmisiones por medios analógicos [30].

Tecnología GPRS (General Packet Radio Services)

Permite un mayor tráfico de datos que la tecnología (GSM) con velocidades que van

desde los 56 Kbps hasta 144 Kbps y mayores servicios como mensajes multimedia y WWW (Word Wide Web), está destinada para la transmisión de datos y las empresas proveedores facturan este servicio en base a la cantidad de información enviada [40].

Tecnología EDGE (Enhanced Data Rates for GSM Evolution)

La ventaja que posee es que es idónea para las aplicaciones en las que es primordial por así decirlo un mayor ancho de banda y mayor volumen de transmisión de datos como es el caso de videos, puede llegar alcanzar una velocidad de 384 Kbps. Esta tecnología es un enlace entre la segunda tercera generación de comunicaciones móviles y funciona sin mayores contratiempos con sistemas GSM que hayan incorporado GPRS [40].

Tecnología UMTS (Universal Mobile Telecommunication System)

Es también llamada la tercera generación de las redes celulares (3G) aunque realmente puede ser usada en otras aplicaciones y se concibió como la sucesora de la tecnología GSM debido a que ésta ya no podía actualizarse para brindar servicios más robustos como contenidos multimedia, video llamadas y conexiones a internet más rápidas como es el caso de (3G) que puede llegar hasta 2 Mbps a cada usuario. Adicional a esto la tecnología

3G es usada en módems portátiles o mediante un teléfono celular que comparte la conexión con una computadora simulando el trabajo de un router [30].

Tecnología HSPA (High Speed Packet Access)

Es la versión mejorada de UMTS ya que optimiza el uso del espectro radioeléctrico asignado a las operadoras, y está conformado por HSUPA (High Speed Upload Packet Access) y HSDPA (High Speed Dwonload Packet Access) que mejora la carga y descarga de datos.

EVOLUCIÓN SISTEMAS INALÁMBRICOS					
CARACTERÍSTICAS	1 G	2 G	2.5 G	3 G	3.5 G
ESTÁNDARES	AMPS Y TACS	GSM, IS-136, IS-95, IDEN y PDC	GPRS, EDGE e IS-95B	UMTS y CDMA2000	HSDPA, CDMA2000 (1xEVDO y 1xEV-DV)
TÉCNICA DE ACCESO AL MEDIO	FDMA	TDMA Y CDMA	TDMA	WCDMA	WCDMA
MODULACIÓN	–	GSMK	GSMK, 8-PSK	QPSK, 16QAM	QPSK, 16QAM
SERVICIOS	VOZ	VOZ Y SMS	VIDEO, E-MAIL, NAVEGADOR DE INTERNET	STREAMING, VIDEO, MMS, APLICACIONES MULTIMEDIA	VO-IP, VIDEO CONFERENCIAS
TRANSFERENCIA MÁXIMA DE DATOS	–	–	GPRS (160 Kbps) y EDGE (473.6 Kbps)	2 Mbps	14 Mbps

Tabla 2.1 Resumen de evolución de las redes inalámbricas móviles

Fuente: <http://tesis.bnct.ipn.mx:8080/bitstream/handle/123456789/12977/TESIS.pdf?sequence>

=1

2.2 Internet y servidor web

Para dar inicio a ésta etapa es necesario recopilar las ideas fundamentales para estructurar y dar forma a la aplicación, para lo cual se siguen los puntos que se enumeran a continuación.

- Casos de uso
- Base de datos
- Flujograma
- Interfaces

2.2.1 Casos de uso

Los casos de uso permitirán definir el funcionamiento del sistema en base a los requerimientos del usuario y su respectiva interacción con la base de datos, es así que tomando en cuenta las funciones que se quiere proveer en la aplicación, la figura 2.5

muestra los casos de uso de la aplicación.

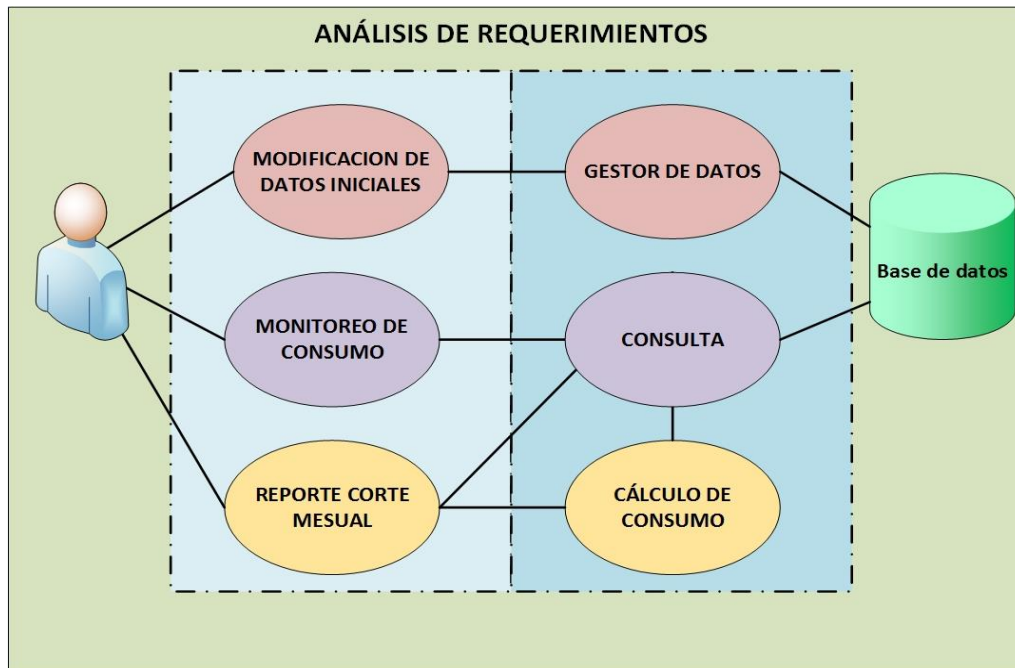


Figura 2.5 Casos de uso de la aplicación

Fuente: Grupo de investigación

Por medio de la aplicación Android se requiere tener acceso a una base de datos la cual alberga información del usuario así como datos relacionados al medidor asignado para la medición de consumo de energía, adicional a esto también reposan los datos de consumo enviados por el medidor inteligente los cuales son extraídos y presentados al usuario de forma entendible al momento de realizar la consulta a la base de datos.

La gestión de los datos y posterior presentación es realizada dentro de la aplicación de conformidad a la forma de programación que se realiza, esta programación contempla la presentación de gráficos evolutivos del consumo, cortes mensuales de consumo y notificaciones para optimizar el consumo de la energía

Por la concepción de la aplicación en la que se está trabajando se requiere necesariamente que la Tablet o teléfono inteligente ingrese a una WLAN para que pueda tener acceso a los datos proporcionados por el medidor inteligente, es importante que la aplicación organice de forma coherente la información obtenida para presentarla de forma clara y sin correr el riesgo de pérdida de datos o de una mala interpretación.

Es por esta razón que en la medida de lo posible se debe procurar que el acceso al internet cuente con las medidas de seguridad pertinentes para proteger la información allí almacenada lo cual conlleva a conocer más a fondo el funcionamiento de la internet.

Inicialmente el internet fue pensado como un medio de ayuda para fines científicos con la cual ingenieros e investigadores tenían la opción de conectarse a equipos remotamente [41]. Actualmente el internet no se limita a la idea inicial sino que está prácticamente en todos los ámbitos de la vida cotidiana que van desde la investigación en centros de estudio pasando por industrias y cimentada con gran fuerza en los hogares de la gran mayoría de las personas, es por esta razón que en los últimos años se ha acuñado la frase “La era de la información” que no está desapegada a la realidad ya que en todo el mundo y a toda hora es subida a la internet información concerniente a diversos temas que pasa a estar a mano de quien lo requiera.

Se convierte así en el medio más eficaz para compartir información sin importar la distancia entre quien requiere y quien genera el conocimiento o temas de interés.

Pero más allá de las ventajas y servicios que se obtienen por medio del internet, se congregan un gran número de conceptos que hacen posible el funcionamiento de la web, reflejados en flexibilidad que presenta para enlazar e interpretar la información, el éxito se da gracias a la incorporación de dos protocolos que han sido los pilares fundamentales en la evolución de la web, dichos protocolos son HTTP (Hípertexto Transfer Protocol) y HTML (Hípertexto Mark-up Protocol) [42].

HTTP ofrece la posibilidad de implementar de manera fácil y sencilla un sistema de comunicaciones para el envío de cualquier tipo de ficheros logrando así que servidores de potencias limitadas logren atender miles de peticiones sin que esto afecte sus costos de despliegue, y HTML ofrece un método de enlace de páginas simple y muy eficiente por medio de marcas al interior del texto pudiendo así referenciar otros recursos como enlaces a otros documentos, imágenes, etc.

Si se compara los servicios que presentan ahora las aplicaciones de la red, están muy alejadas a lo que fueron en sus inicios ya que en un principio la web no era más que un

repositorio de documentos estáticos los cuales estaban disponibles para su consulta o descarga. La evolución se da gracias a la incorporación de CGI (Common Gateway Interface) ayudando así a que las páginas soportasen la presentación de contenido dinámico por medio de mecanismos que ayudasen a pasar la información entre los servidores HTTP y programas externos sin importar el lenguaje de programación escogido por el desarrollador.

Pese a las ventajas iniciales que presentaba la incorporación de CGI se detectó un punto que consistía básicamente en una sobrecarga al servidor debido a la necesaria ejecución de un Run-Time Environment por ejemplo Java.

Una de las opciones más aplicadas en internet es el lenguaje interpretado por un servidor PHP, este lenguaje inserta dentro del programa el protocolo HTML.

2.2.2 Servidor web

“Un servidor web es un programa que atiende y responde a las diversas peticiones de los navegadores, proporcionándoles los recursos que solicitan mediante el protocolo HTTP o el protocolo HTTPS (la versión segura, cifrada y autenticada de HTTP). Un servidor web básico tiene un esquema de funcionamiento muy sencillo, ejecutando de forma infinita el bucle siguiente:”[42].

- Espera las peticiones en el puerto TCP (puerto 80).
- Recibe la petición.
- Busca el recurso en la cadena de petición.
- Envía la respuesta por la misma conexión por la que recibió la petición.
- Regresa al punto dos y realiza la secuencia nuevamente.

Si el servidor web que ejecuta los pasos anteriormente descritos se ajusta a los requerimientos mínimos de un protocolo HTTP con la restricción de que lo haría únicamente para ficheros del tipo estáticos.

Las características que definen a los servidores web obviamente son más extensas debido a que los programas servidores de HTTP son variados, y que se diferencian en el

tipo de peticiones que pueden atender.

Los servidores web deben tener la capacidad de atender ficheros estáticos como requerimiento básico, así como la posibilidad de especificar que parte del disco es la que se servirá, la gran mayoría de servidores proveen seguridades para especificar que usuarios, direcciones, pueden ver o revisar un directorio específico.

La manera más sencilla es de asegurar la seguridad es la inclusión de ficheros .htaccess en el que se especifican que máquinas, usuarios, etc., tienen acceso a los directorios y subdirectorios.

La capacidad de soportar y servir contenido dinámico es una parte primordial al momento de escoger un servidor web, por el simple hecho de que la mayor parte contenido no proviene de páginas estáticas sino que más bien se generan constantemente. El servidor web debe además estar en la capacidad de soportar algunos lenguajes de programación (PHP, JSP, ASP, etc.)

2.2.3 Servidor Apache

Apache es un servidor web ampliamente usado por programadores, esto gracias a su robustez y las ventajas ligadas al código libre, como son la detección y corrección de errores oportunamente. Gracias a módulos de autenticación, autorización y control de acceso al web service provee una buena base para la seguridad del sistema

2.2.4 PHP

Es un lenguaje de programación desarrollado para la creación de páginas web para contenido dinámico, requiere de un servidor web con un módulo procesador de PHP para ser interpretado. Tiene acceso a información almacenada en bases de datos pero destaca la facilidad de enlace con MySQL 2.4 Recuperación de datos.

2.3 Recuperación de datos.

La recuperación de datos se la realiza por medio de comandos que básicamente hacen la petición desde la programación del cliente, los resultados de esa petición de los

representa como tablas, la sentencia antes mencionada es SELECT, al igual que ésta MySQL incorpora funciones tanto propias como auxiliares con la finalidad realizar procesos internos, es decir permite que el gestor manipule resultados en base a la programación asignada.

Para que se realicen los procesos el programador debe hacer uso de condicionantes u operadores lógicos de tal manera que el programa verifique y compare las condiciones de verdad y se ejecuten las operaciones necesarias, al igual que en la mayoría de plataformas de programación, MySQL incorpora las mismas funciones lo que varía en algunos casos no será más que la representación al momento de programar.

Algunos de los comandos que más se usan sobre las bases de datos se presentan en la Tabla 2.2.

Comando	Descripción
show databases;	Mostrar las bases de datos
select database();	Mostrar las bases de datos seleccionadas
show tables;	Muestra las pablas que contiene una base de datos
use nombre_base;	Seleccionar una base de datos
describe nombre_tabla;	Describir la estructura de campos de una tabla

Tabla 2.2 Consultas generales sobre las bases de datos

Fuente: <http://ggyma.geo.ucm.es/docencia/documentos/informatica/Comandos-mysql.pdf>

A cada sentencia o comando se le asigna un punto y coma al final, esto es para indicar que la sentencia termina e indica que se debe proceder a ejecutarlo.

Para la creación de las bases de datos se puede hacer uso de comandos pero MySQL incorpora una interface gráfica que facilita dicha tarea, Workbench es la funcionalidad de la que se habla aunque no es la única.

Cuando ya se tiene creada la base de datos con las respectivas tablas es necesario establecer la forma y los comandos con los cuales hacer los requerimientos para que la

base de datos devuelva los valores pedidos, estos comandos tienen una estructura básica y dependiendo la información que se quiera recuperar existirán varias posibilidades.

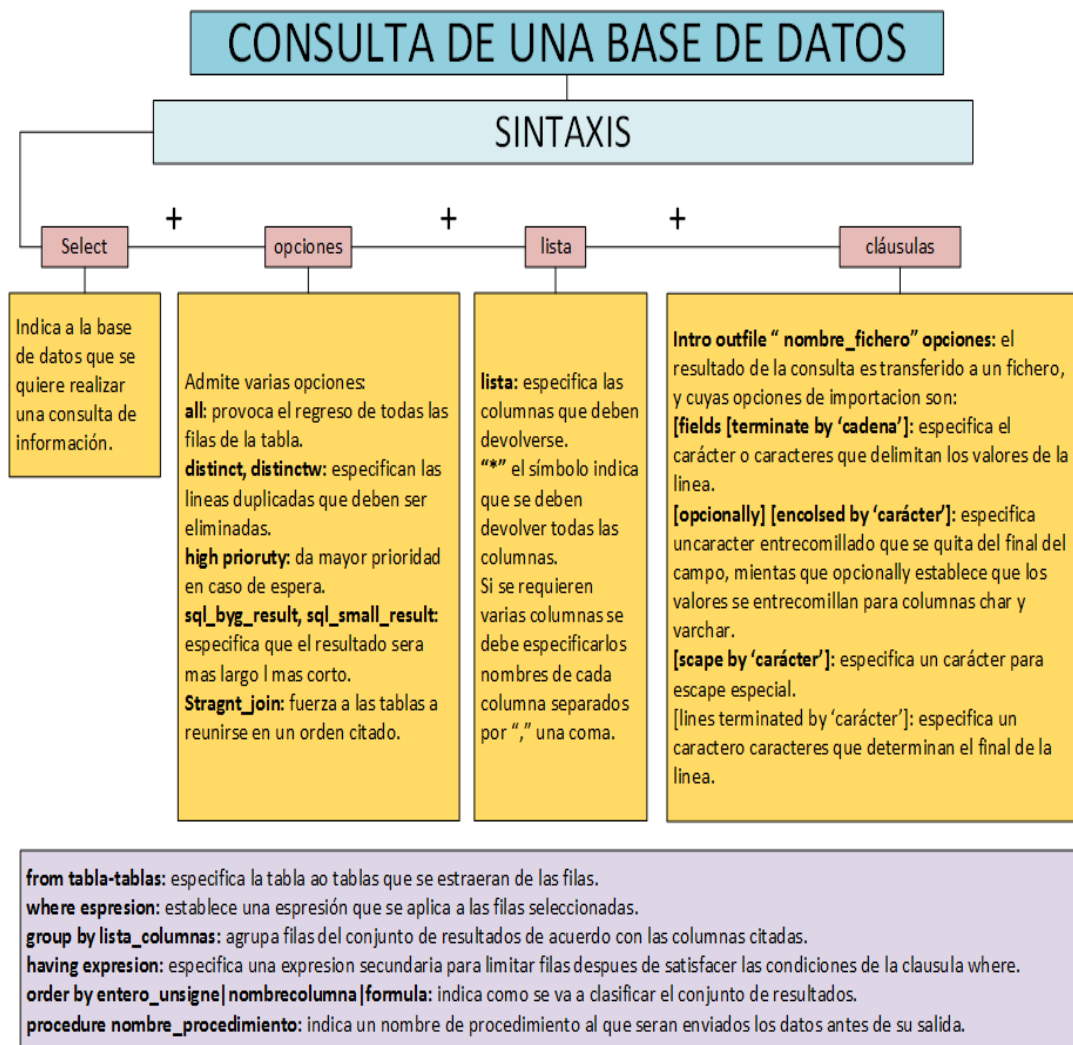


Figura 2.6 Comandos necesarios para consulta de Bases de datos

Fuente: <http://ggyma.geo.ucm.es/docencia/documentos/informatica/Comandos-mysql.pdf>

2.4 Organización y presentación de datos.

2.4.1 Base de datos

Las bases de datos permiten almacenar información de una forma estructurada con la finalidad de ponerla a disposición de un usuario o muchos usuarios, garantizando la seguridad de los datos almacenados, para la creación de una base de datos se requieren varias herramientas y conceptos asociados a esta tecnología.

Una parte muy importante en el desarrollo de las bases de datos es el uso de los Sistemas Gestores de Bases de Datos (SGBD) como por ejemplo MySQL cuya ventaja radica en que su libre distribución, es decir trabajan bajo la licencia de Software libre además posee características de uso simplificado y potencia que lo hacen llamativo para soluciones comerciales y de entretenimiento [43].

Características de MySQL

- Gran parte de su lenguaje de programación es realizado en SQL, provee gran seguridad con la conectividad de la base de datos
- MySQL, a través de tablas estructuradas, está en la capacidad de administrar tareas simples o muy complejas.

En el presente caso la base de datos contiene tres tablas, denominadas de la siguiente manera: “cliente”, “consumo” y “factura”, cada una de las tablas a su vez, está constituida por varios campos con características propias y relacionadas con el tipo de dato que va almacenar.

2.4.2 Tabla Cliente:

Clie_id: es un campo primario de la tabla el cual tiene la característica de ser irrepetible dentro y sirve para establecer la relación entre tablas, el tipo de variable que almacena es “INT” con una dimensión de 11 caracteres.

Clie_cedula: campo en el cual reposa el número de cedula del cliente, es del tipo “VARCHAR” de longitud (10).

Clie_apellidos: almacena los apellidos del cliente, de longitud (64) tipo “VARCHAR”.

Clie_nombres: almacena los nombres del usuario, de longitud (64) tipo “VARCHAR”.

Clie_medidor: almacena los datos de medidor como son; número de longitud (16) tipo “VARCHAR”.

Clie_direccion: almacena la dirección del cliente, de longitud (64) tipo “VARCHAR”.

Clie_usuario: almacena el “Nick” que usa el cliente para ingresar a la aplicación, de longitud (16) tipo “VARCHAR”.

Clie_password: almacena la contraseña que debe verificarse para ingresar a la aplicación, de longitud (8) tipo “VARCHAR”.

Clie_telefono: almacena el número de teléfono del usuario, de longitud (10) tipo “VARCHAR”.

Clie_suministro: almacena el número de suministro proporcionado por la empresa distribuidora, de longitud (10) tipo “VARCHAR”.

Clie_valor_notif: almacena el valor referencial a en base al cual se presentan las notificaciones, de tipo “double”.

Cada uno de los campos descritos anteriormente, poseen privilegios “SELECT”, “INSERT”, “UPDATE” Y “REFERENCES”, necesarios para recuperar, insertar nuevos registros en la tabla existente, actualizarlos y referenciarlos para así extraer su información.

2.4.3 Tabla Consumo

Cons_id: campo primario de la tabla, con atributos de auto incremento, tipo “BIGINT” (20)

Clie_id: hace referencia id principal de la tabla cliente.

Cons_fecha: toma la fecha y hora desde el sistema cada 15 minutos y lo inserta en la tabla, no es posible modificar la longitud del dato, la variable es del tipo “TIMESTAMP”

Cons_horas: representa los minutos en horas para proporcionar la variable necesaria para el cálculo de la potencia y es del tipo “DOUBLE”

Cons_corriente: almacena un valor que es generado aleatoriamente por una función insertada en base de datos, del tipo “DOUBLE”.

Cons_kwh: almacena el resultado de la corriente, voltaje y el tiempo transcurrido en horas, variable tipo “DOUBLE”.

Cons_costo: guarda el valor a pagar en base a la energía consumida y los valores establecidos en pliegos tarifarios, variable tipo “DOUBLE”.

Cons_voltaje: refleja el valor del voltaje para un servicio monofásico del tipo “DOUBLE”

2.4.4 Tabla factura

Fact_id: campo primario de la tabla factura con propiedades de autoincremento,

variable de tipo “INT” de dimensión (11).

Clie_id: hace referencia id principal de la tabla cliente.

Fact_anio: almacena el año de una factura pagada o por pagar, variable del tipo “VARCHAR” de dimensión (4).

Fact_mes: corresponde al mes de la factura emitida, variable de tipo “VARCHAR” de dimensión (2).

Fact_consumo: refleja la cantidad de Kwh consumidos en un mes “DOBLE”

Fact_valor: guarda el valor a pagar en base a la energía consumida y los valores establecidos en pliegos tarifarios, variable tipo “DOUBLE”.

Fact_estado: muestra si la factura esta o no cancelada. “VARCHAR” (2).

2.4.5 Flujoograma de la aplicación:

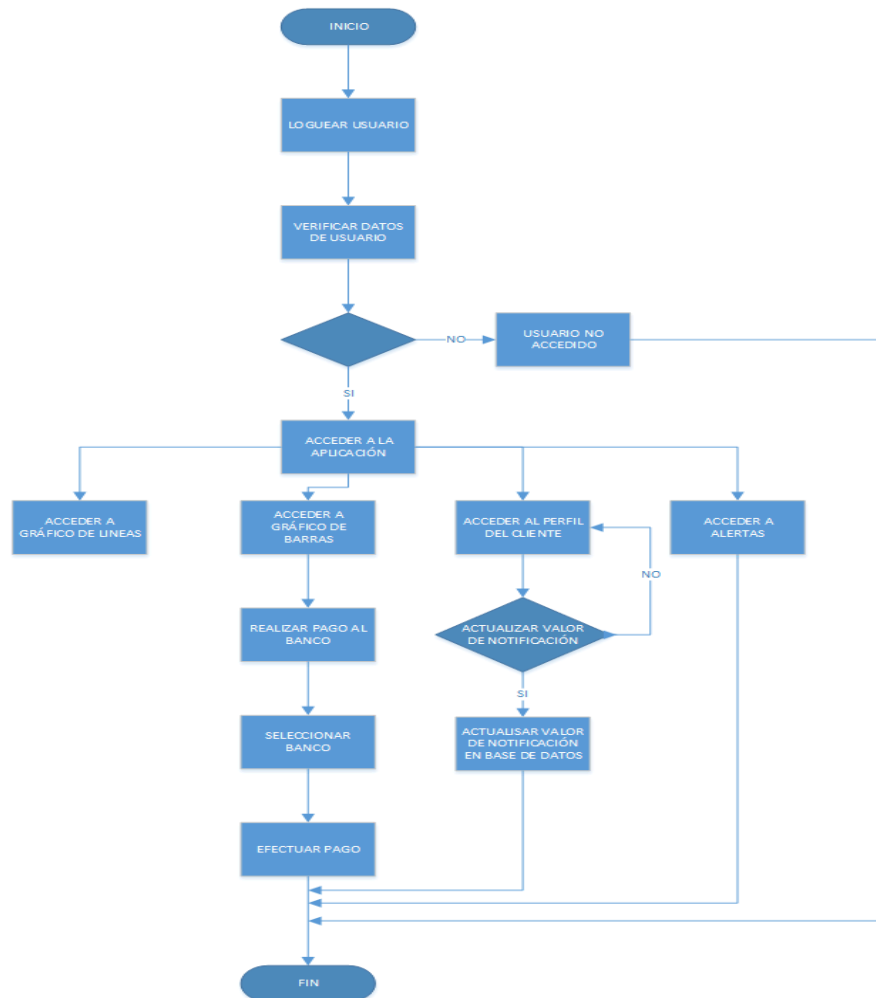


Figura 2.7 Flujoograma de la aplicación

Fuente: Grupo de investigación

CAPÍTULO III

MEDICIÓN INTELIGENTE DE ENERGÍA ELÉCTRICA

3.1 Hogares y ciudades inteligentes

El comienzo de los hogares inteligentes surgió hace algún tiempo en Japón y Estados Unidos con la implementación de la domótica, que no es más que la ciencia que estudia la factibilidad de las edificaciones y permite el control de sus equipos electrodomésticos desde una distancia determinada.

Una casa inteligente es la que realiza acciones que acreditan este planteamiento, llevando a cabo diseños arquitectónicos propios de ellas tales como estar fresca en verano y tibia en invierno, ahorra electricidad de manera programada y la que de forma general controla sus aparatos eléctricos a distancia, que sin duda deben ser equipos basados en las nuevas tecnologías de la información y las comunicaciones.

Esta tecnología puede implementarse en departamentos de grandes ciudades y zonas rurales, haciendo de estos lugares centros de éxito y que sus habitantes disfruten de grandes comodidades.

Son muchas las experiencias tangibles de los lugares donde está implementada esta tecnología que muestran resultados satisfactorios, pues son varias las ventajas que aportan las viviendas con esta tecnología, resultando algunas de ellas las que se muestran a continuación [44]:

- Las viviendas creadas se irán adaptando al nivel de vida de sus propietarios y a medida que pase el tiempo podrán ser asequibles por la mayoría de la población.
- Estas casas llevarán la cuenta exacta del consumo energético y permitirán una evaluación continua de la tecnología utilizada junto a la medida en que son efectivas en el hogar.
- Cuentan con equipos sofisticados que le brindan una gran seguridad a sus propietarios como son: alarmas para intrusión y pánico, control de fuego y humos, vigilancia interna y remota, etc.

- Favorece la disminución del consumo de energía ya que provee una gama de controles como el de la iluminación de la vivienda y gasto de los equipos eléctricos, contribuyendo a lograr un menor consumo eléctrico de la vivienda y protección del medio ambiente.
- Ofrece una gran comodidad a través de la utilización de equipos climatizados.
- Posee áreas estratégicamente creadas para mantener la limpieza del hogar.
- Ayuda a organizar las funciones dentro de la vivienda facilitando a los dueños tener más tiempo libre y emplearlo en otras funciones.

Aunque el precio de estas instalaciones no es asequible para la mayoría de la población, los especialistas en domótica trabajan para reducir los mismos y que la mayoría de los ciudadanos puedan contar con una instalación cómoda y lista para su uso.

Las infraestructuras urbanas, por su parte, para dar respuesta a estos cambios deben enfocarse en reunir condiciones que ofrezcan respuesta a tales desafíos, como son: deficiencia de agua y energía necesaria, contaminación del medio ambiente, tránsito congestionado por la existencia de pocas vías, hurto, arrojar desechos en lugares no establecidos para hacerlo, restauración de instalaciones urbanas en mal estado a causa de poca preocupación por parte de los directivos del país encargados de esas funciones entre otras. Es por ello que se debe trabajar por una cultura próspera, así como una economía sostenible en todas las ciudades, y que la preocupación por reparar y mejorar estos daños sea de toda la población, es decir una concientización global.

En el plano empresarial también se puede plantear que una ciudad inteligente se adecua a las prioridades y necesidades de las instituciones y de sus trabajadores en diferentes áreas, tales como social, ambiental y sobre todo económica, para que así los empleados sientan placer al trabajar y de esta forma mejorar su rendimiento y obtener mayores resultados que beneficien a todos.

Para lograr una ciudad inteligente es necesario ante todo elaborar un plan de desarrollo donde se establezcan metas reales de la visión que se pretende. Dentro de estas no puede faltar la utilización de la tecnología, el cual es un factor determinante al hablar de desarrollo, y para ello se debe conocer cuál es la tecnología más idónea para

desarrollar funciones en la sociedad.

Existen equipos diseñados para estos hogares que ayudan a desempeñar funciones específicas en las viviendas de forma óptima. Tal es el caso de los medidores inteligentes los cuales permiten que los usuarios realicen las medidas de las lecturas de su medidor de forma remota y emiten informaciones detallando el cálculo del uso de los combustibles utilizados, así como el agua y la electricidad en la vivienda. Otra de sus ventajas es que mediante ellos se puede realizar cortes y desconexiones eléctricas en caso de superar el cálculo del valor estimado en cuanto al consumo eléctrico en el hogar[45].

Haciendo un análisis de estos equipos, se pretende que cualquier sistema de medición inteligente que sea implementado, además de registrar el consumo instantáneo, debe monitorear eficientemente otras variables como la calidad de la energía que se emplea en el hogar junto al amperaje demandado, y consecuentemente que estos valores sean transmitidos a la empresa eléctrica como un aviso en caso de algún inconveniente, para en base a estos valores plantear una estrategia de ahorro de energía.

3.2 Desarrollo de la aplicación de medición inteligente

Para la creación de la aplicación de medición inteligente <<Monitor EE>> se desarrollaron una serie de diagramas de “casos de uso” para describir el negocio y cada proceso que se genera en él.

Mediante el Lenguaje de Modelado Unificado (UML) se representan los casos de uso a través de una notación gráfica integrada por actores, relaciones y acciones a realizar para representar los mismos, siendo este uno de los lenguajes de modelado de sistemas de software más utilizado hoy en día.

Se puede agregar además que UML es un lenguaje de visualización gráfica, utilizado para representar, definir, construir y documentar un sistema. En él se describe mediante gráficos, aspectos conceptuales tales como procesos de negocio, conocido como casos de uso de negocio, funciones del sistema, conocido como casos de uso del sistema y aspectos relacionados como expresiones de lenguajes de programación y diseños de bases de

datos.[46]

Los casos de uso implementados en el desarrollo de la aplicación fueron los siguientes:

1. Adquirir aplicación Monitor EE.
2. Consultar aplicación Monitor EE.
3. Registrar usuario.
4. Consultar consumo eléctrico.
5. Realizar pago-banco.
6. Registrar datos de usuario.
7. Comparar consumo eléctrico.
8. Revisar notificaciones.

En los diagramas de caso de uso del negocio realizado intervine un solo actor que es el usuario, ya que es él, el protagonista de todas las acciones que encierran el uso de la aplicación Monitor EE. Los casos de uso citados no son más que los procesos de negocios realizados por él mismo para el trabajo con la aplicación. A continuación se exponen los casos de uso empleados para el desarrollo de la aplicación:

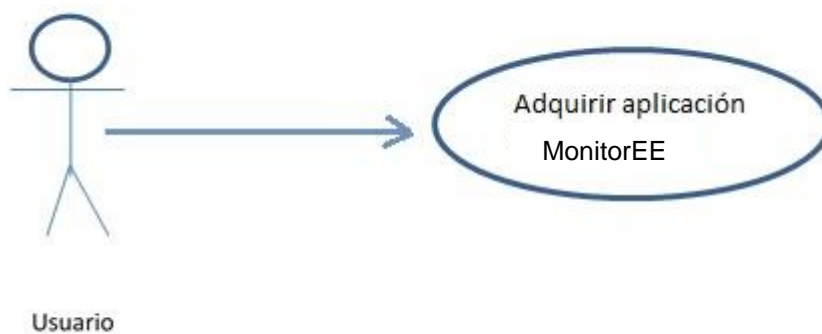


Figura 3.1 Diagrama de caso de uso Adquirir aplicación Monitor EE.

Fuente: Grupo de investigación

El diagrama que se cita a continuación es el diagrama de caso de uso consultar

aplicación Monitor EE que involucra a los restantes casos de uso del negocio, dado que al consultar la aplicación se generan las otras funciones que deben ser seleccionadas por el usuario.

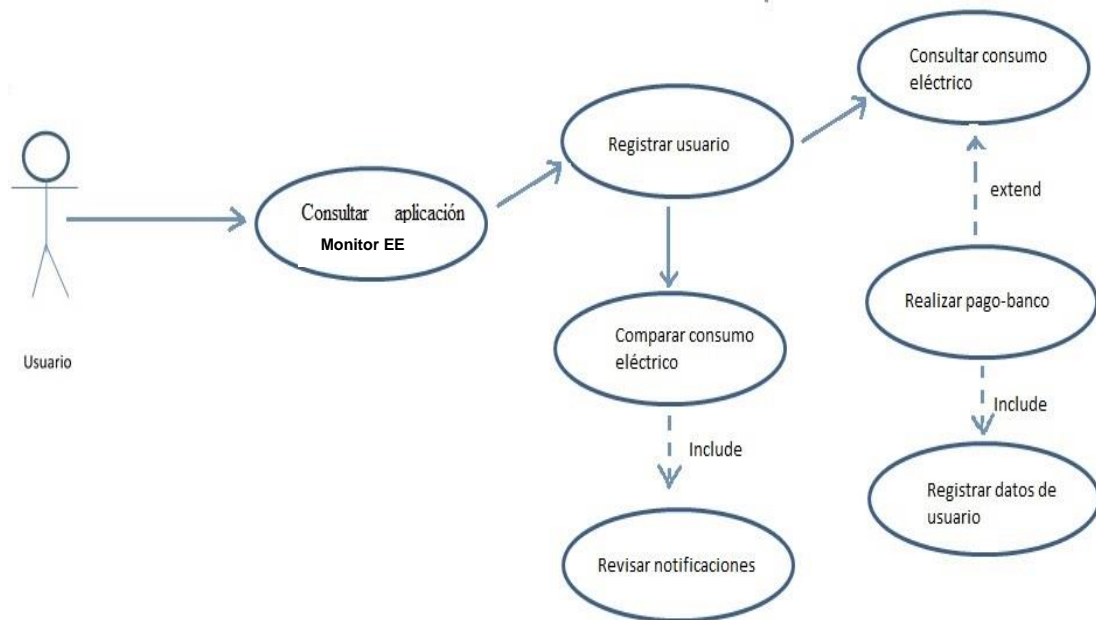


Figura 3.2 Diagrama de caso de uso consultar aplicación Monitor EE.

Fuente: Grupo de investigación

Un caso de uso es una representación gráfica de los pasos o las actividades que deberán realizarse para llevar a cabo algún proceso. A continuación se describe cada uno de los casos de usos identificados en el negocio:

Nombre:	Adquirir aplicación Monitor EE	
Actor:	Usuario	
Descripción:	Proceso de adquirir la aplicación Monitor EE.	
Flujo Principal:	Eventos ACTOR	Eventos SISTEMA
	1. Ingresar a google play vía wifi o en conexión directa desde el movil y buscar la aplicación.	1. El sistema verifica que está conectado y muestra las aplicaciones Android disponibles.

	2. Seleccionar la aplicación Monitor EE y la descarga al móvil	2. El Sistema Confirma el proceso realizado.
		3. El sistema envía al usuario un nombre de usuario y contraseña para posterior uso en la aplicación
Alternativa:		
Precondición:	El usuario debe conectarse	
Postcondición:	El usuario tiene lista la aplicación para ser usada en el móvil.	

Tabla 3.1 Descripción de Caso de Uso: Adquirir aplicación Monitor EE.

Fuente: Grupo de investigación

Nombre:	Consultar Aplicación Monitor EE	
Actor:	Usuario	
Descripción:	Proceso de consultar la aplicación Monitor EE desde el móvil.	
Flujo Principal:	Eventos ACTOR	Eventos SISTEMA
	1. El usuario entra en el móvil y selecciona la aplicación Monitor EE	1. El sistema verifica el proceso que quiere realizar.
		2. El Sistema autoriza el proceso realizado.
Alternativa:		
Precondición:	El usuario debe tener la aplicación Monitor EE en el móvil	
Postcondición:	El usuario está listo para registrarse con el nombre de usuario y contraseña que le fueron asignadas.	

Tabla 3.2 Descripción de Caso de Uso: Consultar Aplicación Monitor EE.

Fuente: Grupo de investigación

Nombre:	Registrar Usuario	
Actor:	Usuario	
Descripción:	Proceso de registrar usuario en la aplicación Monitor EE.	
Flujo Principal:	Eventos ACTOR	Eventos SISTEMA
	1. El usuario entra en la aplicación automáticamente le pide al mismo que se registre con el nombre de usuario y contraseña dada para el proceso	1.El sistema verifica que los datos y están correctos
		2. El Sistema Confirma el proceso realizado
Alternativa:	1. El usuario se equivoca al registrarse con sus datos	1. El sistema no acepta los datos y pone en la interfaz que verifique los mismos hasta que sean válidos.
Precondición:	El usuario debe entrar en la aplicación Monitor EE.	
Postcondición:	El usuario está listo para seleccionar una de las funciones de la aplicación.	

Tabla 3.3 Descripción de Caso de Uso: Registrar Usuario

Fuente: Grupo de investigación

Nombre:	Consultar consumo eléctrico	
Actor:	Usuario	
Descripción:	Proceso de consultar el consumo eléctrico de su vivienda en la aplicación Monitor EE.	
Flujo Principal:	Eventos ACTOR	Eventos SISTEMA

	1. El usuario entra en aplicación y selecciona consultar consumo eléctrico	1. El sistema le muestra el flujo de energía mediante dos gráficas: Consumo en los últimos 6 meses y consumo en el instante.
Alternativa:		
Precondición:	El usuario debe registrarse con su nombre de usuario y contraseña que le fueron asignadas.	
Postcondición:	El usuario tiene en su interfaz el consumo eléctrico que consultó.	

Tabla 3.4 Descripción de Caso de Uso: Consultar consumo eléctrico

Fuente: Grupo de investigación

Nombre:	Realizar pago-banco	
Actor:	Usuario	
Descripción:	El usuario realiza el pago al banco desde una opción que le da la aplicación dentro del caso de uso Consultar consumo eléctrico.	
Flujo Principal:	Eventos ACTOR	Eventos SISTEMA
	1. El usuario indica el nombre del banco del cual es usuario.	1. El sistema lleva al usuario a la página principal del banco del cual es usuario.
	2. El usuario busca la opción de pagar online el consumo eléctrico.	2. El sistema busca la página requerida por el usuario y le pide que ponga sus datos para hacer el proceso.
Alternativa:		
Precondición:	El usuario debe ser cliente de un banco para poder hacer el pago online.	
Poscondición:	El usuario está listo para introducir sus datos y realizar el pago.	

Tabla 3.5 Descripción de Caso de Uso: Realizar pago-banco

Fuente: Grupo de investigación

Nombre:	Registrar datos de usuario	
Actor:	Usuario	
Descripción:	El usuario requiere registrar sus datos en el banco del cual es cliente para efectuar el pago de su consumo eléctrico online.	
Flujo Principal:	Eventos ACTOR	Eventos SISTEMA
	1. El usuario coloca sus datos para realizar la transacción financiera.	1. El sistema verifica que los datos insertados están correctos.
		2. El sistema realiza el trámite con éxito
Alternativa:	1. El usuario coloca mal sus datos.	1. El sistema no acepta los datos y pone en la interfaz que verifique los mismos.
Precondición:	El usuario debe ser cliente de un banco para poder realizar el pago y que los datos registrados sean válidos en el mismo.	
Poscondición:	El usuario realizó el pago al banco satisfactoriamente.	

Tabla 3.6 Descripción de Caso de Uso: Registrar datos de usuario

Fuente: Grupo de investigación

Nombre:	Comparar consumo eléctrico	
Actor:	Usuario	
Descripción:	El usuario inserta manualmente el valor de la notificación(cálculo estimado del consumo eléctrico de la vivienda extraído de la cantidad de equipos electrodomésticos que existen y otros equipos que consumen electricidad)	
Flujo Principal:	Eventos ACTOR	Eventos SISTEMA

	1. Se actualiza el valor de notificación.	1. El sistema se mantiene comparando el valor del consumo insertado por el usuario en el dato suministro de la notificación con el valor actual del consumo eléctrico de la vivienda.
Alternativa:		
Precondición:		
Poscondición:	El sistema está constantemente comprando ese valor con el valor actual del consumo eléctrico	

Tabla 3.7 Descripción de Caso de Uso: Comparar consumo eléctrico

Fuente: Grupo de investigación

Nombre:	Revisar notificaciones	
Actor:	Usuario	
Descripción:	El usuario revisa las notificaciones enviadas por el sistema en caso de que el consumo actual de la electricidad sobrepase el valor de la notificación insertado por él.	
Flujo Principal:	Eventos ACTOR	Eventos SISTEMA
	1. El usuario revisa si existe alguna notificación enviada por el sistema en caso de sobrepasar el consumo eléctrico para tomar medidas en base a ello.	1. El sistema envía notificaciones en caso de que el valor del consumo eléctrico de la vivienda en ese momento sobrepase el valor de la notificación.

Alternativa:	1. El usuario revisa si existe notificación de sobreconsumo eléctrico.	2. En caso que no exista un sobreconsumo eléctrico el sistema no enviará notificaciones.
Precondición:		
Post condición:	Al revisar las notificaciones el usuario sabrá si está superando el valor del consumo normal de la vivienda o no.	

Tabla 3.7 Descripción de Caso de Uso: Enviar notificaciones

Fuente: Grupo de investigación

3.3 Implementación de sistemas de medición inteligente

3.3.1 Herramientas utilizadas para el desarrollo de la aplicación

Para la elaboración de la aplicación Monitor EE es indispensable de la utilización de varias herramientas sin las cuales es imposible llevarla a cabo con éxito. Las cuales se nombran a continuación:

Eclipse como IDE de desarrollo, el cual constituye una plataforma de código abierto que utiliza java como lenguaje de programación. Esta herramienta es un marco de trabajo que incluye un conjunto de servicios enfocados en la construcción de un entorno integrado y unificado de desarrollo.

Su consistencia no está limitada a las herramientas de desarrollo de Java solamente, ya que por ejemplo, los complementos se encuentran disponibles o planificados para incluir soporte para los lenguajes de programación como C/C++ y COBOL. El marco de trabajo de Eclipse puede también utilizarse como base para otros tipos de aplicaciones que no se relacionen con el desarrollo del software, como los sistemas de gestión de contenido.

Al ser este software una herramienta de código libre, sus desarrolladores tienen la

libertad de licenciar sus propias aplicaciones creadas de la manera que lo crean necesario, lo que constituye un aliento para la innovación en el desarrollo de sus aplicaciones.[47]

Como se había expuesto anteriormente el lenguaje de programación utilizado es Java, que esta integrado para el desarrollo de aplicaciones Android, el cual actualmente es el sistema operativo móvil líder a nivel mundial para las nuevas generaciones de teléfonos inteligentes y tablets.

Se puede decir además que java es uno de los lenguajes de programación más utilizados del mundo, pues se caracteriza por ser innovador, orientado a objetos, robusto y por su versatilidad en cuanto a su uso en distintos sistemas operativos. Por poseer la mayor parte de su código libre se puede obtener una gama de herramientas para el desarrollo con este lenguaje.[48]

El gestor de base de datos utilizado en el desarrollo de la investigación es MySQL Server el mismo que gestiona las bases de datos relacionales y constituye un software de código abierto licenciado bajo la GPL de la GNU.

Características principales:

- Constituye un gestor de gran velocidad y robustez.
- Se le pueden incorporar una gran cantidad de datos en las columnas.
- Gestor de base de datos multiplataforma, debido a que soporta gran variedad de sistemas operativos.
- Cada base de datos cuenta con tres archivos: Estructura, datos e índice.
- Flexible, posee sistemas de contraseñas y gestión de usuarios con un buen nivel de seguridad de los datos gestionados.
- El servidor brinda mensajes de error en distintas lenguas.
- Gran facilidad de configuración e instalación.
- Bajos costos en requerimientos para la elaboración de la base de datos, ya que debido a su bajo consumo puede ser ejecutada en una computadora con escasos recursos de hardware.[49]

Para el desarrollo de servicios web se utiliza el lenguaje Personal Home Pages, más conocido como PHP, el cual se caracteriza por ser de código abierto, adecuado para

desarrollo web, pudiendo ser incrustado en HTML si es necesario.

PHP se utiliza para generar páginas web dinámicas lo que significa que generalmente la información está contenida en una base de datos que con solo modificarla el sitio web será capaz de disponer nuevamente de toda la información colocada con solo cargar la portada. Este lenguaje resulta fácil de manejar para los principiantes, pero a su vez ofrece muchas características avanzadas para los programadores profesionales.

Otra de sus características es que soporta gran variedad de base de datos como son: InterBase, mSQL, MySQL, Oracle, PostgreSQL entre otras. Se puede decir además que este lenguaje ofrece la integración con varias bibliotecas externas lo que permite que el desarrollador pueda realizar desde la generación de documentos en pdf hasta analizar código XML.[50]

Para administrar las aplicaciones se usa el servidor independiente de plataforma XAMPP, cuyo nombre proviene del acrónimo X (para cualquiera de los sistemas operativos) junto al uso del servidor Apache, la base de datos MySQL, y los intérpretes para lenguajes de script: PHP y Perl.

Como ya se había planteado, hoy en día, esta herramienta es utilizada como servidor de sitios web, el cual brinda gran versatilidad para que cualquier desarrollador elija su uso. Es básico, flexible, potente e incluye una herramienta para proteger las partes más importantes de la aplicación generada de manera fácil.[51]

3.3.2 Servicios web desarrollados

Lectura gráfico línea: Trae información desde la BD y permite hacer los gráficos de línea necesarios para mostrarle al cliente la manera en que se está consumiendo la energía eléctrica en su hogar.

Lectura gráfico Barra: Trae información desde la BD y permite hacer los gráficos de barra mediante los cuales muestra al cliente su consumo mensual y el valor a pagar de los mismos.

Notificaciones: Alertas que le envía al cliente en caso de que su consumo sea mayor que el valor del consumo mínimo introducido por él en la aplicación Monitor EE.

Actualizar valor mínimo: El cliente puede actualizar manualmente el valor mínimo del consumo en el caso de aumento en la cantidad de equipos electrodomésticos en la vivienda.

3.3.3 Clases desarrolladas en java

LoginActivity.java: Clase que permite que el usuario entre a la aplicación con su respectivo usuario y contraseña para que de esta forma sus datos sean únicamente vistos por él.

ClienteContenedor.java: Clase que administra todas las pestañas superiores en la pantalla de la aplicación Monitor EE (Líneas, barras, datos del cliente, y notificaciones).

BarrasActivity.java: Clase encargada de crear el gráfico de barras en la pantalla, con el cual se le muestra al cliente su consumo mensual y el valor a pagar de los mismos.

MainActivity.java: Clase que muestra el gráfico de líneas mediante el cual se visualiza el consumo de energía en el hogar.

ClientePestañaDatos.java: Clase que administra todos los datos de la pantalla del cliente.

InformaciónServicios.java: Clase que contiene información para conectar la aplicación Monitor EE a los servicios web mediante una dirección URL.

ListadoNotificaciones.java: Clase que regula las notificaciones enviadas al cliente desde la interfaz de notificaciones de la aplicación.

Rastro.java: Clase que permite que el usuario deje un registro de todo lo que realiza en el aplicativo Monitor EE.

XMLParser.java: Clase que lee o administra los datos XML de la aplicación.

3.4 Evolución de consumos sin acciones correctivas

El hombre a través de los tiempos se ha valido de múltiples servicios que le han proporcionado confort a lo largo de su vida, tal es el caso de la energía eléctrica la misma que ha tenido un papel crucial en el desarrollo de la humanidad ya que sin ella no existiría vida moderna ni los avances de la tecnológicos, con sus equipos cada vez más sofisticados que brindan un sinfín de comodidades, los cuales a su vez demandan mayor cantidad de energía, tal es el caso de los electrodomésticos los cuales en el ámbito residencial representan un papel fundamental, pues actualmente se ha creado una gran dependencia de los mismos debido al bienestar y comodidad que ofrecen.

Desde hace muchos años se ha intentado imponer medidas para evitar este gasto de innecesario de energía. Pero, ¿qué sería del mundo sin tomar acciones correctivas para evitar el exceso de consumo eléctrico en todos los sectores de la sociedad?

Sencillamente se tendría un mundo donde las fuentes de energía se llegarían a agotar y esto traería consecuencias como:

- 1 El mundo sin energía eléctrica sería un mundo pobre tecnológicamente hablando, y como consecuencia muchas actividades cotidianas serian imposibles de realizar haciendo que gran parte de la población entre en pánico, puesto que no está preparada para un retroceso en el desarrollo tecnológico.
- 2 Gran parte de las fábricas dejarían de funcionar. En el caso de la medicina se tendría que recurrir a los medicamentos de origen natural y la gran mayoría de sus instrumentos quedarían inutilizados lo cual traería resultados catastróficos.
- 3 Los medios de transporte modernos se verían afectados casi en su totalidad, debido a que actualmente muchos de ellos trabajan en base a la electricidad.
- 4 La producción de alimentos se vería afectada en su mayoría, debido a que en la actualidad muchos de los procesos son automatizados, por ende dependen de la energía eléctrica [52].

Y por ello es de vital importancia tomar conciencia del ahorro energético y desarrollar herramientas tecnologicas que ayuden en este propocito, como es Monitor EE, aplicación

del sistema operativo Android desarrollada en esta investigación.

Es necesario que el usuario tenga conocimiento de la carga estimada a ser utilizada en su hogar ya que en base al resultado puede conocer que equipos son los que consumen mayor cantidad de energía, tomando como base cargas típicas de los aparatos eléctricos se puede determinar el consumo estimado de energía, la fórmula contempla el número de equipos, la potencia típica del equipo y las horas estimadas que esta en funcionamiento.

$$\text{Consumo eléctrico estimado} = \frac{\text{Numero de equipos} * \text{potencia típica} * \text{horas de uso}}{1000}$$

El resultado estará dado en Kwh para cada grupo de equipos y el resultado de la suma de todos los consumos individuales arroja el consumo eléctrico estimado del hogar.



Figura 3.2 Potencia de equipos en un hogar tipo

Fuente: <http://www.maquinariayocio.com/GENERADORES-Y-MAQUINARIA/TABLA-CONSUMO-ELECTRODOMESTICOS-Cuanto-consumen-tus-electrodomesticos>

El usuario, al conocer su consumo estimado, y si además cuenta con una aplicación que muestre de forma gráfica la evolución del consumo de energía, puede regirse a su presupuesto para eliminar o reducir consumos innecesarios y tomar acciones correctivas como las que se detalla a continuación.

Medidas para ahorrar energía eléctrica en el hogar:

1. Refrigeración

- Descongelación periódica de los medios de refrigeración.
- Evitar el exceso de aperturas del refrigerador de forma innecesaria para impedir la pérdida de frío.
- No guardar alimentos calientes en el refrigerador para conservar la energía.
- Limpiar frecuentemente el equipo evitando de esta forma la acumulación de polvo, el cual disminuye el rendimiento del mismo y aumenta el consumo de energía.

2. Iluminación

- Durante el día utilizar la luz natural y en la noche apagar las luces que no se están utilizando.

3. Lavado

- Utilizar la lavadora durante la mañana.

4. Equipos Electrónicos

- Los equipos electrónicos que no se estén utilizando deben ser desconectados.
- Si se posee computadora en el hogar se debe configurar en función de ahorro de energía y programar el apagado automático del monitor tras una inactividad del mismo de más de treinta minutos.

5. Ajuste de potencia Eléctrica contratada

- Realizar un cálculo para determinar el consumo máximo de la vivienda atendiendo a la cantidad de electrodomésticos que se posee y en base a este valor establecer los límites de consumo eléctrico.[46]

El problema del consumo eléctrico no solo afecta los hogares sino también las empresas, quienes para realizar una buena gestión, necesitan minimizar sus datos de

consumo eléctrico. Para tales efectos, se considera indispensable trabajar en los siguientes temas:

1. Capacitación del consumidor:

Se debe realizar una capacitación previa al consumidor con el fin de explicarle la forma en que se realiza el cálculo de la factura de consumo y las herramientas que necesita para hacer un buen ahorro energético, así como las principales medidas que el trabajador debe emplear para contribuir con el ahorro energético.

2. Asesoría al consumidor:

Contratar un grupo de asesores y consultores especializados en el tema, los mismos que partiendo de un diagnóstico acertado de las instalaciones y equipos con los que cuenta la empresa, determinen la mejor manera de ahorrar energía sin afectar su producción habitual.

3. Monitoreo y control de todos los parámetros eléctricos:

Tomar medidas en cuanto a la instalación de sistemas que capturen, en tiempo real, datos y variables eléctricas desde los medidores (u otros equipos), de manera tal que los mismos puedan ser analizados e interpretados posteriormente para determinar cuáles son los más eficientes y con ellos lograr normalizar ciertos parámetros, obtener una reducción de la factura y aplicar las medidas correctivas para un uso eficiente educado de la energía.

4. Energía alternativa:

El uso de grupos electrógenos constituye una alternativa para el ahorro de energía debido a que éstos utilizan combustible.

5. Incentivos:

Se debe crear incentivos tarifarios para las empresas que contribuyan con el uso adecuado de la energía eléctrica y a su vez ahorro de la misma.

6. Uso de equipamiento y procesos de alta eficiencia:

Para reducir el consumo de energía resulta factible el empleo de equipos y procesos de mejor eficiencia. A continuación, se enumeran algunos ejemplos:

- **Uso de motores eléctricos de alta eficiencia:**

Más del 50% del consumo eléctrico en las plantas industriales se debe a motores eléctricos. De éstos, en la actualidad, no más del 5% del parque de motores son de alta eficiencia. En este sentido, un motor de alta eficiencia permitirá un ahorro del orden del 2% de energía, siendo su costo entre un 10% y 20% superior a su versión estándar. Los datos generados de estudios realizados demuestran que después de un período que va desde los 6 meses de su compra hasta completar el año, el usuario comienza a apreciar beneficios en cuanto a ahorro energético y costos de producción.

- **Iluminación:**

Con la utilización de luminarias de alta eficiencia, es decir, aquellas en las que la relación intensidad lumínica- potencia consumida sea alta, se logran ahorros de hasta un 80%. Es por ello que se debe realizar un plan de sustitución o reemplazo de las fuentes de iluminación existentes de alta demanda energética por estas luminarias.

- **Computadores:**

Reemplazar los computadores de escritorio con monitores CRT por notebooks de última generación, permitirá disminuir hasta 17 veces el consumo de energía eléctrica.

Reemplazar servidores existentes en las empresas de más de tres años de antigüedad ya que esto significa un ahorro de energía de hasta un 50%, además de un mejor desempeño en su funcionamiento.

- **Otros equipos:**

Seleccionar un adecuado equipamiento que ayude a realizar los procesos de la empresa y crear un mejor ambiente laboral como ventiladores, sistemas de climatización, hornos, sistemas de refrigeración, equipos de cómputo, entre otros, con distintos grados de eficiencia, significará un ahorro de energía razonable.[52]

Uno de los temas de los cuales las personas no tienen conocimiento y que repercuten

en el uso irracional de energía por parte de los usuarios, es que la tarifa pagada por el consumo de energía no es plana, y que su costo incrementa a medida que se van superando límites establecidos, es así que el CONELEC anualmente proporciona a las empresas distribuidoras pliegos y cargos tarifarios para para que puedan recuadrar los valores por concepto de comercialización de energía.

Los cargos tarifarios que corresponden al sector residencia se detallan en la Tabla 3.8

RANGO DE CONSUMO	ENERGÍA EN USD/KWH
0 – 50	0,078
51 – 100	0,081
101 - 150	0,083
151 – 200	0,090
201 – 250	0,097
251 – 300	0,099
300 – 350	0,099
351 – 500	0,099
501 – 700	0,1285
701 – 1000	0,1450
1001 – 1500	0,1709
1502 – 2500	0,2750
2501 – 3000	0,4360
SUPERIOR	0,6812

Tabla 3.7 Pliego de tarifas residenciales

Fuente: CONELEC. Cargos tarifarios para categoría residencial de baja y media tensión[53]

Adicional a los valores por energía consumida se le debe sumar el cargo por comercialización que es igual a 1.414 USD.

Si se pone en conocimiento de los usuarios los valores contenidos en la Tabla 3.7 se espera que el usuario ajuste su consumo mensual acorde a su presupuesto.

Como se ha dicho en este apartado el CONELEC es el ente encargado de regularizar las tarifas eléctricas, es por ello que se debe establecer tarifas que apliquen al concepto de medición inteligente y por medio de las cuales se fomente el ahorro energético y la optimización, actualmente se tienen contempladas sanciones y penalizaciones para los usuarios industriales que debido a la maquinaria que usan reflejen un bajo factor de potencia.[54]

CAPÍTULO IV

INGENIERÍA DE PROYECTOS

La Gestión Empresarial, en el contexto del desarrollo institucional y sus continuos cambios acelerados; presenta una gran demanda en cuanto a llevar al día datos como generación de proyectos de Inversión ya sean públicos, sociales o privados, convirtiéndose hoy en un desafío notable para los profesionales que propicien el desarrollo local, regional y nacional con la creación de nuevas empresas o crecimiento de las actuales para bienestar de la sociedad en general. [55]

De esta necesidad surge la ingeniería de proyectos, que no es más que la contemplación de una fase dentro del proyecto que está en surgimiento, donde se definen aspectos importantes para el comienzo de una empresa.

El estudio de la Ingeniería del proyecto aporta un conjunto de datos técnicos y económicos, los cuales tienen el objetivo al comenzar la empresa definir sus costos de construcción y explotación. En ella también debe quedar definido la identificación de problemas de carácter técnico que pudieran limitar la empresa en una de sus funcionalidades en algún momento determinado del desarrollo de la misma.

Objetivos de la ingeniería de Proyectos:

- Brindar a los profesionales un conjunto de técnicas y conocimientos para un adecuado manejo de la identificación, formulación, evaluación y administración de proyectos públicos, sociales y privados.
- Brindar una preparación eficiente al profesional, que esté guiada por una formación académica sólida en el campo de proyectos de inversión, permitiéndole resolver diferentes problemas tecnológicos, económicos y financieros que pudiera enfrentar la empresa.
- Brindar la oportunidad a los egresados de las diversas carreras profesionales, la oportunidad de formar equipos y superarse, siendo capaces en forma grupal, de formular y darle inicio a diferentes proyectos de inversión a partir de varios criterios del grupo.

Aspectos básicos de la ingeniería de proyecto

A continuación se hace referencia más a la fase técnica del estudio más bien que a la realización, tratándose brevemente los aspectos básicos que hay que tener en cuenta a la hora de realizar una buena ingeniería. Los puntos son los siguientes:

1. Realización de investigaciones preliminares.
2. Selección del proceso de producción para el desarrollo de la empresa.
3. Especificación de los equipos de funcionamiento y montaje.
4. Edificaciones y su distribución en el terreno (planificación de todos los locales).
5. Distribución de los equipos en los edificios pertenecientes a la empresa.
6. Estudio de proyectos similares creados hasta el momento.
7. Rendimientos y alcance de la empresa.
8. Establecimiento de un valor medio en la capacidad de producción el cual debe ser flexible.
9. Elaborar programas de trabajo de estricto cumplimiento.

4.1 Análisis Costo-Beneficio

El análisis de costo-beneficio es una técnica importante dentro del ámbito de la teoría de la decisión. Con estos datos se pretende determinar la factibilidad de un proyecto mediante la enumeración y valoración posterior en términos monetarios de todos los costos y beneficios derivados directa e indirectamente de dicho proyecto. Este método es aplicable a un sinnúmero de proyectos desarrollados con planes de negocios definidos, prestando especial atención a la importancia y cuantificación de sus consecuencias sociales y/o económicas que esto deriva.

Se utiliza el análisis Costo-Beneficio, en la medida que un proyecto se establece, para determinar entonces sus beneficios, que se pueden acumular en algún momento y el costo en el cual se incurrirá. Un proyecto se puede continuar únicamente si ya después declarados los mismos, los beneficios son mayores que los costos del proyecto y por ende la relación beneficio-costos en este caso es mayor a uno. Es importante señalar que los términos de beneficios, que tiene un valor en el mercado, también deben ser representados en valores monetarios.

Esquema general para el análisis de costo-beneficio

1. Identificación de los beneficios y costos de cada alternativa analizada.
2. Cuantificar en la medida de lo posible, los beneficios extraídos convirtiéndolos en valores monetarios, de manera que se pueda establecer una comparación entre los diferentes beneficios y los costos a obtener de los mismos.
3. Calcular el costo total para cada alternativa extraída.
4. Restar los costos de la alternativa de inversión más baja de aquellos de la alternativa de costos más altos (la cual se considera que debe ser la alternativa justificada) se asigna la letra C a este valor en la razón de B/.C
5. Calcular los beneficios totales de cada alternativa.
6. Restar los beneficios para la alternativa de costo menor de los beneficios. Para la alternativa de costo mayor se debe prestar atención en los signos algebraicos. Al final se utiliza el valor como B en la razón B/C.
7. Si $B/C \geq 1$ se justifica la inversión incremental; se debe seleccionar entonces la alternativa de la inversión más grande. De no ser así se debe realizar lo contrario.

A continuación se describen las variables que deben ser expresadas en términos monetarios:

- Beneficio (B): ventajas experimentadas por el propietario.
- Beneficios negativos (BN): desventajas para el propietario cuando el proyecto en consideración es implementado.
- Costos (C): gastos anticipados, operación, mantenimiento, entre otros, menos cualquier valor de salvamento.

4.4.1 Cálculo de costos-beneficios

1. Lo primero que se debe realizar al iniciar el cálculo de la razón costo-beneficio de un proyecto es convertir todos los beneficios que se tienen en unidades comunes de valores monetarios.

2. Calculado los beneficios, se comienzan a realizar los cálculos de los costos del proyecto.
3. Una vez tanto que el numerador (beneficios) como denominador (costos) se tengan, se puede aplicar la razón convencional de B/C que es la siguiente:

$$B/C = \frac{\text{BENEFICIOS (POSITIVOS - NEGATIVOS)}}{\text{COSTOS}}$$

Nota: el valor de B/C puede cambiar considerablemente si los beneficios negativos se toman como costos. Además el B/C debe ser mayor o igual a 1.0 para que sea un proyecto económicamente ventajoso y se pueda continuar con su desarrollo.[56]

En este proyecto de creación de la aplicación Monitor EE para Android los únicos beneficios negativos que se tienen son solamente los costos del proyecto, por lo que se puede aplicar la regla dada anteriormente y B/C sería igual al cúmulo de beneficios positivos obtenidos en el mismo, por lo que se puede decir entonces que el proyecto económicamente es rentable y ventajoso.

Costos del producto equivalentes a beneficios negativos:

- Tablet con sistema operativo Android para correr la aplicación y probarla: 234 USD.
- Navegación en internet para búsqueda de información necesaria para el desarrollo de la investigación: 12 meses, equivalente a 276 USD.

Costo total: 510 USD

Dato: Al ser el sistema operativo Android libre no hubo costo en licencias ni permisos.

Android Es un sistema operativo que ha venido desarrollando y dominando el desafío de una nueva estrategia para resolver con sus aplicaciones estos problemas cotidianos,

conduciendo al cliente hacia sus pantallas móviles e identificando que resulta beneficioso para él.

Actualmente, entre un sitio web bien estructurado y una aplicación muy sencilla, los usuarios prefieren la aplicación, ya que ésta ventajas alertas, la rapidez y la movilidad.

Un estudio realizado en Octubre de este año por “Marketing guide for Mobile and Data”, a 400 compañías de diferentes productos y servicios, sobre cómo los usuarios perciben las aplicaciones, el impacto recibido del plan móvil, de la ventaja de utilizarlo, de sus métodos, de sus estrategias y la práctica, develó los siguientes datos:

- 85% de los empresarios entrevistados, aseguran que el mercado del móvil es un excelente puente al éxito, hacia nuevos mercados o comercios y audiencias.
- 80% respondieron que las aplicaciones no solo están transformando sus empresas, sino además sus carreras, su visión de estructurar estrategias y visualizar la empresa para estar a la vanguardia estando al tanto de las tendencias actuales y la tecnología.
- 40% respondieron que el móvil es una prioridad para su marca y el núcleo de su estrategia global de marketing
- Un 34% adicional activó una visión estratégica para móviles y continúan probando nuevas tácticas con el fin de adquirir experiencia con el canal.

Hoy en día muchas de las empresas le apuestan a desarrollar un negocio asociado al móvil puesto que es lo que el usuario prefiere y los resultados de la encuesta demuestran que es en lo primero que se debe pensar al proyectar una estrategia de marketing.

Las aplicaciones del Smartphone han seducido a los usuarios, puesto son de gran ayuda en las actividades sus diarias.

Estas aplicaciones tienen la ventaja ser herramientas fácilmente descargables, casi siempre gratuitas, de fácil acceso y son más rápidas para navegar. Poseen interfaces muy fáciles de usar lo mismo para usuarios que son expertos como para los que no lo son.

Datos históricos que registran que el sistema operativo Android, el 22 de octubre 2008 contaba con tan solo 50 aplicaciones. Hoy en día la cifra es superada con más de 6 billones de estas. En cinco años ha avanzado en número y en ventajas tanto para el usuario como para negocios al interconectarse con las aplicaciones [64].

Hoy en día ya no solo se tiene llamadas y mensajes de texto, ahora existen muchas otras ventajas gracias a aplicaciones que con la ayuda de este sistema operativo disminuye costos hasta cierto punto, entre las ventajas se puede mencionar:

- **Interacción social:** La preferida por muchos usuarios de Android, permite que los usuarios utilicen las redes sociales desde su equipo móvil como promotor de su producto.
- **Movilidad:** A través de su Smartphone la masificación Android permite trabajar y revisar datos desde casi cualquier lugar del mundo, a través de una conexión de Internet.
- **Bajo Costo:** Por la manera libre de realizar las aplicaciones y la distribución de las mismas, el usuario puede encontrar aplicaciones acordes a su presupuesto.
- **Herramientas útiles:** Android se compone por un cúmulo de aplicaciones que forman parte de las herramientas útiles para cada usuario de este sistema operativo [65].

Ventajas de las aplicaciones desarrolladas para móviles con sistema operativo Android

Beneficios para los usuarios

- Muchas son las aplicaciones desarrolladas en el sistema operativo Android con un sinnúmero de beneficios para usuario, permitiéndole conseguir un grado de utilidad de las mismas al que no estaban acostumbrado.
- Conocer el consumo eléctrico del hogar, como es una de las principales funciones de la aplicación Monitor EE desarrollada en esta investigación, es beneficioso para todos, puesto que mediante ésta aplicación el usuario puede regular su consumo de energía.

- La rapidez y la facilidad con la que se accede a la información presente en estas aplicaciones constituye una de las mayores ventajas para el usuario ya que de esta forma el mismo se siente más cómodo en su uso.
- El almacenamiento de manera segura de sus datos personales, sin riesgos de ser observados por otro usuario o ser eliminados, es otra importante ventaja. De esta manera el usuario, ahorra tiempo al acceder de forma más rápida a sus datos.
- Muchas aplicaciones están creadas con el objetivo de entretener a los usuarios lo que le facilita diversión sin realizar grandes gastos.

Beneficios para las marcas

- El sistema Android posee aplicaciones de negocios las cuales son capaces de detectar indicadores que van en contra de la empresa en diversas áreas, y una vez detectados se establecen medidas que erradiquen este problema y aseguren la continuidad de la misma, aumentando la eficiencia, reduciendo costos, haciendo crecer las ventas y obteniendo múltiples ganancias.
- La utilización continua de las aplicaciones en los móviles de los clientes permite a las marcas ganar en distinción y dinero al ganar cada día más usuarios que la utilizan.
- La utilización de estas aplicaciones como herramienta de marketing le proporciona al cliente la posibilidad de que en cualquier lugar y en cualquier momento puede acceder a toda la información de la marca solo con disponer de un dispositivo móvil.
- La creación de aplicaciones útiles para el usuario genera mayor expectación, es decir, y se tendrá entonces un grupo de personas esperando a que salga al mercado un nuevo producto de la compañía que supondrá un aumento seguro en las ventas.
- La conexión casi permanente de los usuarios a internet desde el móvil le permite a la marca establecer un nuevo canal de venta directa e inmediata las 24 horas del día.

Beneficios para el sector turístico y comercial

- El diseño gráfico de las aplicaciones y su buena presentación actúa como un reclamo publicitario, atrayendo la atención de aquellos turistas que necesitan de información novedosa y útil.

- El desarrollo de recursos multimedia con diferentes funcionalidades desarrolladas como rutas de interés o itinerarios para llegar a su destino es de vital importancia para estas personas que no tienen pleno conocimiento del país.
- Muchas de estas aplicaciones le permiten a los viajeros guardar experiencias durante su viaje turístico a través de fotos y videos relacionados con el mismo [48].

4.2 Servicios a ser implementados sobre el sistema Android

Android es un sistema operativo principalmente móvil con una calidad de desarrollo en cuanto su rendimiento como en sus características únicas, con una visión de su futuro muy extensa llena de éxitos, de código abierto, estable, robusto y que obliga a desarrollar servicios con un excelente hardware para permitir que sus servicios sean ejecutados por los usuarios con eficacia y rápidamente.

Desde hace algunos años, este sistema operativo ganado terreno rápidamente en el mundo de la tecnología ya que sus creadores han hecho convenios y acuerdos para que este software sea utilizado en nuevos dispositivos por la calidad esperada y velocidad al trabajar con pocos recursos y si a eso se le suma que la programación para este es muy sencilla, crea uno de los software móviles más potentes del mercado.

A diferencia de los demás este tendrá ciertas ventajas o características que se transforman en aplicaciones prácticas y que lo hacen exclusivo [57].

El servicio ofrecido por Android es mediante aplicaciones que corren de forma automática, sin la presencia de los usuarios, lo que hace que ellos sean independientes y satisfagan en su totalidad a los usuarios [58].

4.2.1 Aplicaciones para ahorrar el consumo energético en el hogar en el sistema operativo Android.

El éxito de las aplicaciones móviles ha llegado también al hogar. Uno de los factores que encarecen nuestra factura de la luz es la energía que consumimos sin darnos cuenta, es decir, la que consumen la mayoría de aparatos eléctricos por el simple hecho de estar conectados sin necesidad alguna.

Muchas son las aplicaciones que se han creado en función a esto. A continuación se citan algunas de ellas:

1. Standby Check:

Aplicación que cuenta con un elemento que está constantemente activado cuya funcionalidad es decir cuánta energía en el hogar se podría ahorrar, introduciéndole unos pocos datos.

2. App:

Permite llevar un control total sobre la electricidad que se consume mientras se duerme y ayuda a tomar medidas en vista a estos datos obtenidos con el objetivo de ahorrar. Todos estos progresos y consumos energéticos, tanto en tu casa como en el trabajo se registran en esta aplicación de forma que se podrá evaluar si se lo está haciendo correctamente.

3. Electrocalculator:

Detecta los datos del consumo que más engrosa la factura para dedicarle especial atención a intentar reducirla. El Electrocalculator nos hace el cálculo del gasto de cada equipo electrodoméstico que se tiene en el hogar en concreto o si es del interés hace la suma de varios. No es un coste exacto, ya que habría que sumarle la proporción de impuesto eléctrico e IVA.

Dada la gran importancia del ahorro de energía en los hogares en esta investigación se desarrolla Monitor EE. Se trata de una aplicación para Android la cual mide el consumo eléctrico al colocar los datos que emite el medidor de energía en la aplicación manualmente y da consejos para reducirlo. Además la misma ayuda a identificar, según los datos de los equipos electrodomésticos que se tiene en casa, si su consumo está por encima de lo normal enviándole al usuario algunas notificaciones.

Monitor EE está diseñado de una forma sencilla y fácil de usar, compuesto por una serie de gráficas y números que le dan al usuario una mejor idea de su consumo energético para su mejor entendimiento y de esta manera intentar reducir consumo de energía.

4.3 Impacto Social de las aplicaciones Android en medición inteligente

La energía es un factor que juega un papel determinante para el desarrollo de los países. Hoy en día existe la idea que consumir energía es un factor necesario para el desarrollo económico y social de un país pero también se tiene la sensación de que el ritmo de consumo actual en todas las esferas en desarrollo pone en peligro el estilo de vida de las personas debido al alto consumo de electricidad.

Sin consumir energía no es posible el desarrollo social, industrial o comercial, no es posible superar la pobreza ni mejorar la salud puesto que la energía permite hoy acceder a una mejor calidad de vida.

- Conservar mejor los alimentos, refrigerándolos, y protegiendo de bacterias que perjudican la salud.
- Acceder a información necesaria en muchas ocasiones para mantener a informada a la población mediante la TV, radio e internet.
- Transportar los productos desde los centros de producción a los de distribución y venta de los mismos para su futura refrigeración en casos necesarios.

Por otro lado, la producción y la forma en que se usa la energía, genera un gran impacto ambiental. El desarrollo industrial ha generado un consumo intensivo de energía exigiendo cada vez más la explotación de recursos naturales para su producción [63].

El ahorro de energía es de gran importancia social y ha llegado a convertirse en un tema de interés para muchos investigadores y profesionales. Es por ello que muchas empresas trabajan hoy en día en función de combatir el uso innecesario de la energía.

Para ello es necesario el desarrollo de herramientas que permitan controlar estos consumos de forma sencilla y en el hogar.

Cuando se habla de herramientas útiles no se puede dejar de mencionar la idea de crear una aplicación que permita controlar el consumo de energía hogar, lo que es claramente un beneficio para el usuario.

Existen algunas aplicaciones para teléfonos inteligentes que están diseñadas para ayudar a saber cuánta energía se utiliza para así ahorrar en combustibles caseros.

Por ejemplo, Home Energy Performance Pro la cual después de realizar un diagnóstico profundo de una vivienda, ofrece recomendaciones para reducir su consumo. Analiza el aislamiento, la electricidad y la energía (que se produce y que se consume).

Otra de las aplicaciones que sirven para este menester en la plataforma Android es Energy Efficient Home, la cual asegura que el consumo del hogar con su uso puede bajar hasta en un 40%. Otra de ellas es Energy Savings Calculator con la cual se sabe lo que se ha ahorrado al poner en marcha consejos los cuales deben ser cumplidos. Por último, Energy Saver Free que funciona como una especie de consejero estudiando su consumo inicial y sus necesidades en función al mismo.[66]

Es por ello que en esta investigación se desarrolla Monitor EE, la cual al colocar los datos de notificación en la aplicación, compara permanentemente el valor dado por el usuario que no es más que un cálculo estimado del consumo máximo de los equipos electrodomésticos que se tiene en casa, con el valor actual del consumo eléctrico y envía notificaciones en caso de superar el mismo. El usuario revisa estas notificaciones para informarse y tomar medidas en caso que sea necesario. Además la aplicación brinda la posibilidad ecoger el banco al cual es asociado para que pueda pagar con su número de tarjeta electrónica.

Su manejo es sencillo, se basa en medias, gráficos y estadísticas, así como la posibilidad de mostrarle al cliente en qué momento del día el consumo es mayor o menor.

Monitor EE tiene un diseño muy fácil de manejar y atractivo para el cliente tanto en su interfaz gráfica como en la importancia que le da el usuario por la función tan útil que desempeña.

No está mal pensar que en ahorrarse algunos dólares al mes gracias a una aplicación gratuita o bajo costo como una de las nombradas anteriormente. Por supuesto, el ahorro de energía no se trata de no poder disponer de electrodomésticos indispensables sino de tener la conciencia en que el uso incorrecto de la energía es uno de los mayores problemas

que enfrenta hoy en día la sociedad.

Uno de los mayores obstáculos a la hora de ahorrar energía eléctrica en casa es que el consumo eléctrico no es detectable a simple vista. De hecho casi nadie podría calcular en un momento dado cuánta electricidad está consumiendo, ni las medidas de ahorro necesarias aplicables para la reducción de ese consumo.

Es por ello que esta aplicación puede ser de gran impacto social puesto que conocer el consumo eléctrico es el primer paso para poder reducirlo, ahorrando de esta forma energía y dinero. En la aplicación Monitor EE del sistema Android, estos datos se presentan de forma visual mediante gráficos y colores que hacen muy fácil ver cuándo el consumo es normal o cuando es excesivo o incluso si está cerca del límite de potencia máxima establecida. Según el país desde el que se utilice el medidor el consumo eléctrico se muestra en la moneda correspondiente y según las tarifas eléctricas aplicables

Tras unos días de uso es posible conocer con bastante exactitud cuál es la rutina de consumo. Del mismo modo que la factura eléctrica se hincha sumando pequeñas cantidades de consumo eléctrico, estas cifras pueden reducirse al tomar medidas correctivas como la desconexión de equipos innecesarios [67].

La aplicación que se desarrolló en esta investigación es de gran importancia social puesto que es un medidor de la corriente eléctrica en casa y desarrolla otras funciones de gran interés como identificar cuáles de los equipos electrodomésticos que tenemos en la vivienda son los más consumidores de corriente, así como definirnos cuál es el momento del día en que más debemos ahorrar electricidad. Para este proyecto de desarrollo el siguiente plan de negocio:

4.4 Plan de negocios para implementar proyectos Android para medición inteligente.

Un plan de negocio es un conjunto de objetivos trazados al inicio del desarrollo de un proyecto para su futura proyección y evaluación. En él se hace una planificación de la empresa y sirve en muchas ocasiones para convencer a bancos o futuros inversionistas de la realidad de la empresa mediante sus datos.

En otras palabras se puede decir que un plan de negocio es una guía verbal y gráfica del modelo que se seguirá y se aprobará en la empresa donde será implementado, es decir que es una explicación escrita del modelo de negocio de la compañía que se pretende crear.

Es de especial atención realizar una práctica común en función de renovar y actualizar el plan de negocios, puesto que usualmente queda obsoleto. Algunas veces quedan desestimados y se cree que lo más importante dentro de este proceso de planificación es que con él, el administrador de la empresa adquiere un mejor entendimiento de los principales procesos de negocio dentro de la misma.[68]

El prototipo del plan de negocio es:

1. Tener definido el modelo de negocio y sus acciones estratégicas.
2. Determinar la viabilidad económico-financiera del proyecto empresarial.
3. Definir la imagen general de la empresa ante terceras personas.

Las principales características que presenta un plan de negocio son las siguientes:

- Constituye una herramienta de gran utilidad para el propio equipo de promotores ya que permite detectar errores y planificar adecuadamente y con anterioridad la creación y desarrollo del negocio al comienzo de la inversión.
- Facilita la obtención de la financiación bancaria, ya que contiene la previsión de estados económicos y financieros del negocio e informa adecuadamente sobre su viabilidad y solvencia.
- Facilita la negociación con proveedores.
- Captación de nuevos socios o colaboradores.
- Definir diversas etapas que faciliten la medición de sus resultados.
- Establecer metas a corto y mediano plazo.
- Definir con claridad los resultados finales esperados.
- Establecer criterios de medición para saber cuáles son sus logros.
- Identificar posibles oportunidades para aprovecharlas en su aplicación.
- Involucrar en su elaboración a los ejecutivos que vayan a participar en su aplicación.
- Nombrar un coordinador o responsable de su aplicación.

- Prever las dificultades que puedan presentarse y las posibles medidas correctivas.
- Tener programas para su realización.
- Ser claro, conciso e informativo.

Hoy en día, la utilización y descarga de aplicaciones es una de las tendencias más marcadas que dominan el mercado digital y el mundo de los negocios. Es por ello que se puede decir que esto representa una gran oportunidad para las empresas en su búsqueda para llegar a clientes potenciales y con ellos llegar a convertirse en instituciones competitivas.

Independientemente del camino por el que se opte, se debe tener en cuenta una serie de consejos al momento de implementar una estrategia de negocios para dispositivos móviles.

Enganchar a los clientes

Incentivar a los clientes con algún descuento u obsequio estimula a los mismos a descargar y utilizar la aplicación creada y con ellos difundirla y compartirla ganando de esta forma mayor cantidad de usuarios.

Gratuidad

El hecho de permitir que se realicen las descargas de las aplicaciones de forma gratuita ayudará a la empresa a ganar mas clientes clientes, reforzando el negocio ante la posibilidad de aumentar ventas en otras áreas.

Darla a conocer

Apostar por las redes sociales para promover la aplicación creada es una de las principales estrategias que se deben tomar. Es por ello que se hace necesario la implementación de una estrategia efectiva de publicidad con la cual se logre captar la atención del cliente.

Links Sociales

Por la utilidad de las redes sociales más navegadas hasta el momento como es el caso de Facebook, Twitter, Google+, Youtube y Pinterest se deben hacer enlaces inteligentes desde las aplicaciones creadas las cuales ayudarán a darle publicidad a la aplicación.

Conocer el mensaje que se quiere dar

Antes de desarrollar la aplicación se debe aclarar el concepto de la misión, es decir qué es lo que se quiere ofrecer y cuál es la imagen que se busca dar a los clientes.

Además, al momento de sacarlo al mercado es preciso estar seguro que se desarrolló un producto fuerte y que gustara a la mayoría de los usuarios, para que así, la marca no sufra pérdidas.

Elegir bien los dispositivos

Antes de salir al mercado con la nueva aplicación se debe tener bien definidos en qué dispositivos móviles correrá la aplicación sin problemas, puesto que no todos iguales ni poseen las mismas características. El mercado de los teléfonos inteligentes está siendo cada día más dominado por Android y iPhone, por ende éstos con las más recomendados para el uso de esta aplicación.[69]

4.4.1 Plan de negocio para la implementación de la aplicación Monitor EE del sistema Android para medición inteligente

Las aplicaciones del sistema operativo Android hoy en día son muy utilizadas, al punto que han llegado a formar parte de las actividades diarias, siendo indispensables para sacarle todo el partido posible al móvil. Muchas son las funcionalidades de las aplicaciones realizadas hasta este momento y Monitor EE, aplicación creada en esta investigación, cumple con todas las exigencias necesarias para estar dentro de los parámetros establecidos de calidad.

Previa la creación de Monitor EE se ha planteado varios objetivos para ayudar a verificar la viabilidad del proyecto, los mismos que se detallan a continuación.

- Recuperar la inversión realizada para el emprendimiento empresarial teniendo en cuenta el costo proyecto.
- Desarrollar un mercado atractivo con la aplicación creada para extender su uso a la mayoría de los usuarios de diferentes partes del país.
- Desarrollar otras aplicaciones en un futuro breve que satisfagan las expectativas del consumidor, alcanzando un 90% de satisfacción del cliente.

La visión principal de este proyecto es que a mediano plazo se proporcionen las facilidades necesarias para la implementación de sistemas de medición inteligente en sectores de la urbe y por medio de los resultados obtenidos, expandir el sistema a más sectores de la sociedad.

Debido al constante aumento del uso de teléfonos inteligentes y tablets, el potencial del mercado de desarrollo de aplicaciones móviles sigue en aumento. Actualmente 1200 millones de usuarios utilizan aplicaciones móviles y se espera que para 2017 haya 4400 millones.

De acuerdo al mercado se pronostica que para fines del 2015 las descargas de aplicaciones a través de teléfonos inteligentes tendrán los siguientes datos: el 89% corresponderán a Android, 34% iOS, 4% Windows Phone y 3% Blackberry iOS. Por el lado de las tablets, se prevén unas 14 mil millones de descargas, de las cuales el 54% Android, 42% serán de iPad, 2% Kindle de Amazon y 2% dispositivos de Windows. Basándose en estas estadísticas, en el periodo 2012-2017 el número de usuarios de aplicaciones móviles crecerá a una tasa anual del 29.8%. Como se puede observar las cifras son muy ventajosas para el sistema Android por lo que se requiere seguir trabajando en función de esto.

Promover un sistema de capacitación al personal basado en un enfoque de servicio al cliente. Establecer de políticas de servicio alineadas en enfoques de satisfacción integral para los clientes. Destacar el valor agregado que se prestara a través de un servicio diferenciado y estrategias de comunicación adecuadas.

Es necesario plantear una estrategia encaminada a conocer el mercado y el ámbito social en el que la aplicación va a ser usada, es por ello que a continuación se detallan algunas de las ideas para lograr introducir a Monitor EE en la vida cotidiana de los usuarios.

- Obtener información referente al consumidor, mercado, principales competidores, hábitos de compra, gustos, preferencias y canales de distribución relacionados al mercado potencial en sector urbano.
- Interacción con el mercado objetivo a través de redes sociales tales como twitter y Facebook para conocer las preferencias y necesidades del cliente.

Se dará a conocer de este nuevo servicio de Android a través de la publicidad y promoción.

Dado que el costo final del proyecto fue de 510 USD, y que las ganancias negativas que existen solo se pueden ver como estos gastos, quedarían únicamente las ganancias positivas, por lo que se puede decir que el proyecto es rentable. Determinar la viabilidad financiera del proyecto, mostrar las proyecciones de ingresos y gastos para conjugarlos con la inversión, y así obtener los estados financieros necesarios para aplicar herramientas de evaluación de proyectos y determinar si la idea de negocio es factible, estos datos deben ir de la mano de todo proyecto inversionista.

4.4.2 Pruebas a la aplicación

Las pruebas de software son técnicas clásicas para la verificación y validación del mismo, lo cual garantiza su calidad. Estas pruebas se ejecutan con el fin de detectar defectos del sistema. Para ello, el diseño de las pruebas se basa en la creación de casos de prueba cuya ejecución permitirá observar posibles síntomas de defectos.

Las pruebas realizadas al producto fueron pruebas de caja negra, las cuales se centran en los requisitos funcionales del software e intenta encontrar errores en las siguientes categorías:

- Funciones incorrectas o ausentes.
- Errores de interfaz.
- Errores en estructura de datos.
- Errores de rendimiento.
- Errores de inicialización y de terminación.

Existen varias técnicas para realizar las pruebas de caja negra. Las cuales son:

- Particiones de Equivalencia.
- Análisis de Valores Límite.
- Tablas de Decisión.

- Arreglos Ortogonales.
- Pruebas exploratorias

La técnica de partición de equivalencia intenta seleccionar el subconjunto con la probabilidad más alta de encontrar la mayoría de los errores dividiéndose en dos pasos: Identificación de las clases de equivalencia, las cuales son detectadas tomando cada condición de la entrada y repartiéndola en dos o más grupos, y definición de los casos de prueba las cuales se realizan a partir de los mismos.

La técnica de análisis de valores límite, se enfoca en la identificación de los casos de prueba asociados con los valores límites del dominio de la función o elementos de entrada al software.

La técnica de tablas de decisión representa relaciones lógicas entre las condiciones (entradas) y las acciones (salidas). Esta técnica es utilizada para registrar reglas del negocio complejas basadas en un conjunto de condiciones, las cuales deben ser implementadas en el sistema. Estas tablas creadas son muy rigurosas en su definición de un análisis lógico.

La técnica de arreglos ortogonales es en proceso en 2 dimensiones. Con ella se hace una matriz gráfica con la cual se garantiza que todas sus filas que corresponden a casos de pruebas sean probadas. Esta técnica es utilizada en situaciones donde se tienen muchos casos de prueba para diseñar y ejecutar.

La técnica de pruebas exploratorias se utiliza para determinar cómo funciona realmente el software en procesos de negocio sencillos, donde únicamente puede existir una respuesta real.

Para el desarrollo de las pruebas en esta investigación se utilizaron dos técnicas: partición de equivalencia y pruebas exploratorias.

Casos de pruebas	Descripción del caso de prueba	Técnica utilizada
------------------	--------------------------------	-------------------

Caso de prueba #1: Registrar Usuario	Este caso de prueba pretende que el usuario antes de entrar a la aplicación Monitor EE se registre con su usuario y contraseña para acceder a sus datos.	Partición de equivalencia
Caso de Prueba #2: Acceder a gráfico lineal	En este caso de prueba se accede al gráfico lineal donde muestra el consumo de electricidad mediante él.	Pruebas exploratorias
Caso de Prueba #3: Acceder a gráfico de barra	En este caso de prueba se accede al gráfico de barras donde muestra el consumo mensual con su respectivo valor a pagar.	Pruebas exploratorias
Caso de Prueba #4: Acceder a perfil del cliente	En este caso de prueba se accede al perfil del cliente donde se encuentra el contenedor de sus datos.	Pruebas exploratorias
Caso de Prueba #5: Acceder a alertas	En este caso de prueba se accede a las alertas que informa la aplicación las cuales son enviadas en caso de algún cambio en el valor de la notificación introducida por el usuario.	Pruebas exploratorias
Caso de Prueba #6: Realizar pago al banco	En este caso de prueba se le brinda al usuario la posibilidad de realizar el pago al banco donde es afiliado y pagar con tarjeta de crédito o debito.	Partición de equivalencia

Caso de Prueba #7: Actualizar dato Valor de notificación	En este caso de prueba el usuario puede actualizar el valor de la notificación manualmente que no es más que el consumo energético promedio de su vivienda	Partición de equivalencia
---	--	---------------------------

Tabla 4.1 Definición de casos de pruebas

Fuente: Grupo de investigación

Caso de Prueba #1: Registrar Usuario

Este caso de prueba está compuesto por el caso de uso registrar usuario y se pretende que el usuario se logee en el sistema antes de acceder a la aplicación Monitor EE.

Caso de Prueba: Registrar Usuario	
Técnica:	Partición de Equivalencia
Proceso:	Registrar el usuario con sus respectivos datos de nombre de usuario y contraseña.
Variable:	Usuario y contraseña
Clases	Valida: Datos correctos de usuario y contraseña
	Inválida: Datos incorrectos de usuario y contraseña

Caso de Prueba #2: Acceder a gráfico lineal.

En este caso de prueba se accede a la página donde se encuentra el gráfico lineal que muestra el comportamiento del consumo de energía eléctrica del hogar.

Técnica de prueba utilizada: Pruebas exploratorias.

Resultado esperado

Presentación de la interfaz con el gráfico de líneas y su respectivo comportamiento de la energía eléctrica que está consumiendo el hogar.

Caso de Prueba #3: Acceder a gráfico de barra

En este caso de prueba se accede a la página donde se encuentra el gráfico de barras que muestra el consumo mensual de energía eléctrica con su respectivo valor a pagar.

Técnica de prueba utilizada: Pruebas exploratorias.

Resultado esperado

Presentación de la interfaz con el gráfico de barras, donde muestra el consumo mensual del hogar con su respectivo valor a pagar.

Caso de Prueba #4: Acceder a perfil del cliente

En este caso de prueba se accede a la interfaz del perfil del cliente donde se encuentran sus datos personales.

Técnica de prueba utilizada: Pruebas exploratorias.

Resultado esperado

Presentación de la interfaz del perfil del cliente con sus datos expuestos de manera correcta.

Caso de Prueba #5: Acceder a alertas

En este caso de prueba el usuario accede a las alertas para observar si posee alguna, estas se envía en caso que el valor de notificación sea superado por un consumo excesivo.

Técnica de prueba utilizada: Pruebas exploratorias.

Resultado esperado

Acceder a la interfaz de las alertas de la aplicación y mostrar las mismas en caso de que el sistema enviase alguna.

Caso de Prueba #6: Realizar pago al banco

Este caso de prueba está compuesto por el caso de uso registrar usuario y se pretende que el usuario se loguee en el sistema antes de acceder a la aplicación Monitor EE.

Caso de Prueba: Realizar pago al banco	
Técnica:	Partición de Equivalencia
Proceso:	Realizar el pago al banco donde es afiliado el usuario con el valor a pagar mostrado, mediante una tarjeta de crédito o débito.
Variable:	Selección de nombre de banco e insertar número de tarjeta de crédito o débito

Clases	Válida: Datos correcto de nombre del banco y número de tarjeta de crédito o débito
	Inválida: Dato incorrecto de nombre del banco y número correcto tarjeta de crédito o débito
	Inválida: Dato correcto de nombre del banco y número incorrecto de tarjeta de crédito o débito
	Inválida: Datos incorrectos de nombre del banco y número de tarjeta de crédito o débito.

Caso de Prueba #7: Actualizar dato Valor de notificación.

Este caso de prueba actualiza el valor de la notificación insertada por el usuario referido al valor promedio del consumo de energía eléctrica en el hogar.

Caso de Prueba: Actualizar dato Valor de notificación.	
Técnica:	Partición de Equivalencia
Proceso:	Actualizar el dato de valor de notificación insertada manualmente por el usuario.
Variable:	Valor de notificación
Clases	Valida: Insertar el dato de valor de notificación y que este sea guardado en la base de datos
	Inválida: Insertar el dato de valor de notificación y que este no sea guardado en la base de datos, por tanto no se actualiza.

Evaluación de los prototipos desarrollados.

Resultado real de las pruebas funcionales después de su planificación para la evaluación de los prototipos.

Anteriormente se hizo una planificación de siete casos de pruebas funcionales de caja negra, de las cuales tres de ellas fueron realizadas utilizando la técnica de partición de equivalencia y el resto con pruebas exploratorias. En el mismo, se determinaron los resultados esperados por el sistema. A continuación se muestran los resultados reales que brindaron las mismas en la aplicación Monitor EE desarrollada en la presente

investigación

Número de Caso de Prueba	Nombre del caso de prueba	Resultados reales
1	Registrar Usuario	Prueba sin detecciones de fallos. Éxito del sistema.
2	Acceder a gráfico linear	Prueba sin detecciones de fallos. Éxito del sistema.
3	Acceder a gráfico de barra	Prueba sin detecciones de fallos. Éxito del sistema.
4	Acceder a perfil del cliente	Prueba sin detecciones de fallos. Éxito del sistema.
5	Acceder a alertas	Prueba sin detecciones de fallos. Éxito del sistema.
6	Realizar pago al banco	Prueba sin detecciones de fallos. Éxito del sistema.
7	Actualizar dato Valor de notificación.	Prueba sin detecciones de fallos. Éxito del sistema.

Tabla 4.2 Resultados reales de los casos de pruebas

Fuente: Grupo de investigación

CONCLUSIONES

A lo largo del presente proyecto se ha conseguido obtener un conocimiento amplio de este nuevo sistema operativo. Su arquitectura, sus componentes y características, así como el funcionamiento y posibilidades ofrecidas por el sistema Android se han ido conociendo gracias principalmente a la extensa y en general, completa documentación que Google ha puesto a disposición del desarrollador. Especialmente en las primeras fases, esta documentación es útil y fácil de asimilar, lo que permite acercarse poco a poco a las formas y la tecnología de esta nueva plataforma.

Una vez concluidas las principales tareas que forman este proyecto fin de carrera, es el momento en el que se puede hacer balance y crítica de los resultados obtenidos. Así pues, repasando los objetivos inicialmente marcados pueden sacarse las siguientes conclusiones:

- Se creó una base de datos de prueba en la que se almacenó toda la información del consumo eléctrico para una posterior visualización o análisis del usuario en el momento que fuese de su interés.
- El proyecto se debe considerar como un prototipo puesto que en esta primera parte solo se consideran los datos de una base y en base a ellos se muestran cuadros comparativos, sin embargo para un futuro se puede ampliar para controlar el consumo eléctrico a distancia mediante la aplicación móvil. Por ejemplo desactivar luces o algún electrodoméstico desde la aplicación.
- La aplicación Monitor EE tiene cierta posibilidad de éxito, en la medida que el usuario tome conciencia que el consumo energético en el hogar tiene mucha relevancia no solo a nivel personal sino a nivel de la sociedad en general, puesto que el uso racional de la energía beneficia a todos.
- La información de la base de datos tiene que ser actualizada permanentemente de ser posible, la actualización debe en línea, es decir instantánea, puesto que de los datos almacenados en ella depende el correcto funcionamiento de la aplicación.

RECOMENDACIONES

Como líneas futuras de trabajo, se recomiendan los siguientes puntos:

- Se recomienda ampliar la base de datos creadas para ofrecerle al cliente otros servicios en la aplicación que sea de gran utilidad para ellos.
- Se recomienda ampliar la línea de investigación en temas relacionados a domótica y control remoto de equipos del hogar para de esa manera ofrecer al usuario un mayor control sobre su consumo diario sin importar que esté presente para tomar las acciones correctivas.
- Se recomienda socializar por parte de las empresas distribuidoras y los profesionales de la rama cuán importante y el beneficioso es el ahorro de energía en los tiempos actuales, esto se lo puede llevar a cabo por medio de planes de incentivo para aquellos usuarios que demuestren un cambio en sus hábitos de consumo.
- Promover la legislación y las normativas para la implementación de sistemas de medición inteligente ya que con las reglas de juego claras, se puede brindar un servicio de calidad a los usuarios sin correr el riesgo de que existan controversias o malas interpretaciones de los alcances de implementación.

REFERENCIAS

- [1] J. Boal, "Smart Grid," *Comun. Ind. Av.*, 2010.
- [2] A. G. P. Sevilla and F. O. A. Fernández, "Evolución de las redes eléctricas hacia Smart Grid en países de la Región Andina," *Rev. Educ. en Ing.*, vol. 8, no. 15, pp. 48–61, 2013.
- [3] E. Inga and J. Rodríguez, "Estrategias de Negocio Para Medición Inteligente Acoplando Energías Renovables," *Prim. Congr. Int. y Expo Científica Investig. Sosten. Energías Renov. y Efic. Energética*, pp. 281–291, 2013.
- [4] C. J. R. G. Sarduy and C. P. R. V. Felipe, "La Universidad en la transformación hacia las Redes Eléctricas Inteligentes en América Latina," *Univ. y Soc.*, vol. 6, no. 2, 2014.
- [5] M. A. S. Fornié, "Pedro Blanco González."
- [6] S. S. Trundle, A. J. Slavin, J.-P. Martin, and D. J. Hutz, "Remote Device Control and Energy Monitoring."
- [7] M. Weiss, C.-M. Loock, T. Staake, F. Mattern, and E. Fleisch, "Evaluating mobile phones as energy consumption feedback devices," Springer, 2012, pp. 63–77.
- [8] A. Developers, "What is android?," *ht tp//developer. android. com/guide/basics/what-is-android. html*, vol. 2, 2011.
- [9] J. T. Gironés, *El Gran Libro de Android*. Marcombo, 2013, p. 578.
- [10] S. Mota Olmos, "Programación de aplicaciones móviles en Android para la evaluación del conocimiento," 2013.
- [11] G. Herraiz Antón, "ANDROID," 2012.
- [12] J. CHACHA PATÍN, "SISTEMAS OPERATIVOS PARA DISPOSITIVOS MÓVILES," 2011.
- [13] P. Blanco, J. Camarero, and A. Fumero, "Metodología de desarrollo ágil para sistemas móviles Introducción al desarrollo con Android y el iPhone," 2009.
- [14] Y. Juliana, J. Sandoval Cardona, T. Franco, and S. Alberto, "Sistema operativo Android: características y funcionalidad para dispositivos móviles," 2012.
- [15] "Java a Tope: J2me (java 2 Micro Edition). - Sergio Gálvez Rojas, Lucas Ortega Díaz - Google Libros." [Online]. Available: http://books.google.com.ec/books?hl=es&lr=&id=USaAQ0hHQWIC&oi=fnd&pg=PR5&dq=que+es+java&ots=7wP-WKgtin&sig=rwlxeOMgIvqZRYGo_a0txHhymo4&redir_esc=y#v=onepage&q=que es java&f=false.
- [16] "Main Page - KVM." [Online]. Available: http://www.linux-kvm.org/page/Main_Page.
- [17] T. Lindholm, F. Yellin, G. Bracha, and A. Buckley, *The Java Virtual Machine Specification, Java SE 7 Edition*. Addison-Wesley, 2013, p. 876.

- [18] X. Martínez de Morentin Iribarren, “Programación Java,” 2014. [Online]. Available: <http://academica-e.unavarra.es/handle/2454/9024>.
- [19] “Glosario de informática Inglés-Español.” [Online]. Available: <http://quark.fe.up.pt/orca/pub-es/glosario.html#P>.
- [20] J. F. Díaz Amado, “Integración de software para el desarrollo de aplicaciones robóticas basadas en plug-ins Eclipse,” 2012.
- [21] “Android SDK | Android Developers.” .
- [22] “Traductor de Google.” [Online]. Available: <http://translate.google.com/ec/translate?hl=es&sl=en&u=http://www.qemu.org/&prev=/search%3Fq%3DQEMU%26biw%3D1366%26bih%3D657>.
- [23] “GeoPost-it : Geo-posicionamiento informativo para móviles - acarballearesTFC0114memoria.pdf.” [Online]. Available: <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/28201/6/acarballearesTFC0114memoria.pdf>.
- [24] “Metodología para el desarrollo de aplicaciones móviles | Gasca Mantilla | Revista Tecnura.” [Online]. Available: <http://tecnura.udistrital.edu.co/ojs/index.php/revista/article/view/767/752>.
- [25] L. Delía, N. Galdamez, P. J. Thomas, and P. Pesado, “Un análisis experimental tipo de aplicaciones para dispositivos móviles,” 2013.
- [26] Á. Cobo, *PHP y MySQL: Tecnología para el desarrollo de aplicaciones web*. Ediciones Díaz de Santos, 2005, p. 536.
- [27] “Ingeniería dirigida por modelos y calidad de software.” [Online]. Available: <http://rodin.uca.es/xmlui/handle/10498/15744>.
- [28] “Curb, the First Whole Home Energy Intelligence System, Unveiled at DEMO.” [Online]. Available: <http://www.marketwired.com/press-release/curb-the-first-whole-home-energy-intelligence-system-unveiled-at-demo-1969759.htm>.
- [29] “Mobile wireless communications device including conformable, shape-retaining vacuum formed film and related methods.”
- [30] W. R. Romero Kanashiro, “Redes inalámbricas y simulación de WLAN mediante OPNET,” 2013.
- [31] J. Sala Fayos, “Redes wireless industriales,” Jun-2014. [Online]. Available: <http://openaccess.uoc.edu/webapps/o2/handle/10609/33381>.
- [32] C. A. L. Macías, J. A. M. García, G. C. Ayala, and H. C. Meza, “Sistema inalámbrico de sensores basado en ZigBee con topología de malla.”
- [33] “HTC One, análisis.” [Online]. Available: <http://www.xataka.com/analisis/htc-one-analisis>.

- [34] G. Guerra and A. Luisa, "Integración de una red empresarial WLAN 802.11 segura de última generación," Oct. 2014.
- [35] M. A. B. Botero, O. A. M. Gonzales, D. A. Murillo, and S. M. Álvarez, "ANÁLISIS COMPARATIVO DE TECNOLOGÍAS DE TRANSMISIÓN INALÁMBRICA," *Rev. QUID*, no. 17, 2014.
- [36] G. Ortega and M. Estefania, "Caracterización de la técnica Jamming para evaluar el nivel de seguridad en redes inalámbricas IEEE 802.11.," 2014.
- [37] L. H. S. Granados and J. N. P. Castillo, "Laboratorio virtual para el diseño de radio enlaces en un ambiente grid," *Cienc. e Ing. Neogranadina*, vol. 18, no. 2, 2008.
- [38] J. M. R. Padilla, "802.16 vs. WiMAX," *Bit*, no. 156, pp. 62–65, 2006.
- [39] D. Rathore, R. Paulus, A. K. Jaiswal, and M. Kumar, "Performance Evaluation of a WiMAX Mobile Networks using Omnidirectional antennas," 2014.
- [40] "TESIS.pdf." [Online]. Available: <http://tesis.bnct.ipn.mx:8080/bitstream/handle/123456789/12977/TESIS.pdf?sequence=1>.
- [41] S. Kiesler, *Culture of the Internet*. Psychology Press, 2014, p. 480.
- [42] R. Barzanallana, "Desarrollo de Aplicaciones Web . Xampp," vol. 04, p. 4, 2012.
- [43] L. Castilla Santillán, M. Gibert Ginestá, and Ó. Pérez Mora, "Bases de datos en MySQL," 2007.
- [44] "Casas inteligentes," 2012. [En línea]. Disponible en: http://www.clubplaneta.com.mx/casas_inteligentes.htm. [Accessed: 31-Jan-2015].
- [45] "SmartRegions," May-2013. [En Línea]. Disponible en: <http://www.smartregions.net/medidores>. [Accessed: 31-Jan-2015].
- [46] José Enrique González Cornejo, "El Lenguaje de Modelado Unificado," Jan-2008. [Online]. Available: <http://www.docircs.com/uml.htm>. [Accessed: 06-Feb-2015].
- [47] David Gallardo, "Iniciándose en la plataforma Eclipse," p. 12.
- [48] P. R. G. A. Efraín Galo Cuzco Simbaña, "Análisis, Diseño e Implementación de una aplicación con realidad aumentada para teléfonos móviles orientada al turismo." UNIVERSIDAD POLITECNICA SALESIANA, Feb-2012.
- [49] Alejandro Alfonso Pérez García, "Desarrollo de herramientas web de gestión docente." Universidad Politécnica de Cartagena., Dec-2007.
- [50] Enrique González, "¿Qué es PHP? y ¿Para qué sirve? Un potente lenguaje de programación para crear páginas web," p. 6, 2015.
- [51] "Análisis, diseño, implementación e implantación de un sistema Web para gestionar la información y evaluación, de estudiantes y padres de familia utilizando como

herramienta de apoyo la telefonía móvil, en los departamentos de Inspección y DOBE de la Unidad,” Quito.

- [52] “Consejos para reducir el consumo energético.” [Online]. Available: <http://www.emb.cl/electroindustria/articulo.mvc?xid=958&tip=7>. [Accessed: 06-Feb-2015].
- [53] N. Tensión, S. Comunitario, A. Y. A. Especiales, S. Comunitario, and A. Y. A. Especiales, “CONELEC PERIODO : ENERO - DICIEMBRE * CARGOS TARIFARIOS ÚNICOS GENERAL CNEL EL ORO-CNEL ESMERALDAS-CNEL GUAYAS LOS RIOS-CNEL LOS RIOS-CNEL MANABI- GALAPAGOS CNEL EL ORO-CNEL ESMERALDAS-CNEL GUAYAS LOS RIOS-CNEL LOS RIOS-CNEL MANABI- GALAPAGOS,” no. 041, pp. 41–44, 2014.
- [54] “doc_10765_Pliego Tarifario.pdf.” .
- [55] “Ingenieria del Proyecto.pdf.” [Online]. Available: http://www.bvcooperacion.pe/biblioteca/bitstream/123456789/4379/2/BVCI0004134_2.pdf. [Accessed: 02-Feb-2015].
- [56] “Evaluación de proyectos por medio del análisis costo beneficio.” [Online]. Available: <http://www.monografias.com/trabajos99/evaluacion-proyectos-medio-del-analisis-costo-beneficio/evaluacion-proyectos-medio-del-analisis-costo-beneficio.shtml>. [Accessed: 02-Feb-2015].
- [57] “¿Cuál será el futuro de android y cuál es el impacto en el medio? - Documentos de Investigación - Wisim444.” [Online]. Available: <http://www.buenastareas.com/ensayos/Cu%C3%A1l-Ser%C3%A1-El-Futuro-De-Android/1007637.html>. [Accessed: 02-Feb-2015].
- [58] “Diario de un androide » Creación de aplicaciones Android, parte 4: Servicios.” [Online]. Available: <http://androide.hijodeblog.com/2010/06/17/creacion-de-aplicaciones-android-parte-4-servicios/>. [Accessed: 02-Feb-2015].
- [59] “10 aplicaciones Android para Ingenieros.” [Online]. Available: <http://eadic.com/blog/10-aplicaciones-android-para-ingenieros/>. [Accessed: 02-Feb-2015].
- [60] “Desarrollo de aplicaciones para Android* con características de reconocimiento de voz | Zona para desarrolladores Intel®.” [Online]. Available: <https://software.intel.com/es-es/articulos/developing-android-applications-with-voice-recognition-features>. [Accessed: 02-Feb-2015].
- [61] “Las Aplicaciones más buscadas para Android | Novedades de Tecnología.” [Online]. Available: <http://tecnologianovedades.com/las-aplicaciones-mas-buscadas-para-android/>. [Accessed: 02-Feb-2015].
- [62] “Las 5 mejores apps para ahorrar batería en tu Android.” [Online]. Available: <http://noticias.universia.es/ciencia-nn-tt/noticia/2014/03/12/1087205/5-mejores-apps-ahorrar-bateria-android.html>. [Accessed: 02-Feb-2015].
- [63] “¿Porque es importante el ahorro de energia?” [Online]. Available: <http://www.taringa.net/posts/apuntes-y-monografias/3994649/Porque-es-importante-el-ahorro-de-energia.html>. [Accessed: 02-Feb-2015].

- [64] Arq. Fabiola Aranda Chávez, “EL IMPACTO DE LAS APLICACIONES EN LA MERCADOTECNIA.” [Online]. Available: <https://www.holaluz.com/blog/3-apps-que-te-ayudaran-a-ahorrar-en-tu-factura-de-la-luz/>. [Accessed: 02-Feb-2015].
- [65] Mónica Lucía Tapia Marroquín, “ESTUDIO Y DESARROLLO DE APLICACIONES PARA DISPOSITIVOS MÓVILES ANDROID.” UNIVERSIDAD TÉCNICA DEL NORTE FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS Nov-2013.
- [66] “Mi móvil me ayuda a ahorrar energía en casa.” [Online]. Available: <http://www.ciudadano00.es/2013/09/06/apps-de-ahorro-de-energia/>. [Accessed: 02-Feb-2015].
- [67] “El móvil permite medir, reducir y ahorrar en el consumo eléctrico del hogar - RTVE.es.” [Online]. Available: <http://www.rtve.es/noticias/20130919/movil-permite-medir-reducir-ahorrar-consumo-electrico-del-hogar/747370.shtml>. [Accessed: 02-Feb-2015].
- [68] “Plan de negocio.” [Online]. Available: http://es.wikipedia.org/wiki/Plan_de_negocio. [Accessed: 02-Feb-2015].
- [69] Milton Renato Vizúete Flores, “Analizar el aporte de las soluciones de aplicaciones móviles para el Comercio Electrónico en la ciudad de Quito.” RSIDAD POLITECNICA SALESIANA, Mar-2012.

ANEXOS

PROGRAMACION DE APLICACIÓN ANDROID

1.-PROGRAMACIÓN DE INFORMACIÓN DE SERVICIOS

```
public static final String WS_ACCION_SOAP = "DefaultNamespace";

//public static final String metodo = "consultarClientesMetroCOM";

public static final String WS_NAMESPACE = "urn:DefaultNamespace";

//public static final String WS_URL =
"http://192.168.0.3:8080/appEEQWSmonitor/services/solicitudesAndroid?wsdl";

public static final String WS_URL =
"http://192.168.0.3:8080/siteEEQMonitor/wsEEQMonitor/wsEEQMonitor.php?wsdl";

public static final String WS_URL_LOGUEO =
"http://192.168.0.3:8080/siteEEQMonitor/wsEEQMonitor/wsLogueo.php?wsdl";

public static final String WS_URL_LINEAS =
"http://192.168.0.3:8080/siteEEQMonitor/wsEEQMonitor/wsConsultarLecturaGraficoLineas.php?wsdl";

public static final String WS_URL_BARRAS =
"http://192.168.0.3:8080/siteEEQMonitor/wsEEQMonitor/wsConsultarLecturaGraficoBarras.php?wsdl";

public static final String WS_URL_PERFIL =
"http://192.168.0.3:8080/siteEEQMonitor/wsEEQMonitor/wsConsultarInformacionClienteEEMonitor.php?wsdl";

public static final String WS_URL_ACT_MINIMO =
"http://192.168.0.3:8080/siteEEQMonitor/wsEEQMonitor/wsActualizaValorMinimoNotificacion.php?wsdl";

public static final String WS_URL_ALERTA =
"http://192.168.0.3:8080/siteEEQMonitor/wsEEQMonitor/wsConsultarNotificacionesEEMonitor.php?wsdl";
```

2.-PROGRAMACIÓN GRÁFICA LINEAS

```
private XYPlot plot;

ArrayList <Double> Vector = new ArrayList<Double>();

public Double DatoX, DatoY;

private ListView mainListView ;

private ArrayAdapter<String> listAdapter ;

//private static final String metodo = "consultarLecturaGraficoLineas";

private static final String metodo = "consultarLecturaGraficoLineas";

private static final String nombreClase = "Activity - GraficoLineas";

Intent intent = null;

String loginUsuario = "";

String nombreUsuario = "";

String codigoUsuario = "";

String idClienteBuscar = "";
```

```

String[] arrayDatosCodigo = new String[150];
String[] arrayDatosValor = new String[150];

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    plot = (XYPlot) findViewById(R.id.grafica);
    plot.setDomainStep(XYStepMode.INCREMENT_BY_VAL,10 ); //x
    plot.setRangeStep(XYStepMode.INCREMENT_BY_VAL, 0.1 ); //y

    //recuperar parámetros desde activity anterior
    SharedPreferences settings = getSharedPreferences("pasoInformacion", MODE_PRIVATE);
    String infoCliente = settings.getString("codigoUsuario","codigoUsuarioDef");
    if (infoCliente == null)
    {
        Toast.makeText(getApplicationContext(), "No se pudo cargar información de
actividad anterior",Toast.LENGTH_SHORT).show();
        finish(); //cerrar pantalla
    }

    idClienteBuscar = infoCliente;
    new ConsultarData().execute();

    DatoX = 0.0;
    DatoY = 0.0;
    for (int i=0; i<100; i++)
    {
        Vector.add(DatoX);
        DatoX += 5;
        //DatoY = Math.random()*5; //randomico con valor máximo
        Vector.add(DatoY);
        DatoY = DatoX + 10;
    }
}

private void Refrescar() {

```

```

        // TODO Auto-generated method stub
        this.runOnUiThread(Graf);
    }
    private Runnable Graf = new Runnable(){
        public void run() {
            DatoX = 0.0;
            DatoY = 0.0;
            for (int i=0; i<100; i++)
            {
                DatoX += 0.5;
                Vector.remove(0);
                Vector.add(DatoX);
                DatoY = Math.random()*5; //randomico con valor máximo
                Vector.remove(0);
                Vector.add(DatoY);
            }
            XYSeries series = new SimpleXYSeries(Vector,
SimpleXYSeries.ArrayFormat.XY_VALS_INTERLEAVED,"señal");
            LineAndPointFormatter seriesFormat = new
LineAndPointFormatter(Color.rgb(127, 255, 0),0x000000,0x000000,null);
            plot.clear();
            plot.addSeries(series, seriesFormat);
            //plot.redraw();
        }
    };

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is present.
        getMenuInflater().inflate(R.menu.main, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();

```



```

        if (id == R.id.action_settings) {
            return true;
        }
        return super.onOptionsItemSelected(item);
    }

    private class ConsultarData extends AsyncTask<String, Float, Integer>
    {
        SoapPrimitive resultado = null;

        protected void onPreExecute()
        {
        }

        protected Integer doInBackground(String... urls)
        {
            try
            {
                Rastro.EscribirEnConsola(nombreClase, "M", "INICIO
EJECUCION", "doInBackground");

                SoapObject request = new
SoapObject(InformacionServicios.WS_NAMESPACE, metodo);

                request.addProperty("idCliente", idClienteBuscar);

                SoapSerializationEnvelope sobre = new
SoapSerializationEnvelope(SoapEnvelope.VER11);

                sobre.setOutputSoapObject(request);

                HttpTransportSE transporte = new
HttpTransportSE(InformacionServicios.WS_URL_LINEAS);

                transporte.call(InformacionServicios.WS_ACCION_SOAP, sobre);

                Object resultadoPHP = sobre.getResponse();

                final String KEY_ITEM = "raiz"; // parent node

                final String KEY_ID = "codigo";

                final String KEY_VALOR = "valor";

                ArrayList<HashMap<String, String>> menuItems = new
ArrayList<HashMap<String, String>>();

                XMLParser parser = new XMLParser();

                Document doc = parser.getDomElement(resultadoPHP.toString()); //
getting DOM element

                NodeList nl = doc.getElementsByTagName("datos");

                Integer tamano = nl.getLength();

```

```

        arrayDatosCodigo = new String[150];
        arrayDatosValor = new String[150];
        for (int i = 0; i < tamaño; i++) {
            Element e = (Element) nl.item(i);
            String codigo = parser.getValue(e, KEY_ID);
            String valor = parser.getValue(e, KEY_VALOR);
            arrayDatosCodigo[i] = codigo;
            arrayDatosValor[i] = valor;
        }
        Rastro.EscribirEnConsola(nombreClase, "M", "FIN
EJECUCION","doInBackground");
        return 0;
    }
    catch (Exception ex)
    {
        Rastro.EscribirEnConsola(nombreClase, "E",
ex.getMessage(),"doInBackground");
        return 1;
    }
}
protected void onProgressUpdate(Float... valores)
{
}
protected void onPostExecute(Integer bytes)
{
    try
    {
        Rastro.EscribirEnConsola(nombreClase, "M", "INICIO
EJECUCION","onPostExecute");

        for (Integer i=0; i<100; i++)
        {
            //Vector.add(Double.parseDouble(i.toString()));

            Vector.add(Double.parseDouble(arrayDatosCodigo[i]));

            Vector.add(Double.parseDouble(arrayDatosValor[i]));
        }
    }
}

```



```

        ArrayList<Number> Vector = new ArrayList<Number>();

private static final String NO_SELECTION_TXT = "Toque para ver Consumo(kw/h).";
private XYPlot plot;

private CheckBox series1CheckBox;
private CheckBox series2CheckBox;
private Spinner spRenderStyle, spWidthStyle, spSeriesSize;
private SeekBar sbFixedWidth, sbVariableWidth;

private XYSeries series1;
private XYSeries series2;
private enum SeriesSize {
    Banco,
    Pichincha,
    Produbanco,
    Guayaquil,
    Bolivariano,
    Pacifico
}

// Create a couple arrays of y-values to plot:
Number[] series1Numbers;
Number[] series2Numbers;
String[] series3Data;
String[] series1Data;

private MyBarFormatter formatter1;
private MyBarFormatter selectionFormatter;
private TextLabelWidget selectionWidget;
private Pair<Integer, XYSeries> selection;

@Override
public void onCreate(Bundle savedInstanceState)
{
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_barras);
}

```

```

// initialize our XYPlot reference:
plot = (XYPlot) findViewById(R.id.mySimpleXYPlot);
formatter1 = new MyBarFormatter(Color.argb(200, 100, 150, 100), Color.LTGRAY);
//formatter2 = new MyBarFormatter(Color.argb(200, 100, 100, 150), Color.LTGRAY);
selectionFormatter = new MyBarFormatter(Color.YELLOW, Color.WHITE);
selectionWidget = new TextLabelWidget(plot.getLayoutManager(), NO_SELECTION_TXT,
    new SizeMetrics(
        PixelUtils.dpToPix(100), SizeLayoutType.ABSOLUTE,
        PixelUtils.dpToPix(100), SizeLayoutType.ABSOLUTE),
    TextOrientationType.HORIZONTAL);

selectionWidget.getLabelPaint().setTextSize(PixelUtils.dpToPix(16));
// add a dark, semi-transparent background to the selection label widget:
Paint p = new Paint();
p.setARGB(100, 0, 0, 0);
selectionWidget.setBackgroundPaint(p);

selectionWidget.position(
    0, XLayoutStyle.RELATIVE_TO_CENTER,
    PixelUtils.dpToPix(45), YLayoutStyle.ABSOLUTE_FROM_TOP,
    AnchorPosition.TOP_MIDDLE);
selectionWidget.pack();

// reduce the number of range labels
plot.setTicksPerRangeLabel(3);
plot.setRangeLowerBoundary(0, BoundaryMode.FIXED);
plot.getGraphWidget().setGridPadding(30, 10, 30, 0);

plot.setTicksPerDomainLabel(2);

plot.setOnTouchListener(new View.OnTouchListener() {
    @Override
    public boolean onTouch(View view, MotionEvent motionEvent) {
        if(motionEvent.getAction() == MotionEvent.ACTION_DOWN) {
            onPlotClicked(new PointF(motionEvent.getX(), motionEvent.getY()));
        }
    }
}

```

```

        return true;
    }
});

spRenderStyle = (Spinner) findViewById(R.id.spRenderStyle);

//ArrayAdapter <BarRenderer.BarRenderStyle> adapter = new ArrayAdapter
<BarRenderer.BarRenderStyle> (this, android.R.layout.simple_spinner_item, SeriesSize.values() );

ArrayAdapter <SeriesSize> adapter = new ArrayAdapter <SeriesSize> (this,
android.R.layout.simple_spinner_item, SeriesSize.values() );

adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
spRenderStyle.setAdapter(adapter);
spRenderStyle.setSelection(BarRenderer.BarRenderStyle.OVERLAID.ordinal());
spRenderStyle.setOnItemClickListener(new OnItemSelectedListener() {
    public void onItemSelected(AdapterView<?> arg0, View arg1,int arg2, long arg3) {
        if (arg2 > 0)
        {
            /*Pichincha,
            Produbanco,
            Guayaquil,
            Bolivariano,
            Pacifico*/

String url = "";

            switch (arg2) {
                case 1: url = "http://www.pichincha.com";
                    break;
                case 2: url = "https://www.produbanco.com/GFPNet/";
                    break;
                case 3: url = "http://www.bancoguayaquil.com";
                    break;
                case 4: url = "http://www.bolivariano.com.ec";
                    break;
                case 5: url = "http://www.pacifico.com.ec";
                    break;
                default:
                    break;
            }

```

```

        Intent browserIntent = new Intent("android.intent.action.VIEW",
            Uri.parse(url));
        startActivity(browserIntent);
    }
}

@Override
public void onNothingSelected(AdapterView<?> arg0) {
}

//updatePlot();

//recuperar parámetros desde activity anterior
SharedPreferences settings = getSharedPreferences("pasoInformacion", MODE_PRIVATE);
String infoCliente = settings.getString("codigoUsuario","codigoUsuarioDef");
if (infoCliente == null)
{
    Toast.makeText(getApplicationContext(), "No se pudo cargar información de
actividad anterior",Toast.LENGTH_SHORT).show();

    finish(); //cerrar pantalla
}

idClienteBuscar = infoCliente;
new ConsultarData().execute();
}

private void updatePlot() {

    Iterator<XYSeries> iterator1 = plot.getSeriesSet().iterator();
    while(iterator1.hasNext()) {
        XYSeries setElement = iterator1.next();
        plot.removeSeries(setElement);
    }

    series1 = new SimpleXYSeries(Vector, SimpleXYSeries.ArrayFormat.XY_VALS_INTERLEAVED,
"Consumo");

    plot.addSeries(series1, formatter1);
    MyBarRenderer renderer = ((MyBarRenderer)plot.getRenderer(MyBarRenderer.class));
    renderer.setBarWidth(40);
}

```

```

    plot.redraw();
}

private void onPlotClicked(PointF point) {
    if (plot.getGraphWidget().getGridRect().contains(point.x, point.y)) {
        Number x = plot.getXVal(point);
        Number y = plot.getYVal(point);

        selection = null;
        double xDistance = 0;
        double yDistance = 0;

        // find the closest value to the selection:
        for (XYSeries series : plot.getSeriesSet()) {
            for (int i = 0; i < series.size(); i++) {
                Number thisX = series.getX(i);
                Number thisY = series.getY(i);
                if (thisX != null && thisY != null) {
                    double thisXDistance =
                        LineRegion.measure(x, thisX).doubleValue();
                    double thisYDistance =
                        LineRegion.measure(y, thisY).doubleValue();
                    if (selection == null) {
                        selection = new Pair<Integer, XYSeries>(i, series);
                        xDistance = thisXDistance;
                        yDistance = thisYDistance;
                    } else if (thisXDistance < xDistance) {
                        selection = new Pair<Integer, XYSeries>(i, series);
                        xDistance = thisXDistance;
                        yDistance = thisYDistance;
                    } else if (thisXDistance == xDistance &&
                        thisYDistance < yDistance &&
                        thisY.doubleValue() >= y.doubleValue()) {
                        selection = new Pair<Integer, XYSeries>(i, series);
                        xDistance = thisXDistance;
                        yDistance = thisYDistance;
                    }
                }
            }
        }
    }
}

```



```

        }
    }
}

} else {
    // if the press was outside the graph area, deselect:
    selection = null;
}

if(selection == null) {
    selectionWidget.setText(NO_SELECTION_TXT);
} else {
    /*selectionWidget.setText("Selected: " + selection.second.getTitle() +
        " Value: " + selection.second.getY(selection.first));*/
    String dataActual = "";
    int indiceActual = (int) selection.second.getX(selection.first);
    if (indiceActual >= 0)
        dataActual = series3Data[(int) selection.second.getX(selection.first)];

    selectionWidget.setText("Consumo(Kw/h):[" + dataActual + ":\n" +
        selection.second.getY(selection.first) + "]");
}
plot.redraw();
}

private void onS1CheckBoxClicked(boolean checked) {
    if (checked) {
        plot.addSeries(series1, formatter1);
    } else {
        plot.removeSeries(series1);
    }
    plot.redraw();
}

/*private void onS2CheckBoxClicked(boolean checked) {
    if (checked) {

```

```

        plot.addSeries(series2, formatter2);
    } else {
        plot.removeSeries(series2);
    }
    plot.redraw();
}*/

class MyBarFormatter extends BarFormatter {
    public MyBarFormatter(int fillColor, int borderColor) {
        super(fillColor, borderColor);
    }

    @Override
    public Class<? extends SeriesRenderer> getRendererClass() {
        return MyBarRenderer.class;
    }

    @Override
    public SeriesRenderer getRendererInstance(XYPlot plot) {
        return new MyBarRenderer(plot);
    }
}

class MyBarRenderer extends BarRenderer<MyBarFormatter> {

    public MyBarRenderer(XYPlot plot) {
        super(plot);
    }

    @Override
    public MyBarFormatter getFormatter(int index, XYSeries series) {
        if(selection != null &&
            selection.second == series &&
            selection.first == index) {
            return selectionFormatter;
        } else {

```

```

        return getFormatter(series);
    }
}

private class ConsultarData extends AsyncTask<String, Float, Integer>
{
    SoapPrimitive resultado = null;

    protected void onPreExecute()
    {
    }

    protected Integer doInBackground(String... urls)
    {
        try
        {
            Rastro.EscribirEnConsola(nombreClase, "M", "INICIO
EJECUCION", "doInBackground");

            SoapObject request = new
SoapObject(InformacionServicios.WS_NAMESPACE, metodo);

            request.addProperty("idCliente", idClienteBuscar);

            SoapSerializationEnvelope sobre = new
SoapSerializationEnvelope(SoapEnvelope.VER11);

            sobre.setOutputSoapObject(request);

            HttpTransportSE transporte = new
HttpTransportSE(InformacionServicios.WS_URL_BARRAS);

            transporte.call(InformacionServicios.WS_ACCION_SOAP, sobre);

            //resultado = (SoapPrimitive)sobre.getResponse();

            Object resultadoPHP = sobre.getResponse();

            final String KEY_ITEM = "raiz"; // parent node
            final String KEY_ID = "codigo";
            final String KEY_VALOR = "valor";
            final String KEY_VALOR_COBRAR = "valorPorCobrar";

            ArrayList<HashMap<String, String>> menuItems = new
ArrayList<HashMap<String, String>>();

```

```

XMLParser parser = new XMLParser();
//String xml = parser.getXmlFromUrl(URL); // getting XML
Document doc = parser.getDomElement(resultadoPHP.toString()); //
getting DOM element

NodeList nl = doc.getElementsByTagName("datos");
// looping through all item nodes <item>
Integer tamaño = nl.getLength();
/*arrayDatosCodigo = new String[150];
arrayDatosValor = new String[150];*/
series1Numbers = new Number[tamaño];
series2Numbers = new Number[tamaño];
series3Data = new String[tamaño];
series1Data = new String[tamaño];
for (int i = 0; i < tamaño; i++) {
    Element e = (Element) nl.item(i);

    String codigo = parser.getValue(e, KEY_ID);
    String valor = parser.getValue(e, KEY_VALOR);
    String dataMesAnio = parser.getValue(e, "infoMesAnio");
    valorCobrar = parser.getValue(e, KEY_VALOR_COBRAR);
    series1Numbers[i] = Float.parseFloat(codigo);
    series2Numbers[i] = Float.parseFloat(valor);
    series3Data[5-i] = dataMesAnio;
}
Rastro.EscribirEnConsola(nombreClase, "M", "FIN
EJECUCION", "doInBackground");
return 0;
}
catch (Exception ex)
{
    Rastro.EscribirEnConsola(nombreClase, "E",
ex.getMessage(), "doInBackground");
return 1;
}
}

protected void onProgressUpdate(Float... valores)

```

```

    {
    }

    protected void onPostExecute(Integer bytes)
    {
        try
        {
            Rastro.EscribirEnConsola(nombreClase, "M", "INICIO
EJECUCION", "onPostExecute");

            //for (Integer i=0; i<series1Numbers.length && i<6; i++)

            TextView objInfoValorCobrar = (TextView)
findViewById(R.id.txtInfo);

            Double valor = Double.parseDouble(valorCobrar);
            valor = (double) Math.round(valor);

            objInfoValorCobrar.setText("Deuda: [ " +
valor.toString() + " ] Pagar en : " );

            int contador = 0;
            for (Integer i=series1Numbers.length-1; i>=0 &&
contador<6; i--)
            {
                Vector.add(contador);
                Vector.add(series2Numbers[i]);
                contador++;
            }

            updatePlot();
            Rastro.EscribirEnConsola(nombreClase, "M", "FIN
EJECUCION", "onPostExecute");
        }
        catch (Exception ex)
        {
            Rastro.EscribirEnConsola(nombreClase, "E",
ex.getMessage(), "onPostExecute");
        }
    }
}

```

```
    }  
}
```

4.- PRORAMACIÓN INFORMACIÓN CLIENTE

```
public class ClientePestanaDatos extends Activity  
{  
    private static final String nombreClase = "ClientePestanaDatos";  
    private static final String metodoBuscar =  
"consultarInformacionClienteEEMonitor";  
    private static final String metodoActualizar =  
"actualizaValorMinimoNotificacion";  
  
    private String infoCliente = "";  
    private String idClienteBuscar = "";  
    private boolean flagEjecucion = false;  
    private boolean flagGuardarCliente = false;  
  
    StringBuilder objStrMsgWarging = null;  
  
    String clie_id = "";  
    String clie_cedula = "";  
    String clie_apellidos = "";  
    String clie_nombres = "";  
    String clie_medidor = "";  
    String clie_direccion = "";  
    String clie_usuario = "";  
    String clie_telefono = "";  
    String clie_suministro = "";  
    String valorMinimoNotif = "";  
    String respuesta = "";  
  
    @Override  
    public void onCreate(Bundle savedInstanceState)  
    {  
        try
```

```

        {
            Rastro.EscribirEnConsola(nombreClase, "M", "INICIO
EJECUCION", "onCreate");

            objStrMsgWaring = new StringBuilder();
            super.onCreate(savedInstanceState);
            setContentView(R.layout.activity_cli_pest_datos);
            SharedPreferences settings = getSharedPreferences("pasoInformacion",
MODE_PRIVATE);

            String infoCliente =
settings.getString("codigoUsuario", "codigoUsuarioDef");

            idClienteBuscar = infoCliente;

            new BuscarCliente().execute("");

            Rastro.EscribirEnConsola(nombreClase, "M", "FIN EJECUCION
EJECUCION", "onCreate");
        }
        catch (Exception ex)
        {
            Rastro.EscribirEnConsola(nombreClase, "E",
ex.getMessage(), "onCreate");
        }
    } //onCreate

    public void cmdActualizar_Onclick(View Botton)
    {
        try
        {
            Rastro.EscribirEnConsola(nombreClase, "M", "INICIO
EJECUCION", "cmdActualizar_Onclick");

            EditText objValorMinimoNotif = (EditText)
findViewById(R.id.txtValorMinimo);

            valorMinimoNotif = objValorMinimoNotif.getText().toString();

            if (valorMinimoNotif.equals(""))
            {

```

```

        Toast.makeText(getApplicationContext(), "No ha ingresado el
valor mínimo.", Toast.LENGTH_SHORT).show();

        objValorMinimoNotif.setFocusable(true);

        return;
    }

    new ActualizarRangoMinimo().execute("");

    Rastro.EscribirEnConsola(nombreClase, "M", "FIN
EJECUCION", "cmdActualizar_Onclick");
}
catch (Exception ex)
{
    ex.printStackTrace();
    System.out.println(ex.getMessage());
}
}

private class BuscarCliente extends AsyncTask<String, Float, Integer>
{
    SoapPrimitive resultado = null;

    protected void onPreExecute()
    {
    }

    protected Integer doInBackground(String... urls)
    {
        try
        {
            Rastro.EscribirEnConsola(nombreClase, "M", "INICIO
EJECUCION", "doInBackground");

            SoapObject request = new
SoapObject(InformacionServicios.WS_NAMESPACE, metodoBuscar);

            flagEjecucion = false;

            request.addProperty("idCliente", idClienteBuscar);

            SoapSerializationEnvelope sobre = new
SoapSerializationEnvelope(SoapEnvelope.VER11);

            //sobre.dotNet = true;

            sobre.setOutputSoapObject(request);

            HttpTransportSE transporte = new

```



```

HttpTransportSE(InformacionServicios.WS_URL_PERFIL);

        transporte.call(InformacionServicios.WS_ACCION_SOAP, sobre);

        //resultado = (SoapPrimitive)sobre.getResponse();
        Object resultadoPHP = sobre.getResponse();

        XMLParser parser = new XMLParser();
        Document doc = parser.getDomElement(resultadoPHP.toString()); //
getting DOM element

        NodeList nl = doc.getElementsByTagName("infoCliente");
        // looping through all item nodes <item>
        Integer tamaño = nl.getLength();
        if (tamaño == 1)
        {
            Element e = (Element) nl.item(0);

            clie_id = idClienteBuscar;
            clie_cedula = parser.getValue(e,
"clie_cedula");
            clie_apellidos = parser.getValue(e,
"clie_apellidos");
            clie_nombres = parser.getValue(e,
"clie_nombres");
            clie_medidor = parser.getValue(e,
"clie_medidor");
            clie_direccion = parser.getValue(e,
"clie_direccion");
            clie_usuario = parser.getValue(e,
"clie_usuario");
            clie_telefono = parser.getValue(e,
"clie_telefono");
            clie_suministro = parser.getValue(e,
"clie_suministro");
            valorMinimoNotif = parser.getValue(e,
"clie_valor_notif");

            flagEjecucion = true;
        }
        else
            flagEjecucion = false;

        Rastro.EscribirEnConsola(nombreClase, "M", "FIN

```

```

EJECUCION", "doInBackground");

        return 0;
    }
    catch (Exception ex)
    {
        Rastro.EscribirEnConsola(nombreClase, "E",
ex.getMessage(), "doInBackground");
        return 1;
    }
}

protected void onProgressUpdate(Float... valores)
{
}

protected void onPostExecute(Integer bytes)
{
    try
    {
        Rastro.EscribirEnConsola(nombreClase, "M", "INICIO
EJECUCION", "onPostExecute");

        TextView objTxtIdCliente = (TextView)
findViewById(R.id.lblIdCliente);

        objTxtIdCliente.setText("ID CLIENTE : " + clie_id);

        EditText objTxtCedula = (EditText)
findViewById(R.id.txtCedula);

        if (clie_cedula == null) clie_cedula = "";
        objTxtCedula.setText(clie_cedula);
        objTxtCedula.setEnabled(false);

        EditText objTxtApellidos = (EditText)
findViewById(R.id.txtApellidos);

        if (clie_apellidos == null) clie_apellidos = "";
        objTxtApellidos.setText(clie_apellidos);
        objTxtApellidos.setEnabled(false);
    }
}

```

```

findViewById(R.id.txtNombres);

EditText objTxtNombres = (EditText)

if (clie_nombres == null) clie_nombres = "";
objTxtNombres.setText(clie_nombres);
objTxtNombres.setEnabled(false);

findViewById(R.id.txtMedidor);

EditText objTxtMedidor = (EditText)

if (clie_medidor == null) clie_medidor = "";
objTxtMedidor.setText(clie_medidor);
objTxtMedidor.setEnabled(false);

findViewById(R.id.txtDireccion);

EditText objTxtDireccion = (EditText)

if (clie_direccion == null) clie_direccion = "";
objTxtDireccion.setText(clie_direccion);
objTxtDireccion.setEnabled(false);

findViewById(R.id.txtCelular);

EditText objTxtTelefono = (EditText)

if (clie_telefono == null) clie_telefono = "";
objTxtTelefono.setText(clie_telefono);
objTxtTelefono.setEnabled(false);

findViewById(R.id.txtSuministro);

EditText objTxtsuministro = (EditText)

if (clie_suministro.equals("")) clie_suministro = "";
objTxtsuministro.setText(clie_suministro);
objTxtsuministro.setEnabled(false);

findViewById(R.id.txtUsuario);

EditText objTxtUsuario = (EditText)

if (clie_usuario.equals("")) clie_usuario = "";
objTxtUsuario.setText(clie_usuario);
objTxtUsuario.setEnabled(false);

findViewById(R.id.txtValorMinimo);

EditText objValorMinimoNotif = (EditText)

if (valorMinimoNotif.equals("")) valorMinimoNotif = "";
objValorMinimoNotif.setText(valorMinimoNotif);

```

```

        Rastro.EscribirEnConsola(nombreClase, "M", "FIN
EJECUCION","onPostExecute");
    }
    catch (Exception ex)
    {
        Rastro.EscribirEnConsola(nombreClase, "E",
ex.getMessage(),"onPostExecute");
    }
}

private class ActualizarRangoMinimo extends AsyncTask<String, Float, Integer>
{

    SoapPrimitive resultado = null;
    protected void onPreExecute()
    {

    }

    protected Integer doInBackground(String... urls)
    {
        try
        {
            Rastro.EscribirEnConsola(nombreClase, "M", "INICIO
EJECUCION","doInBackground");

            SoapObject request = new
SoapObject(InformacionServicios.WS_NAMESPACE,metodoActualizar);

            flagEjecucion = false;

            request.addProperty("idCliente",idClienteBuscar);

            request.addProperty("valorMinimoNotif",valorMinimoNotif);

            SoapSerializationEnvelope sobre = new

```

```

SoapSerializationEnvelope(SoapEnvelope.VER11);

        //sobre.dotNet = true;
        sobre.setOutputSoapObject(request);

        HttpTransportSE transporte = new
HttpTransportSE(InformacionServicios.WS_URL_ACT_MINIMO);
        transporte.call(InformacionServicios.WS_ACCION_SOAP, sobre);

        //resultado = (SoapPrimitive)sobre.getResponse();
        Object resultadoPHP = sobre.getResponse();

        XMLParser parser = new XMLParser();
        Document doc = parser.getDomElement(resultadoPHP.toString()); //
getting DOM element

        NodeList nl = doc.getElementsByTagName("datos");
        // looping through all item nodes <item>
        Integer tamaño = nl.getLength();
        if (tamaño == 1)
        {
            Element e = (Element) nl.item(0);
            respuesta = parser.getValue(e, "respuesta");
            flagEjecucion = true;
        }
        else
            flagEjecucion = false;
        Rastro.EscribirEnConsola(nombreClase, "M", "FIN
EJECUCION", "doInBackground");
        return 0;
    }
    catch (Exception ex)
    {
        Rastro.EscribirEnConsola(nombreClase, "E",
ex.getMessage(), "doInBackground");
        return 1;
    }
}

protected void onProgressUpdate(Float... valores)
{

```

```

    }

    protected void onPostExecute(Integer bytes)
    {
        try
        {
            Rastro.EscribirEnConsola(nombreClase, "M", "INICIO
EJECUCION","onPostExecute");

            if (respuesta.equals("OK"))
                Toast.makeText(getApplicationContext(), "Actualización
exitosa.",Toast.LENGTH_SHORT).show();
            else
                Toast.makeText(getApplicationContext(), "Actualización
NO exitosa.",Toast.LENGTH_SHORT).show();

            Rastro.EscribirEnConsola(nombreClase, "M", "FIN
EJECUCION","onPostExecute");
        }
        catch (Exception ex)
        {
            Rastro.EscribirEnConsola(nombreClase, "E",
ex.getMessage(),"onPostExecute");
        }
    }
} //fin BuscarInfoCliente
} //Fin Clase ClientesPestanaDatos

```

5.- PRORAMACIÓN NOTIFICACIONES

```

public class ListadoNotificaciones extends Activity {

    private ListView mainListView ;
    private ArrayAdapter<String> listAdapter ;
    private static final String metodo = "consultarNotificacionesEEMonitor";
    String[] arrayNotificaciones = new String[] { "NUEVO CLIENTE"};
    String[] arrayNotificacionesAux = new String[] {"LISTADO DE ALERTAS"};
    private static final String nombreClase = "Activity - ListadoNotificaciones";
    Intent intent = null;

```

```

String loginUsuario = "";
String nombreUsuario = "";
String codigoUsuario = "";
String idClienteBuscar = "";

/** Called when the activity is first created. */
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

    try
    {
        Rastro.EscribirEnConsola(nombreClase, "M", "INICIO EJECUCION","onCreate");
        setContentView(R.layout.activity_listado);
        mainListView = (ListView) findViewById( R.id.mainListView );

        intent = new Intent();
        intent.setComponent(new ComponentName(this,ClienteContenedor.class)) ;

        mainListView = (ListView) findViewById( R.id.mainListView );

        ArrayList<String> listaNotificaciones = new ArrayList<String>();

        //recuperar parametros pasados desde activity anterior(mainActivity)
        //recuperar parámetros desde activity anterior
        SharedPreferences settings = getSharedPreferences("pasoInformacion",
MODE_PRIVATE);
        String infoCliente = settings.getString("codigoUsuario","codigoUsuarioDef");
        if (infoCliente == null)
        {
            Toast.makeText(getApplicationContext(), "No se pudo cargar información
de actividad anterior",Toast.LENGTH_SHORT).show();

            finish(); //cerrar pantalla
        }

        idClienteBuscar = infoCliente;
    }
}

```

```

//Find the ListView resource.
mainListView = (ListView) findViewById( R.id.mainListView );

// Create and populate a List of clientes.
ArrayList<String> arrayNotificaciones = new ArrayList<String>();
listaNotificaciones.addAll( Arrays.asList(arrayNotificacionesAux) );

// Create ArrayAdapter using the clientes list.
listAdapter = new ArrayAdapter<String>(this, R.layout.activity_fila_listado,
listaNotificaciones);

new ConsultarNotificaciones().execute("");

Rastro.EscribirEnConsola(nombreClase, "M", "FIN EJECUCION","onCreate");
}
catch (Exception ex)
{
    Rastro.EscribirEnConsola(nombreClase, "E", ex.getMessage(),"onCreate");
}
}

public void cmdRefrescar_click(View Botton)
{
    try
    {
        Rastro.EscribirEnConsola(nombreClase, "M", "INICIO
EJECUCION","cmdRefrescar_click");

        new ConsultarNotificaciones().execute("");

        Rastro.EscribirEnConsola(nombreClase, "M", "FIN EJECUCION","search_Onclick");
    }
    catch(Exception ex)
    {
        Rastro.EscribirEnConsola(this.nombreClase, "E",
ex.getMessage(),"cmdRefrescar_click");
    }
}
}

```



```

private class ConsultarNotificaciones extends AsyncTask<String, Float, Integer>
{
    SoapPrimitive resultado = null;
    protected void onPreExecute()
    {
    }

    protected Integer doInBackground(String... urls)
    {
        try
        {
            Rastro.EscribirEnConsola(nombreClase, "M", "INICIO
EJECUCION","doInBackground");

            SoapObject request = new
SoapObject(InformacionServicios.WS_NAMESPACE,metodo);

            //EditText objTextoBuscar = (EditText) findViewById(
R.id.txtBuscar );

            //String textoBuscar = objTextoBuscar.getText().toString();
            request.addProperty("idCliente",idClienteBuscar);

            SoapSerializationEnvelope sobre = new
SoapSerializationEnvelope(SoapEnvelope.VER11);

            //sobre.dotNet = true;
            sobre.setOutputSoapObject(request);

            HttpTransportSE transporte = new
HttpTransportSE(InformacionServicios.WS_URL_ALERTA);

            transporte.call(InformacionServicios.WS_ACCION_SOAP, sobre);

            //resultado = (SoapPrimitive)sobre.getResponse();
            Object resultadoPHP = sobre.getResponse();

            final String KEY_ITEM = "raiz"; // parent node
            final String KEY_ID = "mensaje";

            ArrayList<HashMap<String, String>> menuItems = new
ArrayList<HashMap<String, String>>();

            XMLParser parser = new XMLParser();

```

```

//String xml = parser.getXmlFromUrl(URL); // getting XML
Document doc = parser.getDomElement(resultadoPHP.toString()); // getting
DOM element

NodeList nl = doc.getElementsByTagName("notificaciones");
// looping through all item nodes <item>
Integer tamaño = nl.getLength();
arrayNotificacionesAux = new String[tamaño];
for (int i = 0; i < tamaño; i++) {
    Element e = (Element) nl.item(i);
    String valor = parser.getValue(e, KEY_ID);
    arrayNotificacionesAux[i] = valor;
}
Rastro.EscribirEnConsola(nombreClase, "M", "FIN
EJECUCION","doInBackground");
return 0;
}
catch (Exception ex)
{
    Rastro.EscribirEnConsola(nombreClase, "E",
ex.getMessage(),"doInBackground");
return 1;
}
}

protected void onProgressUpdate(Float... valores)
{
}

protected void onPostExecute(Integer bytes)
{
    try
    {
        Rastro.EscribirEnConsola(nombreClase, "M", "INICIO
EJECUCION","onPostExecute");

        //for (int j = 0; j < 10; j++)
        listAdapter.clear();
        for (int i = 0; i < arrayNotificacionesAux.length; i++)
            listAdapter.add(arrayNotificacionesAux[i]);
    }
}
}

```

```

            mainListView = (ListView) findViewById( R.id.mainListView );
            mainListView.setAdapter( listAdapter );
            Rastro.EscribirEnConsola(nombreClase, "M", "FIN
EJECUCION","onPostExecute");
        }
        catch (Exception ex)
        {
                Rastro.EscribirEnConsola(nombreClase, "E",
ex.getMessage(),"onPostExecute");
        }
    }
}
}

```

6.- PRORAMACIÓN CONTENEDOR CLIENTE

```

public class ClienteContenedor extends TabActivity{
    private static final String nombreClase = "ClienteContenedor";
    String loginUsuario = "";
    String nombreUsuario = "";
    String codigoUsuario = "";
    String infoCliente = "";

    @Override
    public void onCreate(Bundle savedInstanceState)
    {
        try
        {
                Rastro.EscribirEnConsola(nombreClase, "M", "INICIO EJECUCION","onCreate");

                super.onCreate(savedInstanceState);
                setContentView(R.layout.activity_cliente_contenedor);

                TabHost tabHost = getTabHost(); //creamos el tabhost de esta actividad
                TabHost.TabSpec spec; //se crea recursos para las propiedades de la pestaña
                Intent intent; //intent para abrir pestañas
                Resources res = getResources(); //recuperar los recursos asociados

```

```

        //se abre la primera pestaña por defecto

        //intent = new Intent().setClass(this, ClientePestanaDatos.class);

        intent = new Intent().setClass(this, MainActivity.class);

        //configurar las propiedades de la pestaña

        //spec =
tabHost.newTabSpec("Datos").setIndicator("Datos",res.getDrawable(R.drawable.res_cli_pest_datos))
        .setContent(intent);

        spec =
tabHost.newTabSpec("Lineas").setIndicator("Líneas").setContent(intent);

        tabHost.addTab(spec);

        //intent = new Intent().setClass(this, ClientePestanaChip.class);

        intent = new Intent().setClass(this, BarrasActivity.class);

        //configurar las propiedades de la pestaña

        //spec =
tabHost.newTabSpec("PDV").setIndicator("PDV",res.getDrawable(R.drawable.res_cli_pest_chip)).setContent(intent);

        spec =
tabHost.newTabSpec("Barras").setIndicator("Barras").setContent(intent);

        tabHost.addTab(spec);

        intent = new Intent().setClass(this, ClientePestanaDatos.class);

        //configurar las propiedades de la pestaña

        //spec =
tabHost.newTabSpec("CHIPS").setIndicator("CHIPS",res.getDrawable(R.drawable.res_cli_pest_pdv)).setContent(intent);

        spec =
tabHost.newTabSpec("Perfil").setIndicator("Perfil").setContent(intent);

        tabHost.addTab(spec);

        intent = new Intent().setClass(this, ListadoNotificaciones.class);

        //configurar las propiedades de la pestaña

        //spec = tabHost.newTabSpec("CHIPS").setIndicator("Epa",
res.getDrawable(R.drawable.ic_launcher)).setContent(intent);

        spec =
tabHost.newTabSpec("Alerta").setIndicator("Alerta").setContent(intent);

        tabHost.addTab(spec);

        setTabColor(tabHost);

```

```

//recuperar parámetros desde activity anterior

SharedPreferences settings = getSharedPreferences("pasoInformacion",
MODE_PRIVATE);

//Bundle bundle = getIntent().getExtras();

if ( settings.getString("loginUsuario","loginUsuarioDef") == null
    || settings.getString("nombreUsuario","nombreUsuarioDef") == null
    || settings.getString("codigoUsuario","codigoUsuarioDef") == null
    || settings.getString("infoCliente","infoClienteDef") == null)
{
    Toast.makeText(getApplicationContext(), "No se pudo cargar información
de actividad anterior. Imposible continuar",Toast.LENGTH_SHORT).show();

    finish(); //cerrar pantalla
}
else
{
    //String infoCliente = settings.getString("infoCliente",
"infoClienteDef");

    loginUsuario = settings.getString("loginUsuario","loginUsuarioDef");
    nombreUsuario = settings.getString("nombreUsuario","nombreUsuarioDef");
    codigoUsuario = settings.getString("codigoUsuario","codigoUsuarioDef");
    infoCliente = settings.getString("infoCliente","infoClienteDef");

    TextView objLblUsuarioConectado = (TextView)
findViewById(R.id.lblUsuarioConectado);

    objLblUsuarioConectado.setText("Usuario Activo : " + codigoUsuario + "/" +
loginUsuario + "/" + nombreUsuario);
}

Rastro.EscribirEnConsola(nombreClase, "M", "FIN EJECUCION
EJECUCION","onCreate");
}
catch (Exception ex)
{
    Rastro.EscribirEnConsola(nombreClase, "E", ex.getMessage(),"onCreate");
}
}

private void setTabColor(TabHost tabHost) {
    try {

```

```

        for (int i=0; i < tabHost.getTabWidget().getChildCount();i++) {
            TextView tv = (TextView)
tabHost.getTabWidget().getChildAt(i).findViewById(android.R.id.title); //Unselected Tabs
            if (tv != null) {
                tv.setTextColor(Color.WHITE);
            }
            TextView tv2 = (TextView)
tabHost.getCurrentTabView().findViewById(android.R.id.title); // Selected Tab
            if (tv2 != null) {
                tv2.setTextColor(Color.WHITE);
            }
        }
    } catch (ClassCastException e) {
        // A precaution, in case Google changes from a TextView on the tabs.
    }
}
}
}

```

7.- PRORAMACIÓN LOGUEO

```

public class LoginActivity extends ActionBarActivity{

    private static final String nombreClase = "Activity - MainActivity";
    //private static final String metodo = "logueoUsuarioEEMonitor";
    private static final String metodo = "logueoUsuarioEEMonitor";
    boolean banderaLogueo = false;
    Intent intent = null;

    String respuesta = "";
    String login = "";
    String nombre = "";
    String codigoInterno = "";
    String usuario = "";
    String password = "";

    @Override
    public void onCreate(Bundle savedInstanceState) {

```

```

        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        //getMenuInflater().inflate(R.menu.activity_main, menu);
        return true;
    }

    public void cmdOpciones_onclick(View Botton)
    {
        try
        {
            Rastro.EscribirEnConsola(nombreClase, "M", "INICIO
EJECUCION", "cmdOpciones_onclick");

            EditText oUsuario = (EditText) findViewById(R.id.txtUsuario);
            TextView oPassword = (TextView) findViewById(R.id.txtPassword);

            usuario = oUsuario.getText().toString();
            password = oPassword.getText().toString();

            if (usuario.equals("") || password.equals(""))
            {
                Toast.makeText(getApplicationContext(), "No ha ingresado Usuario o
Password.", Toast.LENGTH_SHORT).show();

                return;
            }

            new LogueUsuariosMetroCOM().execute("");

            intent = new Intent();

            intent.setComponent(new ComponentName(this, ClienteContenedor.class) );

            Rastro.EscribirEnConsola(nombreClase, "M", "FIN
EJECUCION", "cmdOpciones_onclick");
        }
        catch (Exception ex)
        {
            ex.printStackTrace();
            System.out.println(ex.getMessage());
        }
    }

```

```

    }
}

@Override
public void onActivityResult(int requestCode, int resultCode, Intent data)
{
}

private class LogueUsuariosMetroCOM extends AsyncTask<String, Float, Integer>
{
    SoapPrimitive resultado = null;
    protected void onPreExecute()
    {
    }
    protected Integer doInBackground(String... urls)
    {
        try
        {
            Rastro.EscribirEnConsola(nombreClase, "M", "INICIO
EJECUCION", "doInBackground");
            SoapObject request = new
SoapObject(InformacionServicios.WS_NAMESPACE, metodo);
            request.addProperty("p_login", usuario);
            request.addProperty("p_password", password);
            //request.addProperty("categoria", "librosno");
            SoapSerializationEnvelope sobre = new
SoapSerializationEnvelope(SoapEnvelope.VER11);
            //sobre.dotNet = true;
            sobre.setOutputSoapObject(request);
            //HttpTransportSE transporte = new
HttpTransportSE(InformacionServicios.WS_URL);
            HttpTransportSE transporte = new
HttpTransportSE(InformacionServicios.WS_URL_LOGUEO);

            transporte.call(InformacionServicios.WS_ACCION_SOAP, sobre);

            //resultado = (SoapPrimitive)sobre.getResponse();
            Object resultadoPHP = sobre.getResponse();

```



```

final String KEY_ITEM = "raiz"; // parent node

    final String KEY_RESPUESTA = "respuesta";

    final String KEY_LOGIN = "login";

    final String KEY_NOMBRE = "nombre";

    final String KEY_CODIGO_INTERNO = "codigoInterno";

        ArrayList<HashMap<String, String>> menuItems = new
DOM element ArrayList<HashMap<String, String>>();

        XMLParser parser = new XMLParser();

        Document doc = parser.getDomElement(resultadoPHP.toString()); // getting

        NodeList nl = doc.getElementsByTagName("infoLogin");

        // looping through all item nodes <item>

        Integer tamaño = nl.getLength();

        if (tamaño == 1)

        {

            Element e = (Element) nl.item(0);

            respuesta = parser.getValue(e, KEY_RESPUESTA);

            login = parser.getValue(e, KEY_LOGIN);

            nombre = parser.getValue(e, KEY_NOMBRE);

            codigoInterno = parser.getValue(e, KEY_CODIGO_INTERNO);

            if (respuesta.equals("S"))

                banderaLogueo = true;

            else

                banderaLogueo = false;

        }

        else

            banderaLogueo = false;

            //throw new exception;

        Rastro.EscribirEnConsola(nombreClase, "M", "FIN
EJECUCION", "doInBackground");

        return 0;

    }

    catch (Exception ex)

    {

        Rastro.EscribirEnConsola(nombreClase, "E",
ex.getMessage(), "doInBackground");

```

```

        return 1;
    }
}

protected void onProgressUpdate(Float... valores)
{
}

protected void onPostExecute(Integer bytes)
{
    try
    {
        Rastro.EscribirEnConsola(nombreClase, "M", "INICIO
EJECUCION","onPostExecute");

        Rastro.EscribirEnConsola(nombreClase, "M", "info","respuesta
logueo:" + respuesta);

        if (respuesta.equals("S"))
        {
            intent.putExtra( "login",login ); //guarda datos para q reciba
otra activity

            intent.putExtra( "nombre",nombre ); //guarda datos para q reciba
otra activity

            intent.putExtra( "codigoInterno",codigoInterno); //guarda datos
para q reciba otra activity

            startActivityForResult(intent,0); //con parametros de retorno

            //pasar informacion próxima pantalla

            //SharedPreferences settings =
getSharedPreferences("pasoNombreCiudad", MODE_PRIVATE);

            SharedPreferences settings =
getSharedPreferences("pasoInformacion", MODE_PRIVATE);

            SharedPreferences.Editor editor = settings.edit();

            editor.putString("infoCliente", nombre);

            editor.putString("loginUsuario", login);

            editor.putString("nombreUsuario", nombre);

            editor.putString("codigoUsuario", codigoInterno);

            editor.commit();

```

```

        finish();
    }
    else
        Toast.makeText(getApplicationContext(), "Problemas a
realizar el Logueo, no se puede continuar.",Toast.LENGTH_SHORT).show();

        Rastro.EscribirEnConsola(nombreClase, "M", "FIN
EJECUCION","onPostExecute");
    }
    catch (Exception ex)
    {
        Rastro.EscribirEnConsola(nombreClase, "E",
ex.getMessage(),"onPostExecute");
    }
}
}
}
}

```

8.- PROGRAMACIÓN RASTRO

```

import java.text.SimpleDateFormat;
import java.util.Date;

public class Rastro {

    public static boolean EscribirEnConsola(String origen, String tipo, Exception error,
String metodo)
    {
        try
        {
            String mensajeLocal = "";
            Date fechaHoy = new Date();
            SimpleDateFormat formato = new SimpleDateFormat("dd/MMM/yyyy hh:mm:ss");
            String cadenaFecha = formato.format(fechaHoy);

            if (tipo == "E")
                mensajeLocal = "[ ERROR ] " + cadenaFecha + " " + origen + "
" + metodo + " " + error.getMessage();
            if (tipo == "M")
                mensajeLocal = " [ MENSAJE ] " + cadenaFecha + " " + origen + "
" + metodo + " " + error.getMessage();

```

```

        System.out.println(mensajeLocal);
        error.printStackTrace();
        return true;
    }
    catch (Exception ex)
    {
        System.out.println("Error al escribir en log en consola : " +
ex.getMessage());
        return false;
    }
}

public static boolean EscribirEnConsola(String origen, String tipo, String msg, String
metodo)
{
    try
    {
        String mensajeLocal = "";
        Date fechaHoy = new Date();
        SimpleDateFormat formato = new SimpleDateFormat("dd/MMM/yyyy hh:mm:ss");
        String cadenaFecha = formato.format(fechaHoy);

        if (tipo == "E")
            mensajeLocal = "[ ERROR ] " + cadenaFecha + " " + origen + "
" + metodo + " " + msg;
        if (tipo == "M")
            mensajeLocal = " [ MENSAJE ] " + cadenaFecha + " " + origen + "
" + metodo + " " + msg;
        System.out.println(mensajeLocal);
        return true;
    }
    catch (Exception ex)
    {
        System.out.println("Error al escribir en log en consola : " +
ex.getMessage());
        return false;
    }
}
}

```

9.- PROGRAMACIÓN XMLPARSER

```
public String getXmlFromUrl(String url) {
    String xml = null;

    try {
        // defaultHttpClient
        DefaultHttpClient httpClient = new DefaultHttpClient();
        HttpPost httpPost = new HttpPost(url);

        HttpResponse httpResponse = httpClient.execute(httpPost);
        HttpEntity httpEntity = httpResponse.getEntity();
        xml = EntityUtils.toString(httpEntity);

    } catch (UnsupportedEncodingException e) {
        e.printStackTrace();
    } catch (ClientProtocolException e) {
        e.printStackTrace();
    } catch (IOException e) {
        e.printStackTrace();
    }

    // return XML
    return xml;
}

public Document getDomElement(String xml){
    Document doc = null;
    DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();
    try {

        DocumentBuilder db = dbf.newDocumentBuilder();

        InputSource is = new InputSource();
        is.setCharacterStream(new StringReader(xml));
        doc = db.parse(is);
    }
}
```

```

        } catch (ParserConfigurationException e) {
            Log.e("Error: ", e.getMessage());
            return null;
        } catch (SAXException e) {
            Log.e("Error: ", e.getMessage());
            return null;
        } catch (IOException e) {
            Log.e("Error: ", e.getMessage());
            return null;
        }
    }

    return doc;
}

public final String getElementValue( Node elem ) {
    Node child;
    if( elem != null){
        if (elem.hasChildNodes()){
            for( child = elem.getFirstChild(); child != null; child =
child.getNextSibling() ){
                if( child.getNodeType() == Node.TEXT_NODE ){
                    return child.getNodeValue();
                }
            }
        }
    }
    return "";
}

public String getValue(Element item, String str) {
    NodeList n = item.getElementsByTagName(str);
    return this.getElementValue(n.item(0));
}
}

```