

**APLICACIÓN MÓVIL USABLE DIRIGIDA A UN SISTEMA AMI CON SISTEMA  
DE AMARRE AUTOMÁTICO (SAA) EN EL SECTOR ELÉCTRICO EXISTENTE,  
EN EL MARCO DEL PROYECTO ID 4249**



**Trabajo de Grado**

**LEIDY JOHANNA IJAJÍ GUACA  
YENNY ALEXANDRA WAITOTO PÉREZ**

**UNIVERSIDAD DEL CAUCA  
FACULTAD DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES  
INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES  
POPAYÁN  
2017**

**APLICACIÓN MÓVIL USABLE DIRIGIDA A UN SISTEMA AMI CON SISTEMA  
DE AMARRE AUTOMÁTICO (SAA) EN EL SECTOR ELÉCTRICO EXISTENTE,  
EN EL MARCO DEL PROYECTO ID 4249**

**LEIDY JOHANNA IJAJÍ GUACA  
YENNY ALEXANDRA WAITOTO PÉREZ**

Trabajo de grado en la modalidad de investigación para optar el título de  
INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES

Director:  
PhD. PABLO AUGUSTO MAGÉ IMBACHÍ

Codirector:  
MSc. JUAN FERNANDO FLÓREZ MARULANDA

**UNIVERSIDAD DEL CAUCA  
FACULTAD DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES  
INGENIERÍA EN ELECTRÓNICA Y TELECOMUNICACIONES  
POPAYÁN  
2017**

## **AGRADECIMIENTOS**

A la Universidad del Cauca, por facilitarnos la utilización de sus instalaciones (Laboratorio de Automática Industrial), importante para este proceso investigativo.

A los ingenieros Pablo Augusto Magé Imbachí, Juan Fernando Flórez Marulanda y Carlos Felipe Rengifo Rodas, por compartir sus conocimientos y experiencias; por su orientación y colaboración en el desarrollo de esta investigación.

Al equipo de desarrollo del piloto AMI-SAA, los ingenieros Luisa Pineda, Yamir Bolaños y Eduardo Castillo.

Al tester del piloto AMI-SAA, el ingeniero Gonzalo Ernesto Potosí.

Al ingeniero Darío Chamorro, quien nos brindó su apoyo traducido en asesoría para el desarrollo de la aplicación móvil.

Y por último y no menos importante, a Natalia Castillo y Viviana Araujo, quienes nos brindaron el acompañamiento necesario para la realización de la metodología Scrum.

## CONTENIDO

CAPITULO I.....	1
INTRODUCCIÓN .....	1
1.1. CONTEXTO.....	1
1.2. MARCO TEÓRICO .....	3
1.2.1. AMI.....	3
1.2.2. Usabilidad .....	7
1.2.3. Sistema operativo Android .....	9
1.2.4. Interfaz de Usuario.....	10
1.3. ESTADO DEL ARTE.....	10
1.3.1. Objetivos .....	10
1.3.2. Planteamiento del problema.....	11
1.4. REVISIÓN SISTEMÁTICA.....	13
1.4.1. Herramientas de búsqueda .....	13
1.4.2. Estrategia de búsqueda .....	14
1.4.3. Resultados de la búsqueda.....	14
CAPITULO II.....	20
METODOLOGÍA .....	20
2.1. INTRODUCCIÓN.....	20
2.2. METODOLOGIA .....	21
2.3. SCRUM: MARCO DE TRABAJO PARA EL DESARROLLO DE REQUISITOS FUNCIONALES .....	22
2.3.1. Características de Scrum .....	22
2.3.2. Proceso .....	23
2.3.3. Historias de usuario (HU).....	26
2.4. MPLu+a: METODOLOGIA PARA EL DESARROLLO DE REQUISITOS NO FUNCIONALES .....	29
2.4.1. Características de MPLu+a .....	29
2.4.2. Análisis de requisitos.....	30
2.4.3. Diseño .....	31
2.4.4. Implementación .....	31
2.4.5. Lanzamiento.....	31
2.4.6. Prototipado.....	32
2.4.7. Evaluación.....	33
2.5. HERRAMIENTAS DE APOYO.....	33

CAPITULO III .....	35
APLICACIÓN MÓVIL USABLE ATEMSAA .....	35
3.1. INTRODUCCION .....	35
3.2. AMI-SAA .....	37
3.2.1. Equipos de medición .....	39
3.2.2. Tramas de comunicación .....	40
CAPITULO IV .....	42
MPIu+a: METODOLOGÍA PARA IMPLEMETACIÓN DE REQUISITOS NO FUNCIONALES .....	42
4.1. INTRODUCCIÓN .....	42
4.2. ANÁLISIS DE REQUISITOS .....	43
4.2.1. Estudio de los implicados y de la plataforma .....	43
4.2.2. Objetos .....	44
4.2.3. Objetivos .....	44
4.2.4. Análisis de requisitos para ATEMSAA .....	48
4.3. DISEÑO .....	50
4.3.1. Diseño de la actividad .....	51
4.3.2. Diseño de la información .....	52
4.3.3. El diseño aplicado en ATEMSAA .....	53
4.4. IMPLEMENTACIÓN .....	73
4.4.1. Implementación en ATEMSAA .....	73
4.5. LANZAMIENTO .....	74
4.6. PROTOTIPADO .....	75
4.7. EVALUACIÓN .....	79
CAPITULO V .....	80
FUNCIONALIDAD ATEMSAA .....	80
5.1. INTRODUCCIÓN .....	80
5.2. FUNCIONALIDADES ATEMSAA .....	81
5.2.1. Funcionalidades PLC_MMS .....	82
CAPITULO VI .....	84
EVALUACIÓN ATEMSAA .....	84
6.1. INTRODUCCIÓN .....	84
6.2. EVALUACIÓN FUNCIONAL DE ATEMSAA .....	85
6.3. EVALUACIÓN NO FUNCIONAL DE ATEMSAA .....	86
6.3.1. Inspección .....	86
6.3.2. Indagación .....	88
6.3.3. Test .....	89

CAPITULO VII.....92  
CONCLUSIONES Y TRABAJOS FUTUROS.....92  
    7.1. CONCLUSIONES .....92  
    7.2. TRABAJOS FUTUROS.....94  
BIBLIOGRAFÍA.....95

## LISTA DE FIGURAS

Figura 1. Componentes de un Sistema AMI. ....	5
Figura 2. Protocolos Bluetooth.....	6
Figura 3. Etapas del DCU. ....	9
Figura 4. Metodologías usadas.....	21
Figura 5. Esquema de las 5 fases de desarrollo en Scrum.....	23
Figura 6. Método ágil Scrum. ....	24
Figura 7. Roles del equipo AMISAA.....	25
Figura 8. Modelo de Proceso de la Ingeniería de la Usabilidad y la Accesibilidad (MPlu+a). ....	29
Figura 9. Arquitectura del sistema AMI-SAA.....	38
Figura 10. Prototipo no funcional (Boceto).....	48
Figura 11. Ejemplo de cuestionario para acercamiento con el usuario de la aplicación.....	50
Figura 12. Estructura general de un dialogo en Android.....	55
Figura 13. Ejemplos de diálogos utilizados en el sistema operativo Android. ....	56
Figura 14. Alerta sin título .....	57
Figura 15. Alerta con título.....	57
Figura 16. Mensaje Toast en ATEMSAA para conexión con dispositivo PLC (en interfaz de Inicio).....	58
Figura 17. Aspecto coherente en los diálogos. ....	59
Figura 18. Información poco clara.....	59
Figura 19. Información clara y concisa. ....	59
Figura 20. Colores de referencia en Android. ....	60
Figura 21. Las grandes áreas vacías tienen un color neutro: blanco.....	61
Figura 22. De acuerdo a la forma de sostener el móvil, el pulgar tiene mayor o menos dificultad en acceder a ciertas zonas. ....	62
Figura 23. Diagrama de Gutenberg. ....	64
Figura 24. Distribución de pantalla para dispositivo móvil en orientación portrait. .	64
Figura 25. Anatomía del Action Bar. ....	65
Figura 26. Primera organización de pestañas utilizando Stacked Action Bar. ....	67
Figura 27. Stacked action bar para dispositivo PLC_MMS. ....	67
Figura 28. Al contacto con el <i>TextFiel</i> el teclado aparece automáticamente. Con la propiedad <i>Hint</i> se indica la información que debe ingresarse (en este caso el ID del dispositivo PLC). ....	69

Figura 29. Mensaje de error al ingresar el usuario/contraseña de manera errada.	70
Figura 30. Diferentes tamaños de fuentes en Android de acuerdo al uso y jerarquía de los elementos.	71
Figura 31. Ejemplo de cuadro sinóptico en ATEMSAA.	72
Figura 32. Ejemplo de pantalla vacía en ATEMSAA.	73
Figura 33. Primer Storyboard para ATEMSAA.	77
Figura 34. Segundo Storyboard para ATEMSAA.	78
Figura 35. Tercer Storyboard para ATEMSAA.	79
Figura 36. Ejemplo de cuestionario.	90
Figura 37. Ejemplo de estadística de encuesta	91



## LISTA DE TABLAS

Tabla 1. Resultados búsqueda en Google Scholar.....	15
Tabla 2. Resultados búsqueda en IEEE Xplore Digital Library. ....	15
Tabla 3. Resultados búsqueda en ScienceDirect. ....	15
Tabla 4. Comparación de diferentes trabajos relacionados. ....	16
Tabla 5. Visual Story Mapping para HU 1.....	28
Tabla 6. Herramientas de apoyo.....	33
Tabla 7. Estructura de la trama de comunicación para el aplicativo móvil con los dispositivos PLC. ....	40
Tabla 8. Métodos utilizados para el análisis de tareas (algunos).....	52
Tabla 9. Clasificación de las técnicas de prototipado. ....	75
Tabla 10. Funcionalidades en ATEMSAA.....	81
Tabla 11. Funcionalidades del PLC_MMS.....	82
Tabla 12. HU: configuración de dispositivo PLC-MMS. ....	85
Tabla 13. Reglas heurísticas de Nielsen.....	87
Tabla 14. Métodos de evaluación por indagación.....	88
Tabla 15. Método de evaluación por Test.....	89
Tabla 16. Aceptación de la aplicación.....	91

## **GLOSARIO**

**AMI:** Advanced Metering Infrastructure (Infraestructura de Medición Avanzada).

**ATEMSAA:** Aplicación en Terminal Móvil para un Sistema de Amarre Automático.

**BPL:** Banda ancha por línea eléctrica.

**CEO:** Compañía Energética de Occidente.

**DCU:** Diseño Centrado en el Usuario.

**HU:** Historia de Usuario.

**PLC:** Power Line Carrier (Comunicación mediante Líneas de Potencia).

**SAA:** Sistema de Amarre Automático.

**VSM:** Visual Story Mapping.

# CAPITULO I

## INTRODUCCIÓN

### 1.1. CONTEXTO

Hoy en día se presenta una alta demanda en el consumo de energía eléctrica, esto impulsa a que las empresas prestadoras del servicio busquen la mejor forma de proveerlo, sin embargo, hay un riesgo de pérdidas en sus circuitos de distribución durante este proceso. Las empresas del sector eléctrico trabajan continuamente en la solución de este problema, implementando tecnologías que les permitan obtener información de los usuarios y sus respectivos consumos energéticos, con el fin de tener más control en este campo. Una de las tecnologías que facilita este proceso es AMI (Advanced Metering Infrastructure) y su relación con las Smart Grid o redes inteligentes y medidores inteligentes que ayudan a monitorear la corriente eléctrica, la comunicación entre sí de dispositivos y logran una optimización en el uso de la energía.

La preocupación de las empresas del sector eléctrico es que los balances de energía no coinciden con los datos recolectados manualmente, por lo tanto la vinculación de medidores inteligentes con tecnología AMI y la red eléctrica convencional, permiten el control y seguimiento de la red por parte de los proveedores del servicio, accediendo a la información de forma remota.

AMI ayuda a reducir costos financieros, ya que mucha de la información que se recolecta actualmente de forma manual podría ser automatizada, permitiendo la implementación de sistemas de gestión de datos para el almacenamiento de los mismos y su respectivo seguimiento, se mejora el servicio ya que los usuarios conocerán exactamente cuál es su consumo y si el servicio en algún momento falla se reportara automáticamente a la empresa.

Es importante mencionar que aunque en AMI la pieza fundamental es el medidor, se requiere toda una infraestructura de comunicaciones y de sistemas de información y estas deben estar diseñadas de tal manera que garantice la consecución de beneficios buscados [1].

Ya conociendo los beneficios que un sistema AMI puede brindar, depende de cada empresa qué clase de servicios o aplicaciones desea ofrecer; por ejemplo, realizar medición bidireccional del flujo de energía (recepción y entrega); capacidad de

proporcionar los datos de los clientes en intervalos de tiempo específico o gestionar la energía mediante la opción de conexión/desconexión remota para algunos o la totalidad de los medidores; capacidad de diagnosticar, vigilar y controlar el estado de la red de comunicaciones; capacidad de lectura remota de medidores de baja demanda; medición y reporte de eventos y parámetros de calidad de energía (armónicos, interrupciones, tensión mínima, máxima), con capacidades de monitorización en tiempo real, configurable por la empresa; reporte y registros diarios de lectura de contadores, etc. [1].

En este contexto, se hizo necesaria una herramienta móvil que facilitara la interacción con el sistema AMI cuando el acceso remoto a los medidores no se puede realizar; esta herramienta facilita la recolección de datos en campo y en ella se desplegaran todos los procesos a que diera lugar una empresa en particular; la forma en cómo se presentan dichos procesos al usuario de la herramienta móvil, se vuelve fundamental para que se pueda llevar a cabo una buena gestión de la información; así, el diseño de la interfaz toma mayor relevancia.

Con lo planteado anteriormente y teniendo como premisa que el usuario humano se vuelve fundamental en la supervisión de la red eléctrica mediante la herramienta móvil, se puede inferir que el desarrollo de las interfaces se debe realizar pensando en los modelos mentales de los usuarios, en sus necesidades y requerimientos, ya que en última instancia, son estos los que interactúan con el proceso [2]. “El diseño se debe llevar a cabo desde un punto de vista de la centralidad humana, es decir, con respecto a la orientación del usuario, sus procesos cognitivos, sus necesidades y capacidades” [3].

Herramientas tecnológicas como los servicios WEB o en otros casos, los dispositivos móviles, han facilitado la supervisión de procesos de manera remota gracias al desarrollo exponencial que han tenido las comunicaciones. En particular la evolución en los dispositivos móviles ha ido acompañada del desarrollo de sistemas operativos independientes (Android, IOS, Windows Mobile, etc.) con capacidad de procesamiento de información, conexión constante con la web, interacción agradable con el usuario final, etc., siendo Android el más consolidado a nivel mundial, razón por la cual se tomó como referencia para el desarrollo de este proyecto (ver Anexo A).

En este orden de ideas lo que se pretende con este proyecto es proponer una aplicación móvil usable que facilite la interacción con el sistema basado en tecnología AMI<sup>1</sup> con Sistema de Amarre Automático del sector eléctrico existente, teniendo como base la teoría del Diseño Centrado en el Usuario (DCU), los patrones de diseño de Android y las recomendaciones y directrices de personas especializadas en el tema para el diseño de estas interfaces.

## 1.2. MARCO TEÓRICO

El constante avanzar tecnológico en el mundo sugiere que las redes eléctricas se conviertan en redes eléctricas inteligentes y un paso para alcanzar este objetivo es poder contar con medición de la energía a través de vías inalámbricas, lo cual hace necesaria la articulación de tecnologías específicas en el área de las comunicaciones, específicamente en el área eléctrica para brindar este servicio [1].

En este capítulo se mencionara de manera general los conceptos más importantes a tener en cuenta en el desarrollo de este trabajo.

### 1.2.1. AMI

***“AMI -Advanced Meter Infrastructure- (Infraestructura de Medición Avanzada). Un sistema AMI es una solución integral que tiene la capacidad de gestionar el intercambio de información y datos entre el sistema de gestión y las unidades de medida, permite la gestión remota de diferentes funcionalidades como la toma de lecturas, procesos de conexión y desconexión para los medidores que posean dicha capacidad, eventos y alarmas, el control de acceso a las interfaces, entre otras funcionalidades con el fin de ofrecer una solución eficiente para la toma oportuna de decisiones preventivas, de mejora o correctivas. El sistema AMI incluye una amplia gama de aplicaciones que permite gestionar la demanda, optimizar la red de distribución, garantizar la integridad del sistema y proveer servicios de valor agregado” [4].***

---

<sup>1</sup> Este trabajo de grado contribuirá al proyecto AMI-SAA con ID 4249.

AMI no es una sola tecnología, sino una integración de muchas tecnologías que proporcionan una conexión inteligente entre los consumidores y los operadores del sistema. AMI brinda a los operadores de sistemas, una herramienta para mejorar en gran medida la refinación de procesos operativos y de gestión de activos basados en datos AMI. AMI proporciona un eslabón esencial entre la red, los consumidores y sus cargas, y la generación y almacenamiento de datos [5].

AMI está comprendida por medidores inteligentes, redes de comunicaciones inalámbricas y una red de retorno de datos (*data backhaul network*). Todas estas brindan en conjunto la recolección de datos medidos en intervalos de tiempo, el procedimiento para el uso en la medición de ingresos y la facturación de la producción. Los sistemas AMI también brindan funcionalidades auxiliares, incluyendo la información para la administración de cortes y la automatización del corte y reconexión remota. Este componente de las Smart Grids proyecta involucrar el despliegue de medidores inteligentes que manejen conectividad de red (generalmente comunicaciones inalámbricas, PLC<sup>2</sup> o BPL<sup>3</sup>), entre los puntos finales de la red y el centro de control de energía, y cualquier hardware o aplicación que permita la interoperabilidad y la respuesta automática utilizando bases de datos compartidas dedicadas [1] [6].

En la Figura 1, se puede observar los elementos de AMI y el respectivo medio de comunicación entre estos. El cliente tiene una red denominada HAN-*Home Area Networks*- (Redes de los hogares), esta red no es indispensable para AMI, pero si debe tenerse en cuenta para el medidor inteligente; estos equipos proporcionan, mediante el centro de gestión la información, el control de los parámetros de calidad y de programación del servicio junto con la actualización del software de medición de forma telemática.

---

<sup>2</sup> PLC: Power Line Communications (Comunicación mediante Líneas de Potencia).

<sup>3</sup> BPL: Banda ancha por línea eléctrica.

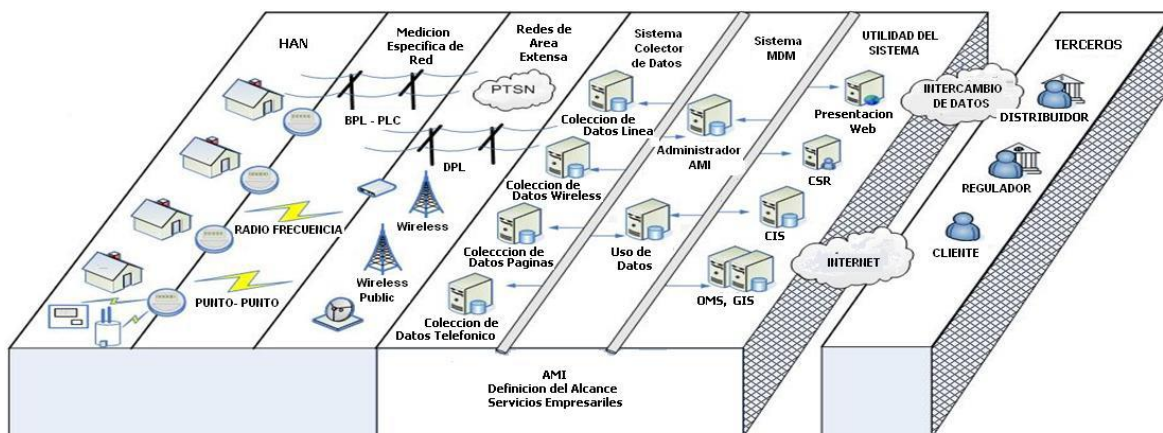


Figura 1. Componentes de un Sistema AMI.  
[Fuente: [7]]

### 1.2.1.1. Tecnología de comunicaciones empleadas en AMI

Los medidores inteligentes deben ser capaces de enviar la información recopilada a la computadora para analizar y recibir comandos operativos desde el centro de operaciones. Por lo tanto, la comunicación estándar es una parte importante de AMI.

La arquitectura más practicada es recoger los datos de los grupos de medidores de concentradores de datos locales, y luego transmitir los datos utilizando un canal de retorno para el comando central donde los servidores, el almacenamiento de datos e instalaciones de procesamiento así como las aplicaciones de gestión y facturación residen [5]. Como diferentes tipos de arquitecturas y redes están disponibles para la realización de AMI, existen diversos medios y tecnologías de comunicación para este propósito también. Algunos ejemplos son:

- ✓ Power Line Carrier (PLC).
- ✓ Banda ancha sobre líneas eléctricas (BPL).
- ✓ Red celular.
- ✓ WiMax.
- ✓ Bluetooth.
- ✓ General Packet Radio Service (GPRS).
- ✓ Internet.

Las tecnologías nombradas anteriormente permiten el flujo de información desde los puntos de acceso (medidores inteligentes) hasta los centros de gestión de información. Si la comunicación desde el centro de gestión está destinado a ser

distribuido a cada uno, o más de los medidores inteligentes, entonces se necesita una mayor tasa de transferencia y más ancho de banda; sin embargo, como el número de dispositivos o velocidad de datos puede aumentar para cubrir los edificios de oficinas o casas grandes, el sistema se puede colapsar. Es aquí donde se brinda otra alternativa de conexión al sistema, una herramienta que va a jugar un papel clave en las nuevas instalaciones, basada en tecnología inalámbrica.

▪ **IEEE 802.15.1, Bluetooth**

Bluetooth ® es una tecnología de conectividad inalámbrica de baja potencia que se utiliza para transmitir audio, transferir datos y transmitir información entre dispositivos [8].

El estándar funciona utilizando radiofrecuencias dentro de un ancho de banda permitido y controlado y fue diseñado para que multitud de dispositivos pudieran conectarse entre si independientemente del fabricante. Por eso todos los dispositivos bluetooth siguen una pila o stack de protocolos implantados dentro del marco del IEEE, que es lo que realmente permite que diferentes tipos de dispositivos puedan conectarse y comunicar [9].

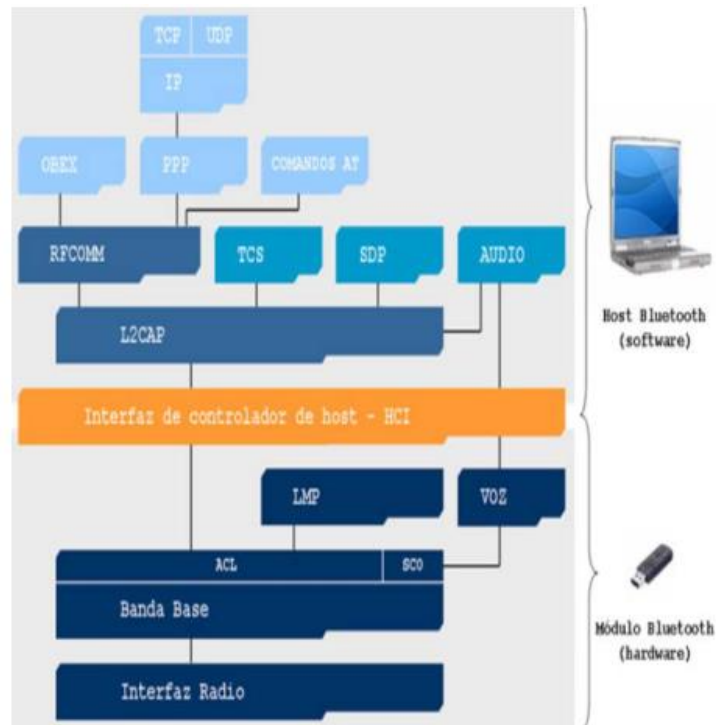


Figura 2. Protocolos Bluetooth.  
[Fuente: [9]]



Bluetooth funciona en el rango de 2400-2483 MHz (banda de 2,4 GHz), esta banda de radiofrecuencia se denomina banda ISM (Industrial, Scientific and Medical), es una banda segura y reservada internacionalmente para uso no comercial [9].

### **1.2.2. Usabilidad**

De acuerdo con la norma internacional ISO 9241, la usabilidad se puede describir como *"el grado en que un producto puede ser usado por determinados usuarios para conseguir objetivos específicos con efectividad, eficiencia y satisfacción en un contexto de uso"* [10].

La usabilidad es un aspecto de calidad relacionado con el esfuerzo que se requiere para usar un producto. Aunque este concepto se puede aplicar a los productos de software, no es una característica particular que viene de la informática. La noción de usabilidad puede utilizarse en diferentes contextos para cualquier tipo de producto [11].

Por otra parte, en la norma internacional ISO 25010, la facilidad de uso se establece como *"la capacidad de un producto de software para ser entendido, aprendido y agradable para el usuario, cuando se utiliza en condiciones específicas"* [12]. Este concepto no sólo está relacionado con la facilidad de uso, sino que también considera un conjunto de atributos que las interfaces de usuario deben cumplir para obtener un sistema utilizable. Entre estos factores se encuentran: comprensibilidad, capacidad de aprendizaje, operatividad y atractivo.

Además, las interfaces de usuario también deben ser intuitivas. Las personas que utilizan el sistema no deben tener que preguntarse sobre el significado de un elemento. El sistema debe permitir un aprendizaje rápido de su funcionalidad para facilitar la interacción a través de los tiempos continuos en los que se utiliza la aplicación de software. Por último, el diseño gráfico debe ser atractivo para los usuarios finales. El cumplimiento de todos estos factores proporcionará un alto grado de usabilidad para una agradable experiencia de usuario. Sin embargo, todavía es necesario tener en cuenta que la usabilidad depende estrictamente de las circunstancias particulares de uso [11].

#### **1.2.2.1. Diseño centrado en el usuario (DCU)**

Es una aproximación al diseño de productos y aplicaciones que sitúa al usuario en el centro de todo el proceso. Así, se puede entender el DCU como una filosofía

cuya premisa es que, para garantizar el éxito de un producto, hay que tener en cuenta al usuario en todas las fases del diseño [13].

El DCU es visto como un proceso cíclico en donde el diseño se realiza siempre teniendo al usuario presente y donde la usabilidad del diseño es evaluada de forma iterativa y mejorada incrementalmente.

El Estándar ISO 13407: *Human-centred design processes for interactive Systems* [14], constituye un marco que sirve de guía para conseguir el desarrollo de sistemas interactivos usables incorporando el DCU durante el ciclo de vida del desarrollo. El estándar describe estas cuatro actividades principales:

- ✓ Entender y especificar el contexto de uso.
- ✓ Especificar los requisitos de usuario y de la organización.
- ✓ Producir soluciones de diseño.
- ✓ Evaluar los diseños en base a los requisitos

Y cuya naturaleza iterativa se refleja en la Figura 3.

Es importante aclarar que la ISO 13407 describe los principios básicos sin estipular métodos específicos. La secuencia de realización o seguimiento de las actividades y el nivel de esfuerzo y detalle apropiado a cada proyecto varía dependiendo del entorno de diseño y el estado del proceso del mismo, por esta razón se torna complejo aplicar este estándar en un desarrollo.

Más allá de los estándares ISO, otras aproximaciones han definido modelos y marcos de trabajo en los que integran las actividades de Diseño Centrado en el Usuario dentro del proceso de desarrollo, como el framework LUCID [15] o el modelo MPlu+a [16]. Siendo este último el que se tendrá en cuenta para el desarrollo de este proyecto.

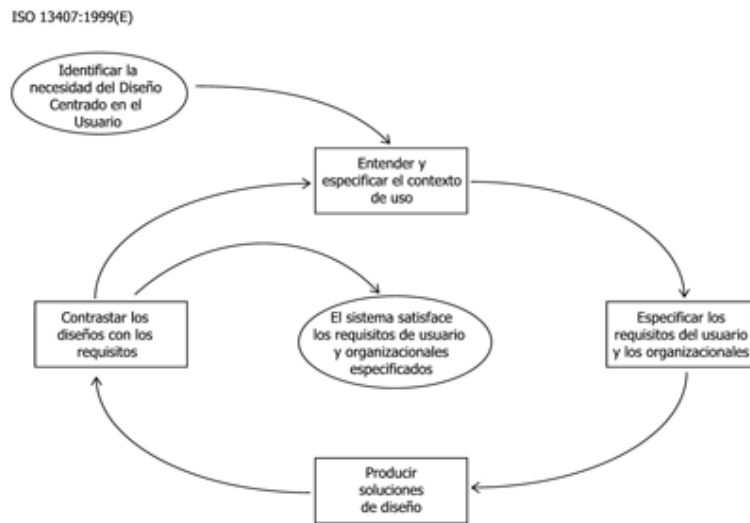


Figura 3. Etapas del DCU.  
[Fuente: [14]]

### 1.2.3. Sistema operativo Android

Inicialmente desarrollado por Android Inc, posteriormente adquirido en 2005 por Google. Solo hasta el 2008 se popularizó gracias a la unión del proyecto de Open Handset Alliance que es un consorcio en el cual se encuentran 48 empresas de desarrollo hardware, software y telecomunicaciones y quienes promocionaron el software libre, aunque es Google quien ha publicado la mayor parte del código fuente del sistema operativo [2].

Android es un sistema operativo y una plataforma software basada en Linux para dispositivos móviles<sup>4</sup>. Para realizar aplicaciones en este, se programa en un entorno de trabajo (framework) de Java y una máquina virtual Dalvik con compilación en tiempo de ejecución.

Para mayor información del sistema operativo Android, remitirse al Anexo B.

---

<sup>4</sup> En un principio se diseñó para Smartphones, luego se expandió para soportar dispositivos como tablets, televisores y hasta relojes.

#### **1.2.4. Interfaz de Usuario**

La interfaz de una aplicación es como la ropa que viste para salir a la calle. Es también la capa que hay entre el usuario y el corazón funcional de la app, el lugar donde nacen las interacciones. En mayor medida está compuesta por botones, gráficos, íconos y fondos [17].

La interfaz de usuario es la principal sección de interacción entre el usuario y el dispositivo, es por medio de ésta que se accede a todas la funcionalidades disponibles. Como bien se conoce dentro de una actividad se encuentra la interfaz de usuario, pero para construir esta interfaz se emplean diferentes objetos los cuales son llamados Views; estos Views se pueden organizar según se requiera mediante el uso de layout, el cual es un recurso que describe lo que se quiere mostrar en la interfaz de usuario. Los layouts se programan en código XML, que tiene un formato similar a HTML. Para asociar el layout a la actividad, este debe ser cargado desde el código de la actividad en el método *onCreate()* [18].

Teniendo en cuenta todos estos conceptos teóricos-introductorios se propone una aplicación móvil usable que facilita la interacción con un sistema basado en tecnología AMI con Sistema de Amarre Automático (AMI-SAA<sup>5</sup>), del sector eléctrico existente, que es el resultado del desarrollo del proyecto.

### **1.3. ESTADO DEL ARTE**

#### **1.3.1. Objetivos**

##### **1.3.1.1. Objetivo general**

Proponer una aplicación móvil usable que facilita la interacción con un sistema basado en tecnología AMI-SAA del sector eléctrico existente.

---

<sup>5</sup> Este trabajo de grado se desarrolla en el marco del proyecto “Diseño e implementación de un sistema de infraestructura avanzada de medición soportado en tecnología de identificación de balances energéticos en transformadores de distribución” (AMI-SAA) con ID 4249.

### **1.3.1.2. Objetivos Específicos**

Capturar requisitos hardware y software para una aplicación móvil que facilite la interacción con un sistema basado en tecnología AMI-SAA en una red eléctrica existente.

Desarrollar una aplicación móvil usable que facilite la interacción con un sistema basado en tecnología AMI-SAA del sector eléctrico existente.

Evaluar la usabilidad y funcionalidad de la aplicación móvil en un sistema basado en tecnología AMI-SAA del sector eléctrico existente.

### **1.3.2. Planteamiento del problema**

#### **1.3.2.1. Definición del problema**

Smart Grid o red inteligente es el despliegue de una infraestructura de energía de última generación capaz de soportar los suministros intermitentes de energía renovable, los flujos bidireccionales de prosumidores (clientes que tanto producen y consumen energía) y la gestión cada vez más inteligentes de los activos [19].

La alta demanda del recurso energético está obligando a las Compañías de energía eléctrica a pensar en nuevas formas de proveer el servicio; es allí donde Smart Grid contribuye a mejorar los procesos de las redes eléctricas convencionales, tales como: las pérdidas de energía durante la distribución, el enganche ineficiente de los usuarios a la red y el costo elevado para su mantenimiento. Esta nueva tecnología soportada de las TIC, permite desarrollar una red eléctrica eficiente y fiable, que mejore la seguridad y calidad del suministro de energía, de acuerdo con los avances de la era digital [20].

En este contexto, surge la tecnología AMI como soporte para mejorar los anteriores procesos. Esta tecnología, es uno de los principales componentes de Smart Grid y está constituida por medidores avanzados o contadores inteligentes que monitorean, se comunican entre sí, y realizan control para optimizar el consumo de energía. AMI implementa sistemas de gestión para almacenar los datos de medición y control de procesos, brinda beneficios financieros, mejora los servicios y permite involucrar las nuevas fuentes de energías renovables a la red eléctrica convencional.

AMI-SAA, es un proyecto ejecutado por Compañía Energética de Occidente (CEO) y la Universidad del Cauca, financiado durante el año 2016 por COLCIENCIAS, el cual pretende tomar la lectura del consumo de los medidores de energía, en una red eléctrica, garantizándose previamente su vínculo a un transformador identificado; permitiendo esto, dar solución a fallas en el cálculo de balances energéticos<sup>6</sup> (por malos amarres en los circuitos de distribución), generando esto un mejor redireccionamiento de la inversiones que hacen las Compañías de energía en la gestión de pérdidas.

En la actualidad, el sector industrial ofrece equipos que permiten obtener información almacenada en los medidores de energía eléctrica de los usuarios [21] pero estos son robustos, con alto valor de adquisición, con baterías no duraderas y con problemas de mantenimiento [22]; estos inconvenientes implican gastos económicos elevados, por esta razón, sería ideal migrar a terminales inteligentes que ayuden a suprimirlos. En este sentido, es necesario definir los lineamientos que permitan el desarrollo de una aplicación móvil, identificando las características que ésta debe tener, soportándose en la tecnología AMI para facilitar la gestión, el apoyo para las brigadas y una mejor toma de medidas, por parte de las Compañías de energía.

### **1.3.2.2. Justificación**

Teniendo en cuenta la problemática anteriormente expuesta, se plantea la pregunta de investigación: ¿Cómo desarrollar una aplicación móvil usable que facilite la interacción con un sistema basado en tecnología AMI con Sistema de Amarre Automático en el sector eléctrico existente? Es pertinente, tener presente que en Colombia, y en el Departamento del Cauca, ya se comenzaron a desarrollar los esquemas necesarios para la incorporación de Smart Grid en los sistemas de potencia [23]; esta investigación se prevé que contribuirá a mejorar la gestión de pérdidas de las Compañías distribuidoras de energía locales y el servicio que estas prestan a los usuarios.

---

<sup>6</sup> *Balance energético*: Parámetro indicador en la gestión de perdidas; obtenido de la deducción de la energía entregada por el transformador y la reportada por los medidores de energía eléctrica vinculados a este.

## **1.4. REVISIÓN SISTEMÁTICA**

Se define revisión sistemática como un estudio integrativo, observacional, retrospectivo, secundario, en el cual se combinan estudios que examinan la misma pregunta [24].

El proceso de búsqueda y selección de información, es un conjunto de técnicas que se adquieren de acuerdo a la información que se necesite y las herramientas que se tengan a disposición, como los motores de búsqueda, buscadores académicos y bibliotecas digitales que se encuentran en la red; con ello se encontraran un sin número de artículos, publicaciones en revistas e información relacionada con el tema a buscar.

En el presente apartado se realiza la revisión sistemática con el fin de filtrar artículos relacionados con la tecnología AMI y las Smart Grid; estos artículos permiten conocer varios conceptos, la arquitectura que maneja AMI en el sector eléctrico y los lugares donde ya se empieza a aplicar Smart Grid.

### **1.4.1. Herramientas de búsqueda**

En este caso, se hizo uso del buscador académico Google Scholar, la biblioteca digital IEEE Xplore y de la plataforma ScienceDirect.

#### **1.4.1.1. Google Scholar**

Google Scholar (GS) es un buscador especializado en recuperar documentos científicos y en identificar las citas que éstos han recibido [25], facilita la búsqueda de trabajos de investigación en grandes áreas como: informática, física, derecho y medicina, debido a que emplea algoritmos diseñados para el entorno académico.

Para la búsqueda se filtró por fecha utilizando el rango comprendido entre el año 2012 y el 2016, se hizo uso de palabras claves donde se incluyeron artículos en inglés y español. Además en Google Scholar se observan las versiones que tiene cada artículo encontrado, cuántas veces fue citado y los artículos relacionados.

#### **1.4.1.2. IEEE Xplore Digital Library**

El IEEE *Xplore Digital Library* es un poderoso recurso para el descubrimiento y el acceso a los contenidos científicos y técnicos publicados por el IEEE (Instituto de Ingenieros Eléctricos y Electrónicos) y sus socios editores. IEEE *Xplore* proporciona acceso a la Web a más de cuatro millones de documentos de texto completo de algunas de las publicaciones más citados del mundo en ingeniería eléctrica, informática y electrónica [26].

#### **1.4.1.3. ScienceDirect**

ScienceDirect es una plataforma electrónica fácil de usar que ofrece artículos en texto completo escritos por investigadores de renombre internacional.

El profesional de información de las empresas puede agilizar su flujo de trabajo ahorrando tiempo con datos fáciles de buscar, además de ofrecer a los usuarios acceso simultáneo a las publicaciones de literatura científica más recientes [27].

#### **1.4.2. Estrategia de búsqueda**

Para la búsqueda, se hizo uso de diferentes estrategias:

- ✓ Búsqueda avanzada
- ✓ Búsqueda por palabras clave. Ej: Infraestructura de medición avanzada (AMI), Smart Grid, AMI, Architecture
- ✓ Búsqueda de frase exacta. Ej: “advanced metering infrastructure”
- ✓ Búsqueda en título del documento Ej: intitle: advanced metering infrastructure
- ✓ Búsqueda con tipo de archivo. Ej: filetype: pdf
- ✓ Búsqueda con operadores AND, OR y NOT

#### **1.4.3. Resultados de la búsqueda**

En las Tabla 1,

Tabla 2 y Tabla 3 se observan las palabras clave que se usaron en los 3 motores de búsqueda para filtrar los artículos relevantes que permitieron la creación del Estado del Arte de este proyecto.



Tabla 1. Resultados búsqueda en Google Scholar.  
[Fuente: propia]

<b>BUSQUEDA Y PALABRAS CLAVES</b>	<b>RESULTADOS GOOGLE SCHOLAR</b>
"advanced metering infrastructure"	8.130
"smart grid" AND "advanced metering infrastructure"	6.650
"smart grid" AND "advanced metering infrastructure" AND "architecture"	3.940
"advanced metering infrastructure" AND "AMI"	8.750
[intitle: advanced metering infrastructure]	31
[intitle: advanced metering infrastructure AND smart grid]	16
[intitle: "advanced metering infrastructure" AND architecture]	4

Tabla 2. Resultados búsqueda en IEEE Xplore Digital Library.  
[Fuente: propia]

<b>BUSQUEDA Y PALABRAS CLAVES</b>	<b>RESULTADOS IEEE XPLORE</b>
("Document title": advanced metering infrastructure)	9145
("Document title": Advanced metering infrastructure AND "Document title": AMI)	0
("Document title": Advanced metering infrastructure OR "Document title": AMI)	9145
("Document title": Advanced metering infrastructure OR "Document title": Smart grid)	29649
("Document Title":Advanced AND "Document Title":metering AND "Document Title":infrastructure OR "Document Title":AMI)	438

Tabla 3. Resultados búsqueda en ScienceDirect.  
[Fuente: propia]

<b>BUSQUEDA Y PALABRAS CLAVES</b>	<b>RESULTADOS SCIENCEDIRECT</b>
TITLE(advanced metering infrastructure)	12
TAK(advanced metering infrastructure)	83
TITLE(advanced metering infrastructure AND Smart grid)	1
TAK(advanced metering infrastructure AND Smart grid)	59
TITLE(advanced metering infrastructure AND "AMI")	0
TAK(advanced metering infrastructure AND "AMI")	51
TITLE(advanced metering infrastructure AND architecture)	0
TAK(advanced metering infrastructure AND architecture)	9

Los motores de búsqueda utilizan operadores booleanos que permiten realizar búsquedas más eficientes como AND, OR y NOT, su correcta asociación con las palabras clave, permitieron que se redujeran el número de artículos encontrados y cada vez más relacionados con el tema de interés.

La mayoría de artículos relevantes en el tema de estudio se encontraron en el motor de búsqueda IEEE, donde se usó la combinación ("**Document Title**":Advanced AND "**Document Title**":metering AND "**Document Title**":infrastructure OR "**Document Title**":AMI), del cual se obtuvieron 438 artículos, que luego fueron filtrados manualmente de acuerdo a su fecha de publicación y su enfoque.

En la Tabla 4 se encuentra un resumen de cada uno de los paper que conforman el estado del arte; ahí se aprecia una breve descripción de ellos y las principales tecnologías que se estudian. Por otra parte se evidencia algunas brechas que existen entre dichos papers y el proyecto que se va a realizar.

Tabla 4. Comparación de diferentes trabajos relacionados.

[Fuente: propia]

NOMBRE	DESCRIPCIÓN	TECNOLOGÍAS	BRECHAS
<b>A Privacy-Aware Communication Scheme in Advanced Metering Infrastructure (AMI) Systems [28]</b>	<p>Este proyecto propone una red AMI que permita la recolección de la información de los usuarios, siempre garantizando la seguridad de la misma. El SM estará conectado con el ES de manera continua, lo que permite que el ES obtenga la información casi en tiempo real. Se propone enlaces celulares o comunicación por medio de redes de sensores para la recolección de la información mediante</p>	<p>Red celular (por ejemplo, GPRS, 3G o LTE). Enlaces mediante Wireless Sensor Network. Time Division Multiple Access (TDMA).</p>	<p>La principal diferencia que se presenta con respecto al proyecto a proponer, es que hay una ausencia en el establecimiento de comunicación por parte de los SM con otro dispositivo que no sea el ES u otro SM, que en este proyecto es un terminal móvil, con el cual se pretende tener acceso de la información de forma segura y confiable pero no de manera constante.</p>

	la red eléctrica.		
<b>Wireless Broadband Architecture Supporting</b> [29]	<p>En este trabajo, se fijan en la implementación de AMI con el uso de tecnologías inalámbricas de acceso de banda ancha, tales como LTE o WiMAX.</p> <p><i>Conectividad directa</i> donde cada SM se conecta al sistema de gestión de datos o con otros SM de forma directa (peer to peer)</p> <p><i>Conectividad indirecta</i> el SM se comunica a un AP de forma inalámbrica por si se necesita datos en campo o por si el sistema de gestión ha sufrido alguna falla.</p>	<p>Conexión de acceso inalámbrico secundaria (por ejemplo WiFi).</p> <p>Conexión inalámbrica de corto alcance (Más bien espectro con licencia o sin licencia), como ZigBee, WiFi, etc.</p>	<p>Existen diferentes tipos de arquitecturas de red de comunicaciones AMI. En este trabajo se estudian tecnologías de banda ancha, pero el proyecto que se propone se basa en la tecnología PLC.</p> <p>Para el caso de estudio que el proyecto plantea, se tiene en cuenta la conectividad directa, debido a que en ella se hace alusión al AP, que es el homólogo al terminal móvil con el cual que se pretende acceder a la información.</p>
<b>A survey on Advanced Metering Infrastructure</b> [30]	<p>En este documento se realiza una introducción a AMI, qué tecnologías se pueden emplear para recoger los datos, qué ventajas y desventajas presentan, qué caracteriza una red AMI, cuales son los dispositivos que conforman AMI (Medidores inteligentes, proveedor de servicio, etc).</p>	<p>Banda ancha por línea eléctrica (BPL). PLC.</p> <p>Redes de radiofrecuencia, así como redes públicas tales como teléfono fijo, celular, etc.</p> <p>Tecnologías inalámbricas son la solución dominante: WiFi 2.4 GHz, (protocolo 802.11), ZigBee (protocolo 802.15.4), tecnológicamente similar a Bluetooth y HomePlug.</p>	<p>Este artículo es una base que permite un acercamiento hacia la tecnología PLC que el proyecto busca implementar para dar solución a la problemática.</p> <p>Se tiene en cuenta, la aclaración que se hace con respecto a los nuevos anchos de banda que se requerirían y que en un futuro podría sustituir a PLC como la tecnología base de AMI.</p>
<b>Advanced Metering Infrastructure for</b>	En este trabajo se estudia la ejecución de	Protocolo avanzado de la IEC, IEC61850	Este proyecto sirve como base para

<p><b>Smart Grid Applications [31]</b></p>	<p>un PQA, el cual sirve según [C. Sellvam] para medir los parámetros eléctricos básicos como el voltaje de fase, corriente de fase, el poder y la energía, junto con los parámetros de calidad de energía, interrupciones, armónicos, desviación de magnitud, desviación de frecuencia, las desviaciones de fase de voltaje y corriente. Las tarifas mensuales también se pondrán a disposición de calcular la cantidad de energía consumida por cada mes.</p>	<p>Internet a través de Ethernet.</p> <p>Aplicación de la base de TCP / IP.</p>	<p>contextualizar al problema que se va a solucionar, pero la brecha más importante está en que la tecnología a usar es Ethernet, a diferencia de la solución que se busca ofrecer que se fundamenta en la tecnología PLC. Hay que rescatar la importancia del PQA, con el cual se evalúan diferentes magnitudes que permiten realizar gestión a la red y que es un parámetro que se busca brindar en la solución del proyecto a realizar.</p>
<p><b>Advanced Metering Infrastructure in the context of Smart Grids [19]</b></p>	<p>AMI es una tecnología que facilita la obtención de datos de los usuarios de una red eléctrica mediante el uso de SM. Es una tecnología que está avanzando y para la cual el ETSI está desarrollando estándares para establecer las comunicaciones entre los diferentes dispositivos que conforman una red AMI: el dispositivo final (medidor inteligente), el concentrador / puerta de enlace, y la plataforma de</p>	<p>ZigBee (IEEE 802.15.4), Wi-Fi y Home tapón (PLC).</p> <p>Protocolo modelo de datos IEC 62056.</p> <p>El protocolo TCP o UDP en la capa de transporte, y el protocolo IP en la capa de red.</p>	<p>Con relación a lo que se busca en el proyecto, este artículo ayuda a complementar acerca de lo que la tecnología AMI pretende en el futuro de las redes eléctricas tradicionales. La comunicación entre medidores inteligentes presenta un soporte a las comunicaciones que se desea entre un medidor inteligente y un dispositivo móvil que permita realizar gestión a la red.</p>

	servicios.		
<b>Análisis de factibilidad para la implementación de un sistema AMI (Advanced Metering Infrastructure) mediante contadores inteligentes por parte de la Empresa Eléctrica Azogues C.A. [32]</b>	<p>El objetivo de esta tesis es el estudio, diseño y análisis de factibilidad para la implementación de un sistema AMI que permitirá realizar: lecturas, cortes y reconexiones de manera remota.</p> <p>Además se presentan las diferentes tecnologías que permiten la transmisión de datos entre el sistema comercial de la empresa y el sistema de medición de cada cliente.</p>	<p>Tecnología PLC Tecnología GPRS Tecnología Wi-Fi</p>	<p>Es un trabajo que permite identificar las diferentes tecnologías en las cuales una red AMI se podría apoyar. Es importante tener presente que en este documento se hace mención a la tecnología PLC, la cual, es la base del proyecto que se va a realizar.</p>

## CAPITULO II

### METODOLOGÍA

#### 2.1. INTRODUCCIÓN

La metodología para la gestión del proyecto se basará en el Modelo Integral para el Profesional en Ingeniería [33], la cual consta de las siguientes fases: información, preposicional y de análisis, verificación y documentación y divulgación. Según la metodología que se utilice, existen varias clasificaciones de los tipos de requisitos [34]. Una de ellas es la que distingue a los requerimientos entre funcionales y no funcionales [34].

*Requisitos funcionales:* Declaraciones de los servicios que proveerá el sistema, de la manera en que éste reaccionará a entradas particulares y de cómo se comportará en situaciones particulares. Pueden declarar lo que el sistema no debe hacer [34]. Para la implementación de los requisitos funcionales, se empleará una adaptación de la metodología de desarrollo **Scrum** [35] en contextos no ideales<sup>7</sup>.

*Requisitos no funcionales:* Son aquellos requisitos que no se refieren directamente a las funciones específicas que entregará el sistema, sino a las propiedades emergentes de éste tales como la fiabilidad, el tiempo de respuesta, la capacidad de almacenamiento, etc. En general se refieren al sistema como un todo y no a rasgos particulares, por lo tanto, una falla de un requerimiento no funcional, podría inutilizar un sistema [34]. Para la implementación de los requisitos no funcionales se empleará la metodología **MPlu+a** (*Modelo de Proceso de la Ingeniería de la Usabilidad y de la Accesibilidad*), en la cual los requisitos no funcionales tienen prioridad.

Para el desarrollo de productos software es ideal garantizar su calidad, por esta razón se considera la usabilidad como un factor importante dentro de este contexto. Existen métodos que permiten el análisis de la usabilidad, inicialmente desarrollado para aplicaciones de escritorio.

---

<sup>7</sup> Proyecto “Una ruta de trabajo para la adopción de Scrum en pequeñas organizaciones de la industria de software”, Viviana Araujo y Natalia Castrillón estudiantes del programa de Ingeniería de Sistemas.

El desarrollo masivo de dispositivos móviles y múltiples aplicaciones para los mismos hacen que sea necesario hacer el estudio de usabilidad para el éxito de ellas. Al igual que cualquier desarrollo software sufren de los mismos problemas en cuanto a su constante innovación, cambios de plataforma de desarrollo y su simplicidad, por lo tanto es necesario que se elija una buena metodología de desarrollo que nos permita hacer un seguimiento constante de la aplicación.

Una de las metodologías de usabilidad más utilizadas en aplicaciones es la MPlu+a que garantiza que los sistemas sean usables y accesibles a todas las personas. Lo ideal en un desarrollo software es que la interacción entre el usuario y la aplicación sea intuitiva, efectiva y eficiente, donde la aplicación sea rápida y sencilla, y su uso no implique grandes conocimientos ni un gran proceso de aprendizaje, donde se debe generar una buena experiencia de usuario.

## 2.2. METODOLOGIA

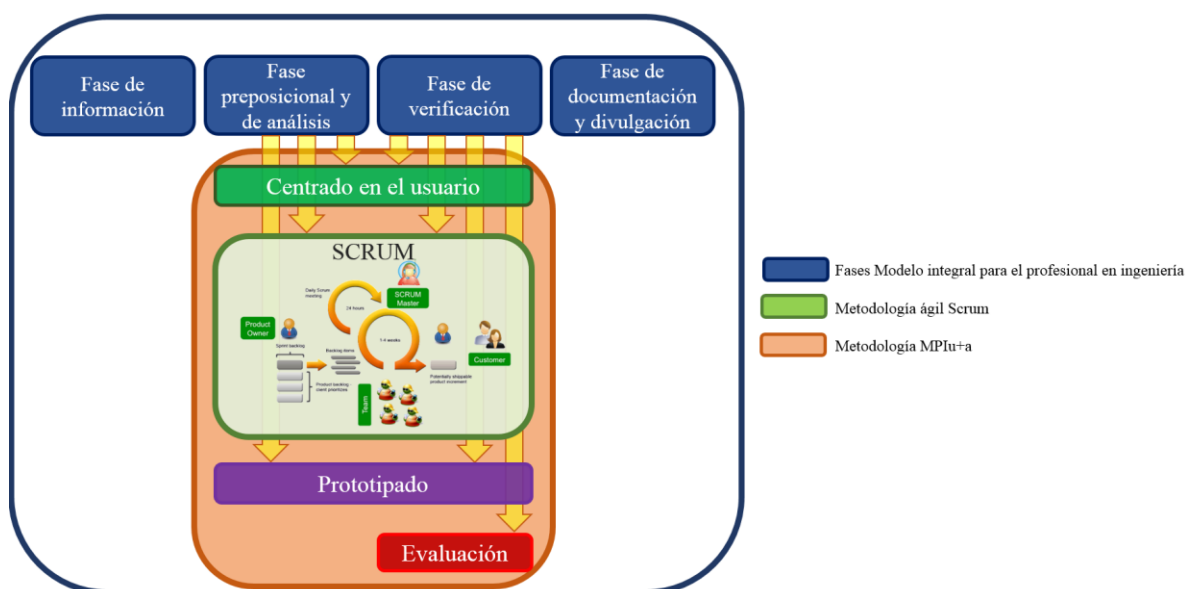


Figura 4. Metodologías usadas  
[Fuente: propia]

En la Figura 4. Metodologías usadas se muestra la relación que hay entre las metodologías utilizadas para el desarrollo del proyecto, en azul se indican las fases del Modelo integral para el profesional en ingeniería usado para la documentación del proyecto, las metodologías SCRUM y MPlu+a se usaron para la implementación de la aplicación móvil dentro del grupo de trabajo del proyecto AMISAA, donde se realizaban reuniones para la asignación de tareas dentro del mismo.

A continuación se dará una descripción de la metodología Scrum y MPlu+a.

## **2.3. SCRUM: MARCO DE TRABAJO PARA EL DESARROLLO DE REQUISITOS FUNCIONALES**

### **2.3.1. Características de Scrum**

Scrum es una metodología ágil para la gestión de desarrollo software, que al ser aplicada de manera correcta genera buenas prácticas para un trabajo colaborativo en equipo, desarrollando productos en menos tiempo, buena calidad y menos costos. Está basado en un proceso iterativo e incremental, donde se hacen entregas parciales y regulares del producto, con el objetivo de obtener resultados lo más rápido posible. Con esta metodología se crean pequeños ciclos para el desarrollo, conocidos como iteraciones, pero llamados en Scrum "Sprints" [36].

Para conocer un poco más sobre el proceso Scrum y sus elementos dirigirse a la sección 2.3.2.

En la Figura 5 se observa una guía del proceso de creación de un proyecto Scrum.

Scrum en cada entrega prioriza beneficios que aportan al buen desarrollo del sistema, entonces será un proceso donde hay una constante realimentación, para la mejora en la adquisición de requerimientos y generando un incremento en el producto final.

Los apartados 2.3.2 y 2.3.3 fueron tomados de [36].



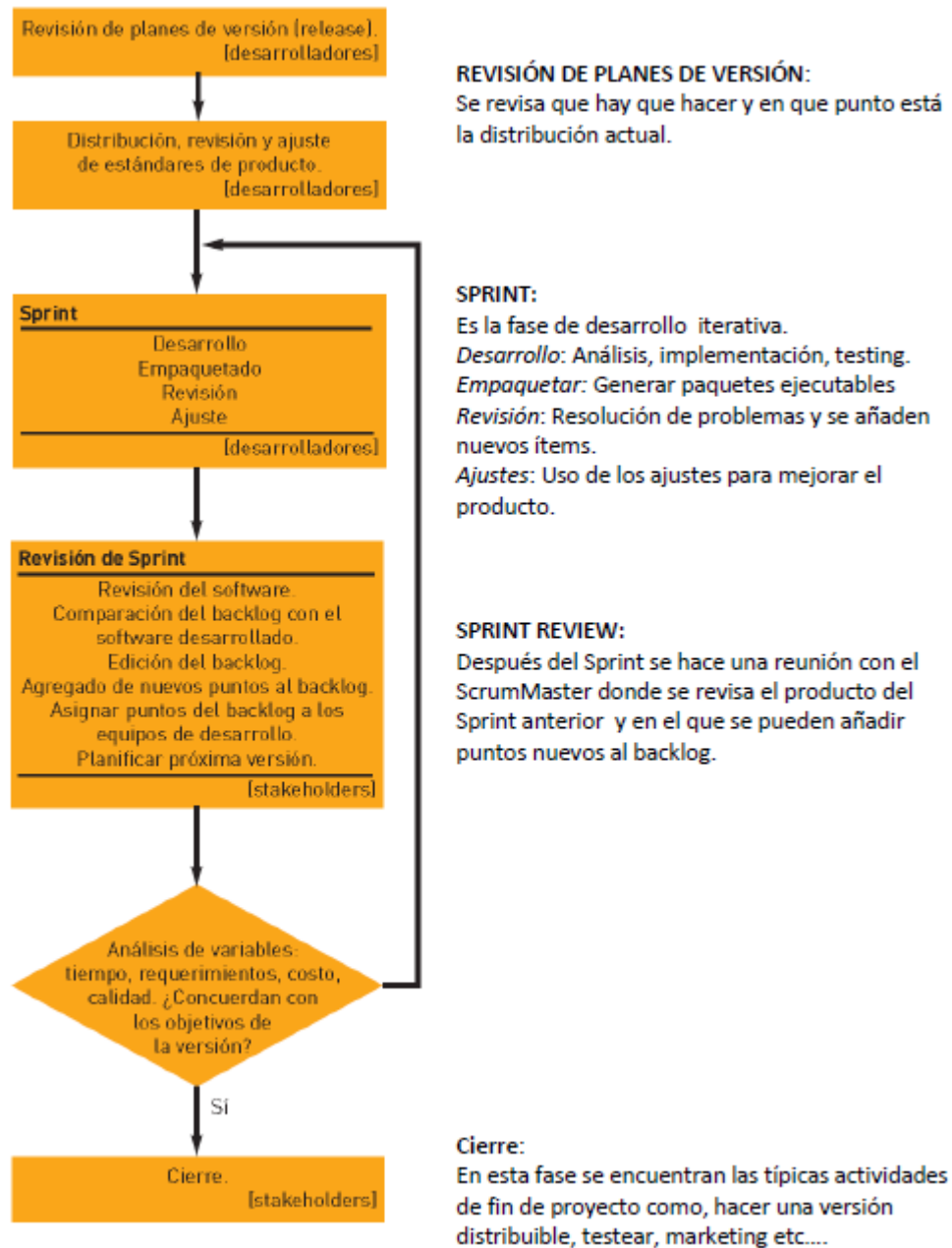


Figura 5. Esquema de las 5 fases de desarrollo en Scrum.  
[Fuente: [36]]

### 2.3.2. Proceso

Scrum se ejecuta en bloques temporales cortos, pero fijos entre 1 a 4 semanas máximo, se gestionan las iteraciones con reuniones diarias, que permiten obtener un producto final en cada sprint, incrementado en cada entrega, con el seguimiento continuo del cliente (Ver Figura 6).

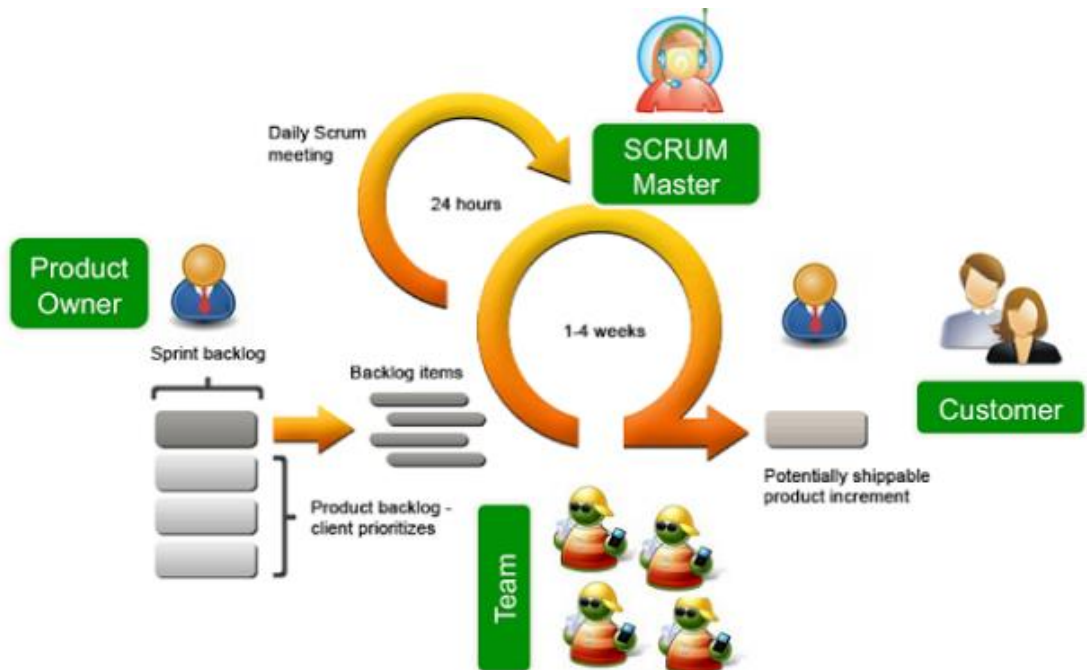


Figura 6. Método ágil Scrum.

[Fuente: <http://programaenlinea.net/wp-content/uploads/2016/06/scrum-diagrama.png>]

El proceso parte de la lista de objetivos/requisitos priorizada del producto, que actúa como plan del proyecto. En esta lista *el cliente prioriza los objetivos balanceando el valor que le aportan respecto a su coste* y quedan repartidos en iteraciones y entregas.

Como se observa en la Figura 6, en los proyectos basados en Scrum se consideran ciertos roles. Los roles para el desarrollo de esta metodología dentro del proyecto, fueron tomados por personas comprometidas con el proyecto AMISAA y ATEMSAA, estos se distribuyeron de la siguiente manera:

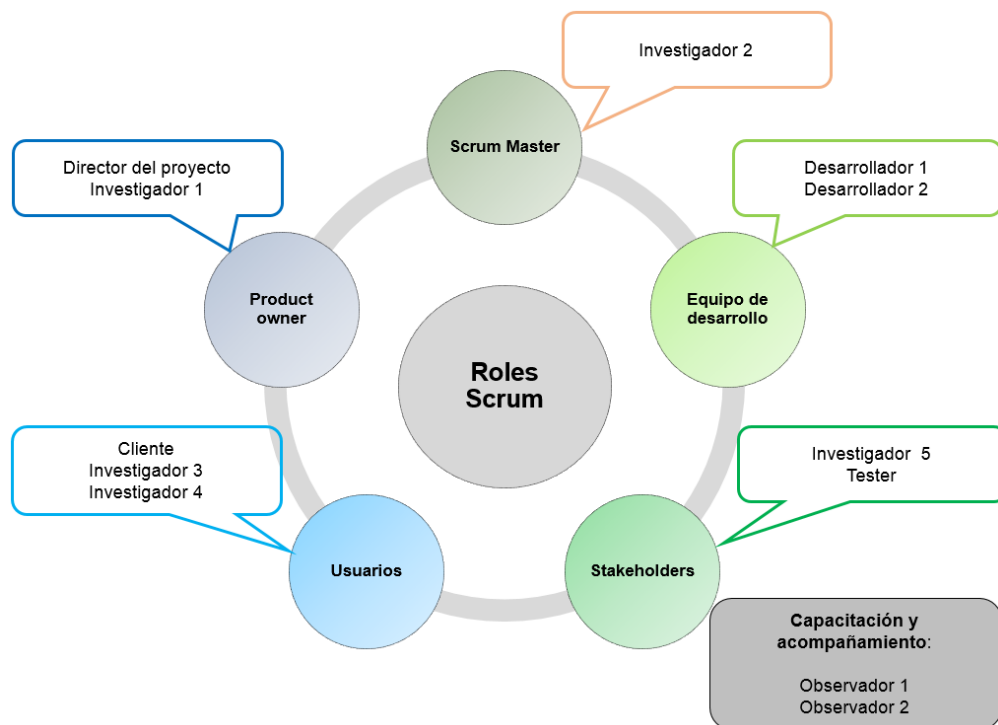


Figura 7. Roles del equipo AMISAA  
[Fuente: propia]

En la Figura 7. Roles del equipo AMISAA se muestran los roles tomados por el personal directamente relacionado con el proyecto AMISAA, el desarrollo de la aplicación se encontraba dentro del rol Equipo de desarrollo, las principales funciones que toma cada rol se muestran a continuación:

**Product Owner:** Es quien determina las prioridades de los entregables, esta persona que toma las decisiones, y en la que realmente conoce el negocio del cliente y su visión del producto. Se encarga de escribir las ideas del cliente, las ordena por prioridad y las coloca en el Product Backlog

**Scrum Master:** Administra y facilita la ejecución del proceso, es el encargado de comprobar que el modelo y la metodología funciona. Eliminará todos los inconvenientes que hagan que el proceso no fluya e interactuara con el cliente y con los gestores.

**Equipo de desarrollo:** Suele ser un equipo pequeño de 5 o 9 personas y tienen autoridad para organizar y tomar decisiones para conseguir su objetivo. Está involucrado en la estimación del esfuerzo de las tareas del Backlog.

**Usuarios:** Es el destinatario final del producto

**Stakeholders:** Trabajan en conjunto para entregar resultados en cada Sprint, son las personas a las que el proyecto les producirán un beneficio. Participan durante las revisiones del Sprint.

La relación directa de cada rol al momento de realizar los Sprints, permitió que el cliente observara en cada momento la evolución del proyecto, logrando retroalimentación, revisión y una correcta planeación del Sprint.

Sus fases de ejecución:

- ✓ **Planteamiento:** un proyecto gestionado bajo el modelo Scrum parte de los objetivos que han trazado con anterioridad el cliente y la empresa. Lo primero es fraccionarlo en entregas parciales, de manera que el cliente pueda replantear aspectos a los que en principio no prestó la importancia debida o que simplemente desconocía. Esos aspectos pueden ser sobre costes, estrategias, producción, etc.
- ✓ **Lista de tareas:** el equipo de trabajo elabora la lista de tareas que debe tener en cuenta para cada entrega de resultados. Es muy importante hacer una estimación del esfuerzo requerido para, de esta manera, fijar plazos proporcionados.
- ✓ **Reuniones:** lo ideal es que cada día el equipo dedique 15 minutos para reunirse y ponerse al tanto de la evolución del plan. En estas sesiones, el líder del proyecto (Scrum Master) debe encargarse de que cada miembro cumpla con las labores que le han sido asignadas y de motivarles para que su productividad no descienda. En caso de que detecte alguna incidencia dentro del grupo, es necesario que tome las opciones más adecuadas, que pueden ir desde un relevo de la función hasta el nombramiento de miembros de apoyo.
- ✓ **Demostración:** una vez ejecutadas las labores de cada fase, el equipo se reúne con el cliente para mostrar los avances correspondientes. El cliente decide si replantea los elementos del proyecto. Si es necesario, el equipo asume nuevos compromisos.
- ✓ **Retrospectiva:** los miembros del equipo se reúnen para valorar el proceso de entrega de resultados y analizan los factores que podrían mejorar de cara al final del proceso. La función del Scrum Master será eliminar dichos obstáculos.

### 2.3.3. Historias de usuario (HU)

Para el desarrollo del prototipo se utilizó la metodología Scrum, la cual ayudó a definir un conjunto de roles, artefactos y eventos que al aplicarse de manera regular generan buenas prácticas para un trabajo colaborativo que permite

mejores resultados en el desarrollo del proyecto debido a las entregas parciales y regulares del producto, que en este caso eran parte de la aplicación móvil.

Scrum fue el indicado para ser aplicado en el desarrollo de la aplicación, pues se requiere obtener resultados lo más pronto posible con el fin de que se definan requisitos y si es necesario cambiarlos en el transcurso del desarrollo, buscando que la aplicación sea más funcional y acorde a lo requerido por el cliente; esta es una ventaja que ofrece el aplicar Scrum, el cliente está en constante contacto con el producto final, en cada una de las fases que se definen, identificando y solucionando contratiempos que se pueden presentar al momento de hacer ciertas pruebas.

Con el uso de Scrum, en este proyecto se definieron sprints cada 15 días donde se entregaron una o dos HU con su respectivo producto que fue probado por el cliente días antes del entrega del mismo. Para hacer un seguimiento al desarrollo de este producto se hicieron reuniones diarias de 15 minutos donde se presentaba lo que se estaba haciendo, los contratiempos y lo que se va hacer.

A continuación se muestra una de las HU, evidenciando cómo se aplicó Scrum para su desarrollo:

***HU 1: yo como brigadista deseo encontrar los dispositivos bluetooth disponibles en el área de cobertura para establecer conexión con el PLC\_MMS.***

*Condiciones de satisfacción:* Se despliega una lista con los dispositivos Bluetooth disponibles en la zona de cobertura. Al encontrar el PLC\_MMS con el que se desea conectar, lo selecciona e ingresa la clave de acceso de seguridad y de esta manera se empareja con el módulo PLC\_MMS. Es necesaria la opción de desconectarse del PLC\_MMS que termina el emparejamiento entre el TU-1 y el PLC\_MMS.

Para el desarrollo de esta HU de plantearon las siguientes actividades:

- ✓ Inicialización del componente bluetooth.
- ✓ Conexión con el dispositivo bluetooth del PLC\_MMS.
- ✓ Pruebas de funcionalidad del aplicativo.
- ✓ Pruebas de usabilidad del aplicativo.

Para representar esta HU gráficamente se usó la técnica **Visual Story Mapping** [37], que permite armar el backlog de manera visual orientado a las HU (ver Tabla 5). El visual story mapping se muestra en dos dimensiones:

- ✓ La horizontal muestra la secuencia de la HU.
- ✓ La vertical muestra las prioridades de la HU.

Tabla 5. Visual Story Mapping para HU 1.  
[Fuente: propia]

Abrir la aplicación	Acceso de usuario	Conexión bluetooth				Desvinculación bluetooth
Abrir la aplicación	Login en la aplicación	Activar bluetooth del dispositivo móvil	Desplegar lista de dispositivos bluetooth asociados al móvil	Buscar dispositivos disponibles en la zona	Conectar al dispositivo bluetooth asociado	Desvincular el dispositivo bluetooth
Abrir aplicación ATEMSAA en background	Login en la aplicación (user: pruebas y pass: gonzalo)	Mostrar ayuda de activación de bluetooth	Ver lista de dispositivos bluetooth asociados al móvil	Escanear dispositivos bluetooth disponibles en la zona	Conectar al dispositivo bluetooth asociado	Desvincular el dispositivo bluetooth en la parte superior con el botón disconnect
Abrir aplicación ATEMSAA por primera vez		Verificar bluetooth activo			Conectar al dispositivo no asociado con la clave de acceso 1234 o 1917	

La parte amarilla muestra las actividades o procesos del negocio del sistema y es llamado Backbone, la primera fila es el Walking Skeleton que representa actividades del sistema que le permiten funcionar de manera correcta. (Esta tabla está basada en [37]).

Para las demás HU se realizó la misma representación gráfica. Para verlas remitirse al Anexo C.

## 2.4. MPlu+a: METODOLOGIA PARA EL DESARROLLO DE REQUISITOS NO FUNCIONALES

### 2.4.1. Características de MPlu+a

La ingeniería de requisitos involucra descubrir cuáles son las metas, necesidades y expectativas de los stakeholders<sup>8</sup>, ajustar las expectativas de los mismos y comunicarlas a los desarrolladores [38].

En el desarrollo de productos software es ideal garantizar su calidad, por esta razón se considera la usabilidad como un factor importante dentro de este contexto [16]. Existen métodos que permiten el análisis de la usabilidad, inicialmente desarrollado para aplicaciones de escritorio.

Una de las metodologías de usabilidad más utilizadas en aplicaciones es la **MPlu+a** que garantiza que los sistemas sean usables y accesibles a todas las personas (ver Figura 8).

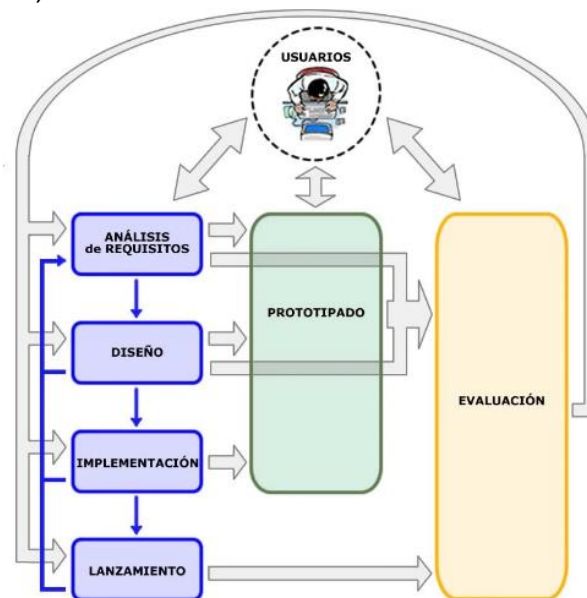


Figura 8. Modelo de Proceso de la Ingeniería de la Usabilidad y la Accesibilidad (MPlu+a).  
[Fuente: [16]]

<sup>8</sup> Grupo de personas afectadas o responsables por la salida de una actividad. Stakeholders de un proyecto son sus miembros, proveedores, clientes, usuarios finales, etc. [69]

Ventajas de MPlu+a:

- ✓ Ayuda a los desarrolladores como guía para el proceso de implementación de un sistema
- ✓ Propone una serie de pautas para el diseño del sistema, basados en tres pilares importantes que son: Ingeniería de software, prototipado y evaluación.
- ✓ El diseño está Centrado en el Usuario.
- ✓ Establece procesos repetitivos que permiten la realimentación del desarrollo del sistema
- ✓ Permite llevar a cabo un trabajo sencillo y eficiente
- ✓ La distribución del personal a cargo del desarrollo del sistemas se hacen por etapas
- ✓ Es flexible ya que no restringe a equipo de desarrollo.

A continuación se describen las fases que componen la metodología MPlu+a (Tomado de [39]).

#### **2.4.2. Análisis de requisitos**

Para no tener inconvenientes de comunicación con los usuarios se aplica en esta metodología técnicas de análisis de requisitos centrados en el usuario, esta fase del modelo de proceso es importante, porque es donde se describe la calidad de los requisitos del sistema desde el punto de vista del usuario, por lo tanto es necesario identificar qué es lo que exactamente desea el cliente y qué condiciones son indispensables para el desarrollo del sistema, se identifican cuál es la necesidad del usuario y va evolucionando de acuerdo al uso del sistema.

Inicialmente se plantean cuáles son los servicios que el sistema debe proporcionar y sus respectivas restricciones para que este funcione de manera correcta, por lo tanto se debe extraer requisitos funcionales, que describen una funcionalidad o servicio del sistema y requisitos no funcionales, que son todas aquellas restricciones que puede tener el sistema para su correcto funcionamiento, garantizando un alto grado de funcionalidad, usabilidad y accesibilidad, aunque el desarrollo está centrado en el usuario es difícil especificar todos los requisitos desde un principio, a medida que el cliente interactúa con el sistema se perfeccionan algunos requisitos y otros son descubiertos, por lo tanto los cambios son inevitables, pero lo ideal será reducir su número y el impacto que tiene sobre el sistema.



### **2.4.3. Diseño**

Es la segunda fase de todo proceso de desarrollo, para llegar a esta fase se hacen varias actividades relacionadas con el análisis de requisitos los cuales proporcionan información al equipo de desarrollo para modelar el sistema, en esta fase se realiza el diseño de la actividad y el diseño de la información.

El diseño de la actividad se relaciona con la especificación funcional, tecnología y las posibilidades del sistema que ofrece para ser usado por las personas, este diseño se consigue analizando las funcionalidades y tareas necesarias para realizarlas.

El diseño de la información da soporte a la percepción, interpretación y comprensión de la información en el sistema, se refiere a elementos interactivos en las interfaces de acuerdo a las tareas, la usabilidad y la consecución funcional de estas tareas.

### **2.4.4. Implementación**

En esta fase se reúne toda la programación del software necesaria para sintetizar la aplicación con sus respectivos procesos que permiten el ensamblaje entre los módulos y dispositivos.

Se debe garantizar la usabilidad y accesibilidad del sistema, por lo tanto deben realizar cuantos prototipos sean necesarios para su correspondiente evaluación.

Es recomendable realizar prototipos software desde los estados iniciales de implementación para su respectiva evaluación por usuarios finales, cuanto antes mejor.

Para acabar, es muy recomendable al final de esta fase y antes de empezar la etapa de lanzamiento evaluar el sistema mediante el método de la evaluación heurística para comprobar la consistencia global del producto justo antes de su puesta en escena.

### **2.4.5. Lanzamiento**

La fase de lanzamiento es la más crítica en cualquier proceso o desarrollo, lo ideal es que no se presenten errores, que no sea complicada usar, que se visualicen

fácilmente las diferentes opciones y sus funcionalidades, además que se obtengan los resultados esperados.

Con la metodología MPlu+a se asegura la usabilidad y la accesibilidad del sistema.

#### **2.4.6. Prototipado**

Cuando se hace un desarrollo es necesario probar las partes que se han implementado, con el objetivo de verificar sus funcionalidades, en este proceso al llegar al final del desarrollo se han realizado diferentes tipos de comprobaciones del sistema.

El MPlu+a no marca ninguna pauta para indicar a los diseñadores en qué situaciones deberán recurrir al uso de una determinada o determinadas técnicas para simular el funcionamiento. Tampoco los limita a poder realizar un primer prototipo en una fase muy inicial del proyecto.

El modelo intenta garantizar que se cumplan los pasos necesarios para disponer de un producto altamente usable y accesible a la vez que concede un alto grado de libertad para que el equipo de desarrollo libremente decida cuándo y cómo deberá aplicar las diferentes técnicas.

Características:

- ✓ Prototipo es una implementación parcial pero concreta de un sistema con el fin de explorar el sistema durante el desarrollo del mismo
- ✓ Los prototipos no se limitan a probar las interacciones con los usuarios, también son útiles en la fase recogida o análisis de requisitos
- ✓ Los prototipos son útiles ya que hay comunicación entre los componentes del sistema, además de la participación activa de los usuarios,
- ✓ Soporte para escoger entre varias alternativas
- ✓ Evaluar el sistema desde sus primeras fases
- ✓ Indispensable para la documentación de conceptos funcionales y tareas concretas del sistema
- ✓ Son el primer paso de una idea abstracta a una concreta
- ✓ Sirve para comprobar la fiabilidad técnica de una idea al responder con la aplicación

## 2.4.7. Evaluación

La evaluación consiste en probar algo que cumpla las expectativas o no de una herramienta.

En MPlu+a, la fase de evaluación del constituye un punto clave para la obtención de sistemas interactivos usables y accesibles. Será en esta fase donde se aplicarán las técnicas necesarias para recibir la realimentación necesaria por parte de los usuarios y/o evaluadores expertos que se verá reflejado en el diseño de las interfaces de los usuarios mejorando sus procesos interactivos. Por tanto, se hablará de la evaluación como:

La actividad que comprende un conjunto de metodologías y técnicas que analizan la usabilidad y/o la accesibilidad de un sistema interactivo en diferentes etapas del ciclo de vida del software

## 2.5. HERRAMIENTAS DE APOYO

Durante el desarrollo del proyecto AMI-SAA fue necesario el uso de herramientas de apoyo con las cuales se buscaba realizar entregas en los tiempos planteados, sin embargo hubo algunas que no arrojaron los resultados esperados, en la Tabla 6. Herramientas de apoyo se muestran las herramientas y/o estrategias usadas:

Tabla 6. Herramientas de apoyo  
[Fuente: propia]

HERRAMIENTA / ESTRATEGIA	FUNCIONO	OBSERVACION
Aplicación Taiga	No	Taiga es una plataforma de gestión de proyectos para metodologías ágiles, a pesar de que era sencilla de usar, no funcionó porque el equipo de trabajo AMISAA, ya tenía una forma de trabajar propuesta y no se adaptó a Taiga
Correo	No	No funcionó porque algunos del equipo de trabajo no siempre estaba pendiente de los correos que se enviaban para hacer revisiones
Whatsapp	No	Inicialmente si funciono, pero después por otras ocupaciones los mensajes no eran respondidos
Reuniones cara a cara	Si	Si funcionó porque se plantaron reuniones con horarios fijos, antes de las entregas de Sprint

Asesores Base de datos	Si	Si funcionó porque permitió plantear la mejor herramienta para el desarrollo de la base de datos para el proyecto AMISAA
Asesores Scrum	Si	Si funcionó porque permitió hacer un correcto seguimiento al desarrollo del proyecto y solución de dudas con respecto a la metodología
Asesores desarrollo móvil	Si	Si funcionó porque permitió solucionar inconvenientes que surgieron al momento de hacer modificaciones en la aplicación móvil

## CAPITULO III

### APLICACIÓN MÓVIL USABLE ATEMSAA

#### 3.1. INTRODUCCION

El suministro de energía eléctrica a los hogares, industria y comercio es el resultado de un proceso de producción, transporte y venta del que hacen parte diferentes actores. Para llevar la energía producida hasta los diferentes puntos de consumo se utilizan grandes y largas autopistas que conforman la red de transmisión, de la que posteriormente se derivan ramales más pequeños, similares a calles, que componen la red de distribución, y allí es donde se transforma la energía a los niveles adecuados para la comercialización o venta de la energía necesaria para el uso de los aparatos electrodomésticos, maquinarias de producción industrial o simplemente iluminación. Como en toda cadena de transporte de un bien, se pueden presentar pérdidas desde el punto de producción hasta la entrega a los usuarios finales [40].

Las pérdidas de energía eléctrica se clasifican en dos categorías en función de su naturaleza: 1) *pérdidas técnicas*, asociadas a características propias de las redes<sup>9</sup>, producidas en los transformadores, conductores y equipos eléctricos, que corresponden a la diferencia entre la energía que se entrega a un sistema y la que es utilizada por el usuario final (independiente del pago de ésta), y 2) *pérdidas no técnicas* asociadas con ineficiencias administrativas y comerciales como facturación y gestión deficientes, o por prácticas de los usuarios como fraude o conexiones ilegales [40].

Las pérdidas no técnicas, debidas principalmente por la falta de honradez de los consumidores de electricidad es un problema que enfrentan todos los servicios públicos de energía. Encontrar medidas eficientes para detectar el consumo de electricidad fraudulenta ha sido un área de investigación activa en los últimos años. [41]. Una manera en que tienen los comercializadores pueden determinar las pérdidas totales es utilizando balances de energía. Estos balances se

---

<sup>9</sup> Características como impedancia, voltaje, longitud de las redes, topología, planeación y operación, entre otros.

pueden realizar en todos los niveles de un sistema de distribución y comercialización de energía.

En el nivel más alto, uno de los datos más importantes es el cálculo total de la energía facturada a los clientes y compararlo con el valor de la energía comprada a los diferentes agentes generadores. Como resultado de esta comparación, se logra obtener un nuevo valor para las pérdidas técnicas y de las pérdidas no técnicas [42].

A nivel de los transformadores de distribución, comparando la energía entregada por estos con la energía registrada por los medidores pertenecientes a su circuito de distribución. La principal dificultad para el desarrollo de estos balances de energía proviene de las asociaciones incorrectas entre transformadores y usuarios. En los operadores de red de nuestro País y en general de países en vías de desarrollo es común que no se conozca cual transformador es el que suministra energía a un determinado usuario. Conscientes del problema de asociación entre transformadores y usuarios y del vacío tecnológico existente para solucionar tal problema, el Grupo de Automática Industrial de la Universidad del Cauca y la Compañía Energética de Occidente realizaron en 2013-2014 un proyecto conjunto de I+D denominado **Sistema de Amarre Automático (SAA)**, el cual consistió en desarrollar una tecnología compuesta por dos tipos de dispositivos electrónicos denominados respectivamente PLC\_TU y PLC\_MMS. Los primeros se conectan en las borneras de los medidores de energía y los segundos en la salida de baja tensión de los transformadores de distribución. Una vez por día los PLC\_MMS solicitan a los PLC\_TU que envíen su código de identificación. El PLC\_MMS registra entonces en un medio de almacenamiento no volátil la lista de códigos de identificación recibidos junto con la correspondiente fecha y hora.

Cuando el SAA se encuentra instalado y probado, la unidad PLC\_MMS se encarga de establecer comunicaciones programadas de forma periódica dos veces cada 24 horas en intervalos predefinidos y en modo encuesta, lo anterior con el fin de monitorear la presencia de los diferentes nodos de usuario (PLC\_TU). Luego de realizar la encuesta a cada PLC\_TU, la unidad PLC\_MMS debe almacenar en un archivo (.cvs) el estado de amarre de las diferentes unidades PLC\_TU, donde al finalizar un mes, que corresponde a un periodo de monitoreo, un operario debe descargar el archivo que contiene la información recolectada.

Esta operación se realiza a través de una unidad móvil de usuario (TU).

En la búsqueda de complementar el sistema de asociación usuario-transformador descrito anteriormente y con miras a lograr un prototipo de infraestructura de medición capaz de efectuar, además de la asociación usuario-transformador, balances de energía de manera automática, surge el sistema denominado **AMI-SAA**.

Dentro de la arquitectura diseñada para el sistema AMI-SAA, se hace necesario tener en cuenta otros terminales. En el caso del terminal de usuario, ahora llamado TU-1, es la *unidad móvil* que permite descargar la información de amarre del sistema de monitoreo y realizar ciertas interacciones en campo del personal del operador de red; el terminal TU-2, que es una unidad encargada de acceder a la *aplicación de gestión* a través de una conexión a internet y un terminal TU-3 que es la unidad encargada de la *configuración de parámetros* tanto del PLC\_MMS como de los otros dispositivos PLC del sistema.

Pensando en la integración de las funcionalidades de los terminales TU-2 y TU-3 al terminal TU-1, surge **ATEMSAA** (*Aplicación en Terminal Móvil para un Sistema de Amarre Automático*), la cual es una aplicación móvil desarrollada para dispositivos con sistema operativo Android bajo criterios y recomendaciones planteados por varios autores y organizaciones a través de los años. Además de pautas, sugerencias, y parámetros creados y utilizados por los desarrolladores del sistema operativo, para la implementación de interfaces gráficas de usuario (GUI, por sus siglas en inglés), lo cual constituye parte fundamental dentro del desarrollo de cualquier aplicación móvil.

ATEMSAA permite establecer comunicación vía inalámbrica con los diferentes módulos PLC desarrollados en el proyecto AMI-SAA, los cuales se encuentran equipados con un módulo Bluetooth [43]. Por otra parte, ATEMSAA permite acceder a la aplicación de gestión a través de una conexión a internet.

### **3.2. AMI-SAA**

La compañía Energética de Occidente S.A.S. E.S.P. y la Universidad de Cauca, desarrollan y evalúan un proyecto, financiado durante los años 2016-2017 por COLCIENCIAS, que tiene como propósito generar una tecnología orientada a la reducción de pérdidas no técnicas eléctricas, específicamente generadas por malos vínculos usuario-transformador. Esta tecnología instalada en el transformador de distribución, utiliza un dispositivo maestro, el cual se comunica con dispositivos esclavos instalados en cada uno de los medidores de energía

eléctrica de los usuarios, conectados a este transformador. El maestro es el encargado de monitorear la permanencia y mantener la información actualizada de cada uno de los usuarios, asociados al transformador que inicialmente han sido conectados, comunicando en línea la información actualizada de vínculo usuarios-transformador a un centro de gestión, donde es analizada a través de una plataforma de visualización y seguimiento Web. La tecnología desarrollada se pone a prueba en un piloto ubicado en un sector residencial de la ciudad de Popayán - Cauca, generando un significativo volumen de información del comportamiento de la tecnología en la identificación de los vínculos usuarios-transformador de la zona del piloto.

El prototipo desarrollado de la tecnología AMI-SAA (ver Figura 9) se encarga de brindar la información necesaria para determinar las conexiones existentes entre un transformador de baja tensión y los N usuarios vinculados a este. Las principales unidades que hacen parte de AMI-SAA son:

- ✓ PLC\_MMS
- ✓ PLC\_TU
- ✓ PLC\_MC
- ✓ Aplicación de gestión
- ✓ Terminal Móvil (TU-1/ATEMSAA)

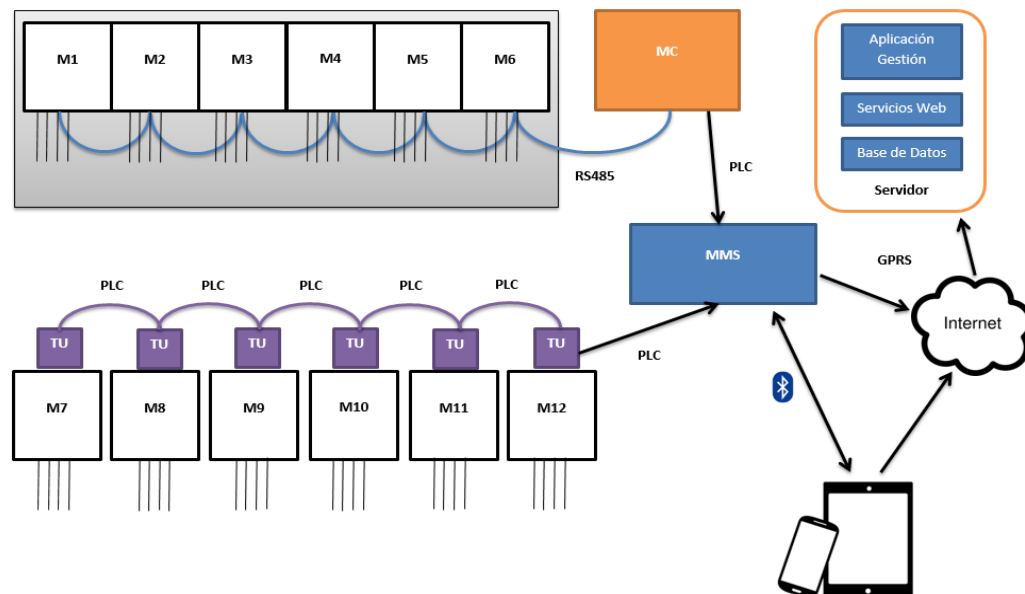


Figura 9. Arquitectura del sistema AMI-SAA.  
[Fuente: propia]



### **3.2.1. Equipos de medición**

A continuación se brinda una pequeña descripción de los diferentes dispositivos que componen el sistema AMI-SAA.

#### **3.2.1.1. PLC\_MMS**

El PLC\_MMS es un dispositivo instalado junto al macromedidor<sup>10</sup>, encargado de encuestar los nodos de usuario y entregar una vez por día la lista de PLC\_TU que enviaron su código de identificación, el valor del consumo de energía registrado por cada uno de los medidores y el valor del consumo de energía registrado por el macromedidor. Una vez procesada la información recolectada del amarre de las diferentes unidades PLC\_TU, el PLC\_MMS almacena en un archivo (.cvs) el estado de amarre de un mes que corresponde a un periodo de monitoreo. Esta información podrá ser recolectada a través de TU-1 o TU-2. Nótese que el terminal TU-1, ahora llamado ATEMSAA, se comunica con el PLC\_MMS mediante un enlace bluetooth (ver Figura 9).

#### **3.2.1.2. PLC\_TU**

El nodo PLC\_TU se encuentra instalado junto al medidor de usuario de una red de distribución, y es el responsable de responder a la encuesta de amarre del PLC\_MMS y enviar la respuesta directamente. El PLC\_TU tienen dos modos de funcionamiento, el primero es en modo directo donde recibe mensajes desde el PLC\_MMS. En el segundo modo, el PLC\_TU se comporta como unidad repetidora según órdenes del PLC\_MMS; en modo repetidor el PLC\_TU es el responsable de retransmitir información de encuesta del sistema de monitoreo.

#### **3.2.1.3. PLC\_MC**

AMI-SAA incluye la realización de medida concentrada a través de un tipo especial de PLC\_TU denominado PLC\_MC que se encargará de transmitir al PLC\_MMS el estado de amarre y el valor del consumo de energía de cada uno de los medidores que conforman el sistema de medida concentrada (El PLC\_MC está conformado por máximo 6 medidores). Dichos medidores se

---

<sup>10</sup> Equipo de medición normalmente instalado en la red de distribución de un transformador:

supondrán conectados entre sí y con el PLC\_MC mediante una red RS 485. Los PLC\_MC al igual que los PLC\_TU se comunican con el PLC\_MMS mediante la red de distribución de energía usando PLC (ver Figura 9).

#### 3.2.1.4. Aplicación de gestión

El software de gestión de balances de energía se encarga de recibir la información entregada por los diferentes PLC\_MMS. Al mismo tiempo, es el aplicativo desde el cual se pueden enviar órdenes a los PLC\_MMS para que efectúen consultas individuales sobre el estado de un medidor.

#### 3.2.1.5. Terminal Móvil (TU-1/ATEMSAA)

El terminal móvil (teléfono inteligente), puede acceder al sistema y enviar órdenes o peticiones a los dispositivos PLC para que efectúen por ejemplo, consultas individuales sobre el estado de asociación de un usuario a su respectivo transformador, entre muchas otras funcionalidades.

### 3.2.2. Tramas de comunicación

La comunicación entre el dispositivo PLC\_MMS y el dispositivo TU-1 se establece mediante la tecnología Bluetooth [43], tecnología regulada bajo el estándar 802.15.1, es una especificación industrial para Redes Inalámbricas de Área Personal (WPAN) que posibilita la transmisión de voz y datos entre diferentes dispositivos mediante un enlace por radiofrecuencia en la banda ISM de los 2,4 GHz [20].

Se establece una estructura de trama que permite la identificación de los mensajes intercambiados entre los diferentes módulos PLC y las aplicaciones con las que interactúan (aplicativo de gestión y terminal móvil). La Tabla 7 muestra la estructura básica de la trama, los campos *tipo* y *carga útil* cambian para cada mensaje [44].

Tabla 7. Estructura de la trama de comunicación para el aplicativo móvil con los dispositivos PLC.  
[Fuente: [44]]

Estructura de la trama						
Llave	Longitud trama	Tipo	Origen	Destino	Carga útil	CRC
2 bytes	1 byte	1 byte	1 byte	1 byte	1-256 bytes	1 byte

La trama está formada por las siguientes partes:

*La llave:* constituye una secuencia de bits específicos que indican cuando se inicia la trama, este es el primer campo de la trama y tiene una longitud de 2 Bytes, en código ASCII: \$@.

*Longitud de trama:* Campo que determina la longitud total de la trama. Se han destinado 8 bits para este campo.

*Tipo:* corresponde al identificador dado a cada uno de los diferentes dispositivos PLC. Tiene una longitud de 1 Byte.

*Origen:* Corresponde a la dirección física del dispositivo al que envía el mensaje, tiene una longitud de 1 Byte.

*Destino:* Corresponde a la dirección física del dispositivo al que va dirigido el mensaje, tiene una longitud de 1 Byte.

*Carga útil:* formado por las variables de medición, determina el tamaño en Bytes de la carga útil, tiene una longitud mínima de 1 Byte y se pueden tener hasta 256 Bytes de longitud para datos.

*CRC, campo detección de errores:* permite verificar si los datos que llegaron son iguales a los datos transmitidos, tiene una longitud de 1 Byte.

## CAPITULO IV

### MPlu+a: METODOLOGÍA PARA IMPLMETACIÓN DE REQUISITOS NO FUNCIONALES

#### 4.1. INTRODUCCIÓN

La especificación de requisitos del software es una descripción completa del comportamiento del sistema software a desarrollar. Incluye la descripción de todas las interacciones que se prevén que los usuarios tendrán con el software. También contiene requisitos no funcionales (o suplementarios). Los requisitos no funcionales son los requisitos que imponen restricciones al diseño o funcionamiento del sistema software (tal como requisitos de funcionamiento, estándares de calidad, o requisitos del diseño) [45].

El contexto de desarrollo de este proyecto se encuentra en el DCU dónde los aspectos de la usabilidad y de la accesibilidad del sistema a implementar tienen una destacada consideración. En este sentido se ha tomado como referencia metodológica el modelo MPlu+a, el cual se fundamenta, como se ha mencionado, en la disciplina DCU y también en la Ingeniería del Software.

MPlu+a está organizado en una serie de fases que repetidamente se van realizando, siendo el *prototipado* y la *evaluación* dos fases claves que se aplican desde el mismo instante que comienza cada nuevo desarrollo. En relación al proyecto que nos incumbe, durante esta fase inicial, que mayoritariamente se ha centrado en la definición de los requisitos tanto funcionales como de usabilidad del sistema, se ha utilizado la técnica de los escenarios como herramienta que ha permitido conocer la manera de trabajar de los usuarios y ser capaces así de incrementar el conocimiento acerca del sistema para poder desarrollarlo de manera que sea realmente útil para las personas que lo utilizarán. Utilizar los escenarios ha permitido pasar de la especificación de procesos y funcionalidades en lenguaje natural en forma de narración textual a especificaciones técnicas de flujos de información descritos con la formalidad de la notación UML [46].

A continuación se describen cada una de las fases que componen el modelo MPlu+a, además se muestra cómo estas fases fueron abordadas a lo largo del desarrollo de la aplicación ATEMSAA.

## 4.2. ANÁLISIS DE REQUISITOS

Debido a que en esta primera fase prima el usuario, sus requerimientos y el contexto en que realiza sus tareas, se deben llevar a cabo una serie de técnicas y recomendaciones para definir de una manera clara y específica todos los requisitos que se puedan generar, esto debido a que el usuario por lo general no sabe en primera instancia qué es lo que quiere. De esta manera se obtendrán resultados positivos que reducirán el grado de error de la aplicación en las posteriores fases del desarrollo.

El proceso de análisis de requisitos, está dividido en diferentes actividades.

### 4.2.1. Estudio de los implicados y de la plataforma

Los implicados son esos participantes (en el proceso de desarrollo) junto a cualquier otro individuo, grupo u organización cuyas acciones pueden influenciar o ser influenciados por el desarrollo y uso del sistema, ya sea directa o indirectamente [47].

El objetivo del análisis de los implicados es estudiar quienes serán los usuarios y el entorno de software y hardware que se va a utilizar.

✓ Audiencia: Siempre que se procede a iniciar un nuevo proyecto de alguna aplicación, se hace pensando en una audiencia determinada. El análisis de la audiencia pretende conocer quiénes son realmente estos futuros usuarios, como interactúan y cuáles son sus necesidades [16].

Se debe ser consciente de que el uso de aplicaciones móviles es cada vez más frecuente entre las personas, esto conlleva a pensar que los implicados son “tecnológicamente sabios” y, por tanto, sus necesidades están “exentas” de problemas tecnológicos, pero existe otro perfil de personas no tan familiarizado con la tecnología y demandan interfaces fáciles de usar y a la vez accesibles.

A la hora de entender a los implicados no sólo se deben tener en cuenta los atributos personales, sino también se debe analizar los distintos dispositivos y plataformas a utilizar para acceder a la información.

✓ Escenarios: El objetivo de los escenarios (o también conocidos como *Historias de Usuarios* (HU) que se establecen para alcanzar un objetivo), es asegurarse que resuelve las necesidades de personas específicas en situaciones reales y

asegura que se han considerado todos los detalles necesarios. Por tanto, los escenarios son una técnica altamente destacada relacionada con la especificación de la audiencia y sus características [16].

Se debe estudiar y documentar el conjunto de posibilidades de plataforma tecnológica a escoger para albergar el sistema, así como las restricciones tecnológicas que dicha plataforma impone. Evidentemente, ello definirá un conjunto de opciones posibles y/o imposibles (posibilidades y/o restricciones) para tenerlas en cuenta a la hora de diseñar la interfaz de usuario [48].

#### **4.2.2. Objetos**

Se considera como objeto (o artefacto) cada cosa, por innecesaria que parezca, que directa o indirectamente intervenga en el proceso de interacción entre la persona y el sistema interactivo [49]. El concepto de objeto que el modelo MPlu+a describe es de dos tipos:

- ✓ **Conceptuales:** pueden ser gestos, mensajes, passwords, firmas, etc.
- ✓ **Físicos:** Bolígrafo, papel, calculadora, reloj, etc.

El uso de objetos o artefactos constituye una valiosa fuente de información para el análisis del sistema. Permite el afloramiento de asunciones de la interacción que aun siendo relevantes para el desarrollo del sistema podrían pasar por alto y emerger en el momento más inoportuno del desarrollo [49].

Para conocer más a fondo el análisis de los implicados y de la plataforma a utilizar, remitirse al Anexo D.

Es importante tener en cuenta que, aunque los objetivos se definen en esta fase del modelo, a medida que avanza el desarrollo del sistema, pueden ir cambiando debido a las condiciones y exigencias de los usuarios.

#### **4.2.3. Objetivos**

No hay ninguna aplicación que se realice sin al menos un objetivo concreto. Una empresa, persona, grupo o institución decide desarrollar una aplicación como consecuencia de haberse marcado unos objetivos por cumplir y ven, por tanto, necesaria la implementación de la aplicación como herramienta para conseguirlo. A la hora de recoger los objetivos de una aplicación tendremos en cuenta tanto los

requisitos funcionales como los no funcionales (tiempos de respuesta, utilización de un determinado lenguaje de programación, etc.), e incluso los objetivos marcados en cuanto a la usabilidad y/o accesibilidad del sistema [50].

Observación: Todos estos objetivos no podrán ser determinados en una primera fase o visita con el cliente. Deberá ser el desarrollador quien, ayudándose de técnicas como el prototipado y la evaluación que el modelo de proceso proporciona junto con la aplicación de las actividades de protección, sea capaz de definir toda esta información lo más detallada posible [50].

#### **4.2.3.1. Objetivos funcionales**

La lista de objetivos funcionales es, en principio, la más fácil de recoger. No porque sea un tema obvio de conseguir, sino porque es el caso más habitual de cualquier aplicación interactiva: Normalmente las aplicaciones sólo responden a criterios de las funcionalidades que tienen que poder realizar [50].

Es evidente que si las aplicaciones no son capaces de realizar dichas funcionalidades de nada servirá el resto. Por más usable y accesible que sea un determinado sistema de nada servirá si no realiza las tareas que se le han encomendado. Aunque si esto sucede realmente la aplicación no puede ser usable puesto que se vio que en la definición de usabilidad está el que los usuarios puedan conseguir objetivos específicos, o sea los objetivos funcionales de dicho sistema.

La definición de los objetivos funcionales se basará en un listado de dichas funcionalidades donde cada una de ellas contará con las especificaciones necesarias que permitan al equipo desarrollador implementar los procesos necesarios para que el sistema pueda ofrecer los resultados esperados a sus usuarios.

#### **4.2.3.2. Objetivos de usabilidad [50]**

No se hablará de nuevo en este apartado acerca de los beneficios y la necesidad que aporta la usabilidad a un sistema interactivo. Tan sólo, y a modo de recapitulación, recordar que la usabilidad es vista generalmente para asegurar que los productos interactivos sean fáciles de aprender, efectivos y agradables para sus usuarios. Todo ello conlleva optimizar las interacciones que las personas

llevan a cabo con sus productos interactivos para poder conseguir realizar sus actividades en el trabajo, la escuela, y, en definitiva, en su vida cotidiana.

Algunos de los objetivos de la usabilidad son:

- **Facilidad de aprendizaje**

- ✓ Los usuarios serán capaces de usar la aplicación la primera vez sin ningún tipo de aprendizaje.
- ✓ La interfaz ha de ser simple, fácil de aprender y utilizar, con funcionalidades accesibles y bien definidas.

- **Consistencia**

- ✓ Se primará el uso de convenciones de diseño siempre que sea posible. Los usuarios se verán presionados a recordar cualquier truco especial de interacción de una visita a otra, dada la cantidad de tiempo que han dedicado a visitar otras aplicaciones. Para entonces, los usuarios habrán acumulado un modelo mental genérico de la forma en la que deben funcionar otras aplicaciones, en base a sus experiencias en los demás sitios.
- ✓ En el texto, se evitarán los fondos oscuros y los colores llamativos. También se debe evitar subrayar las palabras, porque un usuario las podría confundir con hipervínculos.

- **Flexibilidad**

- ✓ Se debe evitar la escritura cuando un botón de selección o un enlace lo pueden hacer.
- ✓ También se debe evitar que el usuario tenga que cambiar constantemente entre hacer clic y escribir.

- **Recuperabilidad**

- ✓ Hay que contemplar los errores del usuario. Debe haber una retroalimentación apropiada del sistema.
- ✓ Se crearán mensajes que reporten los errores y que permitan al usuario saber qué hacer para corregirlo.



### ▪ **Disminución de la carga cognitiva**

- ✓ El usuario debería alcanzar cualquier interfaz en el menor número posible de clics, a ser posible menos de 3.
- ✓ Aunque no importa el número de veces que hay que hacer clic en algo si la opción es mecánica e inequívoca.

### ▪ **Estética**

- ✓ Es preciso proporcionar un entorno agradable que contribuya al entendimiento por parte del usuario de la información presentada.
- ✓ Un aspecto importante a considerar, tal como se ha mencionado antes, es que suele darse el caso de que algún objetivo funcional entra en conflicto con otro de usabilidad. Esta situación se resuelve, por ejemplo, con un preciso análisis de pros y contras en una reunión con los implicados en esta situación.

#### **4.2.3.3. Objetivos de accesibilidad [50]**

La accesibilidad de cualquier sistema debe constituir un objetivo primordial que no puede descuidarse ni dejarse de la mano de la improvisación.

El aspecto realmente importante es definir los objetivos de manera clara y precisa acorde con las discapacidades que se pretenden cubrir, para lo cual se tendrán en cuenta los estándares y las normativas vigentes.

Otro aspecto a considerar son los equipos actuales y la tecnología relacionada. Ya sea por la naturaleza de la tecnología o limitaciones en los equipos, no se cuenta con la capacidad suficiente para poder satisfacer todas las necesidades. Algunos parámetros frecuentes que se han tenido presentes para proporcionar accesibilidad a la aplicación son:

- ✓ Se evitara usar demasiado color para hacer distinciones significativas entre los objetos con el fin de prevenir el cansancio visual.
- ✓ Se elegirán combinaciones de color de modo que la lectura no sea difícil y se hará uso mínimo de colores rojos y verdes para ayudar a los usuarios daltónicos (el 8% de los hombres y 0,5% de las mujeres de Europa y Norteamérica tienen estos problemas visuales).
- ✓ Se hará uso de alto contraste y fuentes altamente legibles para ayudar a los que tienen menos capacidad visual.

#### 4.2.4. Análisis de requisitos para ATEMSAA

Para el caso específico de la aplicación ATEMSAA, se debe tener en cuenta el entorno para desarrollo de aplicaciones Android, además de la arquitectura y tecnología empleada en el piloto AMI-SAA. En primer lugar se recomienda que antes de definir los objetivos del sistema, se realice un primer prototipo de baja fidelidad (Figura 10) teniendo en cuenta solo los requisitos obtenidos hasta este momento<sup>11</sup> (Ver sección 4.6). Esto con el fin de mostrar a los usuarios un panorama (a grandes rasgos) de la aplicación que se desarrollará, para especificar los objetivos de una manera menos dispendiosa.

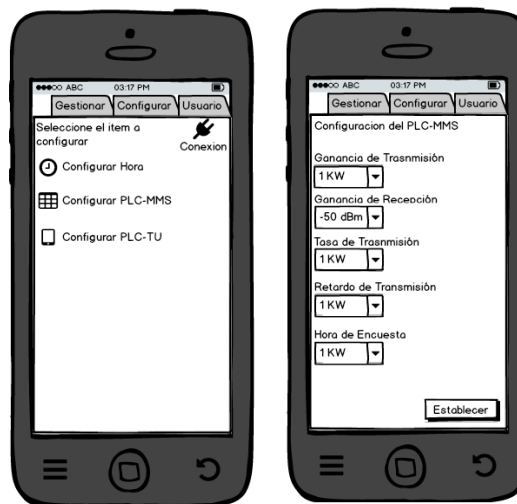


Figura 10. Prototipo no funcional (Boceto).  
[Fuente: propia].

Es importante tener presente que la capacidad de simplificar la interacción de los usuarios con el sistema es el objetivo principal de la metodología propuesta, por tanto, para desarrollar dicha capacidad es necesario conocer muy bien a los usuarios y las características que les identifican para ser capaces de trasladar dichas características al sistema y proporcionarle interfaces suficientemente usables y accesibles, y adaptadas a sus modelos mentales.

---

<sup>11</sup> Una de las ventajas de utilizar el modelo MPLu+a es su flexibilidad. Este no determina de una manera concisa en qué parte del desarrollo se tiene que aplicar una u otra fase, lo que conlleva a que por ejemplo en este caso, se pueda hacer referencia a la fase de prototipado sin ningún inconveniente, siempre teniendo en cuenta al usuario final del sistema.

En este sentido, es importante destacar aspectos muy diversos como por ejemplo el grado de conocimiento/uso de equipos/programas informáticos que tienen, la experiencia profesional, el nivel de estudios, la experiencia en el puesto o tipo de trabajo, el entorno social, etc. Esto servirá para poder tomar decisiones a la hora de diseñar la interfaz de usuario y también para identificar las categorías de usuarios.

Existen varios métodos para obtener el perfil de los usuarios, siendo los cuestionarios y las entrevistas los más utilizados y con más que probada utilidad para esta finalidad. Una vez analizados los resultados, permiten identificar patrones de usuarios (características y necesidades similares) para reflejarlo en el sistema final [51].

Para el caso de ATEMSAA, se realizaron diversos cuestionarios en los cuales se iban exponiendo los diferentes pensamientos que podrían tener los usuarios en cuanto a lo que querían de la aplicación (tanto en función como en apariencia). Un ejemplo se evidencia en la *Encuesta 1* (Anexo F, ver Figura 11), donde se realiza un primer acercamiento con el usuario.

### Encuesta 1

Encuesta dirigida a los operarios

\*Obligatorio

**Escriba su nombre y apellido \***

Tu respuesta

**1. ¿Qué funciones realiza como operario? \***

Tu respuesta

**2. ¿Qué tipo de sistema operativo usa en su celular? \***

Android

IOS

Windows

Otro: \_\_\_\_\_

**3. De acuerdo a su importancia ordene ¿Cuál es el uso que le da a las aplicaciones en su celular? Donde 1 es la mas alta y 5 la mas baja \***

	Redes sociales	Musica y videos	Juegos	Mapas y navegacion	Educacion
1	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

**4. ¿El uso de aplicaciones en su celular ha impactado de alguna manera sus hábitos? \***

Si

No

**5. De acuerdo a la respuesta anterior reponda el por qué \***

Tu respuesta

**6. ¿Qué funciones le brinda el dispositivo que pone a su disposición la empresa? \***

Tu respuesta

**7. ¿Qué funciones considera relevantes? \***

Tu respuesta

**8. ¿Qué ventajas cree que tiene el dispositivo que le brinda la empresa? \***

Tu respuesta

**9. ¿Qué desventajas cree que tiene el dispositivo que le brinda la empresa? \***

Tu respuesta

**10. ¿Cree que una aplicación móvil sería útil para realizar alguna de sus funciones? \***

Si

No

**11. ¿Considera que las aplicaciones móviles permitirían que haga más eficiente su trabajo? \***

Si

No

**12. ¿Cómo cree que es más cómodo usar el celular? \***

Horizontal

Vertical

Ambas

**13. De acuerdo a la respuesta anterior ¿Por qué es mas cómodo el uso del celular que escogió? \***

Tu respuesta

**14. Cuando inicia una aplicación móvil ¿Cuál es su metodo más común de acceso? \***

Pantalla login

Correo

Facebook

Entrada directa

[ENVIAR](#)

Nunca envíes contraseñas a través de Formularios de Google.

Figura 11. Ejemplo de cuestionario para acercamiento con el usuario de la aplicación.  
[Fuente: propia]

### 4.3. DISEÑO

Una vez finalizada la fase de análisis de requisitos se procede a realizar la fase del diseño de la interacción. Esta fase se realiza en repetidas ocasiones tomando en cuenta los requisitos que se obtuvieron en la fase previa.

Durante el desarrollo de un sistema, nuevo o no, una vez resueltas las funcionalidades que debe cubrir junto con el resto de características derivadas del contexto de la interacción se pasa, en esta fase, a realizar el *diseño de la actividad* y el *diseño de la información* como principales actividades que conforman el proceso global de diseño de la interacción [52].

El análisis de estos, proporcionará al equipo desarrollador la información necesaria para llevar a cabo un buen modelo del sistema para su posterior codificación.

#### **4.3.1. Diseño de la actividad**

Directamente relacionado con la especificación funcional (funciones que debe cumplir el sistema). Se consigue analizando todas las funcionalidades y las tareas necesarias que permiten realizarlas (análisis de las tareas) y el modelado a nivel conceptual buscando una aproximación al modelo mental de los usuarios [2] [52].

Lo que se busca con esta parte del diseño es que se defina de una manera clara un modelo de las tareas que se asemeje en gran medida al modelo mental del usuario. En el Anexo D del documento se realiza una ampliación de conceptos como modelo mental del usuario, modelo conceptual y análisis de tareas.

##### **4.3.1.1. Métodos para el análisis de tareas [2]**

Los métodos para llevar a cabo el análisis de las tareas se agrupan en función de sus características:

- **Métodos cognitivos:**

- ✓ Identifican secuencias de comportamiento correctas.
- ✓ Representan el tipo de conocimiento que debe tener el usuario acerca del uso del sistema.
- ✓ Generan una especificación del usuario a partir de la descripción de tareas.

- **Métodos predictivos:**

- ✓ Evalúan el rendimiento humano.
- ✓ Describen secuencias de comportamiento.
- ✓ Análisis centrado en rutinas de comportamiento.

▪ **Modelos descriptivos:**

Permiten obtener una descripción más o menos completa del sistema a partir de la información obtenida de las tareas.

En la Tabla 8 se muestran algunos métodos utilizados con sus principales características:

Tabla 8. Métodos utilizados para el análisis de tareas (algunos).

[Fuente: [2]]

<b>Método</b>	<b>Tipo</b>	<b>Notación</b>	<b>Especificación</b>	<b>Comentarios</b>
HU (Historias de Usuario)	Descriptivo	Textual	Semi-Infomal	Una manera simple de describir una tarea concisa que aporta valor
HTA	Cognitivo	Gráfico	Semi-Infomal	Modelo de descomposición del conocimiento
GOMS	Cognitivo	Textual	Semi-Infomal	Familia de lenguajes para describir el conocimiento
UAN	Cognitivo	Gráfico	Semi-Infomal	Notación para el estilo de manipulación directa
KLM	Predictivo	Textual	Tiempo	Medición del rendimiento humano
TAG	Predictivo	Textual	Esquemas	Medida de la consistencia
CTT	Descriptivo	Gráfico	Lógica temporal	Herramientas de soporte al análisis y verificación.

**4.3.2. Diseño de la información**

El diseño de la información, cuyo principal objetivo es dar soporte a la percepción, la interpretación y la comprensión de la información de los sistemas interactivos, afronta aspectos relacionados con la parte física de la interacción (colores, organización de los elementos...), el lenguaje (visual para las interfaces visuales,

auditivo para las auditivas...), los modelos de la información y la consistencia y la coherencia [2]. No sólo se refiere a qué colores o fuentes tipográficas utilizar, sino a la disposición de los elementos interactivos en la interfaz de acuerdo con las tareas a realizar y en compromiso con los objetivos de usabilidad y a la consecución funcional de dichas tareas. En definitiva, el diálogo con el usuario.

Un aspecto importante en esta fase relaciona los factores humanos con la capacidad de los elementos de la interfaz de transmitir la sensación de poderse utilizar adecuadamente [52].

### **4.3.3. El diseño aplicado en ATEMSAA**

Para el desarrollo de esta aplicación, se recurrió el método de *Historias de Usuario*, el cual es usado en metodologías de desarrollo ágil, en este caso SCRUM. Para mayor información acerca de este método, se puede consultar la Sección 2.3.3

#### **4.3.3.1. Estilo. Estrategias de diseño de la información**

En este apartado es donde se plasma la relación entre las recomendaciones que existen actualmente para los patrones de diseño del sistema operativo Android y las estrategias de diseño del MPlu+a.

Se debe partir de dos aclaraciones que hace el modelo:

- ✓ Suele confundirse el estilo de una interfaz con el diseño gráfico de la misma; el concepto de interfaz es mucho más amplio, y su diseño gráfico es tan solo una parte de la misma.
- ✓ No se debe confundir el diseño de la interfaz de usuario con la propia usabilidad: el estilo de la interfaz es tan solo una de las características de la usabilidad.

#### **▪ Interfaz gráfica de una aplicación móvil**

La Norma UNE – EN ISO 9241, en su parte 10: principios de dialogo, presenta una serie de ideas que pretenden ser una guía para la elaboración de interfaces graficas en equipos con pantallas de visualización de datos (PVD) [2] [16].

- ✓ La aplicación debe estar adaptada a la tarea para la cual se ha diseñado; el dialogo con el usuario debe ser limpio, presentando y exigiendo solamente la información estrictamente necesaria.
- ✓ La aplicación debe informar el progreso al interlocutor de forma comprensible para éste (auto-descriptividad).
- ✓ La aplicación debe poder adaptarse al nivel de capacitación del usuario.
- ✓ La aplicación debe ser controlable por el usuario, no al revés.
- ✓ Las respuestas de interfaces deben ser coherentes y adaptadas al nivel de capacitación del usuario.
- ✓ Debería ser clara y sencilla de utilizar.

Todas estas recomendaciones se elaboran siempre pensando en el usuario, enfocándose en satisfacer sus necesidades, y pensando en la usabilidad de la interfaz desarrollada.

#### **4.3.3.2. Ventanas informativas**

En toda interfaz gráfica los *cuadros de diálogo* o *ventanas informativas* son una gran herramienta para desplegar mensajes importantes. Los diálogos en una aplicación son utilizados con gran frecuencia debido a que facilitan la comunicación con el usuario. Estos diálogos son por lo general ventanas emergentes en las cuales se solicitan datos o respuestas al usuario [2].

Entre las sugerencias más importantes para el manejo de diálogos se tienen:

- ✓ Los diálogos se deben mostrar al operador como consecuencia de una acción de él.
- ✓ Los diálogos deben ser concisos y claros.
- ✓ Los diálogos se deben situar en una parte de la pantalla donde no interfieran con los datos del proceso.
- ✓ Solamente cuando el dialogo es de carácter critico se debe mostrar en el centro de la interfaz.
- ✓ Los diálogos de confirmación de las acciones del operador se deben mostrar solamente cuando se trate de operaciones críticas y/o irreversibles.

##### **▪ Diálogos en Android**

En el sistema operativo Android estos cuadros de diálogos son llamados *diálogos* (una pequeña ventana) cuyo objetivo es solicitar al usuario que tome una decisión



o introduzca información adicional. Estos diálogos no llenan la pantalla y se utilizan normalmente para los eventos modales<sup>12</sup>, aunque su diseño también puede llegar a ser complejos layouts si el desarrollo de la aplicación lo amerita. En la Figura 12 se puede observar la estructura general de un diálogo.

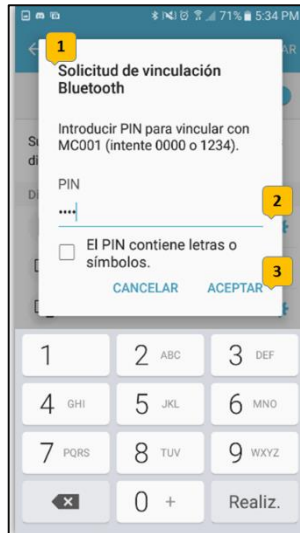


Figura 12. Estructura general de un diálogo en Android.  
[Fuente: propia]

En la región 1 se indica el título del diálogo, este puede ser, por ejemplo, algún ajuste que el usuario desea cambiar, o algún requerimiento en el cual se debe tomar una decisión.

La región 2 indica el área de contenido, los contenidos varían ampliamente entre aplicaciones, se pueden utilizar campos de texto, sliders, Spinners, CheckBoxes, o botones de opciones que permiten al usuario cambiar de aplicaciones o configurar el sistema en algún aspecto. En otros casos, como en las alertas, el contenido consiste en un texto informativo para que el usuario tome una decisión.

La región 3 hace referencia a los action Buttons (botones de acción). Estos son típicamente Cancel (Cancelar) u Ok (Aceptar). Sin embargo si las opciones

---

<sup>12</sup> Se habla de una ventana modal si sólo se puede acceder a la ventana que se muestra, y no al resto de las ventanas de la aplicación [70]. En otras palabras, son las cajas que hasta que no se cierran no dejan continuar con la aplicación.

consisten en acciones específicas, como continuar, esperar confirmación, anular, editar, descargar etc., entonces se deben utilizar verbos activos.

Se deben considerar las siguientes reglas en las acciones de los botones.

- ✓ La acción despectiva de un dialogo siempre está a la izquierda, estas acciones devuelven al usuario a su estado anterior.
- ✓ Las acciones afirmativas se encuentran a la derecha, estas acciones siguen el progreso que el usuario desea para cumplir su objetivo.

En la Figura 13 se muestran algunos ejemplos de cuadros de diálogo comunes.

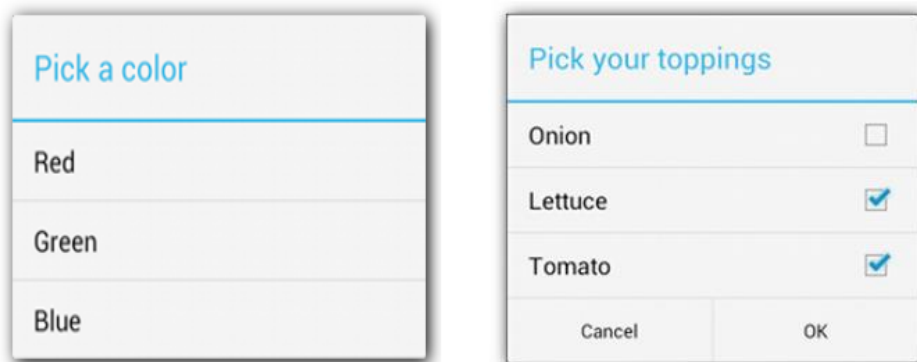


Figura 13. Ejemplos de diálogos utilizados en el sistema operativo Android.  
[Fuente: [53]]

Las alertas informan al usuario acerca de una situación que requiere confirmación por parte de él antes de continuar. Difieren ligeramente en apariencia basada en la gravedad y el impacto del mensaje transmitido.

Se pueden diseñar **alertas sin título** (éstas son muy comunes), por lo general la decisión no tiene un impacto grave y puede resumirse en una o dos frases. Ver Figura 14.

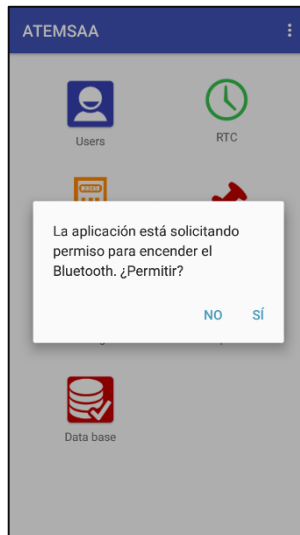


Figura 14. Alerta sin título  
[Fuente: propia]

Las **alertas con título** se deben utilizar con moderación. Son apropiadas solo cuando una operación es de alto riesgo e implica la pérdida potencial de datos, conectividad, cargos adicionales, etc., requiere una pregunta clara o declaración (el título) y alguna explicación adicional (en el área de contenido).

Se debe mantener la pregunta o afirmación lo más corta posible: por ejemplo, "Usar el servicio de localización de Google?" (Ver Figura 15), evitar las disculpas. Los usuarios deben poder comprender las opciones basadas solo en el título y el botón.

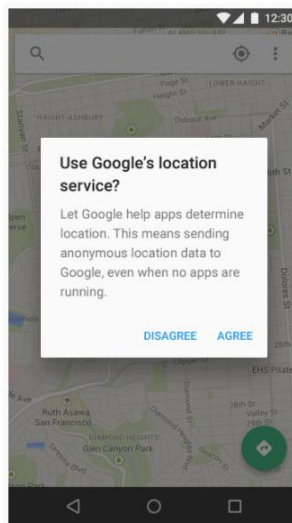


Figura 15. Alerta con título.  
[Fuente: [54]]

## ▪ Toast

Un Toast es una vista que contiene un pequeño mensaje rápido para el usuario. Cuando se muestra la vista al usuario, aparece como una vista flotante sobre la aplicación. Nunca recibirá foco. El usuario probablemente estará en medio de escribir algo más. La idea es ser tan discreto como sea posible, mientras sigue mostrando al usuario la información que desea que vea [55]. Aunque son personalizables, aparecen por defecto en la parte inferior de la pantalla, sobre un rectángulo gris ligeramente translúcido. Por sus propias características, este tipo de notificaciones no deberían utilizarse para hacer notificaciones demasiado importantes al no requerir confirmación por parte del usuario [56].

Por ejemplo, en la aplicación ATEMSAA no se puede acceder a las opciones que brinda la interfaz de inicio, a menos que el usuario se conecte con algún dispositivo PLC (ver recuadro rojo en la Figura 16). Si no se está conectado, el sistema envía un mensaje toast en el cual le solicita que realice dicha conexión.

Siempre que se desarrolle cualquier dialogo es importante que se notifique la acción que se ha realizado por medio de un Toast.

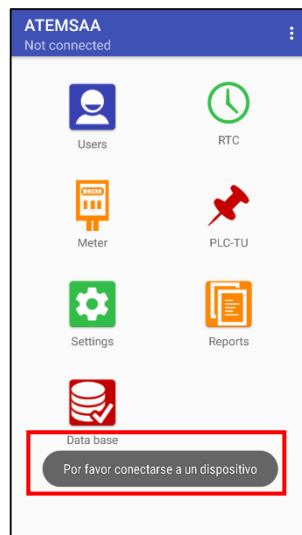


Figura 16. Mensaje Toast en ATEMSAA para conexión con dispositivo PLC (en interfaz de Inicio).  
[Fuente: propia]

Los patrones de Android anteriormente citados se pueden integrar con una serie de principios que se deben tomar en cuenta a la hora de diseñar los cuadros de diálogo de la interfaz gráfica de la aplicación a desarrollar:

- ✓ Tener un aspecto coherente, que responda a las expectativas del usuario y agilice las respuestas (ver Figura 17).

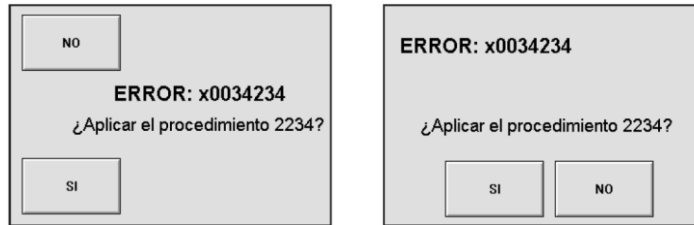


Figura 17. Aspecto coherente en los diálogos.  
[Fuente: [2]]

En la Figura 17 (izquierda) se aprecia una distribución de los botones inusual en el cuadro de mensaje, lo cual hace que este sea casi incomprensible, debido a que se acostumbra a leer de arriba hacia abajo y de izquierda a derecha. En la Figura 17 (derecha) se aprecia una mejor distribución, lo que generalmente se aprecia en cualquier interfaz.

- ✓ Dar indicaciones con claridad, para proporcionar una asimilación correcta.
- ✓ Ser concisa, evitar informaciones superfluas. Si se evitan las explicaciones innecesarias, la información será más clara. Ver Figura 19.

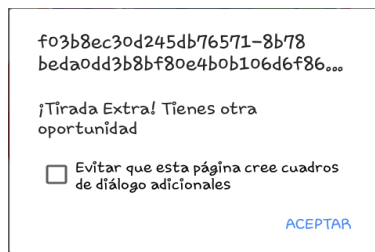


Figura 18. Información poco clara.  
[Fuente: <http://i65.tinypic.com/15nkfia.png>]

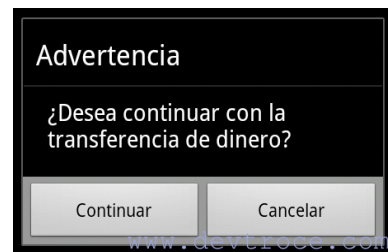


Figura 19. Información clara y concisa.  
[Fuente: <http://www.devtroce.com/wp-content/uploads/2012/01/sino.png>]

La Figura 18 se puede presentar durante algún procedimiento, se observa que ha pasado algo, pero como consecuencia de que la información no es lo suficientemente clara, cualquier usuario no se atrevería a realizar acción alguna.

En la Figura 19 se muestra información clara y concisa, apropiada para que el usuario tome la mejor decisión.

Es importante tener en cuenta que cualquiera sea la ventana informativa que se use, debe tener un aspecto coherente, que responda a las expectativas del usuario, agilice las respuestas y brinde una información clara.

#### 4.3.3.3. Elementos gráficos

Para diseñar aplicaciones en el sistema operativo Android se cuenta con una serie de parámetros que son de gran importancia, a continuación se presentan los más enfocados hacia el sistema operativo.

- **Color y forma como fuentes de información**

El color es un recurso vital en el diseño de una aplicación. Su uso abarca encabezados, textos, botones, fondos y muchos otros elementos que conforman la interfaz. En algunas ocasiones, está asociado a la identidad (color corporativo), y en otras, responde a criterios estéticos y decisiones de diseño. Un color por sí solo, salvo en el caso de colores reservados, no indica mucho. Como parte de un sistema cromático, el uso consistente, consciente y vinculado al contexto donde se aplica, es lo que lo llena de significado para el usuario [17].

Los colores sobre las pantallas y los elementos son importantes, se deben establecer los colores adecuados dependiendo del contexto y de los usuarios a los cuales va dirigido. La buena selección de la combinación de colores ayuda a disponer de una interfaz de usuario agradable. Así mismo es aconsejable que todas las pantallas de una misma categoría tengan el mismo color de fondo, llegando incluso a crear una respuesta automática al usuario [57].

El sistema operativo Android permite el uso de todos los colores del espectro, sin embargo es importante no excederse en la utilización de los mismos. Se debe recordar que los colores en Android se manejan de una forma hexadecimal, partiendo de los colores básicos es sus tonalidades más fuertes, Rojo (#FF0000), Verde (#00FF00) y azul (#0000FF), hasta llegar a la ausencia de los tres colores: el negro (#000000), o la suma de los tres: el blanco (#FFFFFF). Ver Figura 20.

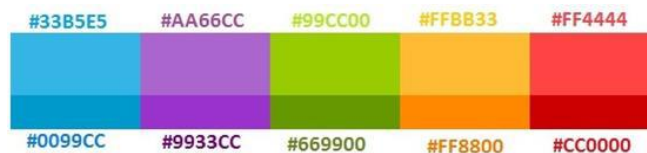


Figura 20. Colores de referencia en Android.  
[Fuente: [54]]

## ▪ Fondo de Pantalla

En el caso del color de fondo, este debe estar en consonancia con el elegido para la tipografía, ya que es necesario un contraste mínimo por cuestiones de legibilidad y accesibilidad. Hay que tener en cuenta que los fondos oscuros suelen cansar la vista más rápidamente, por lo tanto, si la aplicación es de uso frecuente o requiere pasar cierto tiempo leyendo, es conveniente revisar la elección cromática y llevar el color de fondo hacia alternativas más claras. Sin embargo, los colores oscuros en el fondo sí pueden ser una buena alternativa cuando el contenido sea muy visual, como fotografías o videos, ya que ayuda a resaltar estos elementos [17].

Debido a esto, en el caso de ATEMSAA, las grandes áreas de pantalla vacías se rellenaron con un color neutro para no forzar la vista con contrastes excesivos, ver Figura 21.

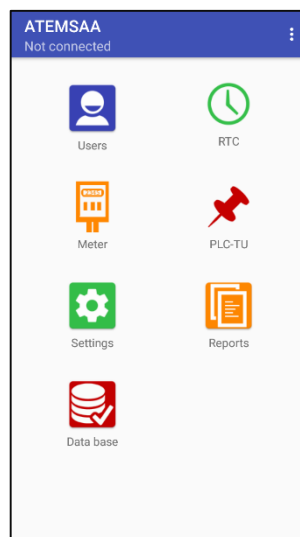


Figura 21. Las grandes áreas vacías tienen un color neutro: blanco.  
[Fuente: propia]

Algunas directrices adicionales que hay que tener en cuenta son las siguientes:

- ✓ Los colores de fondo deben ser contrastantes con los demás elementos.
- ✓ El uso de diferentes colores de fondo puede ser utilizado para diferenciar o agrupar procesos que realice la aplicación.
- ✓ Evitar el uso de colores primarios o fuertes en zonas grandes de la pantalla.

## ▪ Ubicación de Elementos

En [58] se describen una serie de directrices para realizar la distribución de los elementos en la pantalla.

- ✓ Considerar el diagrama de Gutenberg.
- ✓ La información importante debe ir arriba.
- ✓ El centro de la pantalla es un lugar de alta visibilidad.
- ✓ La información miscelánea (Escudos, logotipos, etc.) debe ir en la parte inferior izquierda.
- ✓ Funciones e información crítica deben tener un lugar fijo en la pantalla.
- ✓ La mejor posición para los gráficos es a la izquierda del campo visual.
- ✓ Se debe establecer una estructura de rejilla (grid) regular.
- ✓ Al desarrollar los prototipos de los sinópticos de procesos se debe controlar la densidad de los gráficos, la cual no debe sobrepasar el 50%, esto con el fin que no tome una apariencia aglutinada. (importante en dispositivos móviles).
- ✓ Se debe dar preferencia a las distribuciones simples sobre las complejas.
- ✓ Las anteriores recomendaciones se pueden tomar tal cual para el desarrollo de la interfaz industrial en el dispositivo móvil a utilizar. En este se debe siempre tener en cuenta la manipulación por parte del usuario. Es importante que el diseño del layout satisfaga la postura con la que se maneja el dispositivo, ya sea un Smartphone (su manipulación por lo general se realiza con una mano), o una Tablet (Se utilizan las dos manos).

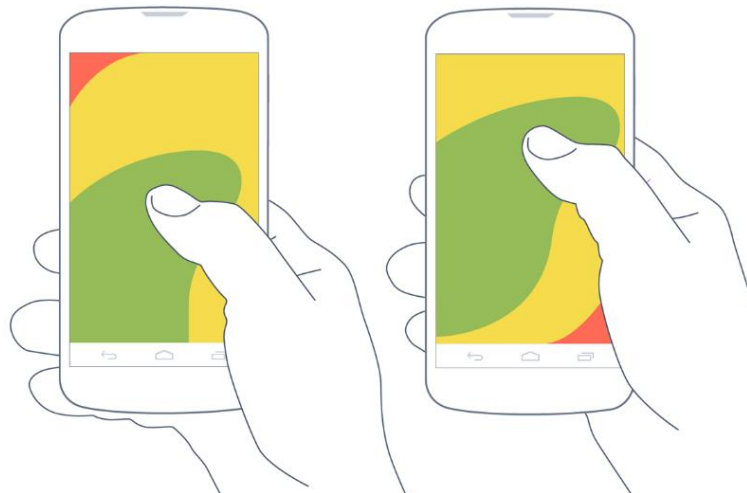


Figura 22. De acuerdo a la forma de sostener el móvil, el pulgar tiene mayor o menos dificultad en acceder a ciertas zonas.

[Fuente: [17]]



En la Figura 22 se puede observar que los campos de acción deben estar situados en la parte inferior del dispositivo, esto debido a las posturas que toma el usuario para su manipulación. Es así que, por ejemplo, los submenús para ir a alarmas o a las gráficas del proceso deben ir ubicados en este lugar, ya que el usuario podrá acceder a ellos rápidamente, conservando así, la ergonomía.

En primera instancia de debe tener en cuenta la composición para distribuir ordenadamente todos los elementos de diseño en una interfaz. En este caso, el diagrama de Gutenberg describe el patrón general seguido por la vista cuando observamos una información homogénea distribuida de manera regular [59].

Si se quiere utilizar este patrón, se debe saber que el Diagrama de Gutenberg divide el medio expositivo en cuatro cuadrantes y cada cuadrante representa una zona óptica [59] (Figura 23):

1. Zona primaria: Situada en el margen superior izquierdo
2. Zona de barbecho fuerte: Situada en la parte superior derecha
3. Zona de barbecho débil: Situada en la parte inferior izquierda
4. Zona terminal: Situada en la parte inferior derecha

En países occidentales se suele utilizar más el Diagrama de Gutenberg, ya que los habitantes suelen leer en función a este patrón, en base a la práctica de leer de izquierda a derecha y de arriba hacia abajo.

Para este patrón de diseño las zonas que presentan menor atención son la esquina superior derecha e inferior izquierda, y en caso de querer atraer atención a esas partes es necesario incluir un elemento llamativo que enfatice de alguna manera la zona. Es por esto que el contenido relevante debe seguir la diagonal que va de la esquina superior izquierda a la inferior derecha [2].

Con lo anterior se puede tener una idea clara de cómo se pueden distribuir las zonas de influencia en la pantalla. Los elementos importantes, tales como alarmas o estados operativos, deben gozar de posiciones privilegiadas en pantalla, en los márgenes superior o inferior preferiblemente.



Figura 23. Diagrama de Gutenberg.

[Fuente: <http://enrique-mingote.blogspot.com.co/2011/06/diagrama-de-gutenberg.html>]

Para el desarrollo de ATEMSAA, se tiene en cuenta la distribución de pantalla en los dispositivos móviles (Figura 24) propuesta por Dueñas y Salazar [2] para la orientación portrait.

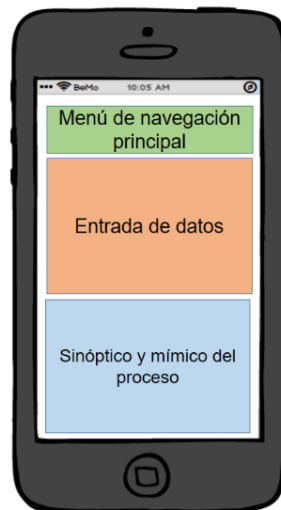


Figura 24. Distribución de pantalla para dispositivo móvil en orientación portrait.

[Fuente: propia]

Se observa un espacio amplio destinado para la entrada de comandos y datos por parte del operario, para que el sistema cumpla los objetivos establecidos. Las características principales que deben tener los comandos son su visibilidad y su facilidad de operación, para esto es imprescindible que su área de acción en pantalla sea de buen tamaño, perfectamente etiquetada y por ello reconocible fácilmente por el usuario.

Las directrices para especificar los comandos del operador se mencionan a continuación:

- ✓ Los comandos deben ser claramente visibles para el operador.
- ✓ Los comandos deben ser claramente etiquetados.
- ✓ El área de acción sobre el comando deber ser suficientemente grande como para que sea fácilmente operado.
- ✓ El operador debe ser realimentado inmediatamente del resultado de su acción.
- ✓ Los comandos que activan una acción crítica o de riesgo deben estar claramente etiquetadas y no deben estar cerca de comandos de uso frecuente.

En segundo lugar se observa el menú de navegación principal, se debe prestar atención a que los botones ubicados en esta zona brinden al operador la posibilidad de hacer clic fácilmente, esto es, que el área de contacto sea lo suficientemente grande para que el usuario no tenga dificultades al utilizarlos [58].

Por último, en el área destinada a desarrollar los prototipos de los sinópticos de proceso, se mostrarán las respuestas de la aplicación a los menús ubicados en la zona del menú de navegación principal. Esta respuesta debe ser clara y debe ocupar un espacio lo suficientemente adecuado para que no se vean muy aglutinados.

#### ▪ Action Bar

El *Action Bar*, también conocido como barra de acciones, es uno de los elementos de diseño más importantes en las actividades de una aplicación, proporciona una estructura visual y elementos interactivos conocidos por los usuarios [60]. Permite el acceso a acciones importantes de forma predecible; por ejemplo, la búsqueda, también la compatibilidad con navegación y cambios de vistas (con pestañas o listas desplegables). Por lo general se organiza como se muestra en la Figura 25 [61].

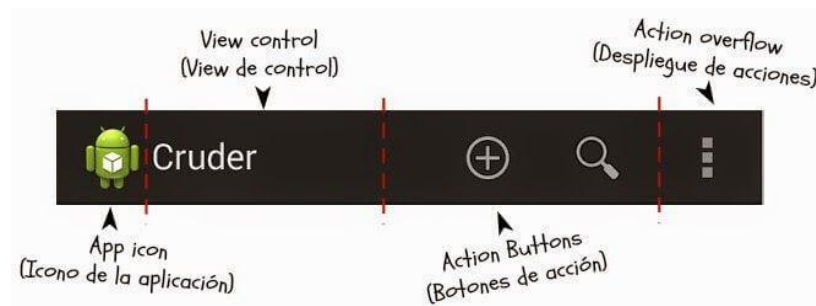


Figura 25. Anatomía del Action Bar.

[Fuente: <http://www.hermosaprogramacion.com/2014/09/android-action-bar/>]

1. **Icono de la aplicación:** en esta ubicación se proyecta el recurso drawable del icono de la aplicación. Por defecto se usa un icono predeterminado por los recursos de Android.
2. **View de control:** Este espacio está diseñado para insertar views que permitan acceder al contenido de la actividad con el fin de mejorar la navegación. Por lo general se usan Spinners, TextViews, SearchViews, etc., para controlar el contenido. Si no es necesario, simplemente se coloca el título de la aplicación o la información que se desee.
3. **Botones de acción:** Representan las acciones más populares dentro de la aplicación, las cuales se pueden ejecutar rápidamente al presionarlos.
4. **Despliegue de acciones:** Este segmento contiene una lista de acciones que no son tan populares, pero pueden ser necesitadas en algún momento por el usuario para tener acceso de forma sencilla

Un aspecto importante al momento de desarrollar la aplicación, y siguiendo los parámetros de diseño para tratar de ahorrar el mayor espacio posible en pantalla, es el de trabajar con pestañas para la navegación a través de la aplicación; se debe tener presente que estas pestañas aparecen en una barra separada, conocida como *Stacked Action Bar* (Barra de acción apilada), cuando la aplicación se encuentra en orientación portrait [60].

En un principio, se tenía la interfaz principal de la aplicación como se muestra en la Figura 26 (izquierda), si se seleccionaba la opción *Settings* (Configuración), aparecía la interfaz como se observa en la Figura 26 (derecha); en esta segunda interfaz se evidencia el uso de la barra de acción apilada, se podía hacer la Configuración de cualquiera de los dispositivos PLC y minimizaba el uso de interfaces para esto.

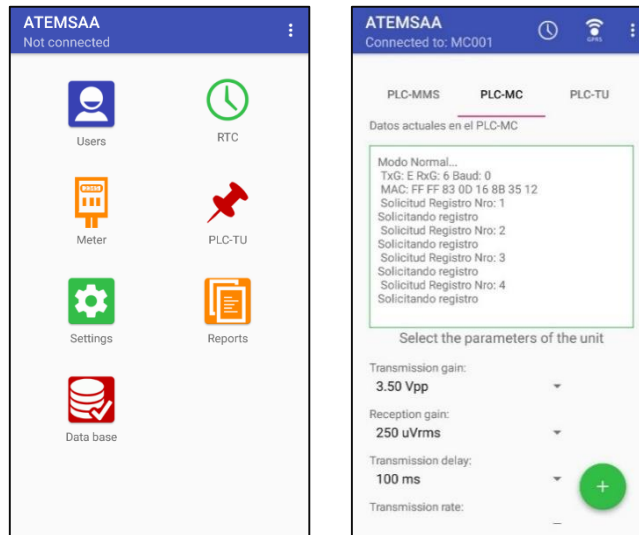


Figura 26. Primera organización de pestañas utilizando Stacked Action Bar.  
[Fuente: propia]

A través de una de las reuniones realizadas con el Product Owner se estableció que la aplicación solo estaría diseñada para la orientación portrait [Anexo F, Encuesta 1]; con esto y teniendo en cuenta las funcionalidades de los dispositivos PLC, se hizo necesario la creación de interfaces independientes por dispositivo; en dichas interfaces, el Stacked Action Bar está conformado por las funcionalidades que tiene el dispositivo. Un ejemplo de esto se evidencia en la Figura 27, en donde se observan las funcionalidades que tiene el dispositivo PLC\_MMS.



Figura 27. Stacked action bar para dispositivo PLC\_MMS.  
[Fuente: propia]

Las pestañas -o tabs- se pueden utilizar para filtrar contenidos o cambiar entre pantallas que, de acuerdo a la arquitectura de información, tienen el mismo nivel de jerarquía, indicando siempre dónde se está y hacia dónde más se puede ir.

Las buenas prácticas indican que es necesario destacar siempre la pestaña seleccionada, mantener el orden y la ubicación inicial -que no cambien de pantalla en pantalla- y no usarlas para incluir otras acciones distintas a la navegación [17].

#### ▪ **Entrada de datos**

De acuerdo a los objetivos del usuario, este le suministra datos al sistema, en general las operaciones que se realizan son: ingresar datos, realizar consultas y visualización de respuestas.

Las principales características de los comandos son su visibilidad y su facilidad de operación. Para cumplir con estos dos parámetros es imprescindible que su área de acción en la pantalla sea de buen tamaño, con sus respectivas etiquetas, y por tanto fácilmente reconocida por el usuario [2]. Android cuenta con *TextFields* (campos de texto) los cuales permiten al usuario escribir cualquier texto en la aplicación. Estos pueden ser de una o varias líneas. Entre las propiedades más importantes de este elemento se encuentran:

- ✓ Si se toca cualquier área del con el dedo, se muestra automáticamente el teclado (Figura 28).
- ✓ Con la propiedad *Hint* se puede mostrar un texto informativo, el cual desaparece cuando se escribe la información requerida.
- ✓ Los campos de texto se pueden personalizar como se desee (fondo, color, tamaño, bordes, etc.).
- ✓ Además de escribir, los campos de texto permiten otras actividades como la selección de texto, cortar, copiar y pegar.
- ✓ Tiene la propiedad de autocompletar.

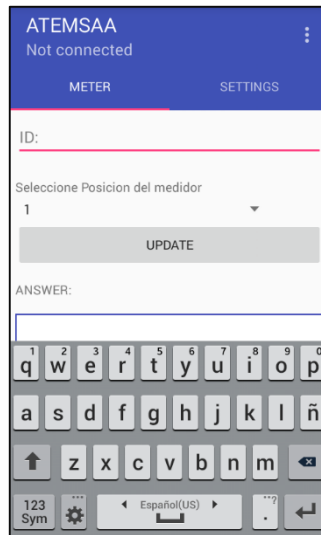


Figura 28. Al contacto con el *TextFiel* el teclado aparece automáticamente. Con la propiedad *Hint* se indica la información que debe ingresarse (en este caso el ID del dispositivo PLC).  
[Fuente: propia]

Otro aspecto importante en los comandos y en la entrada de datos es el de la retroalimentación, lo que implica que el usuario debe recibir una respuesta del sistema ya sea positiva o negativa, inmediatamente después de que ha efectuado una acción. En cuanto a su ubicación, pueden ser localizados junto con los sinópticos siempre y cuando su uso sea muy frecuente, de tal manera que el usuario no tenga que estar cambiando de pantalla a cada momento [2].

Para la entrada de datos se deben tener en cuenta las siguientes directrices:

- ✓ La entrada de datos debe ser visible, identificable y de tamaño adecuado.
- ✓ Se debe confirmar al operador que el dato ha sido aceptado por la interfaz.

En el caso de ATEMSAA, para confirmar o identificar algún inconveniente con la entrada de datos, se hace uso de la herramienta *Toast*, que de manera simple, corta y específica, le describe al usuario cual es el problema (ver Figura 29)

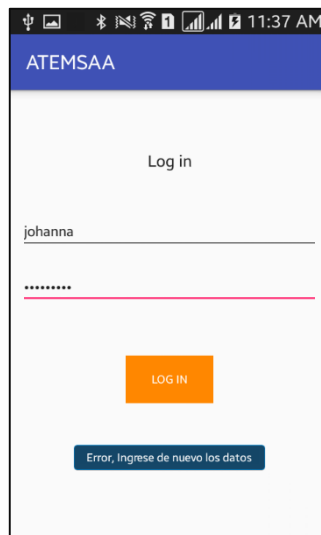


Figura 29. Mensaje de error al ingresar el usuario/contraseña de manera errada.  
[Fuente: propia]

#### ▪ Información textual [58]

La información del proceso es presentada al usuario por medio de varios elementos de los cuales el más comúnmente usado es el texto. Las características del texto que se deben definir para este fin son las siguientes: el uso de fuentes, el tamaño del texto, la alineación, el espaciamiento, los acrónimos y las abreviaturas.

Algunas de las directrices más importantes para la definición de las fuentes son las siguientes:

- ✓ No se deben utilizar más de tres fuentes en la interfaz.
- ✓ El tamaño de la letra debe ser aquel que permita una lectura cómoda desde la distancia para el usuario.
- ✓ El tipo de fuente debe facilitar la lectura, por tal es recomendable utilizar fuentes tipo sans–serif que son aquellas sin pequeños remates en sus extremos, esto facilita su lectura en pantallas, algunos ejemplos de este tipo de letra son: Arial, Arial Narrow, Tahoma y Verdana. Si se habla estrictamente de Android también se podría utilizar el tipo de letra Roboto, que es muy común en muchas aplicaciones.
- ✓ No utilizar énfasis en el texto (subrayado, itálico, sombreado salvo en casos muy especiales).
- ✓ El color del texto debe proporcionar un contraste adecuado con el texto. La combinación de colores con bajo contraste dificulta la lectura de lo escrito, entre la mejor relación de colores se encuentran: negro sobre blanco, negro sobre un



tono de gris claro, negro sobre amarillo. En general es aconsejable usar en el fondo colores claros y suaves y siempre un color de texto oscuro [2].

- ✓ Evitar al máximo el uso de letras mayúsculas en todas las letras del texto, ya que su uso exagerado causa estrés visual.
- ✓ Cuando se presenten grandes cantidades de información, se debe aumentar la separación entre líneas o insertar líneas en blanco para mejorar la legibilidad.

En Android, el tamaño tipográfico se mide en sp (scaledpixels o píxeles escalados), una forma de modificar la escala de las fuentes de acuerdo al tamaño de pantalla y a las preferencias definidas por el usuario en su configuración del teléfono. Los tamaños más comunes van desde 12sp hasta 22sp (Ver Figura 30).

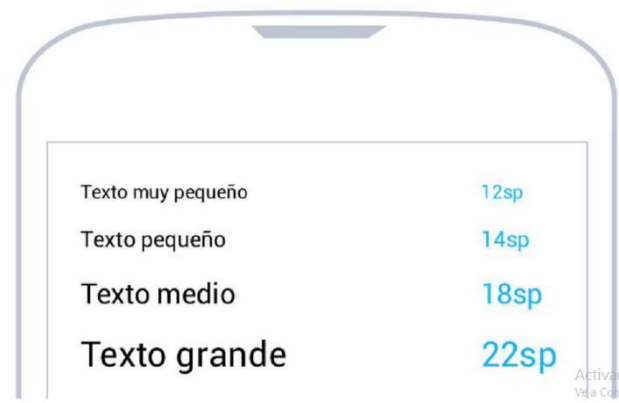


Figura 30. Diferentes tamaños de fuentes en Android de acuerdo al uso y jerarquía de los elementos.

[Fuente: [17]]

### ▪ Cuadro sinóptico

En este espacio se representa las respuestas que se obtienen de cualquiera de los dispositivos PLC a los que ATEMSAA se encuentre vinculada. Es recomendable que el espacio sea claramente diferenciado para que el usuario pueda interpretar sin dificultad la respuesta que está recibiendo del sistema.

Los elementos en pantalla se pueden representar como se deseen, sin embargo, en el caso de ATEMSAA, la respuesta que se obtiene viene predeterminada.

Ya sea para realizar consulta de parámetros o definir una nueva configuración para los dispositivos PLC, realizar descarga de reportes u otras tareas; la aplicación envía una trama con un protocolo de comunicación que el dispositivo PLC reconoce; al recibir la trama, el dispositivo envía la respuesta. Esta respuesta es la que se observa en el cuadro sinóptico.

En la Figura 31 se observa la respuesta que obtuvo ATEMSAA para el caso de “Agregar usuario”.

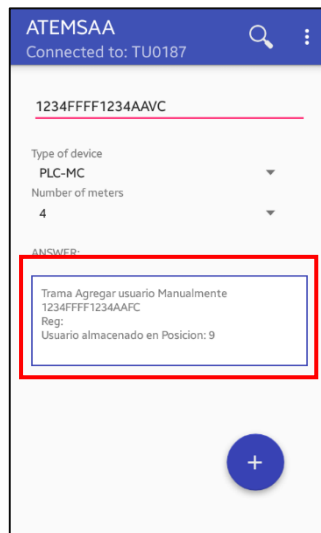


Figura 31. Ejemplo de cuadro sinóptico en ATEMSAA.  
[Fuente: propia]

#### ▪ Pantallas vacías [17]

Los diseños tienen que considerar no solo los escenarios ideales, como una lista que ya está completa o una pantalla llena. Es necesario plantear también el diseño para cuando la aplicación comienza a ser utilizada y aún no se dispone de cierta información. Por ejemplo, cómo se vería una pantalla vacía, sin ítems, cuando todavía no hay nada para mostrar. Un ejemplo de ellos se evidencia en el diseño de la interfaz para el dispositivo PLC\_MC; en esta interfaz, ATEMSAA no tiene datos en el cuadro sinóptico, éste se va a llenar una vez que se realice alguna acción con el dispositivo (leer energía, activar/desactivar relé), debido a esto, se optó por enmarcar con color azul el cuadro sinóptico con el fin de que el usuario supiera que es un espacio que puede ser llenado (ver Figura 32).

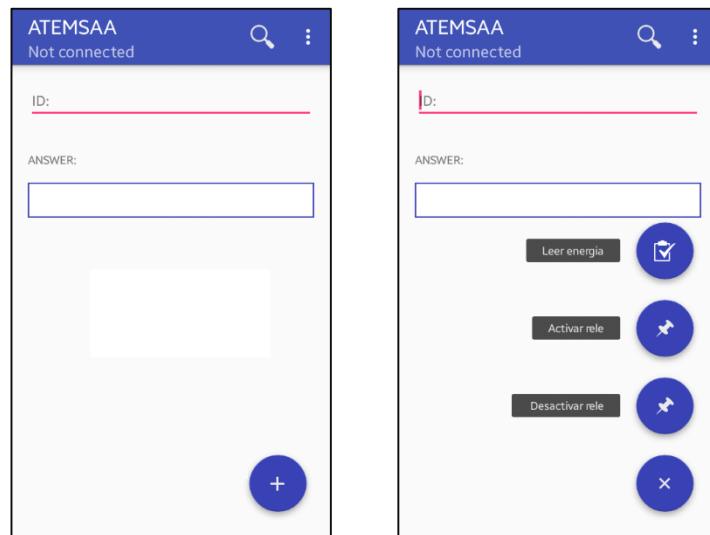


Figura 32. Ejemplo de pantalla vacía en ATEMSAA.  
[Fuente: propia]

## 4.4. IMPLEMENTACIÓN

El modelo MPlu+a no enfatiza mucho en la fase de implementación (desarrollada en este apartado) y la fase del lanzamiento del sistema (en el siguiente apartado), debido a que éstas corresponden más a procesos de ingeniería de software.

La fase de implementación es conocida también como fase de codificación, pues supone todo el proceso de escribir el código software necesario que hará posible que el sistema finalmente implementado cumpla con las especificaciones establecidas en la fase de análisis de requisitos y responda al diseño del sistema descrito en la fase anterior [62].

El modelo hace hincapié en que desde el principio de esta fase se tiene que realizar varios prototipos teniendo siempre en cuenta al usuario final, para poder garantizar la usabilidad y accesibilidad del sistema final [2]. El equipo debe tener siempre presente que cualquier cambio que se realice debe ser consultado con el/los usuarios finales puesto que las aportaciones de evaluar en esta fase son altamente valiosas para no malgastar tiempo en desarrollar software que después deberá, sin ninguna duda, ser cambiado [62].

### 4.4.1. Implementación en ATEMSAA

Cuando se llega a esta fase del modelo MPlu+a, ya se han determinado el o los lenguajes de programación a utilizar para la implementación del proyecto, las

bases de datos correspondientes que se precisen, los sistemas de intercomunicación de procesos, y en general toda la tecnología subyacente [62]. De acuerdo a lo anterior, para ATEMSAA mediante el estudio de los sistemas operativos se definió un sistema operativo Android por razones de licencia, de costo, de mercado, de calidad y de herramientas disponibles para el desarrollo de aplicaciones móviles (Anexo A), además en las encuestas se pudo determinar que este es el sistema más empleado por los futuros usuarios (Anexo F, Encuesta 1).

Para la transferencia de información, la comunicación del dispositivo móvil con el sistema AMI-SAA utiliza Bluetooth debido a que los dispositivos PLC están equipados con un módulo Bluetooth que permite la asociación App-dispositivo.

#### 4.5. LANZAMIENTO

En esta fase es donde se muestran los resultados del producto final y si efectivamente se cumplieron los objetivos trazados antes de empezar la aplicación, por esta razón suele ser una de las fases más críticas.

El éxito del producto dependerá de dos factores importantes:

- La satisfacción y comodidad del usuario con el sistema. Que el sistema no le genere errores, que no le sea complicado utilizarlo, que recuerde donde están las diferentes opciones y sus funcionalidades, etc.
- Que los responsables del proyecto obtengan los resultados esperados. La funcionalidad ofrecida por el sistema es lo que se prima y no se da por bueno si no se cumplen sus especificaciones [63].

Debido a la aplicación satisfactoria del modelo, estas dos premisas se cumplen a cabalidad, ya que el usuario ha estado involucrado en todo el desarrollo y por tanto todo el sistema se habrá hecho para él.

El modelo enfatiza en un concepto muy importante: el **feedback** (realimentación con el usuario), el cual está relacionado con el tiempo de prueba del sistema por parte de los usuarios y las quejas, comentarios, dudas u opiniones que estos generan al momento de probarlo. Una vez ha transcurrido este tiempo de prueba (test), todos los datos y aspectos recogidos son tenidos en cuenta para realizar las mejoras respectivas al sistema con el único fin de generar al usuario una satisfacción total que es el objetivo primordial por parte del grupo desarrollador [63].

## 4.6. PROTOTIPADO

Cuando se realiza un test de usabilidad en un móvil, los usuarios no necesariamente tienen que probar una aplicación completamente terminada. De hecho, como se ha comentado anteriormente, lo más aconsejable es realizar estos test en etapas tempranas del proceso de diseño y desarrollo, por ello, un prototipo puede ser suficiente para las pruebas [17].

Los prototipos son documentos, diseños o sistemas que simulan o tienen implementadas partes del sistema final, constituyen una herramienta muy útil para hacer participar al usuario en el desarrollo y poder evaluar el producto desde las primeras fases del desarrollo [64].

El paradigma de hacer prototipos comienza con comunicación. En primer lugar, al reunirse con el cliente para definir los objetivos generales del software, se identifican la mayoría de requerimientos para la aplicación, con esto se planea rápidamente una iteración para hacer el prototipo, y se lleva a cabo el modelado (en forma de un “diseño rápido”). Éste se centra en la representación de aquellos aspectos del software que serán visibles para los usuarios finales (por ejemplo, disposición de la interfaz humana o formatos de la pantalla de salida). El diseño rápido lleva a la construcción de un prototipo. Éste se entrega y es evaluado por el cliente, que al hacer la retroalimentación para mejorar los requerimientos permite que el prototipo sea afinado para satisfacer sus necesidades [65].

El MPlu+a es muy flexible en cuanto a la utilización de una u otra técnica de prototipado. El modelo clasifica los prototipos o las técnicas de prototipado, en función del coste y el esfuerzo de realización y de la fidelidad de dichos prototipos respecto al sistema final. Ver Tabla 9.

Tabla 9. Clasificación de las técnicas de prototipado.  
[Fuente: [16]]

<b>Prototipo de Baja Fidelidad</b>	<b>Prototipo de Alta Fidelidad</b>
Implementaciones genéricas. Permiten abarcar un espectro mayor de la interacción a realizar. Caracterizado por ser económico, rápido de construir, rápido de arreglar y no precisa de técnicos expertos. Además que es útil para el diseño general de las interfaces.	Implementaciones específicas y de mayor detalle de una o varias tareas del sistema. Caracterizado por el uso de herramientas especializadas de prototipado, requiere de expertos que conozcan dichas herramientas, por tanto más costoso y requiere de más tiempo.

Es importante tener en cuenta que un prototipo no será mejor que otro, por el número de funcionalidades o características que se vean representadas en él. Cada tipología resultara más o menos útil en una u otra circunstancia. Siendo el diseñador o el equipo de desarrollo el encargado de determinar qué será mejor para cada situación y evaluación a realizar [16]. La utilidad real del prototipado se fundamenta en que no tendría sentido empezar a implementar una interfaz de la aplicación si no se ha asegurado antes de que el diseño es usable [66], se trata de verificar si los usuarios son capaces de realizar sus tareas con la interfaz propuesta.

Uno de los métodos más utilizado es el prototipado en papel (boceto) que puede derivar en un Storyboard Navegacional o modelo de navegabilidad. Estas son técnicas de baja fidelidad basadas en la utilización de materiales tan básicos como lápiz, colores, papel y las tijeras. Su finalidad es la creación de prototipos minimalistas, flexibles y enormemente versátiles. Además, permiten probar y refinar un diseño en repetidas ocasiones y realizar pruebas de usabilidad con el usuario antes de que se pase a realizar cualquier línea de código [2].

Algunas de las principales razones por las que se utiliza estas técnicas son:

Permiten mapear todas las interfaces de la aplicación en un tiempo muy corto, esto hace que el ahorro para cualquier empresa sea significativo ya que el tiempo de diseñadores y desarrolladores tiene un alto costo.

- ✓ Debido a su rápida construcción permite rápidas iteraciones en el diseño antes de que se lleve a cabo el lento trabajo de crear diseños visuales o escribir código.
- ✓ Son técnicas de un muy bajo costo, lo cual es beneficioso para el usuario por que no se cohíbe en proponer cualquier arreglo en el diseño.
- ✓ Permite que todos los miembros del equipo de desarrollo y el usuario final se involucren en el proceso, para alcanzar un óptimo diseño.
- ✓ Son prototipos centrados en el usuario.
- ✓ No están sujetos a restricciones impuestas por la tecnología -arquitectura del sistema, la base de datos, el ancho de banda, el sistema operativo-, y a pesar de ello ayuda al equipo a anticipar problemas y decisiones derivadas de la tecnología.

Para el caso de ATEMSAA, los prototipos fueron diseñados en el Software *Balsamiq Mockups*, este software permite la interacción del usuario con la

aplicación mediante una simulación donde se puede observar cómo estarían diseñadas las interfaces y la navegabilidad entre ellas.

En un principio, el sistema AMI-SAA solo contaba con el dispositivo PLC\_MMS, por ende, cuando se realizó un primer prototipo para la aplicación solo se tuvo en cuenta aquellas funcionalidades relacionadas con dicho dispositivo. En la Figura 33 se observa el primer storyboard creado para ATEMSAA, en él se ve tanto la idea de cómo estarían diseñadas las interfaces y cómo sería la navegación. En primer lugar se cuenta con una interfaz que permite autenticar al usuario, después la aplicación solicita el encendido del bluetooth para posteriormente seleccionar el dispositivo PLC al que desea vincularse. Como *interfaz principal* se tiene la pestaña de “Usuarios”, ubicada en el Stacked Action Bar junto con las opciones: configuración, reportes y reloj.

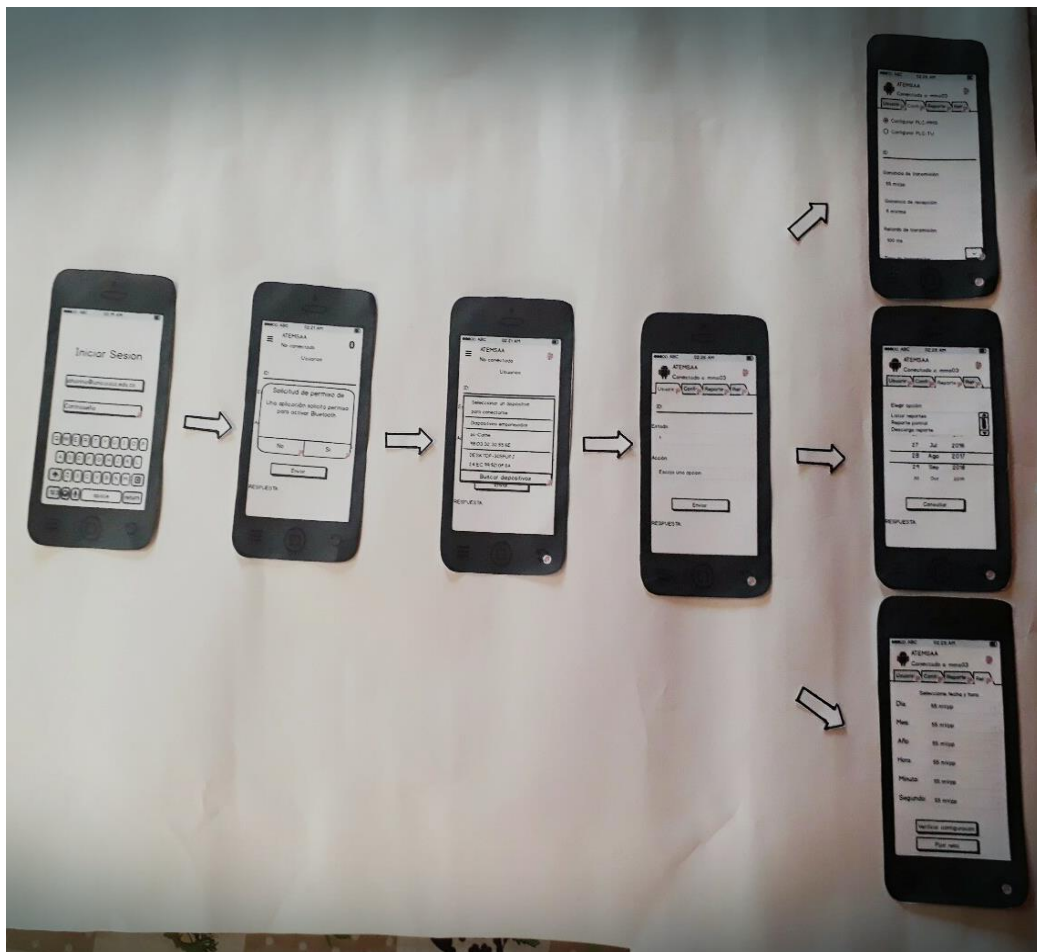


Figura 33. Primer Storyboard para ATEMSAA.  
[Fuente: propia]

Para una segunda fase, el equipo desarrollador de AMI-SAA ya había incluido otras funcionalidades al sistema, además de que se incorporaron nuevos dispositivos PLC; por esta razón, por pedido del product owner y por sugerencia de los desarrolladores del sistema AMI-SAA (Anexo G, Encuestas 3 y 4), la *interfaz principal* se organiza en iconos que agrupan funcionalidades similares de los diferentes dispositivos. Por ejemplo, en la Figura 34 se observa tanto la interfaz principal como la agrupación en el icono “Configuración” donde, como su nombre lo indica, se pueden realizar las configuraciones de los diferentes dispositivos PLC.

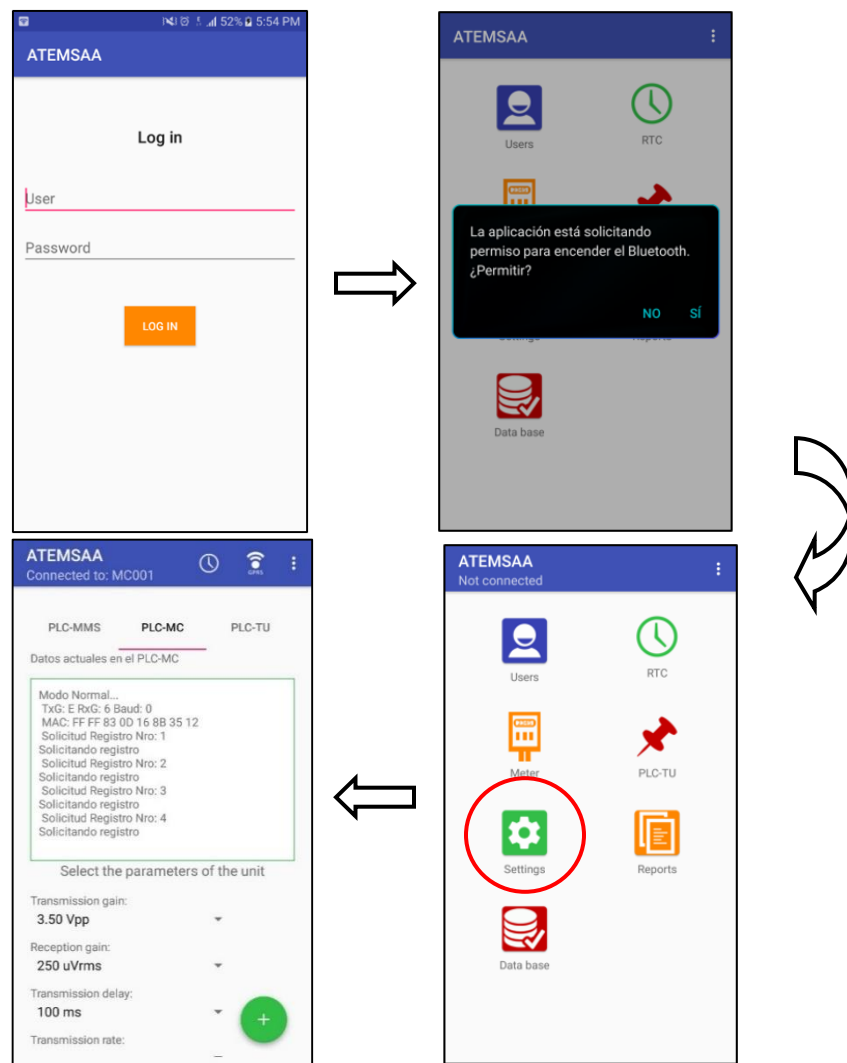


Figura 34. Segundo Storyboard para ATEMSAA.  
[Fuente: propia]

Uno de los obstáculos que surgió del diseño presentado en la Figura 34, es que la aplicación se conecta solo con un tipo de los dispositivos PLC; por esto, las



funcionalidades de los otros dispositivos, aunque pudieran verse, no eran útiles en ese momento. Debido a esto, se presentó un tercer prototipo para la aplicación ATEMSAA (ver Figura 35); en este prototipo la *interfaz principal* se observa un Stacked Action Bar conformado solo por las diferentes funcionalidades que tiene el dispositivo PLC al que la aplicación se encuentra vinculado, para este ejemplo en particular, la aplicación solo muestra las acciones que el usuario puede realizar cuando se encuentra vinculado con un dispositivo PLC\_MC.



Figura 35. Tercer Storyboard para ATEMSAA.  
[Fuente: propia]

## 4.7. EVALUACIÓN

El modelo MPlu+a propone que la evaluación de la aplicación se lleve a cabo en todo el ciclo de vida del proceso de desarrollo [16], siempre teniendo en cuenta los objetivos de usabilidad, accesibilidad y el usuario final del sistema, para que haya una realimentación constante con los evaluadores para la mejora del sistema. Es conveniente que esta evaluación se realice en cada aspecto que sea considerado relevante para el desarrollo del sistema, para que no se cometa el error de evaluar al finalizar todo el proceso o en un solo punto de este. Por otra parte, un aspecto importante dentro del modelo, es que incluye la evaluación de la accesibilidad como un proceso integrado en el ciclo de vida del desarrollo de sistemas software, aspecto que se ha visto separado de dicho ciclo de vida.

En la sección de EVALUACIÓN (CAPITULO VI), se habla sobre los tipos de evaluación que se pueden realizar en una aplicación móvil y con cuál se evaluó ATEMSAA.

## CAPITULO V

### FUNCIONALIDAD ATEMSAA

#### 5.1. INTRODUCCIÓN

El prototipo sistema de amarre automático diseñado por CEO (AMI-SAA), se encuentra desplegado en una zona suburbana de la ciudad de Popayán, Cauca que cubre 200 viviendas. El piloto con el apoyo de infraestructuras tanto de media tensión (MT) y baja tensión (LV), se encarga de brindar la información necesaria para determinar las conexiones existentes entre un transformador de baja tensión y los usuarios vinculados a este. Como se mencionó en el apartado 3.2.1, el sistema está conformado por las siguientes unidades:

- ✓ PLC\_MMS
- ✓ PLC\_TU
- ✓ PLC\_MC
- ✓ Aplicación de gestión
- ✓ Terminal Móvil (TU-1/ATEMSAA)

El sistema de monitoreo, PLC\_MMS, encargado de encuestar los nodos de usuario y procesar la información recolectada de amarre. Está constituido por dos unidades, la de control, PLC\_CPU, encargada del control, procesamiento y monitoreo de la información de amarre y la segunda la unidad transceptora, PLC\_T, pasarela de propósito general que sirve de puente de comunicación vía PLC entre las unidades nodo de usuario (PLC\_TU) y la unidad de control (PLC\_CPU). Ya que los nodos de usuario se encuestan en cada una de las tres fases de un transformador se requiere una unidad transceptora por cada fase. Tanto PLC\_CPU como las tres (3) PLC\_T se encuentran ubicadas en el transformador dentro del sistema de monitoreo (PLC\_MMS). La unidad PLC\_CPU tiene asignado un código (IDM), que lo identifica dentro de las unidades de la red como un dispositivo maestro, el IDM se encuentra asociado al código de transformador al cual está conectado.

Los nodos de usuario, PLC\_TU, en el medidor de usuario son los responsables de responder a la encuesta de amarre del PLC\_MMS. Los PLC\_TU tienen dos (2) modos de funcionamiento, el primero es en modo directo donde reciben mensajes desde el PLC\_MMS y envían la respuesta directamente. En el modo dos los

PLC\_TU se comportan como unidades repetidoras según órdenes del PLC\_MMS; en modo repetidor los PLC\_TU, son las responsables de retransmitir información de encuesta del sistema de monitoreo.

El terminal de usuario, TU-1, es una unidad móvil que permite descargar la información de amarre del sistema de monitoreo. Por su parte el terminal TU-2, es una unidad encargada de la configuración de parámetros en etapa de alistamiento [67].

La herramienta informática para alistamiento y configuración en dispositivo móvil, llamada ATEMSAA, debe tener botones, pantalla de visualización de mensajes de salida y de entrada, entre otros, de manera que le permita al operario facilidad en cuanto a configuraciones de los diferentes dispositivos PLC, solicitud y descarga de información, entre otras labores asignadas.

*Nota: Como primera estrategia para capturar los requisitos funcionales, se recurrió a los casos de uso, los cuales pueden verse en el Anexo I; esto facilitó la creación de las HU.*

## 5.2. FUNCIONALIDADES ATEMSAA

En total ATEMSAA cuenta con 13 funcionalidades macro, la mayoría distribuidas entre los diferentes dispositivos PLC con los que puede establecer conexión inalámbrica; por otra parte, ATEMSAA ofrece la posibilidad de conexión directa a la base de datos de AMI-SAA en donde se pueden acceder a los datos almacenados en el sistema (Ver Tabla 10).

Tabla 10. Funcionalidades en ATEMSAA.

[Fuente: propia]

Funcionalidad	PLC_MMS	PLC_TU	PLC_MC	Base de Datos
Autenticación para acceder a la aplicación	X	X	X	X
Conexión Bluetooth	X	X	X	
Verificación de amarre	X	X		

Consultar parámetros de red PLC <sup>13</sup>	X	X	X	X
Configurar parámetros de red PLC	X	X	X	X
Agregar, buscar, testear usuario	X			
Listar reportes del dispositivo	X			
Descarga de reportes de consumo	X			
Agregar medidor			X	
Leer energía		X	X	
Activar/desactivar relé		X	X	
Cierre de sesión	X	X	X	X
Lista de dispositivos <sup>14</sup>				X

A continuación se detallan las funcionalidades de dispositivo PLC\_MMS. Esta misma descripción se tiene para los diferentes dispositivos PLC (ver Anexo E).

### 5.2.1. Funcionalidades PLC\_MMS

Las acciones realizadas por el dispositivo PLC\_MMS se observan en la Tabla 11:

Tabla 11. Funcionalidades del PLC\_MMS.

[Fuente: propia]

Funcionalidad	Condiciones de satisfacción
Autenticación para acceder a la aplicación	Garantizar que la autenticación cumpla con parámetros de seguridad

<sup>13</sup> Parámetros de red PLC: estos parámetros están conformados por ganancia de transmisión, ganancia de recepción, tasa de transmisión, retardo de transmisión y hora de encuesta.

<sup>14</sup> Se puede realizar la consulta de PLC\_MMS, PLC\_TU, PLC\_MC, Clientes, Macros, Medidores, Productos y Transformadores registrados en la base de datos de AMI-SAA.

Conexión Bluetooth	ATEMSAA despliega una lista con los dispositivos Bluetooth disponibles en la zona de cobertura. Al encontrar el PLC_MMS al cual se desea conectar, lo selecciona e ingresa la clave de acceso de seguridad y de esta manera se empareja con el módulo PLC_MMS. Es necesaria la opción de desconectarse del PLC_MMS que termina el emparejamiento.
Verificación de amarre	Por amarre del trafo se necesita el número total de usuarios vinculados, listado de MAC vinculadas, además el estado actual del vínculo con el trafo.
Consultar parámetros de red PLC	Se despliega la información obtenida de número del PLC_MMS, número del producto asignado, ganancia de transmisión, ganancia de recepción, tasa de transmisión, retardo de transmisión y hora de encuesta con la que el dispositivo PLC_MMS cuenta.
Configurar parámetros de red PLC	Se pueden configurar los parámetros: ganancia de transmisión, ganancia de recepción, tasa de transmisión, retardo de transmisión y hora de encuesta
Agregar usuario	El PLC_MMS primero intentará registrar de manera automática los PLC_TU; los cuales, automáticamente solicitan registro en el PLC_MMS correspondiente, tras ser energizados, junto al medidor de energía eléctrica. Posterior a que el PLC_TU intente registrarse y no lo logre, el brigadista podrá registrar un PLC_TU; entonces se desplegará en el aplicativo móvil el número del PLC_TU, número del producto asignado, ganancia de transmisión, ganancia de recepción, tasa de transmisión, versión del firmware, nivel de ruido para transmisión.
Buscar usuario	En la memoria EEPROM del PLC_MMS. Los campos que debe ingresar un operario mediante interfaz deben ser: ID (MAC del PLC_TU)
Testear usuario	Permitir encuestar individualmente a un usuario PLC_TU, solicitando información de si está o no amarrado. Los campos que debe ingresar un operario mediante interfaz deben ser: ID (MAC del PLC_TU)
Consultar los reportes de consumo	Por dispositivo se necesita conocer el consumo y si está conectado/desconectado.
Descarga de reportes de consumo	Se permite la descarga del reporte en formato .csv
Cierre de sesión	Estando conectado a este dispositivo, ATEMSAA tiene la posibilidad de realizar un cierre total de sesión.

## **CAPITULO VI**

### **EVALUACIÓN ATEMSAA**

#### **6.1. INTRODUCCIÓN**

Aunque el concepto de usabilidad es fácil de asimilar, conseguir que un producto sea usable es más difícil. En general, cuando se diseña un producto, y más en un sistema software, se está más preocupado de la funcionalidad que de la usabilidad [39]. Evaluar la usabilidad implica un constante acercamiento con el usuario lo que incrementa el tiempo de desarrollo, cuesta más dinero y sobretodo porque se tiene un trabajo adicional que puede suponer implementar cambios derivados de dicha evaluación.

La evaluación ejecutada con la presencia de usuarios permite realizar mejoras y correcciones con respecto a prototipos ya presentados, por esta razón es necesario que se realice durante todo el ciclo de vida del proceso de desarrollo, en este sentido el desarrollador no puede asumir que si él se encuentra satisfecho con su diseño los demás usuarios también lo estarán.

El diseño del sistema debe permitir al usuario realizar sus tareas de una forma fácil. Además es importante comprobar el impacto de dicho diseño sobre el usuario, esto incluye aspectos como la facilidad de su aprendizaje, su manejabilidad, etc. Por último se deben identificar los problemas específicos del diseño [57].

En [57] se definen tres objetivos principales en cuanto a la evaluación de sistemas:

- ✓ Comprobar la extensión de la funcionalidad del sistema.
- ✓ Comprobar el efecto de la interfaz en el usuario.
- ✓ Identificar cualquier problema específico con el sistema.

Centraremos esta parte en explicar los métodos de evaluación más relevantes y las conclusiones extraídas a partir de la aplicación de los mismos en el proyecto.

## 6.2. EVALUACIÓN FUNCIONAL DE ATEMSAA

La evaluación de la aplicación móvil ATEMSAA se realizó en el Laboratorio de Automática de la Universidad del Cauca, en donde se realiza también el desarrollo del piloto del sistema AMI-SAA.

En la Tabla 12 se muestra la prueba realizada para evaluar el funcionamiento de la HU “*configuración de dispositivo PLC\_MMS*” (esta HU se encuentra detallada en el Anexo C). Mediante el uso de la metodología Scrum se fueron cumpliendo durante cada sprint, los requisitos funcionales.

Tabla 12. HU: configuración de dispositivo PLC-MMS.  
[Fuente: propia]

<b>Identificador</b>	DPL006
<b>Prueba</b>	Yo como brigadista/desarrollador quiero configurar red del PLC del PLC_MMS para contar con información de parámetros del PLC_MMS
<b>Objetivo</b>	Por cada PLC_MMS se necesita configurar red PLC, número del PLC_MMS, número del producto asignado, ganancia es decir, que se permita la configuración de: transmisión, ganancia de recepción, retardo de transmisión, tasa de transmisión, versión del firmware, nivel de ruido para transmisión en el PLC_MMS.

<b>Lista de chequeo procedimiento de prueba</b>			
<b>Numero</b>	<b>Item</b>	<b>Cumple</b>	
		<b>si</b>	<b>no</b>
1	Abrir la aplicación ATEMSAA	X	
2	Digite user: (pruebas) y password: (gonzalo) e ingrese.	X	
3	La aplicación solicita activación del bluetooth del dispositivo móvil si este no está activo.	X	
4	Aceptar la activación del bluetooth en caso contrario se cerrara la aplicación.	X	
5	En la parte superior derecha encuentra el menú que permite conectar a un dispositivo o cerrar la sesión.	X	
6	Seleccione connect para desplegar una lista de dispositivos bluetooth asociados al móvil.	X	
7	Con scan for devices se muestran los dispositivos bluetooth disponibles en la zona de cobertura	X	

8	Selecciona el dispositivo bluetooth (mc01) con el cual se quiere emparejar y si este no está asociado al móvil solicita la clave de acceso (PIN: 1234).	X	
9	Al ingresar la clave correcta, en la parte superior de la aplicación se muestra que se está conectando al dispositivo. Después mostrará en la parte superior el nombre del dispositivo al cual se encuentra conectado.	X	
10	En el menú principal seleccione Settings para tener acceso a las funcionalidades relacionadas con las configuraciones.	X	
11	Al seleccionar Settings en la pantalla se mostrara la configuración actual del dispositivo PLC_MMS y los parámetros que se pueden configurar.	X	
12	Al elegir nuevos parámetros y seleccionar el botón Record, si no se ha digitado la ID del PLC_MMS y/o el ID del Macro, las solicitará.	X	
13	Con la ID del PLC_MMS (cualquier cadena de 12 caracteres) y el ID del Macro (00000004) y la selección de Record, mostrara los cambios de parámetros del PLC_MMS.	X	
14	Con el botón Check settings, se visualizan en la pantalla los nuevos parámetros que se configuraron.	X	

### 6.3. EVALUACIÓN NO FUNCIONAL DE ATEMSAA

A continuación se realiza una pequeña descripción de los métodos de evaluación de usabilidad más destacados.

#### 6.3.1. Inspección

El termino inspección aplicado a la usabilidad aglutina un conjunto de métodos para evaluar la usabilidad en el que unos expertos conocidos como *evaluadores* analizan el grado de usabilidad de un sistema a partir de la inspección o examen detallado de su interfaz.

Hay distintos métodos que se enmarcan en la clasificación de evaluación por inspección. A continuación se describirá uno de los más representativos:

##### 6.3.1.1. Evaluación Heurística

Uno de los métodos más destacados para la evaluación de usabilidad es la evaluación heurística, este método desarrollado por Molich y Nielsen consiste en



analizar la conformidad de la interfaz con unos principios reconocidos de usabilidad (heurísticos) mediante la inspección de varios evaluadores expertos. Como puede ser difícil que el mismo desarrollador descubra los errores, es necesario que al menos se consideren de 3 a 5 evaluadores [57].

Cada evaluador realiza individualmente una revisión de la interfaz, una vez concluidas estas evaluaciones, se lleva a cabo una reunión entre los evaluadores y se elabora un informe con los resultados de esta.

La aplicación del método se centra en la validación de las “diez reglas heurísticas de usabilidad de Nielsen”, conjunto revisado de reglas heurísticas de usabilidad que han sido complementadas con aspectos relevantes de los dispositivos móviles como la pantalla táctil (ver Tabla 13).

Tabla 13. Reglas heurísticas de Nielsen  
[Fuente: [68]].

1	Visibilidad del estado del sistema	El dispositivo debe mantener siempre informado al usuario y los cambios de estado del proceso, a través de una retroalimentación especifican en un tiempo razonable.
2	Correspondencia entre el sistema y el mundo real	La aplicación deberá hablar el mismo lenguaje que el usuario, seguir convenciones del mundo real mostrando la información de una manera lógica y un orden natural.
3	Control y libertad del usuario:	La aplicación deberá permitir al usuario deshacer y rehacer acciones, deberá tener salidas de emergencia. Las opciones se deben marcar claramente, y el usuario no se deberá forzar al usuario a pasar a través de diálogos extensos.
4	Consistencia y estándares	La aplicación deberá seguir las convenciones establecidas, el usuario deberá hacer las cosas de una manera familiar, estándar y consistente.
5	Prevención de errores	La aplicación deberá tener un diseño de interfaz que prevenga errores, que oculte funcionalidades no disponibles. Se debe permitir acceso a información adicional y se debe advertir a los usuarios cuando los errores son probables de que ocurran.
6	Minimizar la carga de memoria del usuario	Objetos, acciones y opciones visibles, que el usuario no tenga que recordar la información de una parte del dialogo a otra.

7	Personalización y atajos	El sistema debe proporcionar opciones de configuración básica y avanzadas. Debe permitir personalizar y definir, o proporcionar atajos.
8	Eficiencia de uso y rendimiento	La aplicación debe realizar sus tareas en el mínimo tiempo posible. También debe minimizar los pasos para llevar a cabo una tarea determinada. Las animaciones y las transiciones se deben mostrar sin ningún problema.
9	Estética y diseño minimalista	Se debe evitar mostrar información irrelevante o rara vez necesaria.
10	Ayuda al usuario para reconocer, diagnosticar y recuperarse de errores	El dispositivo debe mostrar mensajes de error en un lenguaje natural y familiar para el usuario, indicando el error de una manera precisa.
11	Ayuda y documentación	El sistema debe proporcionar ayuda y documentación fácil de encontrar centrada en el usuario, y sin ser demasiado extensa.
12	Interacción física y ergonómica	Proporcionar botones físicos o elementos de interfaz similares y adecuados para sus principales funciones, deben estar ubicados en posiciones reconocibles.

### 6.3.2. Indagación

Las técnicas de evaluación por indagación tienen como objetivo llegar al conocimiento de un sistema por medio de conjeturas y señales. Por medio de estas técnicas se habla con los usuarios y se observan detenidamente usando el sistema en tiempo real y obteniendo respuestas a preguntas formuladas verbalmente o por escrito.

En la Tabla 14 se muestra una clasificación de los métodos por indagación más importantes:

Tabla 14. Métodos de evaluación por indagación.  
[Fuente: [2]]

Método	Característica
Observación de Campo	Su principal objetivo es entender cómo los usuarios de los sistemas interactivos realizan sus tareas y conocer todas las acciones que estos llevan a cabo durante la realización de las mismas.

Grupo de discusión dirigido (Focus Group)	Técnica de recogida de datos donde se reúnen de 6 a 9 personas (usuarios e implicados) para discutir aspectos relacionados con el sistema. Un experto en usabilidad y/o accesibilidad, hace el papel de moderador, prepara los temas a discutir y recoge la información necesaria.
Entrevistas	Conversación donde uno o más usuarios reales del sistema que se desarrollara o rediseñara responden a una serie de preguntas relacionadas con el sistema que el entrevistador va formulando.
Cuestionarios	Listas de preguntas que el evaluador distribuye entre usuarios para que estos devuelvan respuestas que luego serán debatidas, con la finalidad de sacar conclusiones.
Grabación del uso (análisis de logs)	Conocida también como logging, se basa en grabar o recoger todas las actividades realizadas por el usuario con el sistema para su posterior análisis.

### 6.3.3. Test

Los métodos de evaluación de la usabilidad de un sistema por medio de Test son caracterizados por que usuarios representativos trabajan en tareas utilizando el sistema –o el prototipo- y los evaluadores utilizan los resultados para ver como la interfaz de usuario soporta a los usuarios con sus tareas.

En la Tabla 15 se ilustran los principales métodos de evaluación por Test.

Tabla 15. Método de evaluación por Test.  
[Fuente: [2]]

Método	Característica
Medida de las prestaciones	Basado en la toma de medidas acerca del rendimiento u otro tipo de aspecto subjetivo que afecte la usabilidad del sistema, para esto se ha de disponer del sistema ya implementado o un prototipo que permita evaluar estos aspectos.
Pensando en voz alta (Thinking Alaud)	Método descrito por Nielsen, en el cual se pide a los usuarios de una manera individual que expresen en voz alta y libremente sus pensamientos, sentimientos y opiniones sobre cualquier aspecto (diseño, funcionalidad, accesibilidad, etc.) mientras interaccionan con el sistema o un prototipo del mismo.
Interacción Constructiva	Conocido como aprendizaje por codescubrimiento, es una derivación del Thinking alaud e implica tener, en vez de uno, a dos usuarios realizando conjuntamente cada test del sistema.
Test retrospectivo	Para llevar a cabo este método es necesario tener una

	grabación de la sesión de test, para que luego el usuario pueda revisarlo y genere más comentarios que cuando está realizando el test en tiempo real.
Método del conductor	En este método se conduce al usuario en la dirección correcta mientras se usa el sistema, durante este test el usuario puede preguntar cualquier aspecto relacionado con el sistema y el evaluador le responderá. Es centrado en el usuario inexperto.
Ordenación de tarjetas (Card Sorting)	Utilizada para conocer como los usuarios visualizan la organización de la información. El diseñador utiliza las aportaciones de los usuarios para decidir cómo deberá estructurarse la información en la interfaz.

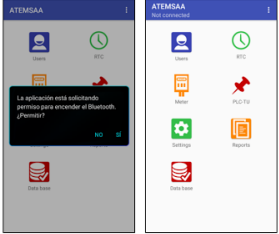
En el caso de ATEMSAA se usó el método de Indagación haciendo uso de cuestionarios que facilitaron la constante realimentación con los usuarios. En la Figura 36 se observa un ejemplo de las encuestas realizadas a los operarios/brigadistas para evaluar usabilidad.

### Encuesta 5

Encuesta dirigida a los operarios

\*Obligatorio

Ventana de diálogo e interfaz principal de la aplicación



1. ¿Qué opina del mensaje emergente para la activación del bluetooth al momento de ingresar a la aplicación? \*

Me gusta

No me gusta

2. ¿Sugiere algún cambio en este mensaje? \*

Tu respuesta

---

3. ¿Cree conveniente que si no se activa el bluetooth, se debe cerrar la aplicación? \*

Sí

No

4. ¿Qué opina de la interfaz desarrollada para el menú de opciones? \*

Me gusta

No me gusta

5. Tras una primera mirada, le queda claro ¿cuál es el objetivo de la interfaz? \*

Sí

No

6. Considera que gráficamente la interfaz es: \*

Simple

Equilibrada

Recargada

7. ¿Qué le llama la atención de la interfaz positiva o negativamente? \*

Tu respuesta

---

8. ¿Sugiere algún cambio en la interfaz? \*

Tu respuesta

---

9. ¿Cuáles deberían ser las funciones del botón usuarios? \*

Tu respuesta

---

ENVIAR

Nunca envíes contraseñas a través de Formularios de Google.

Figura 36. Ejemplo de cuestionario.  
[Fuente: propia]

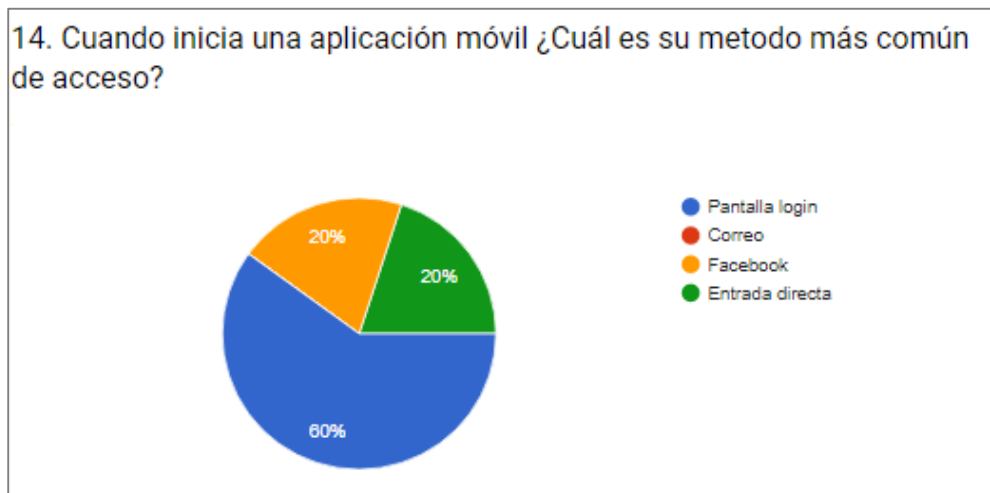


Figura 37. Ejemplo de estadística de encuesta  
[Fuente: propia]

La Figura 37. Ejemplo de estadística de encuesta muestra una de las estadísticas que arroja una de las encuestas, estas estadísticas permitieron la toma de decisiones al momento de realizar el desarrollo de la aplicación móvil y un mejor desempeño

Tabla 16. Aceptación de la aplicación  
[Fuente: propia]

Concepto evaluado	% Aceptación
Método de inicio de sesión	60
Distribución de pantalla	33
Fondo de pantalla	100
Distribución de colores	66.7
Tipografía	100
Despliegue de acciones	90
Visualización de respuestas de los PLC	100

En la Tabla 16. Aceptación de la aplicación se observan los resultados obtenidos por la aceptación de la aplicación por parte del personal encuestado, para el caso de ATEMSAA la población encuestada fue de 6 personas relacionadas directamente con el proyecto AMISAA, que incluía al Director del proyecto, un tester y cuatro investigadores de los cuales 2 representaron al cliente que no podía estar presente en las pruebas de usabilidad.

## CAPITULO VII

### CONCLUSIONES Y TRABAJOS FUTUROS

#### 7.1. CONCLUSIONES

- ✓ Por razón de licencia, costo, mercado, calidad y herramientas disponibles para el desarrollo de aplicaciones móviles, se decidió que el sistema operativo Android era la mejor herramienta para llevar a cabo el proyecto AMI-SAA. Existe una gran expectativa en torno a este sistema operativo, que respaldado por Google ha logrado grandes ventas alrededor del mundo.
- ✓ ATEMSAA surge como una necesidad de conexión entre el usuario y el sistema AMI-SAA sin requerir de cables para tal propósito, por esta razón, la tecnología Bluetooth es altamente funcional, además que permite la transferencia de un alto volumen de datos de manera rápida y eficiente.
- ✓ La usabilidad es un aspecto que proporciona una medida de la calidad de interacción que tienen los usuarios con un producto, evaluando la experiencia de quienes lo utilizan mediante la comprensión de lo presentado en las pantallas y su satisfacción en las respuestas que espera de este. Evaluar la usabilidad no es una tarea sencilla debido a que se requiere analizar y experimentar con el usuario para poder definir el modelo mental que tiene sobre el producto, por esto la interacción constante con el usuario facilita la retroalimentación y con esto mejoras en el producto.
- ✓ El estudio y evaluación de usabilidad en aplicaciones móviles ha tenido auge debido al gran número de usuarios de dispositivos inteligentes; esto ha facilitado el desarrollo de aplicaciones y garantiza la eficiencia de las mismas.
- ✓ El uso de dispositivos móviles en la supervisión y control de la red eléctrica optimiza este proceso gracias a la portabilidad que ofrecen, disminuyendo los esfuerzos que principalmente los operarios/brigadistas de CEO realizan, debido al acceso de la información y actualizaciones de la red de manera remota y en tiempo real.
- ✓ Scrum es una metodología centrada directamente al usuario, que de forma permanente evalúa el desarrollo del software mediante la interacción constante

con el usuario. Se realizan entregas rápidas lo que facilita la realimentación y la realización de productos de buena calidad y bajo costo.

- ✓ La metodología escogida favoreció aspectos como: la escogencia de Android como herramienta de desarrollo de la aplicación; el diseño de las interfaces según el análisis de preferencias; la comunicación constante con el usuario facilitando la realización de los diferentes prototipos. La interactividad de la metodología MPlu+a fue la principal razón de esto.
- ✓ La comunicación entre ATEMSAA y la base de datos de AMI-SAA se realiza bajo parámetros de seguridad debido a que son datos que solo personal autorizado pueden ver. Esta seguridad está definida por el sistema de gestión.
- ✓ En cuanto a evaluación de usabilidad y funcionalidad, debido a la realimentación se fueron satisfaciendo las necesidades del usuario a medida que se presentaban los diferentes sprints, esto favoreció que al término del desarrollo del proyecto se tuvieran resultados satisfactorios.
- ✓ El desarrollo de la aplicación ATEMSAA iba a la par con el desarrollo del piloto AMI-SAA, por esta razón cuando se obtuvo el primer prototipo de laboratorio, ATEMSAA fue de gran ayuda para configurar los distintos dispositivos PLC que conforman el sistema.

## 7.2. TRABAJOS FUTUROS

- ✓ Realizar el Responsive Design para la aplicación. Android se utiliza en cientos de tipos de dispositivos con diferentes tamaños de pantalla, que van desde teléfonos pequeños hasta TV grandes. Por lo tanto, es importante que la aplicación sea compatible con todos los tamaños de pantalla y esté disponible para el mayor número de usuarios posible.
- ✓ Migrar la aplicación móvil ATEMSAA a otros sistemas operativos móviles como: Windows Phone y iOS, con la finalidad de cubrir un mayor número de usuarios.
- ✓ Incluir nuevas funcionalidades a la aplicación ATEMSAA teniendo presente los diferentes avances en el desarrollo del piloto. Por ejemplo, ATEMSAA realiza el listado de reportes que contiene el PLC-MMS hasta el momento de la consulta, con el tiempo se hará necesario consultar/descargar el reporte en algún intervalo de tiempo en particular.
- ✓ Realizar evaluación en campo de la aplicación ATEMSAA. El piloto estará ubicado en una zona residencial de la ciudad de Popayán, Cauca.



## BIBLIOGRAFÍA

- [1] L. A. Guevara Ortiz, “Planificación y optimización de un operador móvil virtual como unidad de negocio para la infraestructura de Medición Avanzada de energía eléctrica.”, 2014.
- [2] A. Dueñas Salazar y D. J. Bravo, “Guía de diseño de HMIs bajo el sistema operativo android.”, Universidad del Cauca, 2014.
- [3] G. Johannsen, “Human-Machine Interaction”, Universidad de Kassel, 2009.
- [4] Insitute Colombiano de Normas Técnicas y Certificación (ICONTEC), *Norma Técnica Colombiana NTC 6079. Requisitos para sistemas de infraestructura de medicion avanzada (AMI) en redes de distribución de energía eléctrica*. Colombia, 2014, p. 3.
- [5] National Energy Technology Laboratory for the U.S. Department of Energy, “Advanced Metering Infrastructure”, 2008.
- [6] J. M. Alvarado Brito, “Servicios de medición avanzada (AMI) para redes inteligentes y su adaptabilidad en el parco de la legislacion ecuatoriana”, 2011.
- [7] E. Morán y L. Ortiz, “Estudio de la Infraestructura de Medición Avanzada (AMI), principales requerimientos y beneficios”, Universidad Politécnica Salesiana, 2012.
- [8] Bluetooth SIG Inc, “Bluetooth”, 2016. [En línea]. Disponible en: <https://www.bluetooth.com/>.
- [9] J. García Ruiz, E. J. Moya de la Torre, y A. P. Méndez, “Detección e identificación de dispositivos Bluetooth para control de Acceso”, en *XXXVI Jornada de Automática*, 2015.
- [10] International Organization For Standardization (ISO), “Ergonomic requirements for office work with visual display terminals (VDTs) — Part 11: Guidance on usability”, *ISO 9241-11:1998*, 1998.
- [11] F. Paz, F. A. Paz, D. Villanueva, y J. A. Pow-Sang, “Heuristic Evaluation as a Complement to Usability Testing: A Case Study in Web Domain”, *12th International Conference on Information Technology - New Generations*, 2015.
- [12] International Organization For Standardization (ISO), “Systems and software

engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models”, *ISO/IEC 25010:2011*, 2011.

- [13] J. Garrido Cobo, “TFC Desarrollo de aplicaciones móviles”, Universidad Abierta de Catalunya, 2013.
- [14] T. Granollers, “13407 ISO: procesos de diseño centrado en el hombre para sistemas interactivos”. [En línea]. Disponible en: <http://www.grihotools.udl.cat/mpiuu/dcu/dcu-iso13407/>.
- [15] C. B. Kreitzberg, “The LUCID framework”, New Jersey, 2008.
- [16] T. Granollers i Saltiveri, “MPlu+a. Una metodología que integra la ingeniería del software, la interacción persona-ordenador y la accesibilidad en el contexto de equipos de desarrollo multidisciplinares”, Universidad de Lleida, 2004.
- [17] J. Cuello y J. Vittone, *Diseñando apps para móviles*, 1a ed. 2013.
- [18] J. E. Amaro Soriano, *El gran libro de programación avanzada con Android*. Marcombo, 2012.
- [19] Z. Popovic y V. Cackovic, “Advanced Metering Infrastructure in the context of Smart Grids”, pp. 1509–1514, 2014.
- [20] J. Boal, “Smart Grid”, pp. 1–22, 2010.
- [21] PDADB.net, “Detailed Technical Datasheet of Psion Teklogix Ikon 7505”, v1.7.90, 2009. [En línea]. Disponible en: [http://pdadb.net/index.php?m=specs&id=1978&view=1&c=psion\\_teklogix\\_ikon\\_7505](http://pdadb.net/index.php?m=specs&id=1978&view=1&c=psion_teklogix_ikon_7505).
- [22] J. Millán, “Conversación con Ingeniero Jorge Millán, Jefe Innovación y Desarrollo - Control de pérdidas, Compañía Energética de Occidente S.A.S. E.S.P., Agosto, 2015”. Popayán, Cauca, 2015.
- [23] G. J. Díaz, D.F.; Gonzalez, J.W.; Restrepo, H.A.; Isaac, I.A.; López, “Esquema de incorporación de las Smart Grids en el sistema de potencia colombiano”, vol. 5, núm. 1, pp. 21–26, 2011.
- [24] O. A. Beltrán G, “Revisiones sistemáticas de la literatura”, *Revista Colombiana de Gastroenterología*, pp. 60–69, 2005.
- [25] D. Torres Salinas, R. Ruiz Perez, y E. Delgado Lopez, “Google Scholar: ¿una herramienta para la evaluación de la ciencia?”, *Anuario ThinkEPI 2010*,

vol. 4, pp. 254–257, 2010.

- [26] IEEE, “About IEEE Xplore® Digital Library”, 2017. [En línea]. Disponible en: <http://ieeexplore.ieee.org/xpl/aboutUs.jsp>.
- [27] Editorial Elsevier, “ScienceDirect”, 2011. [En línea]. Disponible en: [http://www.americalatina.elsevier.com/corporate/es/science\\_direct.php](http://www.americalatina.elsevier.com/corporate/es/science_direct.php).
- [28] S. Chen, K. Xu, Z. Li, F. Yin, y H. Wang, “A privacy-aware communication scheme in Advanced Metering Infrastructure (AMI) systems”, *IEEE Wireless Communications and Networking Conference, WCNC*, pp. 1860–1863, 2013.
- [29] R. Mao y V. Julka, “Wireless Broadband Architecture Supporting Advanced Metering Infrastructure”, 2011.
- [30] R. R. Mohassel, A. Fung, F. Mohammadi, y K. Raahemifar, “Electrical Power and Energy Systems A survey on Advanced Metering Infrastructure”, *International Journal of Electrical Power and Energy Systems*, vol. 63, pp. 473–484, 2014.
- [31] C. Selvam, K. Srinivas, G. S. Ayyappan, y M. Venkatachala, “Advanced Metering Infrastructure for Smart Grid Applications”, pp. 145–150, 2012.
- [32] D. Idrovo y S. Reinoso, “Análisis de factibilidad para la implementación de un sistema AMI (Advanced Metering Infraestructure) mediante contadores inteligentes por parte de la Empresa Eléctrica Azogues C.A.”, Universidad politécnica salesiana. Cuenca, Ecuador., 2012.
- [33] C. Serrano, *Modelo Integral para el Profesional en Ingeniería*, 2a ed. Popayán, Cauca: Editorial Universidad del Cauca, 2005.
- [34] M. M. Sandoval Carvajal y M. A. García Vargas, “La trazabilidad en el proceso de requerimientos software”, Universidad Nacional de Costa Rica, 2008.
- [35] J. R. Laínez Fuentes, *Desarrollo de Software Ágil: Extremme Programming y Scrum*, 2a ed. 2015.
- [36] M. Trigas Gallega, *Gestión de proyectos informáticos. Metodología SCRUM*. 2012.
- [37] M. del P. Gamarra, “Visual story mapping – Técnica para construir backlogs”, *Agile Ninja*, 2017. [En línea]. Disponible en: <http://www.agileninja.org/visual-story-mapping/>. [Consultado: 15-ago-2017].
- [38] B. Nuseibeh y S. Easterbrook, “Requirements Engineering: A Roadmap”,

*Proceeding ICSE '00 Proceedings of the Conference on The Future of Software Engineering*, pp. 35–46, 2000.

- [39] T. Granollers, “MPlu+a. Modelo de Proceso de la Ingeniería de la usabilidad y de la accesibilidad”. [En línea]. Disponible en: [http://www.grihotools.udl.cat/mpiua/fases-mpiua/#caracteristicas\\_mpiua](http://www.grihotools.udl.cat/mpiua/fases-mpiua/#caracteristicas_mpiua).
- [40] D. J. Romero-López y A. Vargas-Rojas, “Modelo de incentivos para la reducción de pérdidas de energía eléctrica en Colombia”, *Revista de la Maestría en Derecho Económico*, vol. 6, Bogota, Colombia, pp. 221–257, nov-2010.
- [41] J. Nagi y K. S. Yap, “Nontechnical Loss Detection for Metered Customers in Power Utility Using Support Vector Machines”, *IEEE Transactions on Power Delivery*, vol. 25, pp. 1162–1171, 2010.
- [42] A. Meffe y C. C. Barioni de Oliveira, “Technical loss calculation by distribution system segment with corrections from measurements”, en *CIGRE 2009. 20th International Conference and Exhibition on Electricity Distribution - Part 1*, 2009.
- [43] L. Pineda, “Definición de los alcances del proyecto y del prototipo. Proyecto: Sistema de Amarre Automático.”, Universidad del Cauca, Popayán, Cauca, 2013.
- [44] L. Pineda, “Diseño de protocolos de comunicación para el sistema SAA”, Universidad del Cauca, Popayán, 2013.
- [45] P. Pytel, C. Uhalde, H. D. Ramón, H. Castello, M. Tomasello, M. F. Pollo Cattaneo, P. V. Britos, R. García Martínez, y 2011, “Ingeniería de requisitos basada en técnicas de ingeniería del conocimiento”, *XIII Workshop de Investigadores en Ciencias de la Computación*. Red de Universidades con Carreras en Informática (RedUNCI), Buenos Aires, Argentina, pp. 426–429, 2011.
- [46] F. Perdrix, E. Piqué, y J. Cal, “Uso de escenarios aplicados a la Ingeniería de los Requisitos para la creación de una hemeroteca digital”, *V Congreso Interacción Persona-Ordenador*. Universitat de Lleida, Lleida, España, 2004.
- [47] T. Granollers, “Análisis de implicados (stakeholders)”, *MPlu+a, Modelo de Proceso de la Ingeniería de la usabilidad y de la accesibilidad*. [En línea]. Disponible en: <http://www.grihotools.udl.cat/mpiua/fases-mpiua/analisis-de-requisitos/stakeholders/>. [Consultado: 10-jul-2017].
- [48] T. Granollers, “Plataforma”, *MPlu+a, Modelo de Proceso de la Ingeniería de la usabilidad y de la accesibilidad*. [En línea]. Disponible en:

<http://www.grihotools.udl.cat/mpiua/fases-mpiua/analisis-de-requisitos/plataforma/>. [Consultado: 10-jul-2017].

- [49] T. Granollers, “Objetos”, *MPIu+a, Modelo de Proceso de la Ingeniería de la Usabilidad y de la Accesibilidad*. [En línea]. Disponible en: <http://www.grihotools.udl.cat/mpiua/fases-mpiua/analisis-de-requisitos/objetos/>. [Consultado: 10-jul-2017].
- [50] T. Granollers, “Objetivos”, *MPIu+a, Modelo de Proceso de la Ingeniería de la Usabilidad y de la Accesibilidad*. [En línea]. Disponible en: <http://www.grihotools.udl.cat/mpiua/fases-mpiua/analisis-de-requisitos/objetivos/>. [Consultado: 10-jul-2017].
- [51] T. Granollers, “Clasificar los Usuarios”, *MPIu+a, Modelo de Proceso de la Ingeniería de la Usabilidad y de la Accesibilidad*. [En línea]. Disponible en: <http://www.grihotools.udl.cat/mpiua/fases-mpiua/analisis-de-requisitos/clasificar-usuarios/>. [Consultado: 13-jul-2017].
- [52] T. Granollers, “Diseño”, *MPIu+a, Modelo de Proceso de la Ingeniería de la Usabilidad y de la Accesibilidad*. [En línea]. Disponible en: <http://www.grihotools.udl.cat/mpiua/fases-mpiua/disenio/>. [Consultado: 12-jul-2017].
- [53] Android, “Cuadros de diálogo”, *Android Developers, Google Services*, 2016.
- [54] Apache, “Material design”, 2016. [En línea]. Disponible en: <https://material.io/guidelines/>. [Consultado: 13-ene-2017].
- [55] Android, “Toast”, *Android Developers, Google Services*, 2016. [En línea]. Disponible en: <https://developer.android.com/reference/android/widget/Toast.html>. [Consultado: 13-ene-2017].
- [56] S. Gómez Oliver, “Notificaciones en Android (I): Toast”, *Curso de programación Android*, 2011. [En línea]. Disponible en: <http://www.sgoliver.net/blog/notificaciones-en-android-i-toast/>. [Consultado: 14-ene-2017].
- [57] T. Granollers i Saltiveri, J. Lorés Vidal, y J. J. Cañas Delgado, *Diseño de sistemas interactivos centrados en el usuario*, Primera Ed. Barcelona, 2005.
- [58] P. Ponsa y A. Granollers, “Diseño de pantalla”, Universidad Politécnica de Cataluña, 2007.
- [59] F. López Montosa, “Diagrama de Gutenberg: Usabilidad Web y patrones visuales”, *Revista Digital INESEM*, 2016. [En línea]. Disponible en:

<https://revistadigital.inesem.es/disenyo-y-artes-graficas/diagrama-de-gutenberg/>.

- [60] Android, “Action Bar”, *Android Developers - Google Services*, 2017. [En línea]. Disponible en: <https://developer.android.com/training/appbar/index.html?hl=es-419>.
- [61] J. Revelo, “Tutorial sobre la Action Bar en Android”, 2014. [En línea]. Disponible en: <http://www.hermosaprogramacion.com/2014/09/android-action-bar/>.
- [62] T. Granollers, “Implementación”, *MPlu+a, Modelo de Proceso de la Ingeniería de la usabilidad y de la accesibilidad*. [En línea]. Disponible en: <http://www.grihotools.udl.cat/mpiua/fases-mpiua/implementacion/>. [Consultado: 10-jul-2017].
- [63] T. Granollers, “Lanzamiento”, *MPlu+a, Modelo de Proceso de la Ingeniería de la usabilidad y de la accesibilidad*. [En línea]. Disponible en: <http://www.grihotools.udl.cat/mpiua/fases-mpiua/lanzamiento-2/>. [Consultado: 13-jul-2017].
- [64] T. Granollers, “Prototipado”, *MPlu+a, Modelo de Proceso de la Ingeniería de la usabilidad y de la accesibilidad*. [En línea]. Disponible en: <http://www.grihotools.udl.cat/mpiua/fases-mpiua/prototipado/>. [Consultado: 13-jul-2017].
- [65] R. S. Pressman, *Ingeniera de Software. Un enfoque practico*, 7a ed. New York: McGraw-Hill Companies, Inc, 2010.
- [66] Y. Hassan, F. J. Martín Fernández, y G. Iazza, “Diseño web centrado en el usuario: usabilidad y arquitectura de la información”, Universitat Pompeu Fabra, 2004.
- [67] L. Pineda, “Diseño de la arquitectura del sistema SAA”, Universidad del Cauca, Popayán, Cauca, 2013.
- [68] R. Inostroza, C. Rusu, y S. Roncagliolo, “Usability Heuristics for Touchscreen-based Mobile Devices”, en *Information Technology: New Generations (ITNG), 2012 Ninth International Conference on*, 2012.
- [69] M. P. Izaurralde, “Caracterización de Especificación de Requerimientos en entornos Ágiles: Historias de Usuario”, Universidad Tecnológica Nacional, 2013.
- [70] M. Rodríguez Penabad, *Delphi - Programación orientada a objetos. Entorno visual*. 2006.

