



UNIVERSIDAD TECNOLÓGICA ISRAEL

TRABAJO DE TITULACIÓN EN OPCIÓN AL GRADO DE:

INGENIERO/A EN SISTEMAS INFORMÁTICOS

**TEMA: SISTEMA DE LEVANTAMIENTO DE ACTIVOS FIJOS EN ANDROID
PARA LA EMPRESA CAYMAN SYSTEMS**

AUTOR/ A: Aguirre Ayo Fabián Patricio

TUTOR/ A: Ing. Marco Vinicio Tupiza Anaguano Mg.

QUITO- ECUADOR

AÑO: 2018

Resumen

El presente proyecto se basó en el desarrollo de una aplicación móvil en Android que permite el levantamiento de activos fijos y sincronización de información con el sistema de gestión de activos fijos que posee la empresa CAYMANSYSTEMS.

La empresa CAYMANSYSTEMS cuenta con un sistema de levantamiento de activos fijos para dispositivos móviles con sistema operativo Windows Mobile 6.5, los equipos móviles que tienen este sistema operativo poseen un costo elevado en el mercado y están en desuso.

Con la finalidad de lograr que el aplicativo móvil sea compatible con los dispositivos con mayor presencia en el mercado según **OS Market** y permita trabajar de forma “OFF-LINE”, se desarrolló el aplicativo para dispositivos móviles que poseen sistema operativo Android.

En el proyecto se emplearon una serie de técnicas e instrumentos de recolección de datos, específicamente el análisis de fuentes documentales, la observación directa y las entrevistas no estructuradas.

Para la elaboración del sistema y el cumplimiento de los objetivos planteados se utilizó la metodología de desarrollo ágil (XP). Adicionalmente, para el desarrollo de la aplicación se utilizaron diversas tecnologías como el lenguaje de programación Java, base de datos SQLite IDE.

El desarrollo del aplicativo redujo el gasto en la adquisición de Hardware y se logró que el aplicativo sea compatible con todos los dispositivos con sistema operativo Android versión 5.1 o superior.

Palabras Claves: Android, Activos Fijos, Dispositivo móvil, SQLite, Off-line.

Abstract

This project was based on the development of a mobile application on Android that allows the removal of fixed assets and synchronization of information with the fixed asset management system owned by the company CAYMANSYSTEMS.

The CAYMANSYSTEMS company has a system for the removal of fixed assets for mobile devices with a Windows Mobile 6.5 operating system. Mobile devices with this operating system have a high cost in the market and are no longer in use.

In order to achieve that the mobile application is compatible with the devices with greater presence in the market according to OS Market and allows working in an "OFF-LINE" way, the application was developed for mobile devices that have an Android operating system.

The project used a series of data collection techniques and instruments, specifically the analysis of documentary sources, direct observation and unstructured interviews.

For the elaboration of the system and the fulfillment of the proposed objectives, the agile development methodology (XP) was used. Additionally, several technologies were used to develop the application, such as the Java programming language, SQLite IDE database.

The development of the application reduced the expense in the acquisition of Hardware and the application was made compatible with all devices with Android operating system version 5.1 or higher.

Keywords: Android, Fixed Assets, Mobile Device, SQLite, Off-line.

TABLA DE CONTENIDO

1	Introducción	1
1.1	Antecedentes	3
1.1	Planteamiento del problema	3
1.2	Formulación del problema	4
1.3	Justificación	4
1.4	Objetivo General	5
1.5	Objetivos Específicos	5
2	Fundamentación Teórica	6
2.1	Activo Fijo	6
2.1.1	Clasificación de activos fijos	6
2.2	Sistema de levantamiento de activos fijos.	8
2.3	Contenido de una base de datos de un sistema de levantamiento de activos fijos.	8
2.4	Sistema operativo Android.	9
2.4.1	Estructura sistema operativo Android.	10
	Marco de trabajo de aplicaciones.	10
	Bibliotecas.	11
	Runtime de Android.	11
	Núcleo Linux.	11
2.4.2	Características sistema operativo Android.	11
2.5	Estándares de codificación	12
2.5.1	CamelCase	12
2.6	Base de datos SQLite	12
2.6.1	Características SQLite.	14
2.7	Metodología de Desarrollo Ágil.	14

2.7.1	Programación Extrema (XP).	15
2.7.1.1	Valores de XP.	15
2.7.1.2	Fases de la Programación Extrema.	16
	Planificación del proyecto.	16
	Fase: Diseño.	17
	Codificación.	18
	Fase: Pruebas.	18
3	Propuesta	22
3.1	Recopilación de información	22
3.1.1	Método Inductivo	22
3.1.2	Método Deductivo	22
3.1.2	Observación	22
3.1.4	Entrevista	23
3.2	Diagramas de procesos	25
3.3	Especificación de requerimientos	32
3.3.1	Ámbito del software	32
3.3.2	Funciones del producto	34
3.3.3	Características de los usuarios del sistema	35
3.3.4	Restricciones	35
3.3.5	Requisitos	35
	No Funcionales	36
4	Implementación	37
4.1	Diseño general	37
4.1.1	Proceso de Desarrollo de XP	38
	Historias de Usuario	38

4.1.1.1	Iteración 1	39
4.1.1.2	Iteración 2	40
4.1.1.3	Iteración 3	40
4.1.1.4	Iteración 4	43
4.1.1.5	Iteración 5	44
4.1.1.6	Iteración 6	46
4.1.2	Tarjetas CRC	48
4.1.3	Plan de entregas	53
4.1.4	Esquema de la base de datos	54
4.1.5	Diagrama de la arquitectura del sistema	55
4.1.6	Diseño de interfaces	56
4.1.7	Estándares de programación utilizados	59
4.1.7.1	Estándar de diseño de base de datos.	60
4.1.7.2	Estándar de desarrollo.	60
4.1.7.3	Estándar de base de datos.	60
4.1.7.4	Estándares de programación	62
5	Conclusiones y Recomendaciones	64
5.1	Conclusiones	64
5.2	Recomendaciones	64
6	Referencias bibliográficas.	65
7	Anexos	67
7.1	Anexo 1: Modelo Ficha Observación	67
7.2	Anexo 2 Entrevista aplicada Gerente Comercial	68
7.3	Anexo 3 Manual de Usuario	70

LISTA DE FIGURAS

Gráfico N° 1 Espina de Pescado	8
Gráfico N° 2 Estructura sistema operativo Android (myfpschool, 2017)	14
Gráfico N° 3 Ciclo de vida de metodología XP	22
Gráfico N° 4 Diagrama de flujo ingreso al sistema.	28
Gráfico N° 5 Diagrama de flujo descarga de activos y catálogos.	29
Gráfico N° 6 Diagrama de flujo ingreso de información activo fijos.	30
Gráfico N° 7 Diagrama de flujo modificación de información de activos fijos.	31
Gráfico N° 8 Diagrama de flujo eliminación de información de activos fijos.	32
Gráfico N° 9 Diagrama de flujo sincronizar información recolectad de activos fijos.	33
Gráfico N° 10 Diagrama de flujo ingreso masivo de activos fijos.	34
Gráfico N° 11 Funciones del aplicativo de levantamiento de información.	37
Gráfico N° 12 Diagrama de la base de datos.	54
Gráfico N° 13 Diagrama de arquitectura del sistema	55
Gráfico N° 14 Pantalla de inicio de sesión	56
Gráfico N° 15 Pantalla principal aplicativo de levantamiento de información de activos fijo.	57
Gráfico N° 16 Pantalla ingreso de información de activos fijos.	58
Gráfico N° 17 Pantalla de proceso de sincronización de catálogos.	59

LISTA DE TABLAS

Tabla N° 1 Roles y perfiles en el proceso de levantamiento de información.	28
Tabla N° 2 Tipo de usuario Perito	39
Tabla N° 3 Historia de Usuario 1- Acceso al sistema	42
Tabla N° 4 Criterio de pruebas de aceptación 1	42
Tabla N° 5 Historia de Usuario 2- Sincronización de catálogos	43
Tabla N° 6 Criterio de pruebas de aceptación 2	43
Tabla N° 7 Historia de Usuario 3- Descargar Activos	44
Tabla N° 8 Criterio de pruebas de aceptación 3	44
Tabla N° 9 Historia de usuario diseño pantalla levantamiento de activo fijo	45
Tabla N° 10 Criterio de pruebas de aceptación 4	45
Tabla N° 11 Historia de usuario ingreso de activo fijo.	46
Tabla N° 12 Criterio de pruebas de aceptación 5	46
Tabla N° 13 Historia de usuario modificación y eliminación en la base de datos local de activos fijos..	47
Tabla N° 14 Criterio de pruebas de aceptación 6	48
Tabla N° 15 Historia de usuario sincronizar información sistema cayman activo web	49
Tabla N° 16 Criterio de pruebas de aceptación 7	49
Tabla N° 17 Tarjeta CRC 1 Ubicación Geográfica.	50
Tabla N° 18 Tarjeta CRC 2 Ubicación Orgánica.	50
Tabla N° 19 Tarjeta CRC 3 Grupo Cuenta Contable.	51
Tabla N° 20 Tarjeta CRC 4 Marca.	51
Tabla N° 21 Tarjeta CRC 5 Modelo.	51
Tabla N° 22 Tarjeta CRC 6 Estructura.	51

Tabla N° 23 Tarjeta CRC 7 Componente.	52
Tabla N° 24 Tarjeta CRC 8 Estado.	52
Tabla N° 25 Tarjeta CRC 9 Color.	52
Tabla N° 26 Tarjeta CRC 10 Tipo etiqueta.	52
Tabla N° 27 Tarjeta CRC 11 Información Técnica.	53
Tabla N° 28 Tarjeta CRC 12 Activo.	53
Tabla N° 29 Plan de entrega.	54
Tabla N° 30 Estándares de Tablas	61
Tabla N° 31 Estándares de campos	61
Tabla N° 32 Estándares de Relaciones	62
Tabla N° 33 Estándares de Claves	62
Tabla N° 34 Estándares de Clases	62
Tabla N° 35 Estándares de Funciones y Procedimientos	63
Tabla N° 36 Estándares de Variables	63
Tabla N° 37 Estándares de Controles Visuales	64

1 Introducción

El presente proyecto consiste en el desarrollo de un sistema de levantamiento de activos fijos para empresa CAYMAN SYSTEMS el cual permitirá la gestión de la información en el proceso de levantamiento de información, desarrollado para dispositivos con sistema operativo Android esto con el fin que la aplicación pueda ser utilizada por la mayoría de usuarios de dispositivos móviles con versión de Android 5.1.1 o superior.

Un punto muy importante del presente proyecto es la posibilidad del trabajo en “off line” registrando la información en un archivo de base de datos SQLite alojado en el dispositivo móvil, para luego sincronizar con el sistema web de gestión de activos fijos que la empresa posee.

Descripción de capítulos:

En el capítulo I describe la Fundamentación Teórica en donde se detalla todas las herramientas e información relevante para el desarrollo de la investigación.

En el capítulo II describe la propuesta la misma que se encuentra estructurada de la siguiente manera: recopilación de información, diagramas de procesos, especificación de requerimientos, ámbito del software, funciones del producto, características de los usuarios del sistema, restricciones, requisitos funcionales y no funcionales.

El Capítulo III describe el diseño, desarrollo del software y los diferentes mecanismos utilizados para el procesamiento de la información tales como: esquema de la base de datos, diagrama de la arquitectura del sistema, diseño de interfaces , estándares de programaciones utilizadas, Implementación, plan de implementación, requerimientos de HW/SW, manual de usuario , manual técnico.

Capítulo IV se detalla las conclusiones y recomendaciones a las que se ha llegado en base a todo la investigación realizada, de acuerdo a los objetivos planteados, además se detalla los anexos y referencias bibliográficas.

1.1 Antecedentes

CAYMANSYSTEMS Ecuador, empresa pionera en desarrollo de tecnología, inicia sus labores en el año 2000, comprometida en brindar soluciones para el óptimo progreso de las empresas. Investigando nuevas tendencias tecnológicas, con el fin de mejorar, controlar y automatizar los procesos importantes de las compañías (CaymanSystems, 2016).

La empresa, se dedica al desarrollo y comercialización de sistemas para la gestión y control de activos fijos a nivel nacional e internacional, uno de estos es el aplicativo móvil para el levantamiento de activos fijos, el cual está desarrollado para dispositivos móviles con sistema operativo Windows Mobile 6.5.

El aplicativo permite a la empresa realizar el levantamiento de información de todos los activos fijos, para luego poder valorar y entregar un informe de la valoración.

Para la valoración, es necesaria la recolección de la información detallada de los bienes con esto la empresa entregara un informe veraz.

1.1 Planteamiento del problema

Debido al avance de la tecnología, el sistema operativo Windows Mobile 6.5 sobre el que se instala el aplicativo ha quedado en desuso, los equipos móviles que tienen este sistema operativo tienen un costo elevado en el mercado lo que provoca a la empresa CAYMANSYSTEMS incurra en altos costos de operación en el levantamiento de la información.

El uso del aplicativo ha dejado de ser una ventaja competitiva para la empresa debido a la pérdida de posicionamiento en el mercado ya que es compatible con muy pocos dispositivos móviles, por este motivo el costo de mantenimiento y mejoras de este aplicativo se ha convertido en un gasto para la empresa.

El Gráfico N°1 Muestra el problema encontrado en la empresa CAYMANSYSTEMS con sus respectivas causas y efectos.

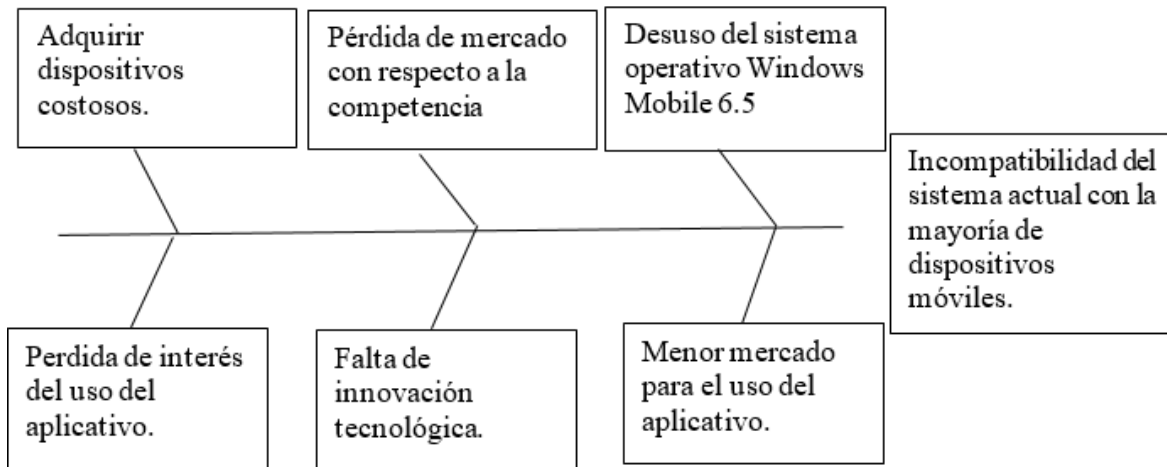


Gráfico N° 1 Espina de Pescado

Autor: Fabián Aguirre

1.2 Formulación del problema

¿Cómo desarrollar un sistema móvil que permita el levantamiento de información de activos fijos y sea compatible con los dispositivos móviles con más presencia en el mercado según **OS Market Share Worldwide?**

1.3 Justificación

El desarrollo de este proyecto permitirá a la empresa contar con un aplicativo útil para el proceso de levantamiento de información que va ser compatible con la mayoría de los dispositivos móviles, permitiendo así reducir el costo operativo en el levantamiento de información para aumentar su ventaja competitiva respecto a la competencia.

Por lo expuesto se justifica el desarrollo del aplicativo móvil para el levantamiento de información de activos fijos desarrollado bajo plataforma Android.

1.4 Objetivo General

Desarrollar una aplicación, para el levantamiento de activos fijos en Android 5.1.1 para la empresa Cayman Systems, que permita realizar la recolección de información e integrar al sistema **CAYMAN ACTIVO V.4 RFID**.

1.5 Objetivos Específicos

- Realizar el levantamiento de información a fin de definir los requerimientos funcionales de la aplicación.
- Diseñar esquema base de datos y prototipos de interfaces, ajustándose a la información recolectada en los requerimientos del usuario.
- Desarrollar el sistema levantamiento de activos fijos, cumpliendo con el estándar de programación CamelCase.
- Realizar pruebas que permitan garantizar la funcionalidad de la aplicación.

2 Fundamentación Teórica

2.1 Activo Fijo

Un activo es un bien que la empresa posee, puede ser tangible e intangible que no puede convertirse en líquido a corto plazo y que normalmente son necesarios para el funcionamiento de la empresa y no se destinan a la venta. (Maestro Financiero, 2013)

Son ejemplos de activos fijos: bienes inmuebles, maquinaria, material de oficina, etc. Se encuentran recogidos en el balance de situación. También se incluyen dentro del activo fijo las inversiones en acciones, bonos y valores emitidos por empresas afiliadas.

2.1.1 Clasificación de activos fijos

Los activos fijos según su naturaleza y plan general contable dentro de las compañías se clasifican en activos fijos tangibles e intangible.

Activo fijo tangible: Dentro de esta categoría se incluyen todos aquellos bienes y materiales tangibles, es decir, se pueden tocar. En función de las características del negocio los activos fijos podrán variar de manera notoria. Según la clasificación establecida por el Plan General Contable de las empresas en el Ecuador, algunos de los bienes tangibles de los que pueden disfrutar las empresas serían los siguientes: (School, EAE Businnes, 2017)

- Terrenos y bienes naturales. Aquellos que posea la empresa, ya sean urbanos o no.
- Construcciones. Inmuebles en general y que son propiedad de la organización.
- Instalaciones técnicas. Este concepto hace alusión a todos aquellos elementos que, en conjunto, constituyen una unidad de uso especializada

necesaria para la actividad de la empresa. Se trata de montajes en cadena y otro tipo de construcciones similares.

- **Maquinaria.** Aquellas máquinas, vehículos industriales y herramientas necesarias para la actividad cotidiana.
- **Mobiliario.** Todas las estanterías, mesas, sillas, mostradores y demás muebles que posee la empresa.
- **Equipos para procesos informáticos.** Compuesto por ordenadores, impresoras, escáner y demás aparatos electrónicos.
- **Elementos de transporte.** Medios de transporte que formen parte de los bienes de la compañía.
- **Otros.** Aquellos bienes que no se puedan incluir dentro de ninguna de las categorías anteriores.

Activo fijo Intangible: Hacen referencia a aquellos bienes y derechos que no son físicos o palpables como tal. Se trata de bienes como marcas, permisos, patentes, derechos de traspaso, fondos de comercio o gastos de investigación (School, EAE Businnes, 2017).

- **Marcas registradas.** Una marca registrada es un derecho que puede ser adquirido, vendido o arrendarse.
- **Patentes.** Es un derecho que te otorga un permiso especial y exclusivo, para vender o fabricar un producto o servicio como un software, una app, un curso de formación para jóvenes talentos o para directivos senior, etc.
- **Derechos de autor.** Con este derecho se garantiza al autor su derecho a explotar sus productos.

- Franquicias. Por medio de este derecho, la empresa adquiere permiso para poder hacer uso de la marca y productos de otra empresa durante un tiempo determinado.
- Licencias y permisos. Se trata de autorizaciones a través de las que se concede el uso de bienes diferentes, como el caso de recursos software para la empresa.

2.2 Sistema de levantamiento de activos fijos.

Es importante que todas las empresas, establezcan unos mecanismos de control para verificar la existencia real del activo fijo y que dichos datos podamos verificarlos o cotejarlos con el aspecto contable en cualquier momento o periodo con la satisfacción de que todo está en orden y al día. (HTK Identificación Inteligente, 2017)

Esto, aunque parezca sencillo y práctico, es muy complicado para muchas empresas que han optado por el descuido, no seguir el debido proceso para el control de las entradas, salidas y del control diario que requieren ciertos activos fijos de acuerdo a su actividad económica de la empresa (educaconta, 2012).

Los sistemas de levantamiento de activos buscan ayudar a las empresas hacer más fácil y menos costoso este proceso, ya que en la actualidad las empresas contratan servicios de terceros para poder realizar el levantamiento de información de sus activos fijos.

La función principal de los sistemas de esta índole es el de recolectar toda la información de los activos fijos y enviarlos a una base de datos.

2.3 Contenido de una base de datos de un sistema de levantamiento de activos fijos.

Los atributos más relevantes que debe poseer un sistema de levantamiento de Activos Fijos según la empresa CAYMANSYSTEMS son:

- Código de Identificación física de cada activo
- Código de Equipo y Modelo
- Descripción amplia del activo
- Fecha de adquisición
- Documento contable de adquisición y número de registro contable
- Área usuaria a la que pertenece
- Centro de costo contable al que pertenece
- Costo de adquisición
- Valor neto del activo
- Estado actual
- Vida útil estimada

2.4 Sistema operativo Android.

Según Figueredo (2006), un Sistema Operativo para dispositivos móviles es considerado el programa principal y éste es capaz de administrar todos sus recursos para ser utilizados de manera eficiente, cómoda y sin interrupciones, de tal manera que el usuario pueda mantener una comunicación sin problema haciendo uso de los recursos que el hardware le suministra (Kristel & Beauperthuy Taibo, 2011).

Android es un sistema operativo con una plataforma abierta para dispositivos móviles adquirido por Google y la Open Handset Alliance, su finalidad es satisfacer la necesidad de los operadores móviles y fabricantes de dispositivos, además de fomentar el desarrollo de aplicaciones, cualidad que ningún otro sistema operativo incluye en sus conceptos (Google, 2010).

Android bajo la definición de Google se considera un “software stack” o una pila de software, ya que está conformada por:

- El sistema operativo, donde todas las funciones se desarrollan.
- El middleware que permite la conexión entre redes
- Las aplicaciones o API's que constituyen todos los programas que el teléfono puede ejecutar.

2.4.1 Estructura sistema operativo Android.

Android es un sistema operativo basado en el núcleo Linux, está organizado en varias capas, siendo la más baja el núcleo y la más alta las aplicaciones, como se puede visualizar en el Gráfico N° 2.



Gráfico N° 2 Estructura sistema operativo Android (myfpschool, 2017)

Marco de trabajo de aplicaciones.

Los desarrolladores tienen acceso completo a las mismas API del entorno de trabajo usados por las aplicaciones base. La arquitectura está diseñada para simplificar la reutilización de componentes (Google, 2010).

Bibliotecas.

Android incluye un conjunto de bibliotecas de C/C++ usadas por varios componentes del sistema. Estas características se exponen a los desarrolladores a través del marco de trabajo de aplicaciones de Android (Google, 2010).

Runtime de Android.

Android incluye un set de bibliotecas base que proporcionan la mayor parte de las funciones disponibles en las bibliotecas base del lenguaje Java. Cada aplicación Android corre su propio proceso, con su propia instancia de la máquina virtual Dalvik. Dalvik ha sido escrito de forma que un dispositivo puede correr múltiples máquinas virtuales de forma eficiente (Google, 2010).

Núcleo Linux.

Android depende de Linux para los servicios base del sistema como seguridad, gestión de memoria, gestión de procesos, pila de red y modelo de controladores. El núcleo también actúa como una capa de abstracción entre el hardware y el resto de la pila de software (Google, 2010).

2.4.2 Características sistema operativo Android.

- Código abierto.
- Núcleo basado en el Kernel de Linux.
- Adaptable a muchas pantallas y resoluciones.
- Utiliza SQLite para el almacenamiento de datos.
- Ofrece diferentes formas de mensajería.
- Navegador web basado en WebKit incluido.

- Soporte de Java y muchos formatos multimedia.
- Soporte de HTML, HTML5, Adobe Flash Player, etc.
- Incluye un emulador de dispositivos, herramientas para depuración de memoria y análisis del rendimiento del software.
- Catálogo de aplicaciones gratuitas o pagadas en el que pueden ser descargadas e instaladas (Google Play).
- Bluetooth.
- Google Talk desde su versión HoneyComb, para realizar video llamadas.
- Multitarea real de aplicaciones.

2.5 Estándares de codificación

En programación a nivel general, existen unos estándares o convenciones para programar a nivel universal, es en este punto donde un programador sin importar el lugar de procedencia puede interpretar el código de otro con facilidad y coherencia.

2.5.1 CamelCase

Es un estilo de escritura que se aplica a frases o palabras compuestas. El nombre se debe a que las mayúsculas a lo largo de una palabra en CamelCase se asemejan a las jorobas de un camello. El nombre CamelCase se podría traducir como Mayúsculas/Minúsculas Camello. El término case se traduce como "caja tipográfica", que a su vez implica si una letra es mayúscula o minúscula y tiene su origen en la disposición de los tipos móviles en casilleros o cajas (Bruno).

2.6 Base de datos SQLite

SQLite apareció en mayo del año 2000 de la mano de su creador D. Richard Hip, quién ha liberado las diferentes versiones de SQLite en base a la licencia GPL por lo que su

código es de dominio público y puede ser modificado por cualquier persona. Gracias a esto, SQLite ha sido mejorada a lo largo de 7 años por un gran número de colaboradores y también ha sido migrada a diversas plataformas. (Filein, 2017)

SQLite es una herramienta de software libre, que permite almacenar información en dispositivos empotrados de una forma sencilla, eficaz, potente, rápida y en equipos con pocas capacidades de hardware, como puede ser una PDA o un teléfono celular. SQLite implementa el estándar SQL92 y también agrega extensiones que facilitan su uso en cualquier ambiente de desarrollo.

Es un ligero motor de bases de datos de código abierto, que se caracteriza por mantener el almacenamiento de información persistente de forma sencilla. (Reviso, 2017)

A diferencia de otros Sistemas gestores de bases de datos, SQLite tiene las siguientes ventajas:

- No requiere el soporte de un servidor: SQLite no ejecuta un proceso para administrar la información, si no que implementa un conjunto de librerías encargadas de la gestión (hermosaprogramacion, 2014).
- No necesita configuración: Libera al programador de todo tipo de configuraciones de puertos, tamaños, ubicaciones (hermosaprogramacion, 2014).
- Usa un archivo para el esquema: Crea un archivo para el esquema completo de una base de datos, lo que permite ahorrarse preocupaciones de seguridad, ya que los datos de las aplicaciones Android no pueden ser accedidos por contextos externos (hermosaprogramacion, 2014).

- Es de Código Abierto: Está disponible al dominio público de los desarrolladores al igual que sus archivos de compilación e instrucciones de escalabilidad (hermosaprogramacion, 2014).

2.6.1 Características SQLite.

Estas son algunas de las características principales de SQLite (Rómmel, 2011):

- La base de datos completa se encuentra en un solo archivo.
- Puede funcionar enteramente en memoria, lo que la hace muy rápida.
- Tiene un footprint menor a 230KB.
- Es totalmente auto contenido (sin dependencias externas).
- Cuenta con librerías de acceso para muchos lenguajes de programación.
- Soporta texto en formato UTF-8 y UTF-16, así como datos numéricos de 64 bits.
- Soporta funciones SQL definidas por el usuario (UDF).
- El código fuente es de dominio público y se encuentra muy bien documentado.

2.7 Metodología de Desarrollo Ágil.

Las metodologías de desarrollo ágil buscan elaborar software totalmente funcional en el tiempo o plazo establecido para el desarrollo del proyecto. Utilizan un proceso ágil, es decir que, si los requerimientos del software cambian en cualquier etapa en la que se encuentre el proyecto, el equipo debe adaptar el producto a estos cambios ya que la agilidad como tal es la respuesta efectiva al cambio. (Cevallos, 2015)

Existen diferentes metodologías de desarrollo ágil tales como: programación extrema XP, Scrum, Cristal entre otras, todas con el mismo objetivo, pero con diferentes formas de trabajo (Cevallos, 2015). (Gimson, 2012)

2.7.1 Programación Extrema (XP).

Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. (Letelier, 2006)

XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico (Gimson, 2012).

2.7.1.1 Valores de XP.

- **Comunicación.**- Comunicación cara a cara es la mejor forma de comunicación, entre los desarrolladores y el cliente (Gimson, 2012).
- **Simplicidad.**- La simplicidad ayuda a que los desarrolladores de software encuentren soluciones más simples a problemas, según el cliente lo estipula (Gimson, 2012).
- **Retroalimentación.**- La retroalimentación continua del cliente permite a los desarrolladores llevar y dirigir el proyecto en una dirección correcta hacia donde el cliente quiera (Gimson, 2012).
- **Valentía.**- Requiere que los desarrolladores vayan a la par con el cambio, porque sabemos que este cambio es inevitable, pero el estar preparado con una metodología ayuda a ese cambio (Gimson, 2012).

- Respeto. - El equipo debe trabajar como uno, sin hacer decisiones repentinas. Cada integrante del proyecto (cliente, desarrolladores, etc.) forman parte integral del equipo encargado de desarrollar software de calidad (Gimson, 2012).

2.7.1.2 Fases de la Programación Extrema.

Planificación del proyecto.

Historias de usuario: Definir las historias de usuario con el cliente. Las historias de usuario tienen la misma finalidad que los casos de uso (programacionextrema, 2012)

Release planning*: Después de tener ya definidas las historias de usuario es necesario crear un plan de publicaciones, en inglés "Release plan", donde se indiquen las historias de usuario que se crearán para cada versión del programa y las fechas en las que se publicarán estas versiones (programacionextrema, 2012).

Iteraciones Todo proyecto que siga la metodología X.P. se ha de dividir en iteraciones de aproximadamente 3 semanas de duración. Al comienzo de cada iteración los clientes deben seleccionar las historias de usuario definidas en el "Release planning" que serán implementadas (programacionextrema, 2012).

Velocidad del proyecto: La velocidad del proyecto es una medida que representa la rapidez con la que se desarrolla el proyecto; estimarla es muy sencillo, basta con contar el número de historias de usuario que se pueden implementar en una iteración; de esta forma, se sabrá el cupo de historias que se pueden desarrollar en las distintas iteraciones (programacionextrema, 2012).

Programación en pareja: La metodología X.P. aconseja la programación en parejas pues incrementa la productividad y la calidad del software desarrollado (programacionextrema, 2012).

Reuniones diarias. Es necesario que los desarrolladores se reúnan diariamente y expongan sus problemas, soluciones e ideas de forma conjunta (programacionextrema, 2012).

Fase: Diseño.

Diseños simples: Sugiere que hay que conseguir diseños simples y sencillos. Hay que procurar hacerlo todo lo menos complicado posible para conseguir un diseño fácilmente entendible que a la larga costará menos tiempo y esfuerzo desarrollar (programacionextrema, 2012).

Glosarios de términos: Usar glosarios de términos y una correcta especificación de los nombres de métodos y clases ayudará a comprender el diseño y facilitará sus posteriores ampliaciones y la reutilización del código (programacionextrema, 2012).

Riesgos: Si surgen problemas potenciales durante el diseño, X.P sugiere utilizar una pareja de desarrolladores para que investiguen y reduzcan al máximo el riesgo que supone ese problema (programacionextrema, 2012).

Funcionalidad extra: Nunca se debe añadir funcionalidad extra al programa, aunque se piense que en un futuro será utilizada. Sólo el 10% de la misma es utilizada, lo que implica que el desarrollo de funcionalidad extra es un desperdicio de tiempo y recursos (programacionextrema, 2012).

Re factorizar. Es mejorar y modificar la estructura y codificación de códigos ya creados sin alterar su funcionalidad (programacionextrema, 2012).

Tarjetas C.R.C. El uso de las tarjetas C.R.C (Class, Responsibilities and Collaboration) permiten al programador centrarse y apreciar el desarrollo orientado

a objetos olvidándose de los malos hábitos de la programación procedural clásica (programacionextrema, 2012).

Las tarjetas C.R.C representan objetos; la clase a la que pertenece el objeto se puede escribir en la parte de arriba de la tarjeta, en una columna a la izquierda se pueden escribir las responsabilidades u objetivos que debe cumplir el objeto y a la derecha, las clases que colaboran con cada responsabilidad (programacionextrema, 2012).

Codificación.

El cliente es una parte más del equipo de desarrollo; su presencia es indispensable en las distintas fases de X.P. A la hora de codificar una historia de usuario su presencia es aún más necesaria.

La codificación debe hacerse atendiendo a estándares de codificación ya creados. Crear test que prueben el funcionamiento de los distintos códigos implementados nos ayudará a desarrollar dicho código (programacionextrema, 2012).

Fase: Pruebas.

Uno de los pilares de la metodología X.P es el uso de test para comprobar el funcionamiento de los códigos que vayamos implementando (programacionextrema, 2012).

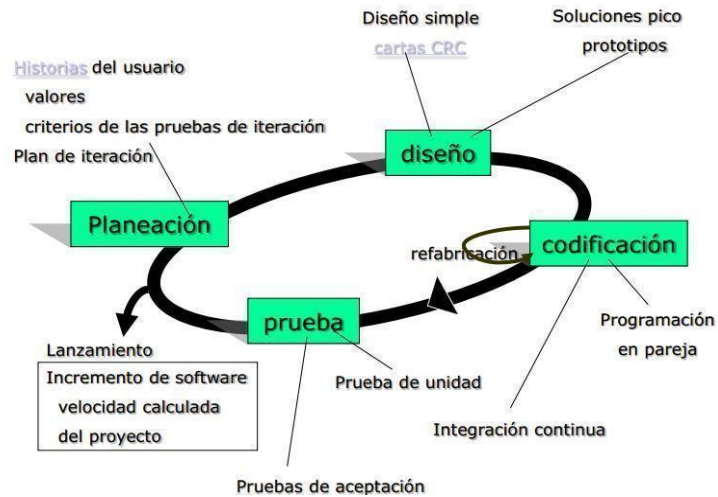


Gráfico N° 3 Ciclo de vida de metodología XP

2.8 IDE de Programación.

IDE es un entorno de programación que ha sido empaquetado como un programa de aplicación, o sea, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (proyectosdesoftwareblog, 2012).

Los IDEs pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes (proyectosdesoftwareblog, 2012).

2.8.1 Características IDE de Programación.

Los IDE ofrecen un marco de trabajo amigable para la mayoría de los lenguajes de programación tales como C++, Python, Java, C#, Delphi, Visual Basic, etc. En algunos lenguajes, un IDE puede funcionar como un sistema en tiempo de ejecución, en donde se

permite utilizar el lenguaje de programación en forma interactiva (proyectosdesoftwareblog, 2012).

Un IDE debe tener las siguientes características:

- Multiplataforma
- Soporte para diversos lenguajes de programación
- Integración con Sistemas de Control de Versiones
- Reconocimiento de Sintaxis
- Extensiones y Componentes para el IDE
- Integración con Framework populares
- Depurador
- Importar y Exportar proyectos
- Múltiples idiomas
- Manual de Usuarios y Ayuda

2.8.2 Android Studio (IDE).

Se va utilizar este IDE de desarrollo ya que está orientado y fue creado para programar en Android por lo cual nos va permitir tener un código más ordenado, estructurado mejora el diseño de interfaces.

Android Studio es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de aplicaciones para Android y se basa en IntelliJ IDEA (Google, 2018)

Android Studio ofrece aún más funciones que aumentan tu productividad durante la compilación de apps para Android, como las siguientes: (Google, 2018)

- Un sistema de compilación basado en Gradle flexible
- Un emulador rápido con varias funciones

- Un entorno unificado en el que puedes realizar desarrollos para todos los dispositivos Android
- Instant Run para aplicar cambios mientras tu app se ejecuta sin la necesidad de compilar un nuevo APK
- Integración de plantillas de código y GitHub para ayudarte a compilar funciones comunes de las apps e importar ejemplos de código
- Gran cantidad de herramientas y frameworks de prueba
- Herramientas Lint para detectar problemas de rendimiento, usabilidad, compatibilidad de versión, etc.
- Compatibilidad con C++ y NDK
- Soporte incorporado para Google Cloud Platform, lo que facilita la integración de Google Cloud Messaging y App Engine

3 Propuesta

3.1 Recopilación de información

Para este estudio se utiliza los siguientes métodos de investigación los cuales permiten determinar el estado actual del proceso y la factibilidad del desarrollo del proyecto en la empresa, así como la aceptación que tendrá el desarrollo como en los clientes.

3.1.1 Método Inductivo

Se aplica el método inductivo, con la finalidad de alcanzar conclusiones generales, se elaboró un banco de preguntas para realizar una entrevista, con el fin de obtener información sobre el proceso de recolección de datos y las especificaciones sobre el aplicativo que será desarrollado, el cual se observa en el **Anexo 2**.

3.1.2 Método Deductivo

Para el presente estudio se aplica el método deductivo, su objetivo es conocer la apertura y apoyo con la información que se requiere para el desarrollo del sistema de levantamiento de información de activos fijos conjuntamente conocer las buenas prácticas de producción y planes de pruebas, esta teoría es necesaria para el desarrollo del sistema. Además se va a utilizar el método exploratorio que nos ayudará a realizar evaluaciones continuas del sistema que se está empleando y también conocer acerca de posibles fallos que puedan provocarse en el aplicativo **Anexo 2**.

3.1.2 Observación

Se acude al sitio para obtener información de cómo realiza el levantamiento de información de activos fijos, se desarrolla una ficha de observación la misma que permite recopilar información relevante para el desarrollo del sistema, en esta parte se pudo

observar las necesidades de la empresa tales como, registro de información, listado de información recolectada, sincronización de catálogos, usuarios y perfiles, sincronizar información recolectada para visualizar en el sistema web de la empresa, este estudio la información recopilada se encuentra en la ficha de observación en el **Anexo 1**.

3.1.4 Entrevista

La entrevista fue realizada al gerente comercial la misma que permitió conocer el estado en el que se encuentra el proceso de levantamiento de información con sistema móvil que la empresa posee en la actualidad, Agustín Carrasco manifiesta que estaría dispuesto a desarrollar un software para dispositivos móviles con sistema operativo Android, que permite el realizar el levantamiento de información de activos fijos ya que actualmente este proceso se realiza con el sistema móvil actual, el cual ha perdido aceptación en el mercado debido a que no es compatible con los dispositivos móviles que poseen los clientes, además comentó que la empresa se encuentra en proceso de innovación tecnológica y de procesos, para lo cual una de sus metas es contar con un sistema móvil que pueda ser instalado en la mayoría de dispositivos que están en el mercado, también se pudo determinar los roles que ejerce cada empleado, información recopilada que servirá de relevancia en el desarrollo del sistema.

En la tabla se muestra los perfiles y funciones que realizan, de las personas que trabajan en el proceso de levantamiento de información.

Tabla N° 1 Roles y perfiles en el proceso de levantamiento de información.

Función	Actividad
Jefe Proyecto	Persona encargada de la gestión de catálogos, usuarios y perfiles dentro del sistema web.
Perito	Persona encargada de la recolección de información y sincronizar la información con el sistema web de la empresa.
Supervisor	Persona encargada de validar la información en el sistema web.

De manera general en base a la entrevista se recopila la siguiente información que será de gran ayuda para el desarrollo del sistema de levantamiento de información, de lo expuesto puede ver en el **Anexo 2**

3.2 Diagramas de procesos

3.2.1 Situación actual

Los diagramas fueron desarrollados en base a la información recopilada en la entrevista, la misma que fue de uso para detallar en los siguientes diagramas de proceso la situación actual del proceso de levantamiento de activos utilizando el aplicativo que posee la empresa CAYMANSYSTEMS.

En el gráfico N° 4 muestra el proceso de ingreso al aplicativo móvil de recolección de información, en la cual se valida los datos del usuario con el servicio web.

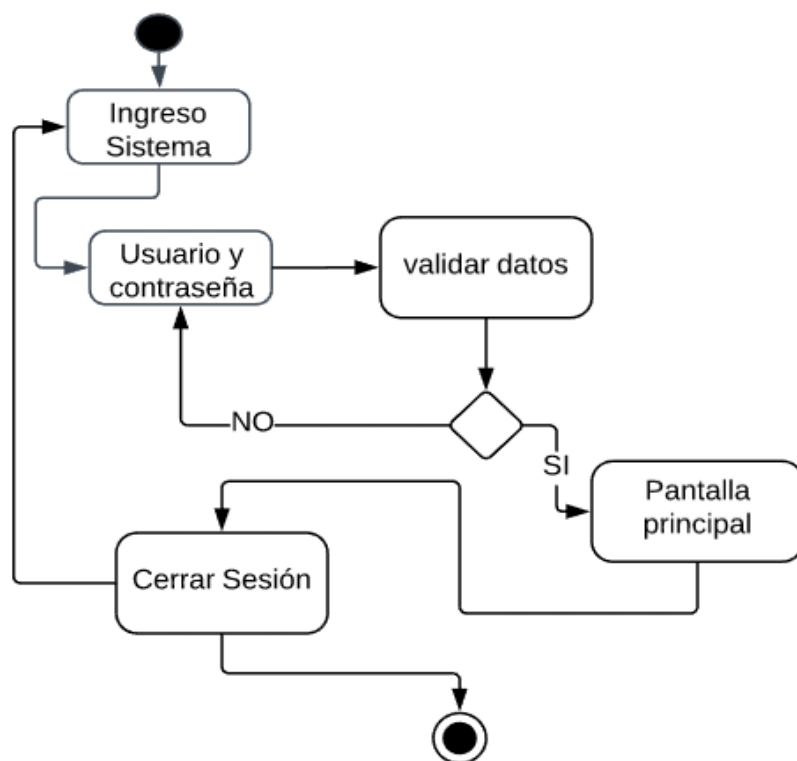


Gráfico N° 4 Diagrama de flujo ingreso al sistema.

Autor: Fabián Aguirre

En el gráfico N° 5 se muestra el proceso de descarga de datos catálogos y activos, los catálogos son parametrizados desde el sistema web para luego ser descargados desde el aplicativo móvil, para lo cual el aplicativo debe tener conexión con el servidor web, si se necesita realizar una conciliación en campo se procede a parametrizar la información de la base de datos de activos fijos del cliente para poder descargar esta información en el dispositivo, toda la información descargada es ingresada a la base de datos del dispositivo.

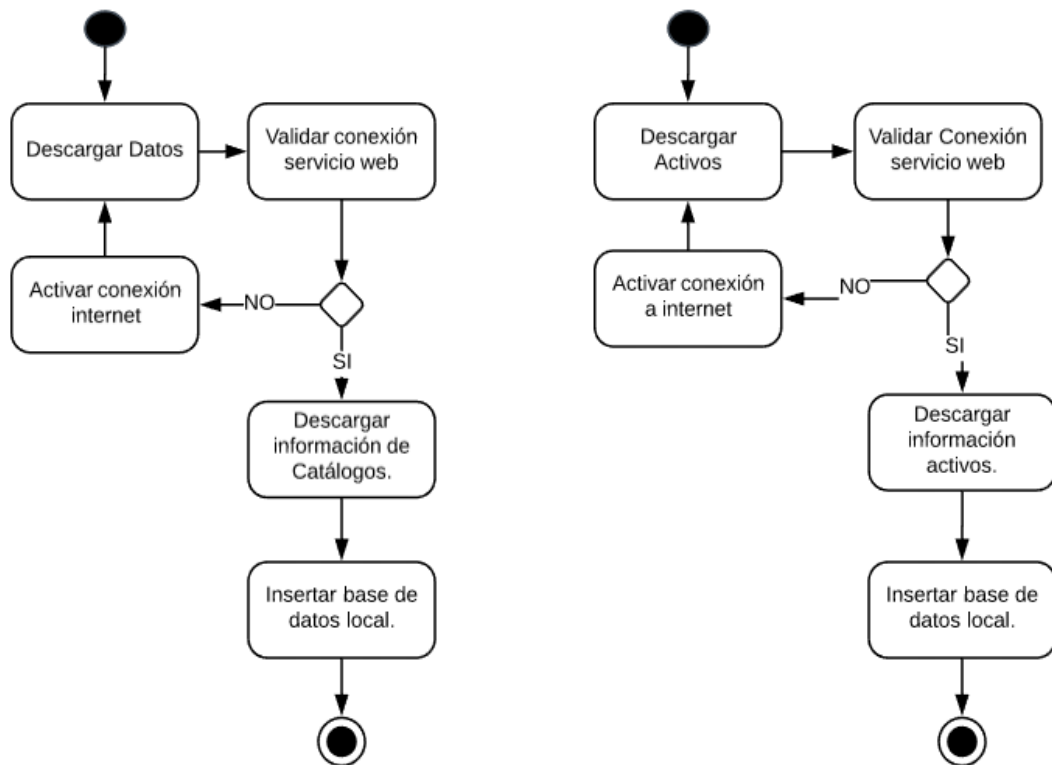


Gráfico N° 5 Diagrama de flujo descarga de activos para conciliación en campo y descarga de datos (catálogos de parametrización del sistema).

Autor: Fabián Aguirre.

En el gráfico N° 6 se muestra el ingreso de información en el aplicativo móvil para lo cual antes de ingresar a la pantalla de registro de información válida que esté descargada la información de catálogos para presentar el formulario antes de guarda la información se valida los campos obligatorios.

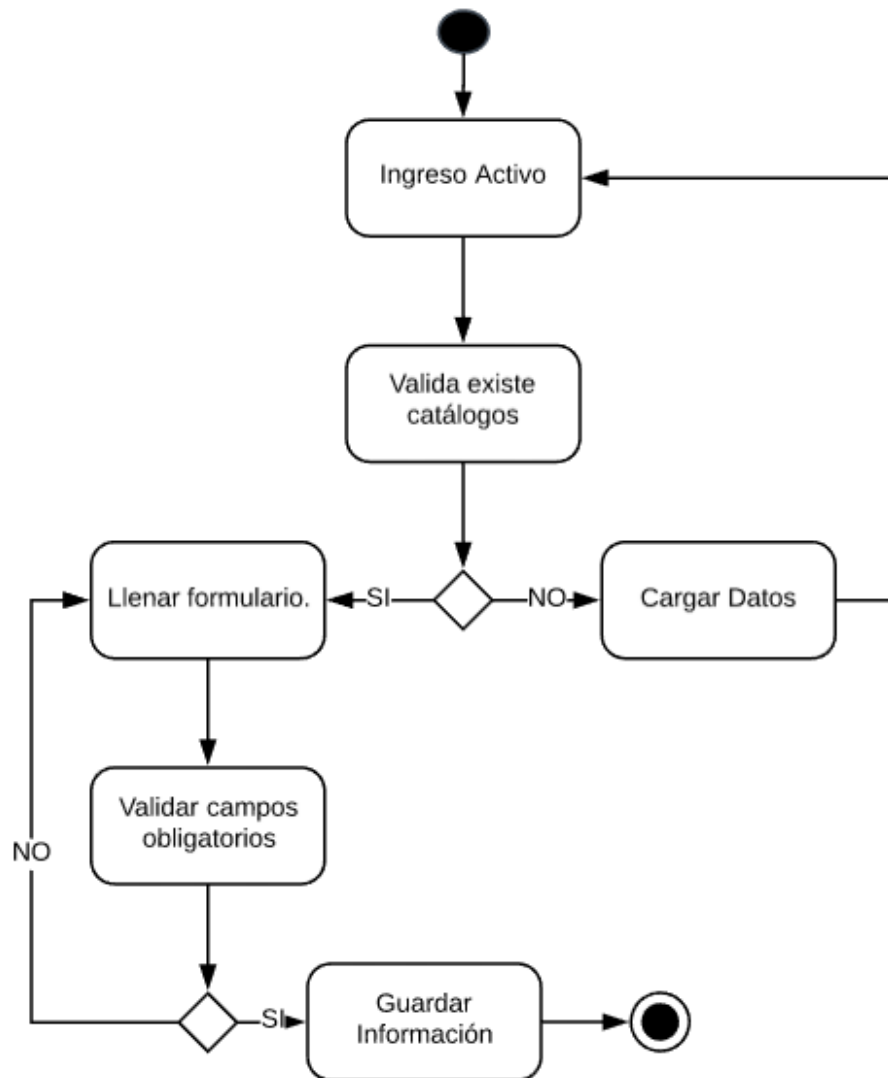


Gráfico N° 6 Diagrama de flujo ingreso de información activo fijos.

El gráfico No. 7 muestra el proceso de modificación de información para lo cual el primer paso es buscar el activo, si existe se presenta el formulario con la información de la activo a modificar se valida la los campos obligatorios antes de guardar se debe tener en cuenta que este proceso se realiza sobre la base de datos local.

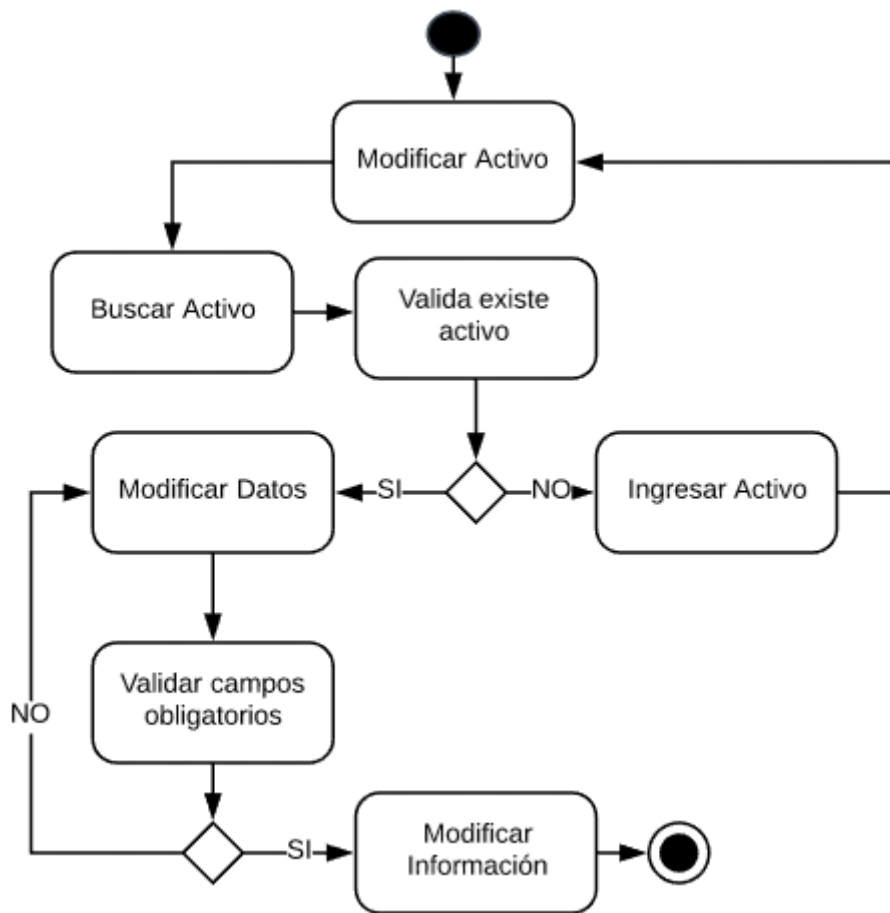


Gráfico N° 7 Diagrama de flujo modificación de información de activos fijos.

En el gráfico N° 8 muestra el proceso de eliminación de información para lo cual el primer paso es buscar el activo, si existe se presenta un mensaje de confirmación se debe tener en cuenta que este proceso se realiza sobre la base de datos local.

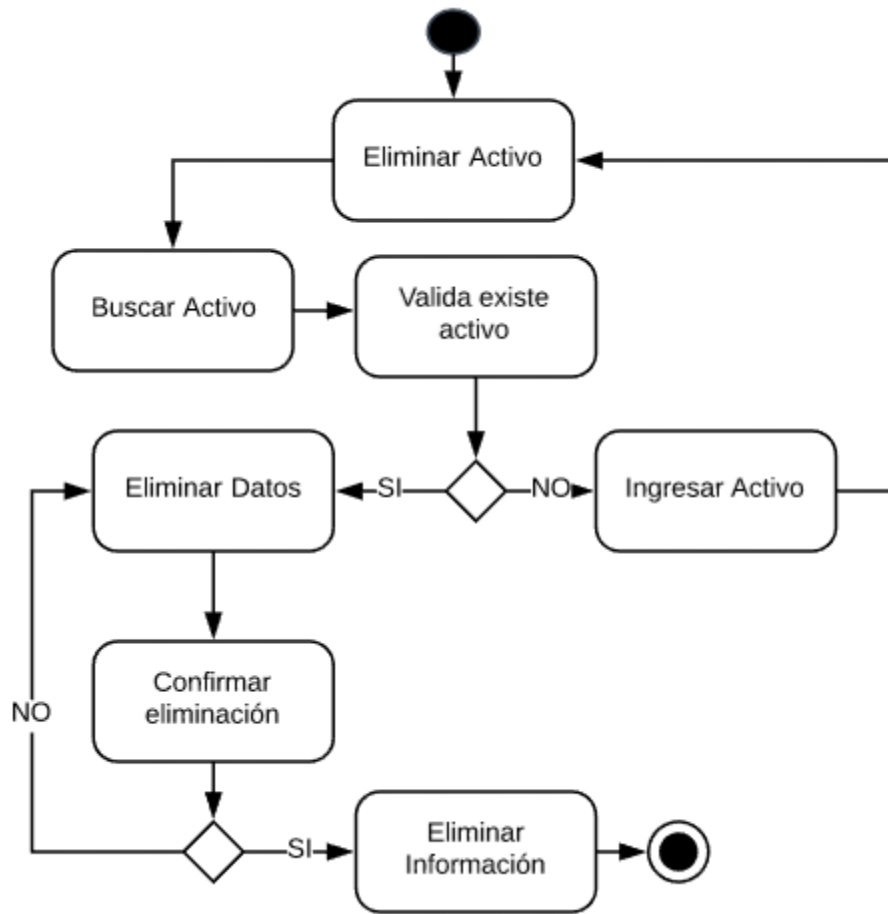


Gráfico N° 8 Diagrama de flujo eliminación de información de activos fijos.

En el gráfico N° 9, muestra el proceso de sincronizar información recolectada, para esto se valida la conexión al servicio web, una vez enviada la información y el servicio web devuelve un mensaje de confirmación de inserción del registro se procede a eliminar la información de la base de datos local.

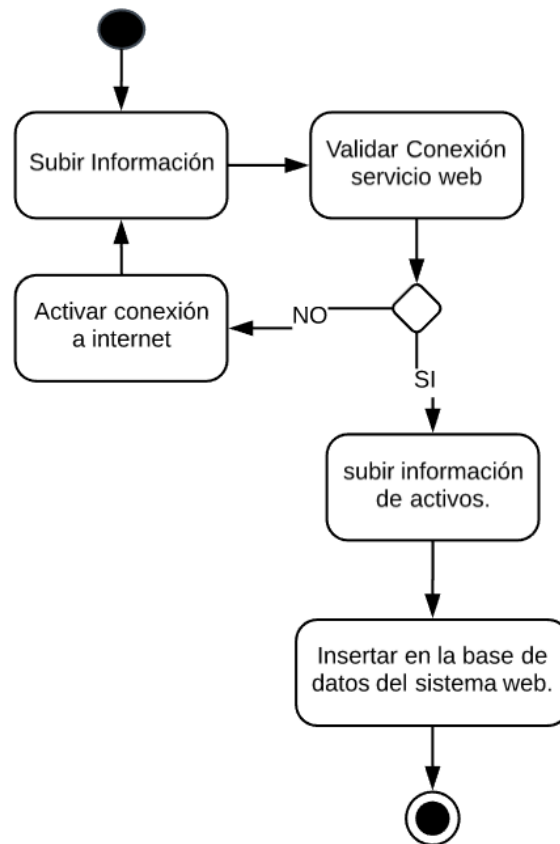


Gráfico N° 9 Diagrama de flujo sincronizar información recolectados de activos fijos.

En el gráfico N° 10 se muestra una funcionalidad que se va agregar a en el proceso de levantamiento de información el cual va permitir el ingreso de información de activos fijos de forma masiva o por lotes, para esto se podrá agregar el número de activos a ingresar según sea el caso pero no podrán ser menos de uno.

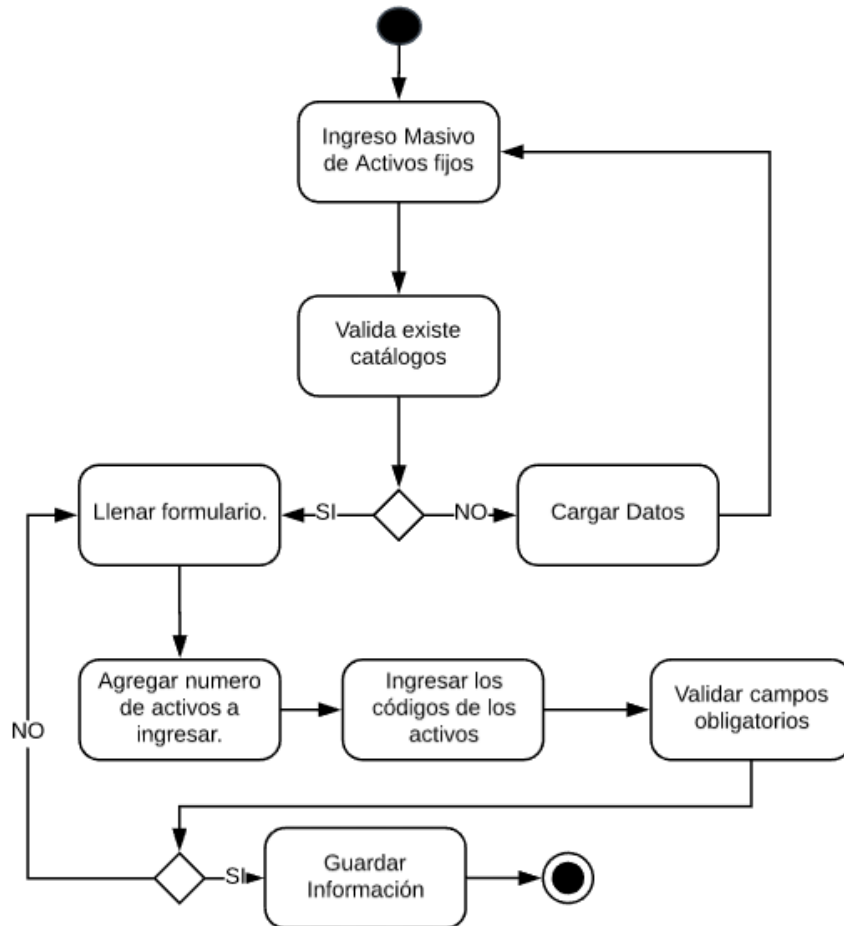


Gráfico N° 10 Diagrama de flujo ingreso masivo de activos fijos.

3.3 Especificación de requerimientos

En esta parte se desarrollará la fase 1 (Planificación del Proyecto) como lo indica la metodología XP, los requerimientos han sido identificados luego de la entrevista realizada a los involucrados en el proceso de levantamiento de información de activos fijos.

3.3.1 Ámbito del software

El aplicativo de levantamiento de activos fijos, será una herramienta que permitirá a la empresa Cayman Systems el levantamiento de información, para al final del día sincronizar esta información con el sistema web CAYMAN ACTIVO V.4 RFID.

Para cumplir con esta meta el aplicativo contará con la siguiente funcionalidad:

- El usuario administrador del sistema CAYMAN ACTIVO V.4 RFID será el encargado de gestionar usuarios, perfiles y parametrizar catálogos antes de iniciar el levantamiento de información.
- El usuario con rol de perito será el encargado de utilizar el aplicativo de levantamiento de información en sus dispositivo, se le asignará un usuario y contraseña para poder ingresar, antes de iniciar el día de trabajo tendrá que sincronizar los catálogos al realizar esta acción el aplicativo móvil creará una base de datos local en el dispositivo para ser utilizada en el ingreso de información.
- Procederá con el levantamiento de información para este proceso, podría necesitar una nueva opción en los catálogos, el usuario pedirá al administrador del sistema web que lo haga para luego sincronizar los catálogos.

- En el sistema móvil no se podrá ingresar nuevos catálogos, por problemas de duplicidad de la información ya que varios peritos podrían ingresar el mismo catálogo al mismo tiempo, además con esto evitaremos que los peritos clasifiquen de manera errónea los bienes en sus respectivas cuentas.
- Al final del día el usuario perito procederá con la sincronización de la información recolectada, si existe un activo ya ingresado en el sistema web con el mismo código el web services enviara un mensaje de que el activo ya está registrado, si ocurre un error al enviar la información el aplicativo no borrarla la información hasta que la respuesta del web services sea positiva. una vez que se reciba el mensaje de confirmación del sistema web el sistema móvil borrarla toda la información de los activos para no sobrepasar el límite de almacenamiento de la base de datos SQLite

Una vez realizada la sincronización el perito no podrá modificar la información en el sistema web, ya que esta información pasa al proceso de validación y los usuarios validadores serán los únicos que podrán modificar datos de los activos sincronizados.

Los usuarios peritos podrán modificar, eliminar datos de los activos desde el dispositivo móvil siempre y cuando estos estén en la base de datos local del dispositivo una vez sincronizados ya no podrá modificar información del activo, para realizar modificación se deberá pedir a los usuarios validadores que realicen la modificación desde el sistema web.

3.3.2 Funciones del producto

Las funciones de producto están basadas en las necesidades de la empresa CAYMASYSTEMS convirtiéndolas de manera técnica a los requerimientos funcionales del usuario y agrupando cada uno de ellos en las características correspondientes con los que contará el sistema, los requerimientos se trabajaron juntamente con la parte Jefe Área Técnica, personas que manejan el proceso de levantamiento de información de activos obteniendo la siguiente representación:

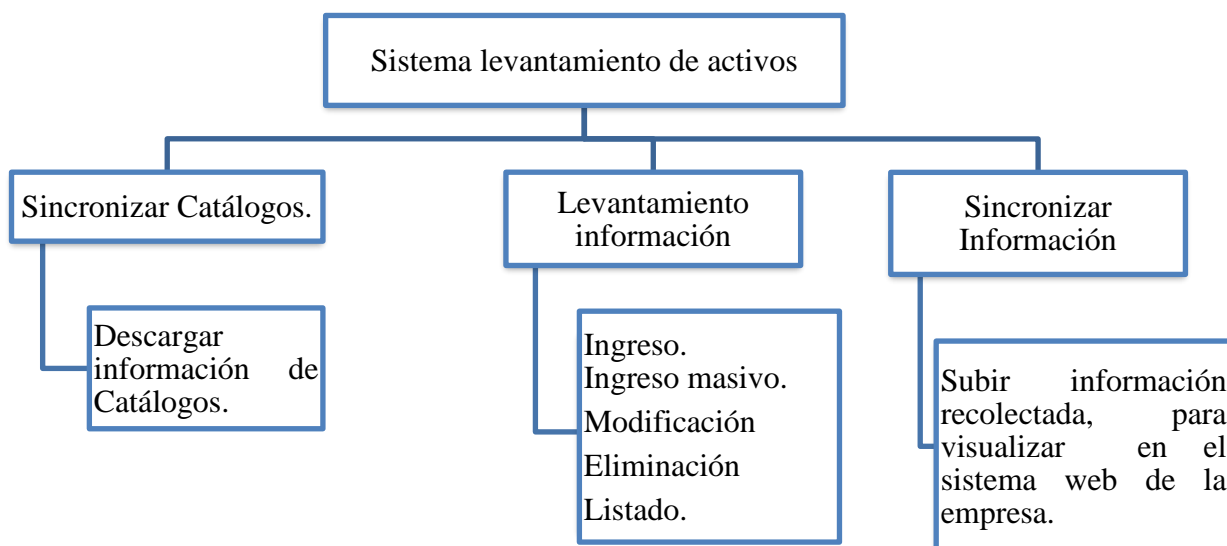


Gráfico N° 11 Funciones del aplicativo de levantamiento de información.

3.3.3 Características de los usuarios del sistema

Usuarios operacionales

Tabla N° 2 Tipo de usuario Perito

Usuarios operacionales	
Tipo de usuario	Perito
Formación	Formación en conocimientos en, atención al cliente.
Habilidades	Manejo de teléfonos móviles con sistemas operativo Android.
Actividades	Realizar recolección de información, sincronización de información.

3.3.4 Restricciones

El sistema se lo va realizar usando el lenguaje de programación Java (Android Studio) y la base de datos SQLite, para el funcionamiento de sistema se necesitará un dispositivo móvil con sistema operativo Android con la versión 5.0(Lollipop) o superior, la gestión de usuarios se realizará desde el sistema web que posee la empresa.

3.3.5 Requisitos

Funcionales

Los requerimientos funcionales se establecen en las historias de usuario descritas en este proyecto de investigación

RF1.- Permite al usuario registrar, modificar, buscar y eliminar información de la Base de Datos.

RF2.- Permitirá la gestión de activos fijos en el proceso de levantamiento de información.

RF3.- Permite sincronizar catálogos, usuarios, perfiles desde el sistema web.

RF4.-Permite generar listados de activos por código de barras, serie, fecha de levantamiento.

RF5.- Permite sincronizar información recolectada con el sistema web.

RF6.- Permite realizar lectura de códigos de barras mediante escáner del dispositivo móvil.

No Funcionales

NRF1.- El dispositivo debe tener conexión a Internet para sincronizar la información recolectada.

NRF2.- El tiempo máximo de respuesta de la aplicación tiene que ser de un segundo en el 95% de las operaciones y en sincronizar información no mayor a 3 minutos.

NRF 3.- El aplicativo soporta tantas entradas a la base de datos como se desee en el día de trabajo, además de nuevas funcionalidades.

NRF 4.- El dispositivo debe tener una batería cargada al 100%, para utilizar el aplicativo por 8 horas seguidas.

4 Implementación

4.1 Diseño general

El sistema de levantamiento de activos fijos para la empresa CAYMANSYSTEMS será implementado mediante el modelo MVC (Modelo Vista Controlador), para ello se utilizará Java(Android Studio) el cual está basado en las mejores prácticas de estructura de código para facilitar su implementación y obtener un mejor rendimiento.

MVC separa en capas definidas el desarrollo de una aplicación, estas son:

- El modelo encargado de la lógica del negocio y la persistencia de los datos
- Las vistas que son los responsables de mostrar al usuario el resultado que obtienen del modelo a través del controlador.
- El controlador que es el encargado de gestionar las peticiones del usuario procesarlas invocando al modelo de mostrarlas al usuario a través de las vistas.

Una vez analizado el tipo de diseño a implementar se aplicará el proceso de desarrollo XP.

4.1.1 Proceso de Desarrollo de XP

Historias de Usuario

Las historias de usuarios se realizan en base a la entrevista (anexo 1), ejecutada al personal que maneja el proceso de levantamiento de información de activos fijos que maneja la empresa “CAYMANSYSTEMS”, para lo cual se detalla a continuación:

Tabla N° 3 Historia de Usuario 1- Acceso al sistema

Historia de Usuario	
Número:01	Usuario: Perito
Nombre historia: Acceso al sistema	
Prioridad del negocio: Alta	Riesgo en desarrollo: Alta
Días estimados: 2	Iteración asignada: 1
Programador responsable: Fabián Aguirre	
Descripción: En la pantalla de ingreso, muestra un formulario donde se puede ingresar usuario y contraseña, validar información ingresada mediante un servicio web.	
Observaciones: Es necesario informar con un mensaje de alerta si el usuario coincide o no con la información existente en la base de datos.	

4.1.1.1 Iteración 1

Tabla N° 4 Criterio de pruebas de aceptación 1

HISTORIA DE USUARIO: Acceso al Sistema.

Prueba de aceptación:1

CRITERIOS:

- La interfaz presentada le pareció adecuada
 - Al llenar el formulario debe validar usuario y contraseña obligatoria.
 - Presentar mensaje de contraseña o usuario incorrecto.
 - Al ingresar información correcta del usuario mostrar la pantalla principal de la aplicación
 - Finalmente aprobó el perito la parte del inicio de sesión luego de haber realizado las pruebas respectivas
-

Tabla N° 5 Historia de Usuario 2- Sincronización de catálogos

Historia de Usuario

Número:02 **Usuario:** Perito

Nombre historia: Sincronización de catálogos

Prioridad del negocio: Alta

Riesgo en desarrollo: Bajo

Días estimados: 1

Iteración asignada: 2

Programador responsable: Fabián Aguirre

Descripción:

En el menú principal debe contener una opción de Descarga de datos, que permitirá la descarga de catálogos utilizando un servicio web, este proceso no re direccionará a otra pantalla para no perder información ingresada en las pantallas de ingreso de información de activos, una vez finalizado este proceso se podrá dar clic en ok en el mensaje de finalización.

Observaciones:

Luego de la primera descarga de catálogos, solo se descargan catálogos nuevos o modificados.

4.1.1.2 Iteración 2

Tabla N° 6 Criterio de pruebas de aceptación 2

HISTORIA DE USUARIO: SINCRONIZACIÓN DE CATÁLOGOS.

Prueba de aceptación:2

CRITERIOS:

- Verificar opción de Descarga de datos en el menú principal.
 - Validar conexión al servicio web.
 - Ejecución del proceso sin re direccionar a otra pantalla.
 - Mensaje de avance del proceso.
 - Presentación de mensaje de finalización y permitir dar clic en ok para cerrar el mensaje.
 - El perito aprueba el proceso.
-

Tabla N° 7 Historia de Usuario 3- Descargar Activos

Historia de Usuario

Número:03

Usuario: Perito

Nombre historia: Descargar Activos

Prioridad del negocio: Alta

Riesgo en desarrollo: Alta

Días estimados: 2

Iteración asignada: 3

Programador responsable: Fabián Aguirre

Descripción:

En el menú principal está la opción descargar activos la cual, permite que se descargue todos los activos asignados al dispositivo mediante servicios web.

Los activos serán insertados en la base de datos local del dispositivo para poder realizar una conciliación en campo con esta información.

Observaciones:

Esta información sólo se cargará si el cliente de la empresa tiene una base de datos de activos para cargar en los dispositivos.

4.1.1.3 Iteración 3

Tabla N° 8 Criterio de pruebas de aceptación 3

HISTORIA DE USUARIO: Descarga de Activos.

Prueba de aceptación:3**CRITERIOS:**

- Validar existencia de la opción descargar activos en el menú principal.
 - Validar conexión al servicio web para descargar información.
 - Mensaje de proceso de descarga de activos.
 - Visualizar activos cargados en el reporte de activos.
 - Finalmente el perito aprobó el proceso luego de las pruebas respectivas
-

Tabla N° 9 Historia de usuario diseño pantalla de levantamiento de activo fijo

Historia de Usuario

Número:04

Usuario: Perito

Nombre historia: Diseño pantalla levantamiento de activo fijo

Prioridad del negocio: Alta

Riesgo en desarrollo: Alta

Días estimados: 2

Iteración asignada: 4

Programador responsable: Fabián Aguirre

Descripción: La pantalla va estar distribuida por secciones ubicación geográfica (Ubicación Geográfica 1, Ubicación Geográfica 2, Ubicación Geográfica 3, Piso/Nivel todos estos van hacer combos de selección)

- Siguiete sección es ubicación orgánica (Ubicación Orgánica 1, Ubicación Orgánica 2, Ubicación Orgánica 3, Custodio todos estos van hacer combos de selección).
- Siguiete sección codificación (Tipo Activo combo de selección) (código barras, código barras padre, código secundario todas cajas de texto).
- Siguiete sección es Clasificación (Grupo/Cuenta, Subgrupo, Descripción todos estos van hacer combos de selección).
- Siguiete sección es características (marca, modelo (combo de selección), serie cajas de texto, estructura, componente (combo de selección)).
- Siguiete sección de estado (estado, color, año de fabricación, mes, tipo de etiqueta).
- Siguiete sección Fotografía (botón para capturar imagen).
- Siguiete sección observaciones (2 cajas de texto múltiple línea).

Siguiete sección información técnica (cajas de texto todas las características por grupo/cuenta).

Observaciones:

Se debe tomar en cuenta los campos obligatorios para diseñar la pantalla.

4.1.1.4 Iteración 4

Tabla N° 10 Criterio de pruebas de aceptación 4

HISTORIA DE USUARIO: Diseño pantalla Levantamiento de activo fijo.

Prueba de aceptación:4

CRITERIOS:

- La interfaz presentada le pareció adecuada
 - Al llenar el formulario debe validar información obligatoria.
 - Finalmente aprobó el jefe de proyectos el diseño de la pantalla levantamiento de información.
-

Tabla N° 11 Historia de usuario ingreso de activo fijo.

Historia de Usuario

Número:05

Usuario: Perito

Nombre historia: Ingreso de activo fijo

Prioridad del negocio: Alta

Riesgo en desarrollo: Alta

Días estimados: 2

Iteración asignada: 5

Programador responsable: Fabián Aguirre

Descripción: En la pantalla principal el perito selecciona la opción Activo.

En la pantalla procederá a llenar y seleccionar toda la información que se va recolectando en el proceso de levantamiento.

Si en el proceso de levantamiento aparece una nueva opción de catálogos que no fue tomando en cuenta con en el proceso de parametrización, se procederá a comunicar al jefe de proyecto para que la cree, una vez creada se podrá seleccionar la opción cargar datos en el menú principal para actualizar los catálogos.

Una vez ingresada toda esta información se procederá a guardar la información en la base de datos local.

Observaciones:

En el caso de que no llene todos los campos obligatorios no le permitirá guardar la información.

4.1.1.5 Iteración 5

Tabla N° 12 Criterio de pruebas de aceptación 5

HISTORIA DE USUARIO: INGRESO DE ACTIVO FIJO

Prueba de aceptación:5

CRITERIOS:

- El perito ingresa los activos.
 - Registra en cada uno de los campos obligatorios que el sistema solicita.
 - El perito intenta grabar la información de los activos sin completar lo cual el sistema no permitió.
 - Nuevamente tuvo que completar la información requerida para el proceso de grabar la información.
 - Historia de usuario fue aceptada por el jefe de proyecto.
-

Tabla N° 13 Historia de usuario modificación y eliminación en la base de datos local de activos fijos.

Historia de Usuario

Número:06

Usuario: Perito

Nombre historia: Gestión de activo fijo

Prioridad del negocio: Alta

Riesgo en desarrollo: Alta

Días estimados: 2

Iteración asignada: 6

Programador responsable: Fabián Aguirre

Descripción: En el menú principal está la opción listado en la pantalla se muestra al usuario una caja de texto con un botón para buscar.

Si el activo existe en la base de datos se carga en una lista, caso contrario se presentara un mensaje de que el activo no está ingresado en la base de datos. Para modificar un activo se puede seleccionar el botón ver de la lista el cual va re direccionar a una pantalla cargada con toda la información del activo se validaron los mismos campos que en el proceso de ingreso se procederá actualizar la información del activo.

Para eliminar el activo se procede a seleccionar en la lista el botón Eliminar al dar clic se va eliminar el activo de la base de datos local.

Observaciones:

En el caso de que no llene todos los campos obligatorios no le permitirá modificar la información.

4.1.1.6 Iteración 6

Tabla N° 14 Criterio de pruebas de aceptación 6

HISTORIA DE USUARIO: Gestión de activo fijo

Prueba de aceptación:6

CRITERIOS:

- El perito busca los activos, selecciona el filtro de búsqueda.
 - Selecciona el activo a modificar.
 - El perito intenta modificar la información de los activos sin completar los campos obligatorios lo cual el aplicativo no permitió.
 - Nuevamente tuvo que completar la información requerida para el proceso de modificación de la información.
 - Selecciona el activo a eliminar, pregunta si está de acuerdo con eliminar el activo.
 - Todo este procesado va trabajar de forma local, consultas, modificaciones, eliminaciones a la base de datos local.
 - Historia de usuario fue aceptada por el jefe de proyecto.
-

Tabla N° 15 Historia de usuario sincronizar información sistema cayman activo web

Historia de Usuario

Número:07

Usuario: Perito

Nombre historia: Sincronizar información sistema cayman activo web

Prioridad del negocio: Alta

Riesgo en desarrollo: Alta

Días estimados: 2

Iteración asignada: 7

Programador responsable: Fabián Aguirre

Descripción: En la pantalla principal el perito selecciona la opción Enviar Activos.

En la pantalla se presenta un mensaje con el avance del proceso de sincronización de información.

Al subir la información se validará la conexión a internet del dispositivo,

Si el dispositivo tiene conexión a internet se procederá a subir todos los activos levantados al sistema web mediante el uso de un servicio web, si el servicio nos devuelve que el proceso fue terminado con éxito se procederá a borrar los datos de los activos en la base de datos local.

Observaciones:

Toda la información subida tendrá que ser validada en el sistema web.

4.1.1.7 Iteración 7.

Tabla N° 16 Criterio de pruebas de aceptación 7

HISTORIA DE USUARIO: Sincronizar información sistema cayman activo web

Prueba de aceptación:7

CRITERIOS:

- El perito Sincroniza información.
 - Valida el dispositivo tiene conexión a internet se procederá a subir todos los activos levantados al sistema web mediante el uso de un servicio web.
 - Jefe de proyecto válido que la información esté en el sistema activo web.
 - Historia de usuario fue aceptada por el jefe de proyecto.
-

4.1.2 Tarjetas CRC

Las tarjetas CRC sirven para la representación de sistemas orientados a objetos u constan de los siguientes partes:

- Nombre de la clase
- Responsabilidades de las clases que se basa en:
 - Atributos
 - Métodos
- Colaboradores: son aquellas clases con las cuales se van a trabajar conjuntamente

A continuación se presentarán las tarjetas CRC para el sistema de levantamiento de activos fijos para la empresa CAYMANSYSTEMS.

Tabla N° 17 Tarjeta CRC 1 Ubicación Geográfica.

Ubicación Geográfica.	
Responsabilidad	Colaboración
<ul style="list-style-type: none">● Guardar información de las ubicaciones geográficas.● Verificar información.● Listar información.	

Tabla N° 18 Tarjeta CRC 2 Ubicación Orgánica.

Ubicación Orgánica.	
Responsabilidad	Colaboración
<ul style="list-style-type: none">● Guardar información de las ubicaciones orgánicas.● Verificar información por Ubicación Geográfica.● Listar información.	Ubicación Geográfica.

Tabla N° 19 Tarjeta CRC 3 Grupo Cuenta Contable.

Grupo Cuenta Contable.	
Responsabilidad	Colaboración
<ul style="list-style-type: none"> ● Guardar información del grupo cuenta contable. ● Verificar información. ● Listar información. 	

Tabla N° 20 Tarjeta CRC 4 Marca.

Marca.	
Responsabilidad	Colaboración
<ul style="list-style-type: none"> ● Guardar información marca. ● Verificar información. ● Listar información. 	

Tabla N° 21 Tarjeta CRC 5 Modelo.

Modelo.	
Responsabilidad	Colaboración
<ul style="list-style-type: none"> ● Guardar información modelo. ● Verificar información por marca ● Listar información. 	Marca

Tabla N° 22 Tarjeta CRC 6 Estructura.

Estructura.	
Responsabilidad	Colaboración
<ul style="list-style-type: none">● Guardar información Estructura.● Verificar información● Listar información.	

Tabla N° 23 Tarjeta CRC 7 Componente.

Componente.	
Responsabilidad	Colaboración
<ul style="list-style-type: none">● Guardar información componente.● Verificar información● Listar información.	

Tabla N° 24 Tarjeta CRC 8 Estado.

Estado.	
Responsabilidad	Colaboración
<ul style="list-style-type: none">● Guardar información Estado.● Verificar información● Listar información.	

Tabla N° 25 Tarjeta CRC 9 Color.

Color.	
Responsabilidad	Colaboración
<ul style="list-style-type: none"> ● Guardar información color. ● Verificar información ● Listar información. 	

Tabla N° 26 Tarjeta CRC 10 Tipo etiqueta.

Tipo etiqueta.	
Responsabilidad	Colaboración
<ul style="list-style-type: none"> ● Guardar información tipo etiqueta. ● Verificar información ● Listar información. 	

Tabla N° 27 Tarjeta CRC 11 Información Técnica.

Información Técnica.	
Responsabilidad	Colaboración
<ul style="list-style-type: none"> ● Guardar información técnica. ● Verificar información grupo cuenta. ● Listar información. 	Grupo cuenta contable.

Tabla N° 28 Tarjeta CRC 12 Activo.

Activo.	
Responsabilidad	Colaboración
<ul style="list-style-type: none"> ● Guardar información activo. ● Verificar información ubicación geográfica. ● Verificar ubicación orgánica ● Verificar grupo cuenta contable ● Verificar marca. ● Verificar modelo. ● Verificar estructura. ● Verificar componente. ● Verificar estado. ● Verificar color. ● Verificar tipo etiqueta. ● Listar información. ● Modificar Activo. ● Eliminar Activo. 	<p>Ubicación geográfica.</p> <p>Ubicación orgánica</p> <p>Grupo cuenta contable</p> <p>Marca</p> <p>Modelo</p> <p>Estructura</p> <p>Componente.</p> <p>Estado.</p> <p>Color.</p> <p>Tipo Etiqueta.</p>

4.1.3 Plan de entregas

Basándonos en las historias de usuario definidas para el desarrollo del sistema web, se ha elaborado el siguiente plan de entrega, el cual muestra las historias de usuario que se llevarán a cabo en cada iteración. Para este plan de entrega se ha tomado en cuenta la prioridad y el esfuerzo de cada historia de usuario.

Tabla N° 29 Plan de entrega.

Historias	Iteración	Prioridad	Esfuerzo	Fecha	Fecha
				Inicio	Final
Historia 1	1	Alta	2	01/01/18	01/01/18
Historia 2	3	Alta	3	02/01/18	08/01/18
Historia 3	2	Alta	2	09/01/18	09/01/18
Historia 4	3	Alta	2	10/01/18	18/01/18
Historia 5	3	Alta	3	19/01/18	20/01/18
Historia 6	3	Alta	3	21/01/18	31/01/18
Historia 7	4	Alta	3	01/02/18	05/02/18

4.1.4 Esquema de la base de datos

El esquema de base de datos representa las principales tablas del sistema de levantamiento de activos fijos junto con las tablas relacionadas como Activo, Grupo, Marca, modelo las mismas que están relacionadas de acuerdo a la necesidad del sistema a ser implementado.

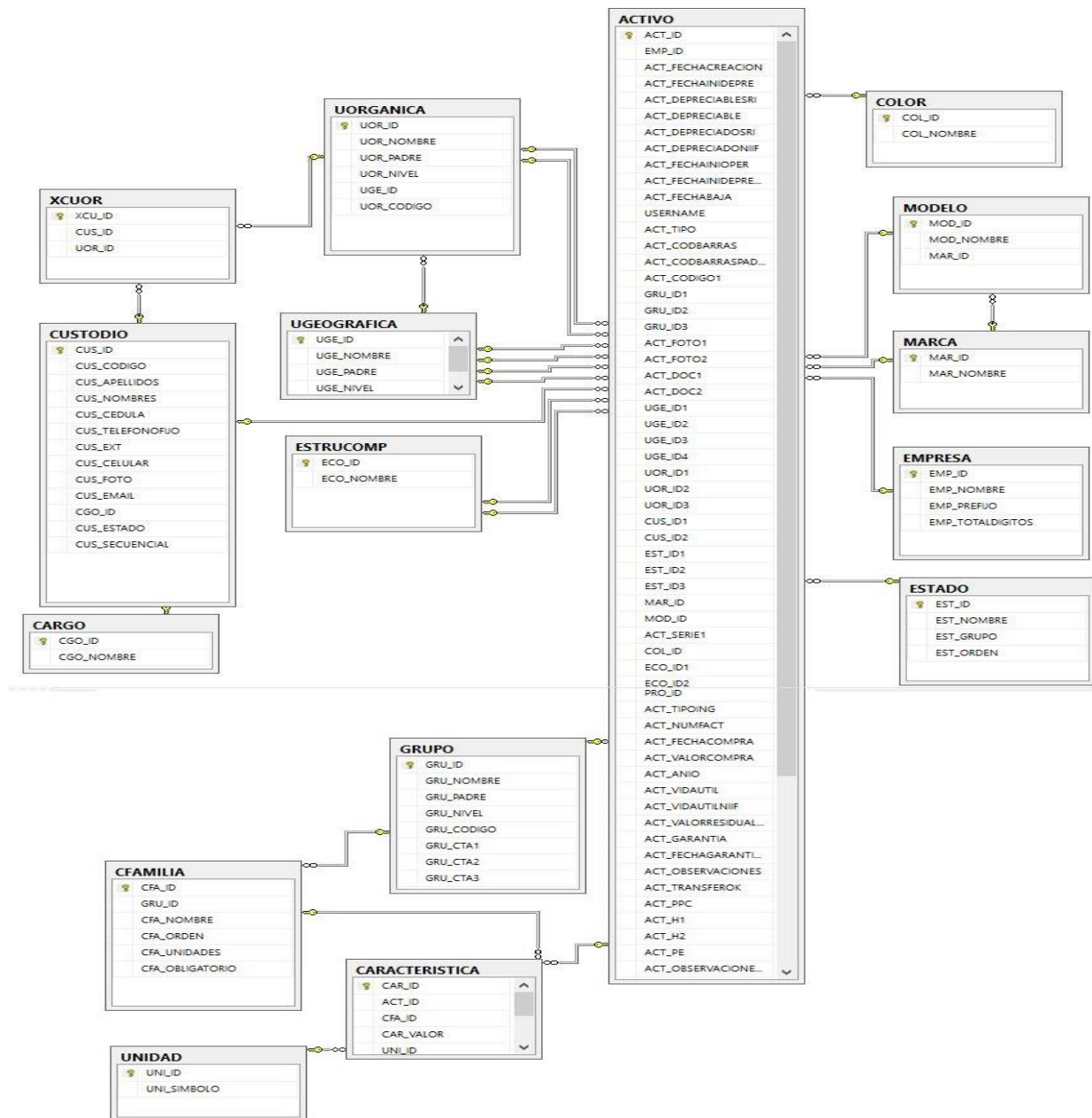


Gráfico N° 12 Diagrama de la base de datos.

4.1.5 Diagrama de la arquitectura del sistema

En el gráfico N° 13 se muestra el diagrama del aplicativo el cual está instalado en un dispositivo móvil con sistema operativo Android 5.1.1 o superior y va guardar información en un archivo de base de datos SQLite alojado en la memoria interna del dispositivo, este se conecta mediante internet a un web services el publicado en un servidor aplicaciones IIS.

El web services permite la carga y descarga de información desde una base de datos SQL Server hacia los dispositivos móviles mediante la utilización del aplicativo de levantamiento de activos.

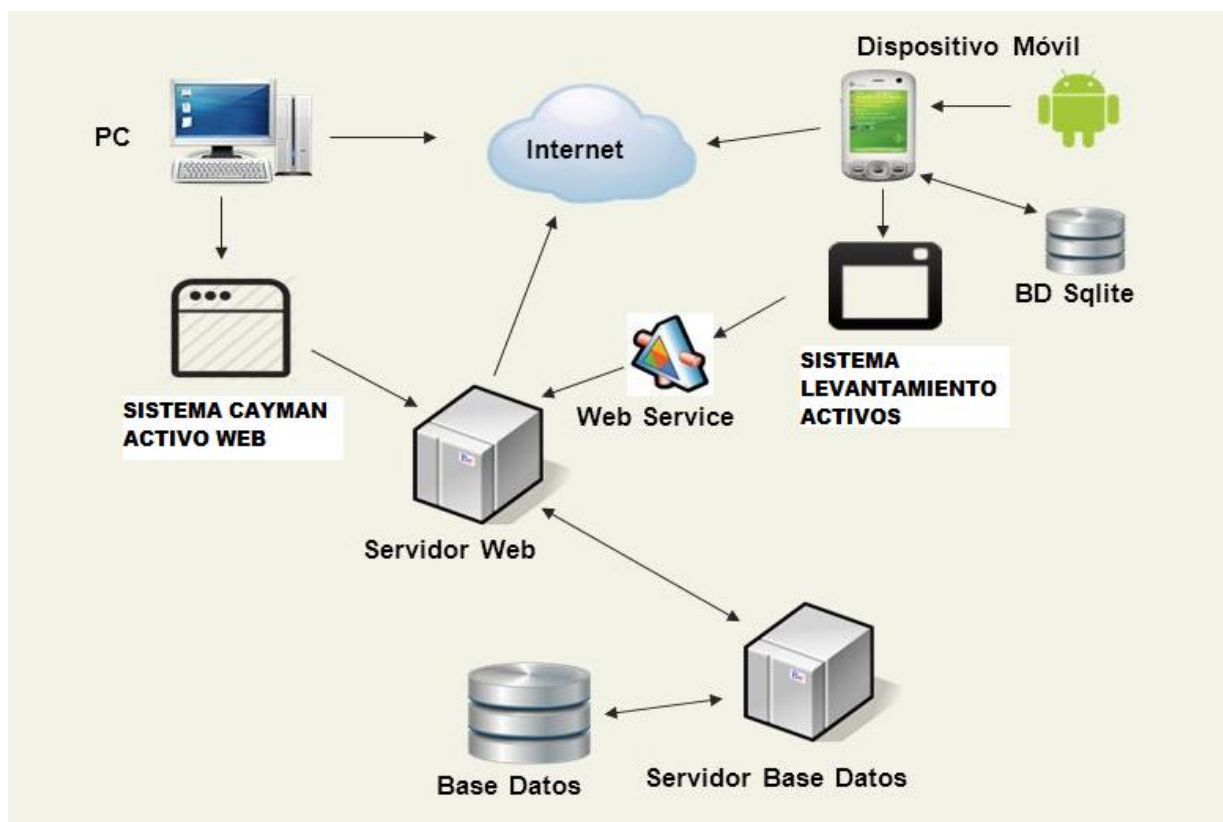
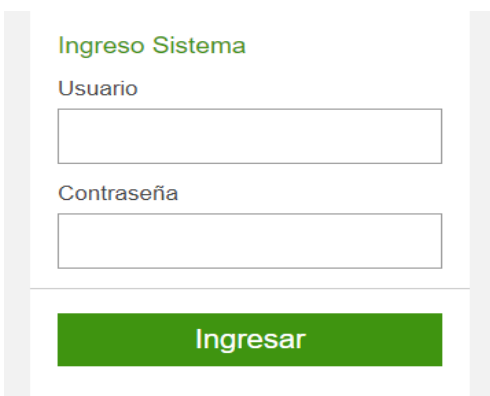


Gráfico N° 13 Diagrama de arquitectura del sistema

Autor: Fabián Aguirre

4.1.6 Diseño de interfaces

En el gráfico N° 14, se muestra el diseño de la pantalla de inicio de sesión la cual consta de dos campos de texto para ingresar el usuario y contraseña.



Ingreso Sistema

Usuario

Contraseña

Ingresar

Gráfico N° 14 Pantalla de inicio de sesión

En el gráfico N°15, se muestra el diseño de la pantalla principal del aplicativo consta de un encabezado donde se muestra el usuario que ingreso al sistema, y el menú con las opciones tipo slider el cual se activa dando clic en la barra superior.

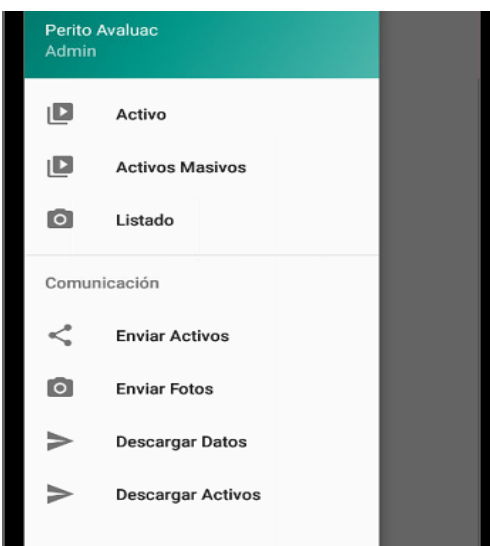


Gráfico N° 15 Pantalla principal aplicativo de levantamiento de información de activos fijo.

En el gráfico N°16, se muestra un bosquejo del diseño de la pantalla de ingreso, modificación de activos fijo la cual como va estar separada por secciones para organizar de mejor manera los campos del formulario, se validan todos los campos obligatorios al momento de guardar la información, este bosquejo está ajustado al tamaño de pantalla 5,1 pulgadas con resolución de 1920x1080 el cual es mínimo para visualizar el sistema.

Para la empresa CAYMANSYSTEMS las ubicaciones geográficas hacen referencia a la zona donde se encuentra el activo fijo, las ubicaciones orgánicas es el lugar del activo dentro de los departamentos de la empresa, por lo cual estas ubicaciones se distribuyen de la siguiente manera:

Ubicación geográfica 1: Provincia.

Ubicación geográfica 2: Cantón.

Ubicación geográfica 3: Dirección.

Ubicación geográfica 4: Piso.

Ubicación orgánica 1: Centro de Costo.

Ubicación orgánica 2: Área de la empresa.

Ubicación orgánica 3: Sub Área de la empresa.

Ubicación Geográfica

Ubicación Geográfica 1: (Seleccione...) ▼

Ubicación Geográfica 2: (Seleccione...) ▼

Ubicación Geográfica 3: (Seleccione...) ▼

Ubicación Geográfica 4: (Seleccione...) ▼

Ubicación Orgánica

Ubicación Orgánica 1: (Seleccione...) ▼

Ubicación Orgánica 2: (Seleccione...) ▼

Ubicación Orgánica 3: (Seleccione...) ▼

Custodio 1:

Custodio 2:

Codificación

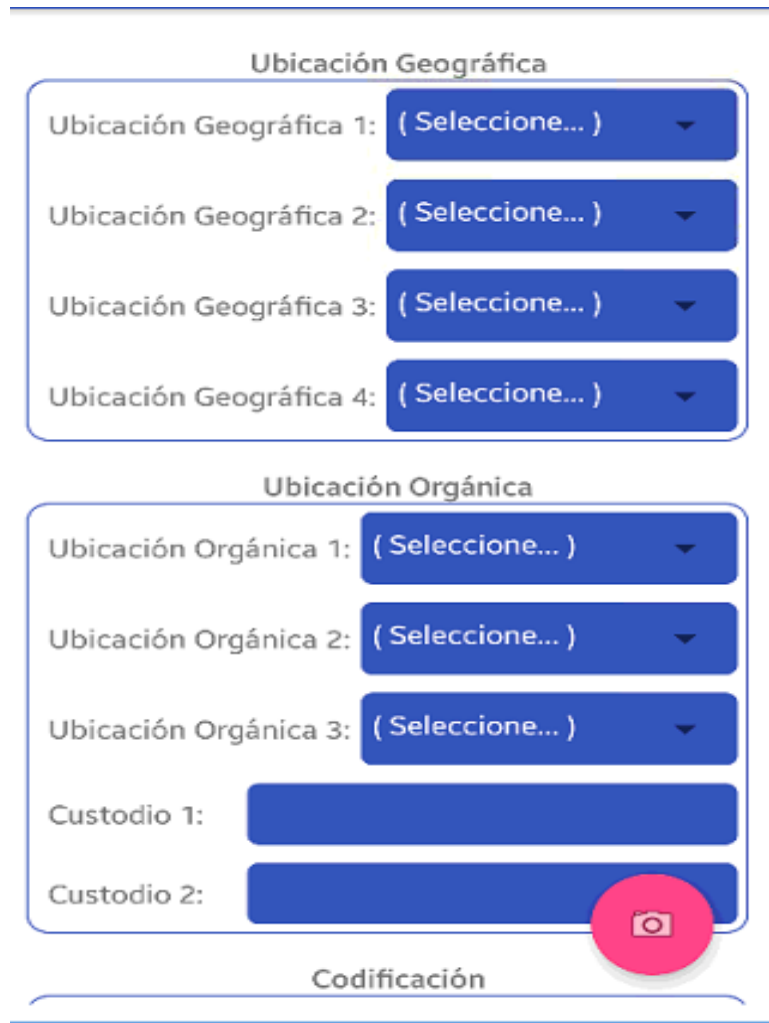


Gráfico N° 16 Pantalla ingreso de información de activos fijos.

En el gráfico 17, se muestra la descarga de información de descarga de todos los catálogos de la base de datos del sistema web así al aplicativo de levantamiento de información.

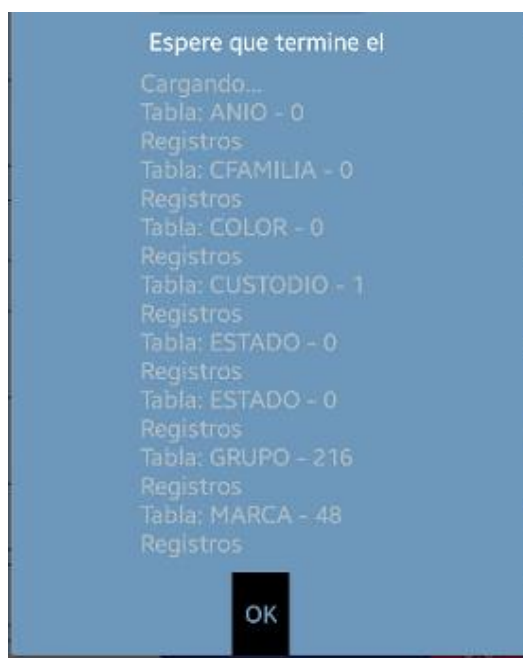


Gráfico N° 17 Pantalla de proceso de sincronización de catálogos.

4.1.7 Estándares de programación utilizados

Un estándar de programación es una forma de "normalizar" la programación para que al trabajar en un proyecto cualquiera, las personas involucradas en el mismo tengan un acceso rápido y una correcta comprensión del código. Un estándar nos define la escritura y organización del código fuente de un programa.

Para la descripción de los estándares utilizados en el desarrollo del proyecto, se toma como referencia los estándares de camelCase.

4.1.7.1 Estándar de diseño de base de datos.

El uso de estos estándares tiene innumerables ventajas, entre ellas:

- Asegurar la legibilidad del modelo de datos, inclusive para personas que no están relacionadas con el ambiente informático, en etapas de análisis y diseño;
- Facilitar la portabilidad entre motores de bases de datos, plataformas y aplicaciones;
- Facilitar la tarea de los programadores en el desarrollo de los sistemas.

Los nombres de tablas y campos deben especificarse bajo el estándar que maneja la empresa las tablas y campos llevarán el nombre completo en mayúsculas, los campos tendrán las primeras letras de cada tabla seguida del guion bajo y el nombre del campo en sí.

4.1.7.2 Estándar de desarrollo.

Los estándares a seguir para la representación del sistema mostrados en las historias de usuarios será la siguiente:

El nombre de los componentes que intervienen en la historia de usuario llevará su descripción en la parte inferior de su representación gráfica.

4.1.7.3 Estándar de base de datos.

Tablas.

El nombre de las tablas se escribirá en singular, la primera letra en mayúscula y las demás en minúsculas, sin espacios, sin tildes ni la letra “ñ”. Ejemplo:

Tabla N° 30 Estándares de Tablas

Nombre	Nomenclatura
Tabla de Usuario	USUARIO
Tabla de Activo	ACTIVO
Tabla de Grupo	GRUPO

Campos

Los campos tendrán un formato las tres primeras letras del nombre de tabla que pertenece, un guion bajo y el nombre del atributo. Ejemplo:

Tabla N° 31 Estándares de campos

Nombre	Nomenclatura
Nombre de usuario	USU_NOMBRE
Apellido de usuario	USU_APELLIDO
Dirección de usuario	USU_DIRECCION

Relaciones

Las relaciones deben ser según los requerimientos que se encuentran en cada tabla, especificar la clave foránea que es heredada y cómo se llama la relación tanto de una tabla otra, especificando dicha relación.

Tabla N° 32 Estándares de Relaciones

Nombre	Nomenclatura
Relación entre Activo y Usuario	fk_ActUsuID_UsuId

Claves.

Todas las claves primarias y foráneas deberán ser especificadas el tipo de claves para poder identificar. Ejemplo:

Tabla N° 33 Estándares de Claves

Nombre	Nomenclatura
Código de la Usuario	Usu_Id PK

4.1.7.4 Estándares de programación

Dentro de la aplicación va a existir distintos tipos de objetos, pueden ser: variables, constantes, procedimientos, botones, cajas de texto etc.; los nombres de los mismos estarán compuestos de un prefijo seguido del nombre del objeto.

Clases.

Los nombres de las clases poseen un sufijo en conjunto con el nombre propio de la clase, el nombre es significativo y está en singular.

Tabla N° 34 Estándares de Clases

Nombre	Nomenclatura
Clase de Usuario	Usuario.class
Clase de Imagen	Imagen.class

Nombre de las funciones y procedimientos

El nombre de una función o procedimiento debe contener las siguientes especificaciones: Inicia en mayúscula, Identifica claramente el contenido, Inicia con un verbo.

Tabla N° 35 Estándares de Funciones y Procedimientos

Nombre	Nomenclatura
Guardar Usuario	GuardarUsuario ()

Variables

Las variables dependiendo de su alcance se dividen en globales y locales, el nombre de la variable consta de un prefijo seguido del nombre de la variable. El nombre debe de tener la siguiente sintaxis:

Tabla N° 36 Estándares de Variables

Nombre	Nomenclatura
Variable Local	Private pass
Variable Pública	Public aux

Controles Visuales

Los controles visuales poseen un nombre estándar, el cual se identificará la primera letra en mayúscula del control seguido con un prefijo todo en minúscula después se identifica por el nombre del control.

Tabla N° 37 Estándares de Controles Visuales

Nombre	Prefijo	Ejemplo
TextView	Lbl	Lbl_Nombre
EditText	Txt	Txt_Telefono
Spinner	Cbo	Cbo_Estado
LinearLayout	Tbl	Tbl_Publicidad
Button	Btn	Btn_Confirmar

5 Conclusiones y Recomendaciones

5.1 Conclusiones

- El uso del aplicativo para levantamiento de información de activos fijos, permite a la empresa reducir costos ya que no es necesario adquirir nuevos dispositivos, puesto que la mayoría de usuarios disponen de un Smartphone de buenas características.
- Con el desarrollo del aplicativo se reduce tiempos y recursos en la etapa de levantamiento de información, ya que con la funcionalidad de ingreso masivo se optimiza el tiempo de registro de activo.
- El proceso de recolección de datos “off line” permite ahorrar batería del dispositivo ya que no necesita tener encendido los datos móviles, es importante que la batería del dispositivo permite trabajar las 8 horas diarias al perito.

5.2 Recomendaciones

- Para evitar el traslado de activos en el levantamiento se recomienda enviar vía correo electrónico al custodio y encargado del área de activos fijos el acta de activos fijos al terminar la recolección de información.
-

6 Referencias bibliográficas.

androidos.readthedocs. (17 de Septiembre de 2017). <http://androidos.readthedocs.io>.

Obtenido de <http://androidos.readthedocs.io>:

<http://androidos.readthedocs.io/en/latest/data/caracteristicas/>

Borja López, Y. (12 de 12 de 2015). *Metodología Ágil Desarrollo de Software XP*.

Obtenido de <http://www.runayupay.org>:

http://www.runayupay.org/publicaciones/2244_555_COD_18_290814203015.pdf

Cevallos, K. (08 de mayo de 2015). *Metodología de Desarrollo Ágil: XP y Scrum – INGENIERÍA DEL SOFTWARE*. Obtenido de Metodología de Desarrollo Ágil: XP y Scrum – INGENIERÍA DEL SOFTWARE:

<https://ingsoftwarekarlacevallos.wordpress.com/2015/05/08/metodologia-de-desarrollo-agil-xp-y-scrum/>

Debitoor. (28 de 11 de 2017). <https://debitoor.es/>. Obtenido de <https://debitoor.es/>:

<https://debitoor.es/glosario/definicion-de-activo>

Filein, R. (11 de 10 de 2017). SQLite: La Base de Datos Embebida. *Sg Buzz*, SG #17.

Obtenido de SQLite: La Base de Datos Embebida:

<https://sg.com.mx/revista/17/sqlite-la-base-datos-embebida#.WjAcLdFrz4Y>

HTK Identificación Inteligente. (12 de 12 de 2017). <http://htk-id.com/>. Obtenido de <http://htk-id.com/>: <http://htk-id.com/algo-que-tu-y-tu-empresa-deben-entender-sobre-el-inventario-de-los-activos-fijos/>

Myfpschool. (12 de 12 de 2017). <http://myfpschool.com/>. Obtenido de

<http://myfpschool.com/>: <http://myfpschool.com/como-es-la-estructura-de-un-sistema-operativo-movil/>

Programacionextrema. (12 de 12 de 2015). *Programacionextrema*. Obtenido de programacionextrema: <http://programacionextrema.tripod.com/fases.htm>

Revelo, J. (12 de 12 de 2017). *Tutorial De Bases De Datos SQLite En Aplicaciones Android*. Obtenido de Tutorial De Bases De Datos SQLite En Aplicaciones

Android: <http://www.hermosaprogramacion.com/2014/10/android-sqlite-bases-de-datos/>

Reviso. (12 de 12 de 2017). <https://www.reviso.com/es/>. Obtenido de <https://www.reviso.com/es/: https://www.reviso.com/es/que-es-un-activo>

educaconta. (01 de 11 de 2012). *educaconta*. Obtenido de educaconta: <http://www.educaconta.com/2012/11/control-de-los-activos-fijos.html>

Gimson, L. (01 de 06 de 2012). *docplayer*. Obtenido de docplayer: <http://docplayer.es/5603334-Metodologias-agiles-y-desarrollo-basado-en-conocimiento.html>

hermosaprogramacion. (01 de 10 de 2014). *hermosaprogramacion*. Obtenido de hermosaprogramacion: <http://www.hermosaprogramacion.com/2014/10/android-sqlite-bases-de-datos/>

Kristel, M. P., & Beauperthuy Taibo, J. L. (01 de 01 de 2011). *revistanegotium*. Obtenido de revistanegotium: <http://www.revistanegotium.org.ve/pdf/19/art4.pdf>

programacionextrema. (01 de 01 de 2012). *programacionextrema*. Obtenido de programacionextrema: <http://programacionextrema.tripod.com/index.htm>

proyectosdesoftwareblog. (01 de 01 de 2012). *proyectosdesoftwareblog*. Obtenido de proyectosdesoftwareblog: <https://proyectosdesoftwareblog.wordpress.com/unidad-2/>

Rómmel, F. (2011). SQLite: La Base de Datos Embebida. SG, 1.

7 Anexos

7.1 Anexo 1: Modelo Ficha Observación

FICHA DE OBSERVACIÓN					
<p>Fecha: 22 Octubre 2017</p> <p>Lugar: Quito, Amazonas y Orellana.</p> <p>Hora: 10: am</p>	<p>Fabián</p> <p>Observador: Aguirre</p>				
PREGUNTAS PLANTEADAS	SIEMPRE	A VEC ES	CASI NUNC A	NO APLIC A	OTROS
1.-Realiza el levantamiento de información de activos fijos utilizando el aplicativo móvil que posee la empresa.		x			
2.-El número de equipos que posee la empresa satisface al número de peritos que trabajan en el levantamiento de información.			x		
3.-El aplicativo que posee la empresa le permite la descarga de catálogos y trabajar “off line”.	x				
4.-La información recolectada en el dispositivo es sincronizada al final del día.	x				

RESUMEN DE LO OBSERVADO:

El líder de proyecto, organiza y dirige al equipo de trabajo, el número de equipos de trabajo es proporcional al número de activos fijos y ubicaciones geográficas que posee la empresa que contratante.

El proceso de levantamiento de información de activos fijos en este momento se realiza de forma mixta un porcentaje de forma manual y la otra con los equipos móviles que posee la empresa, en la actualidad la empresa no ha podido adquirir dispositivos para poder realizar todo el proceso de forma automática con el aplicativo móvil ya que el costo de los dispositivos móviles compatibles con el aplicativo es elevado.

El líder del proyecto es el encargado de parametrizar toda la información que se va descargar en el aplicativo móvil para realizar la recolección de información.

Una vez realizada la parametrización los peritos en campo descargan la información en el aplicativo para recolectar la información.

Al final del día el perito sincroniza toda la información al sistema web que posee la empresa.

7.2 Anexo 2 Entrevista aplicada Gerente Comercial

Entrevistado: Agustín Carrasco.

Entrevistado por: Fabián Aguirre

Objetivo: Con la finalidad de conocer los procesos que realiza la empresa, para el desarrollo de un aplicativo móvil en Android para el levantamiento de información de activos fijos.

1.- ¿Cuál son los usuarios que intervienen en el proceso de levantamiento de información?

Los usuarios que interviene en el proceso son: Jefe Proyecto, Perito, Supervisor.

Función	Actividad
Jefe Proyecto	Persona encargada de la gestión de catálogos, usuarios y perfiles dentro del sistema web.
Perito	Persona encargada de la recolección de información y sincronizar la información con el sistema web de la empresa.
Supervisor	Persona encargada de validar la información en el sistema web.

2.- ¿Qué tiempo se demora en el levantamiento de cada activo fijo?

1-10 minutos

10-20 minutos

25-30 minutos

Manifiesta que el tiempo está entre 1 a 10 minutos ya que el tiempo varía según el activo fijo.

3.- ¿El aplicativo móvil que posee la empresa es compatible con la mayoría de dispositivos móviles?

No

Si

Manifiesta que el sistema es compatible con equipos móviles que están en desuso y tienen un costo elevado, por tal motivo el uso del aplicativo en el proceso de levantamiento de información se está convirtiendo en un gasto, por lo que la empresa está perdiendo competitividad en el mercado.

4.- ¿La empresa estaría a dispuesta a desarrollar un aplicativo móvil que funcione en dispositivos móviles con sistema operativo Android?

SI

NO

La empresa está en una innovación tecnológica la cual y una de sus metas es contar con un aplicativo que sea compatible con la mayoría de dispositivos móviles, para así poder adquirir dispositivos de menor costo para ejecutar el aplicativo.

5.- ¿La funcionalidad del nuevo aplicativo móvil tendría que ser igual al aplicativo que posee la empresa en la actualidad?

SI

NO

6.- ¿De manera general puede mencionar la funcionalidad que va tener el aplicativo?

Pantalla de ingreso al aplicativo utilizando una autenticación a un servicio web

Sincronizar catálogos.

Ingreso modificación y eliminación de activos fijos todo esto de forma “off line”.

Listado de información recolectada.

Tomar foto de activos fijos.

Sincronizar información de activos fijos con el sistema web de la empresa.

7.3 Anexo 3 Manual de Usuario