

**UNIVERSIDAD NACIONAL AUTÓNOMA DE NICARAGUA  
UNAN- LEÓN**



**FACULTAD DE CIENCIAS Y TECNOLOGÍA  
DEPARTAMENTO DE COMPUTACIÓN**

Tesis para optar al título de Ingeniero en Sistemas de Información

**DESARROLLO DE UNA APLICACIÓN MÓVIL PARA ACCEDER AL  
CATÁLOGO DEL SISTEMA DE BIBLIOTECAS DE LA UNAN- LEÓN.**

**Autores:**

Br. Ada Georgina Hernández Acosta.  
Br. Miurel Virginia Quiroz Valverde.  
Br. Edys Narciso Meza Palacio.

**Tutor:** Ing. OTON CASTILLO NAVAS.

León, Junio 2015.

**“A La Libertad Por La Universidad”**

## **Agradecimiento**

Nos complace a través de este trabajo exteriorizar nuestro sincero agradecimiento:

A Dios nuestro Padre por derramar sobre nosotros sabiduría e inteligencia que nos ha permitido finalizar con éxitos nuestros estudios universitarios.

Nuestros padres por su constante apoyo moral y económico para con nosotros, porque sin su ayuda no hubiese sido posible culminar nuestros estudios.

La Facultad de Ciencias y Tecnología de la Universidad Nacional Autónoma de Nicaragua y en ella a los distinguidos docentes quienes con su profesionalismo y ética puesto de manifiesto en las aulas enrumban a cada uno de los que acudimos con sus conocimientos que nos servirán para ser útiles a la sociedad.

Nuestro tutor, Ing. Otón Castillo Navas, por habernos guiado con sus conocimientos y experiencia, hasta finalizar este trabajo monográfico.

## DEDICATORIA

Dedicamos el presente trabajo monográfico a:

Nuestro Padre Celestial, quién nos ha guiado por el buen camino, nos dio fuerzas para seguir adelante y no desmayar en las situaciones difíciles de nuestra vida, enseñándonos a enfrentar las adversidades sin perder nunca la dignidad ni desfallecer en el intento.

A nuestros padres y madres por darnos su apoyo, consejos, comprensión, amor, ayuda en los momentos difíciles, y por ayudarnos con los recursos necesarios para nuestros estudios.

A nuestros maestros que con paciencia y amor nos comparten la enseñanza que necesitamos para ser jóvenes de bien y preparados para los retos de la vida.

## INDICE

<b>1. INTRODUCCIÓN</b> .....	6
<b>2. ANTECEDENTES</b> .....	7
<b>3. JUSTIFICACIÓN</b> .....	8
<b>4. PLANTEAMIENTO DEL PROBLEMA</b> .....	9
<b>5. OBJETIVOS</b> .....	10
5.1 OBJETIVO GENERAL .....	10
5.2 OBJETIVOS ESPECÍFICOS .....	10
<b>6. METODOLOGÍA</b> .....	11
6.1 METODOLOGÍA DE TRABAJO .....	11
6.1.1 MODELO INCREMENTAL .....	11
6.1.1.1 ETAPAS DEL MODELO INCREMENTAL .....	11
<b>7. RECURSOS</b> .....	14
7.1.1 RECURSOS HARDWARE .....	14
7.1.2 RECURSOS SOFTWARE .....	14
<b>8. MARCO TEÓRICO</b> .....	15
8.1 ¿QUÉ ES SIABUC? .....	15
8.2 SIABUC 9.....	17
8.3 HISTORIA DE ANDROID.....	18
8.4 ¿QUE ES ANDROID? .....	18
8.5 VERSIONES DE ANDROID .....	19
8.5.1 ANDROID PETIT .....	19
8.5.2 ANDROID CUP CAKE .....	19
8.5.3 ANDROID DONUT .....	19
8.5.4 ANDROID ECLAIR.....	20
8.5.5 ANDROID FROYO.....	20
8.5.6 ANDROID GINGERBREAD .....	20
8.5.7 ANDROID HONEYCOMBS.....	21
8.5.8 ANDROID ICE CREAM SANDWITCH .....	21
8.5.9 ANDROID JELLY BEAN .....	21
8.5.10 ANDROID KIT KAT .....	22
8.6 ANDROID VS OTROS S.O .....	22

8.7	GOOGLE MAPS .....	23
8.8	HERRAMIENTAS DE DESARROLLO .....	25
8.8.1	ECLIPSE .....	25
8.8.2	CARACTERISTICAS .....	25
8.8.3	PLUGIN DE ECLIPSE .....	26
8.9	SDK ANDROID .....	26
8.10	PHP .....	27
8.10.1	¿QUE ES PHP? .....	27
8.10.2	VISION GENERAL DE PHP .....	27
8.11	SQLITE ANDROID .....	27
8.11.1	CARACTERISTICAS .....	28
8.12	JSON .....	29
8.13	ANALISIS Y DISEÑO .....	30
8.13.1	DIAGRAMA DE CASO DE USO (INTERFAZ USUARIO) .....	30
8.13.2	DIAGRAMAS DE SECUENCIAS.....	30
8.13.3	DISEÑO DE DATOS .....	33
8.13.4	DIAGRAMA DE ESTRUCTURA DE FICHEROS.....	34
<b>9.</b>	<b>CONCLUSIÓN .....</b>	<b>39</b>
<b>10</b>	<b>RECOMENDACIONES.....</b>	<b>40</b>
<b>11</b>	<b>BIBLIOGRAFÍA .....</b>	<b>41</b>
<b>12</b>	<b>ANEXOS .....</b>	<b>42</b>
12.1	ESPECIFICACION DE REQUISITOS .....	41
12.2	FUNCIONAMIENTO DE LA APLICACIÓN.....	49

## INTRODUCCIÓN

La Universidad Nacional Autónoma de Nicaragua (UNAN- León), es la universidad más antigua de Nicaragua, fue fundada en 1812 en la ciudad de León y en 1947 fue elevada a Universidad Nacional por el presidente Anastasio Somoza.

La UNAN-León fundó su primera biblioteca en 1816 con 4,000 volúmenes. Originalmente, la Biblioteca Central custodiaba las colecciones de Derecho, Medicina y Farmacia, y en la medida en que fue creciendo la población estudiantil y fueron creándose más carreras, hubo necesidad de separar las colecciones de acuerdo a las especialidades. Esto dio como resultado la creación de nuevas bibliotecas, quedando en el acervo de la Biblioteca Central sólo documentos de cultura general y del área de ciencias, parte de los libros "fundadores e históricos" y el Archivo Municipal.

Actualmente las bibliotecas cuentan con un sistema para la automatización de las tareas diarias que se realizan, el cual fue donado por la Universidad de Colima México.

SIABUC (Sistema Integral Automatizado de bibliotecas de la Universidad de Colima, México) es un software que está basado en módulos, cada módulo corresponde a una tarea específica dentro de la biblioteca, los principales módulos son adquisición, análisis y consulta; en nuestro trabajo de tesis desarrollaremos una aplicación en la plataforma de android utilizando el módulo de consulta.

El módulo de consulta contempla un servicio web que nos proveerá para la búsqueda de las fichas catalográficas; éstas podrán ser guardadas localmente, aparte que permitirá la localización de cada biblioteca perteneciente a dicha universidad.

## ANTECEDENTES

Las Bibliotecas han venido desempeñando un papel fundamental en la difusión de conocimiento desde que se les reconoció en el siglo XIX una función social estrechamente vinculada a la educación, la universidad, la investigación y el desarrollo de facultades humanas.

Con el advenimiento de las nuevas tecnologías de la información y comunicación, se está requiriendo nuevos cambios en las bibliotecas, para redefinir su identidad y el rol que tiene el profesional de la información dentro de la sociedad". Ya que según UNESCO, el lugar de las Bibliotecas en las Sociedades del conocimiento se centra en multiplicar y mejorar los entornos de aprendizaje.

La UNAN-León ha tomado un papel importante para la población estudiantil al proporcionarle recursos tan importantes como son las bibliotecas en todas las facultades que ella posee, y aparte ha facilitado a los estudiantes un catálogo de biblioteca web donde sus estudiantes pueden consultar los materiales de este catálogo.

Como una opción más para el usuario nos ha inspirado crear una aplicación dirigida a los dispositivos móviles para tener un mejor acceso a los catálogos, implementando nuestros conocimientos y nuestra capacidad de investigación y de esta forma contribuir con el crecimiento de nuestra universidad.

Existen aplicaciones de bibliotecas como: **Bibliotecas de Navarra .apk**, **Universitarios CLM .apk**, **Biblioteca UCLM .apk** que permite realizar todo tipo de acciones en una biblioteca desde consultar el catálogo de biblioteca, renovar préstamos de libros, ubicar las distintas sedes de las bibliotecas entre muchas otras operaciones.

Cada día el campo de las aplicaciones móviles abre paso y dan lugar a las más increíbles soluciones digitales para facilitar la vida del ser humano.

## JUSTIFICACIÓN

La UNAN- León, se ha caracterizado por ser una Institución de Educación Superior prestigiosa, de calidad, altamente crítica y participativa. Ha jugado un papel destacado en las transformaciones de la sociedad, impulsando y fortaleciendo programas educativos pertinentes para la formación de profesionales con sólidos conocimientos científicos, humanísticos e investigativos.

El presente trabajo de investigación es desarrollado con la finalidad de poder ofrecer a los usuarios (estudiantes) de la Universidad Nacional Autónoma de Nicaragua (UNAN-LEON) una nueva alternativa de realizar consultas al catálogo de las bibliotecas a través de nuestra aplicación móvil desarrollada en android.

Se ha elegido android como sistema operativo porque según reportes del Dpto. de Informática de la UNAN-León en el último mes del corriente año, se han conectado el 48.1 % de usuarios con este tipo de dispositivos, siendo éste el porcentaje más alto según sus estadísticas, esto nos demuestra que los estudiantes acceden a la web más con sus móviles que con sus computadores u otros sistemas, creemos que este es un motivo potencial para desarrollar la aplicación bajo esta plataforma demostrando nuestros conocimientos y habilidades.



## PLANTEAMIENTO DEL PROBLEMA

Normalmente un estudiante hace una consulta del libro que desee obtener únicamente cuando éste se encuentra dentro de la biblioteca en los equipos que se nos ha proporcionado para dicha tarea, porque no poseen su laptop o simplemente por ignorar que existe un sitio web donde puede acceder y realizar consultas y otras funcionalidades que ofrece.

La mayoría de los estudiantes actualmente tienen acceso a Smartphone, tabletas, etc, que son herramientas que se han convertido en un objeto personal de nuestra vida diaria y son adquiridos a costos más cómodos como son los de S.O android y la mayoría de las personas siempre están buscando aplicaciones para sus teléfonos, de esta forma podría ser que nuestra aplicación complemente las necesidades de los usuarios.

Siendo así nuestro objetivo brindar a los estudiantes una nueva forma de acceso a las consultas de catálogos que ofrece la UNAN-León, y aparte nos proporciona una opción de almacenamiento local de fichas favoritas, también nos ofrece la ubicación de una biblioteca en caso de no saberla, y la mejor forma de poder llegar hasta ella.

De esta forma hacemos que nuestra aplicación sea más interesante para los estudiantes a consultar sus fichas offline antes de entrar a la biblioteca en la comodidad de su teléfono.

## OBJETIVOS

### OBJETIVO GENERAL

- Brindar un nuevo acceso al catálogo de bibliotecas de la UNAN-León a través de una aplicación móvil android.

### OBJETIVOS ESPECÍFICOS

- Proporcionar un almacenamiento local de fichas catalográficas favoritas.
- Utilizar la herramienta de Google maps para la localización de las bibliotecas de nuestra universidad.

## **METODOLOGÍA DE TRABAJO**

La metodología empleada para nuestro trabajo de tesis está basada en el modelo incremental. Este modelo es de naturaleza interactiva brindando al final de cada incremento un resultado de proyecto a desarrollar completamente operacional.

### **ETAPAS DEL MODELO INCREMENTAL**

#### **Análisis:**

Para reunir información de cómo podríamos iniciar nuestra tesis nos dirigimos a la dirección de bibliotecas en el edificio central, ahí nos asesoraron y nos brindaron su apoyo.

Iniciamos analizando el Sistema SIABUC que es el software que usa UNAN León para ofrecer a los estudiantes y trabajadores una mejor organización de los libros, materiales o contenidos de una biblioteca, SIABUC contiene varios módulos y el que nosotros estamos interesados es el módulo de consulta, que está orientado hacia el usuario, generalmente se coloca una o varias terminales a disposición de los usuarios de la biblioteca sustituyendo de esta manera al tradicional catálogo físico de fichas catalográficas, SIABUC está alojado en un servidor IIS, y su base de datos corre en el gestor de base de datos Postgresql que es un gestor de base datos basado en objetos y libre.

Para poder utilizar el servicio de SIABUC en nuestra aplicación editamos algunos archivos .php, que proveen los creadores de SIABUC, ya que no permite realizar una nueva conexión a la base de datos a parte de la que tiene; se agregaron unas variables y líneas de código para poder usar lo que el sistema en si devuelve al hacer una consulta y se obtiene el resultado de la consulta, convirtiéndolo en objeto JSon y ya podríamos adaptarlo a nuestra necesidad, de esta forma realizamos una conexión indirecta al servidor, y así mostramos los resultados a nuestras pantallas.

#### **Diseño:**

Diseñamos la aplicación de forma que el usuario se sienta más cómodo usándola de manera sencilla debido a su fácil manejo.

Al proceder a crear nuestra aplicación le hemos integrado 3 funciones que son:

La búsqueda, que es la función principal de la aplicación, hemos hecho un sólo tipo de búsqueda que es general y se puede realizar una búsqueda por una o más palabras o por autor si se desea, siempre nos devuelve un mismo resultado y resuelve la búsqueda por biblioteca y dependiendo la biblioteca donde estemos ubicados o la biblioteca donde queramos hacer la búsqueda, clickeamos y nos muestra los resultados de todos los libros catalogados en esa

biblioteca, elegiremos el libro que busquemos y se mostrará los datos de la ficha que el libro u otro material posee.

Segunda función de nuestra aplicación es:

Guardar una ficha bibliográfica de un libro o documento que se ha buscado y almacenarla en una base de datos local con SQLite que es un motor de bases de datos muy popular en la actualidad por ofrecer características tan interesantes como su pequeño tamaño, no necesitar servidor, precisar poca configuración, ser transaccional, y por supuesto ser de código libre, y de esta forma cumplir con el objetivo de ofrecer este servicio offline, podemos guardar cuanta fichas queramos y borrarlas cuando no la deseemos tener más.

Como última función de la aplicación es ubicarse y localizar una determinada biblioteca, tenemos 2 opciones de vista las cuales son: primeramente se muestra la pantalla de bibliotecas y como resultado están todas las bibliotecas que pertenecen a la universidad y clickeamos y se nos mostrará la biblioteca ubicada en el mapa de google maps, pero si en nuestro caso no sabemos cómo llegar a la biblioteca destino entonces proveemos la opción de “Cómo llegar...” y con la ayuda de google maps creamos una ruta de camino desde donde estamos ubicados con nuestros celulares hasta donde está la biblioteca destino para esto debemos activar el GPS de nuestro teléfono, y de esta forma se traza la ruta que nos ayudará a llegar sin perdernos.

Es de esta manera que pretendemos facilitar esta herramienta que esperamos que sea muy útil para nuestros compañeros y cualquier usuario que desee usar dicha aplicación.

### **Código:**

Para poder utilizar el servicio de SIABUC en nuestra aplicación editamos algunos archivos .php, se agregaron variables y líneas de código como son las siguientes:

```
if(isset($_POST["tbusqueda"]) && $_POST["tbusqueda"]=="getbiblio"){ $client =
new SoapClient($ruta_ws_acervo); //Instanciación del servicio alojado en el servidor
$resp = $client->ListarBibliotecas();

$resp = $resp->ListarBibliotecasResult;

settype($resp, "array");

echo json_encode($resp); }
```

Además codificamos el diseño de la aplicación para que ésta sea funcional en nuestros dispositivos móviles.

**Prueba:**

En cada ciclo se realizan pruebas para verificar la correcta funcionalidad de la aplicación y de esta forma revisar si los objetivos se van cumpliendo, esta aplicación ha sido probada en los siguientes teléfonos: LG con versión 4.4 de android, HUAWEI con versión 4.2.2 de android y Verykool con versión 4.2.2 de android.

## RECURSOS

Para el desarrollo de nuestra aplicación necesitaremos de los siguientes recursos.

### RECURSOS HARDWARE

- PC: 2.3 GHz
- RAM: 4 GB
- Disco Duro: 260 GB
- Dispositivos móviles (Sistema Operativo Android)

### RECURSOS SOFTWARE

- Windows 7 ó 8
- Eclipse o
- Android Studio
- PHP
- SQLite
- Rational rose
- Sublime text
- Servidor de bibliotecas de SIABUC9

## MARCO TEÓRICO

### ¿QUÉ ES SIABUC?

#### **Sistema Integral Automatizado de Bibliotecas de la Universidad de Colima (SIABUC)**

Fue en el año de 1983 cuando la Universidad de Colima incursionó en el ámbito tecnológico, durante ese año se creó la Dirección de Desarrollo Bibliotecario, cuyo objetivo principal fue estructurar un sistema de bibliotecas para apoyar con servicios de información bibliográfica y documentar las labores sustantivas de docencia e investigación en todos los campos de la Universidad, centralizando sus procesos de adquisición, catalogación y clasificación.

Debido a esto surgió la necesidad de automatizar dichos procesos y servicios de información). Para satisfacer esas necesidades se realizó un estudio de experiencias sobre sistemas de automatización bibliotecaria y sus casos de éxito en las bibliotecas mexicanas, al identificar la carencia de este tipo de tecnología informática y el desconocimiento por parte de los bibliotecarios se diseñó SIABUC (Sistema Integral Automatizado de Bibliotecas de la Universidad de Colima).

En aquel entonces, el objetivo principal de SIABUC era generar un fichero electrónico que facilitará el proceso de impresión de los catálogos topográficos de la entonces única biblioteca que tenía la Universidad, sin embargo, debido a la carencia de este tipo de software se plantea el proyecto a nivel nacional ante Consejo Nacional de Ciencia y Tecnología (CONACYT) y la Secretaría de Educación Pública (SEP), con la finalidad de obtener recursos económicos para mejorarlo y posteriormente ofrecerlo a las bibliotecas públicas mexicanas.

SIABUC es un software que fue creado en la Universidad de Colima como respuesta a una necesidad interna de preparar juegos de fichas catalográficas, sin embargo, el proyecto creció y fue cubriendo gradualmente cada una de las tareas comprendidas en el diagrama de flujo de actividades de las bibliotecas. El impacto y trascendencia que SIABUC ha tenido es evidente puesto que actualmente es utilizado en más de 2500 instituciones públicas y privadas de México, América Latina y el Caribe.

Presencia de SIABUC en Latinoamérica La primera institución usuaria y representante de SIABUC en Costa Rica fue la Escuela Centroamericana de Ganadería, en 1989, en Balsa de Atenas, provincia de Alajuela. Durante ese mismo año el Instituto Tecnológico de Costa Rica inicia con el uso de SIABUC,

actualmente cuentan con la versión SIABUC8 en funcionamiento. En la actualidad, más de 50 bibliotecas son usuarias de SIABUC.

Otro país centroamericano que utiliza SIABUC es Colombia, aquí las instituciones pioneras son la Universidad de Cauca y la Universidad de Caldas. Durante el 2003 se firmó un convenio entre la Universidad de Colima y el Ministerio de Cultura de Colombia para un proyecto denominado “Apoyo para la Sistematización de la Red Nacional de Bibliotecas Públicas”, esto con la finalidad de que SIABUC8 fuera la herramienta de automatización utilizada en más de 500 bibliotecas en aquel país.

Mientras que en **Nicaragua**, la primera institución en utilizar SIABUC fue la UNAN-León de Nicaragua y posteriormente la Universidad Centroamericana (UCA). Actualmente son 42 las instituciones que tienen convenio firmado con la Universidad de Colima para la utilización de SIABUC.

El funcionamiento de SIABUC está basado en módulos, cada módulo corresponde a una tarea específica dentro de la biblioteca, los **módulos** son:

### **Adquisiciones**

Nos lleva un control de las compras y las donaciones, pudiéndose capturar desde la solicitud de compra, imprimir las listas de pedidos, registrar la recepción de material y llevar un catálogo de los proveedores. Este módulo genera otros productos impresos como son: Listas de pedidos por fecha, Lista de donaciones, Lista de material ingresado, etc.

### **Análisis**

En este módulo se realiza la catalogación o procesos técnicos al material adquirido. Dicha catalogación está basada en el formato MARC. Se realiza también el etiquetado del material y, de ser necesario, la impresión de la ficha o juego de fichas catalográficas. Algunos de los productos que nos genera este módulo son: Impresiones de etiquetas con clasificación y/o códigos de barras, impresión de tarjetas de préstamos, listados, consultas simples, control de productividad de los capturistas, etc. La captura de publicaciones periódicas está incluida dentro de éste módulo, el cual nos permite además manejar un catálogo de fichas analíticas (artículos o interiores de la revista).

### **Consultas**

Este módulo está orientado hacia el usuario. Generalmente se coloca una o varias terminales a disposición de los usuarios de la biblioteca sustituyendo de esta manera al tradicional catálogo físico de fichas catalográficas.

### **Inventario**



El proceso de realizar un inventario se agiliza bastante con este módulo que nos permite, además, la identificación exacta del material faltante y su posterior descarte. Se recomienda utilizar el código de barras para aprovechar al 100% este módulo.

### **Préstamos**

En este módulo se lleva un estricto control de los préstamos de material, incluye las siguientes funciones: Catálogo de Usuarios, el cual puede incluso, importarse desde otro sistema; Préstamo a domicilio o en sala, Bloqueos de libros y de usuarios conflictivos, Reportes de usuarios morosos, Reportes de préstamos, Constancias de no adeudo, etc.

### **Estadísticas**

Esta funcionalidad se encuentra dentro de cada uno de los diferentes módulos de SIABUC y presenta diversos reportes sobre las tareas realizadas en cada uno de ellos. Algunas de las estadísticas incluyen: Préstamos, Consultas, Productividad, Compras, Donaciones, etc.

### **Publicaciones en WEB**

Permite poner los catálogos de SIABUC disponibles a través de una página WEB (sólo para servidores basados en Windows)

### **SIABUC 9**

SIABUC 9 es uno de los software más rentables por su precio accesible, gastarán lo mismo si compran SIABUC 9 o si agregan 10 nuevos libros a su colección; además el soporte técnico es totalmente gratuito y directo con los desarrolladores del sistema, por eso hay una gran cantidad de instituciones que lo utilizan y recomiendan tanto en México como en América Latina. SIABUC 9 fue desarrollado por la Universidad de Colima, México, que es una universidad pública, y que tiene más de 20 años de experiencia en compartir y contribuir al desarrollo del país y de la región latinoamericana. Se ha procurado que cada versión de SIABUC 9 sea mejor, por su facilidad de uso, que no se requiera tener conocimientos avanzados en bibliotecología ni informática para automatizar los procesos de su biblioteca; que tampoco se requiera de equipo de cómputo sofisticado; que por su precio sea accesible a grandes, medianas o pequeñas bibliotecas, está en idioma español y realizado por mexicanos, con soporte técnico en línea así como asesoría directa y personalizada.

### **Características**

- SIABUC9 Software mexicano y en español, diseñado para plataformas Windows sin necesidad de gran equipamiento de cómputo.

- Utiliza PostgreSQL como gestor de base de datos, PostgreSQL es un Sistema de gestión de bases de datos relacional orientado a objetos y libre, publicado bajo la licencia BSD.
- Tiene el más bajo costo de inversión.
- Actualizaciones y soporte técnico gratuito.
- Se imparten cursos durante todo el año en sus instalaciones o en las del cliente.
- Anualmente se realiza una Reunión de Usuarios de SIABUC en Colima a la cual se invita a todos los usuarios registrados

## **HISTORIA DE ANDROID**

En el año 2003, Andy Rubin, Rich Miner, Nick Sears y Chris White daban forma a Android Inc. En sus inicios, únicamente trascendió que la actividad de la empresa se centraba en “el desarrollo de software para teléfonos móviles”.

Android Inc. pasó casi dos años trabajando “en la sombra”, hasta que Google comenzó a “reclutar” a fuerza de talonario a algunas “startup” (término que se refiere a nuevas compañías con un futuro prometedor) del sector móvil, con la clara intención de replicar su éxito de la Web en el futuro de las telecomunicaciones inalámbricas.

Al principio Android era un sistema operativo para móviles prácticamente desconocido. Hasta noviembre de 2007 sólo hubo rumores, pero en esa fecha se lanzó la Open Handset Alliance, que agrupaba a muchos fabricantes de teléfonos móviles, chipsets y Google, se proporcionó la primera versión de Android, junto con el SDK para que los programadores empezaran a crear sus aplicaciones para este sistema.

Aunque los inicios fueran un poco lentos, debido a que se lanzó antes el sistema operativo que el primer móvil, rápidamente se ha colocado como el sistema operativo de móviles más vendido del mundo, situación que se alcanzó en el último trimestre de 2010.

## **¿QUÉ ES ANDROID?**

Android es un sistema operativo basado en Linux diseñado originalmente para dispositivos móviles, tales como teléfonos inteligentes, pero que posteriormente se expandió su desarrollo para soportar otros dispositivos tales como tablets, reproductores MP3, notebooks, PCs e incluso televisores. El sistema permite programar aplicaciones en una variación de Java llamada Dalvik. El sistema operativo proporciona todas las interfaces necesarias para desarrollar aplicaciones que accedan a las funciones del teléfono (como el GPS, las

llamadas, la agenda, etc.) de una forma muy sencilla en un lenguaje de programación muy conocido como es Java.

## **VERSIONES DE ANDROID**

### **Android Petit Four**

En febrero de 2009 llegó la primera actualización para Android, unos tres meses después del lanzamiento del G1. La versión 1.1 fue dedicada básicamente a reparar errores y a implementar las actualizaciones "over the air" que hasta ese momento ninguna plataforma estaba haciendo.

### **Android cup cake**

Android 1.5 es más conocido por su nombre en clave, Cup cake, fue la primera versión en utilizar nombre de postres. Cada versión después de Cup cake ha sido nombrada con un nombre de postre continuando el orden alfabético. En esta versión comenzamos a ver algunos cambios en la interfaz de usuario, por poco que se puedan apreciar, como son los cambios en la barra del buscador y en la barra del menú, también cambio el logo del navegador. Las primeras versiones de Android no contaban con un teclado virtual, ya que el G1 disponía de un teclado físico, en la 1.5 se introdujo el teclado virtual (teclado QWERTY virtual) coincidiendo con la salida del primer Android con pantalla táctil y sin teclado físico, el HTC Mágic.

### **Android donut**

Con la llegada de Donut vino el soporte para redes CDMA haciendo que Android llegara a Estados Unidos y Asia. Pero tal vez la mejora más significativa fue la posibilidad de correr el sistema operativo en múltiples resoluciones de pantalla y relaciones de aspecto, a raíz de esta actualización es cuando podemos disfrutar hoy en día de pantallas con resolución QVGA, HVGA, WVGA, FWVGA, QHD y 720p. Donut también introdujo la búsqueda rápida, generalmente conocida en el mundo de móvil como búsqueda universal. Antes de Donut la búsqueda se limitaba a Internet pero con las mejoras introducidas en la versión 1.6 se podría buscar además contenido propio del teléfono (contactos, aplicaciones, etcétera) incluso por voz, todo desde el mismo widget. Otras mejoras: Nuevo diseño de Android Marquet al

estilo de la mascota de Android en color blanco y verde, introducción de listas de aplicaciones gratuitas y de pago. Nueva interfaz de la cámara con mejor integración en la galería.

### **Android Eclair**

Hablamos ya de un año después del lanzamiento del G1, noviembre del 2009. Fue ofrecido en exclusiva con Verizon y el Motorola Droid, un teléfono que marcó un antes y después para Android y con el que Motorola volvió a ser la gran marca que fue. GHA HDI -10- El Droid de Motorola fue el teléfono más potente que se había visto en el mercado hasta la fecha, con una pantalla con resolución de 854 x 480. Pero no solo fue el Droid el que impulsó las ventas de Android sino las mejoras que se introdujeron en la versión 2.0. Después del Droid/Miles toné prácticamente la mayoría de teléfonos lanzados llegaron con Android 2.1, una corrección de errores y que Google no renombro dejándola con el nombre de Eclair.

### **Android Froyo**

Lanzado a mediados de 2010 trajo una gran cantidad de cambios. La pantalla de inicio fue rediseñada, se ampliaron los 3 paneles existentes desde el inicio a 5 con un nuevo grupo de accesos directos dedicados y se agregaron unos puntos para saber en cada momento en la pantalla donde nos encontrábamos. El Nexus One fue el primer teléfono en actualizarse a Android 2.2. Froyo también introducía una galería completamente rediseñada con imágenes en 3D que aparecen al inclinar el teléfono. Además se introdujo soporte para hotspot móvil Wifi (compartir la conexión 3G), algo que muchas compañías decidieron desactivar o activarla con la opción de pagar un coste extra. Se mejoró también el soporte para copiar y pegar en Gmail incorporando también Microsoft Exchange (servidor de comunicación basado en el correo electrónico), mejoras en el video llamado y en la memoria siendo terminales más potentes. En esta versión se agregó la posibilidad de poner una contraseña o PIN en la pantalla de bloqueo para los usuarios que no les gustaba el patrón de desbloqueo.

### **Android Gingerbread**

Un año y medio después del lanzamiento de Froyo y el Nexus One (el primer teléfono de Google fabricado por HTC), Google volvió con un nuevo móvil de marca propia pero esta vez en colaboración con Samsung, el Nexus S y aprovechó para lanzar la nueva versión del sistema operativo, Android 2.3 Gingerbread. Con el Nexus S llegó la pantalla curvada y el fin del TrackBall. Gingerbread fue una actualización menor en muchos sentidos pero trajo algunos cambios importantes en la interfaz de usuario.

Mejor control en copiar y pegar, Teclado mejorado, Maximización de la batería y herramientas de gestión de desarrollo, Soporte para cámara frontal (video online), Juegos, Otras características: Apoyo a la tecnología NFC integrada en una antena incrustada en la tapa de la batería.

### **Android Honeycombs**

La versión de Android para Tablet, que presentó de la mano de Motorola junto con el Xoom. Cambio de color, del verde típico de Android al azul que se utilizó para la batería, el widget del reloj, indicadores de señal y algunas otras características de la interfaz.

Android 3.1 y 3.2 fueron versiones de mantenimiento, prueba de ello es que Google no las renombró y continuaron llamándolas Honeycomb. Aunque algunas mejoras introducidas en estas actualizaciones se han ido implementando en la mayoría de Tablet con Android 3.0 del mercado, como la posibilidad de modificar el tamaño de los widgets al presionar sobre ellos.

### **Android Ice Cream Sandwich**

Ha sido lanzada junto con el Galaxy Nexus, el nuevo Smartphone Google fabricado por Samsung. Ice Cream Sandwich toma prestadas muchas características de Honeycomb como los botones virtuales o la transición de tonos verdes a azules, la multitarea con una lista desplegable de miniaturas y las barras de acción dentro de las aplicaciones. Comprensiblemente sigue siendo Multiplataforma (Tablet, teléfonos móviles y notebooks).

. Además la pantalla principal recibe una bandeja de favoritos que puede ser configurada por el usuario. Y mejora en el soporte 3D.

### **Android Jelly Bean**

Nadie tiene noticias al respecto de ninguno de los dos, hasta el día 9 de octubre de este mismo año 2012, cuando Google lanzó la versión 4.1.2 conocida como Jelly Bean, la cual incorpora última versión de Nexus. Las novedades son pocas, exceptuando las típicas correcciones de bugs y mejoras en la estabilidad. No obstante, en la Nexus 7 esta actualización incorpora la posibilidad de rotar la pantalla de inicio, algo que venía deshabilitado de fábrica y que sólo se podía conseguir mediante aplicaciones de terceros. Otra de las funciones que incluye esta actualización es poder expandir las notificaciones enriquecidas de Jelly Bean con un sólo dedo.

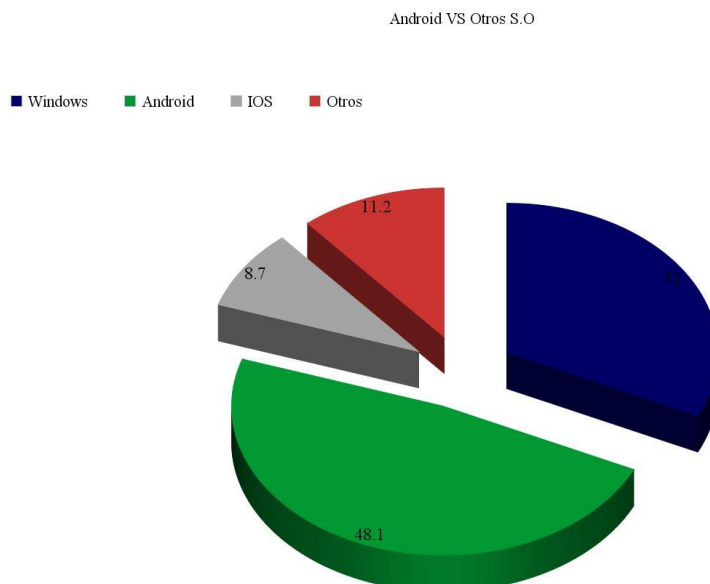
## Android Kitkat

Aunque se esperaba la versión número 5.0 y con el nombre Key Lime Pie, Google sorprendió con el cambio de nombre, que se debió a un acuerdo con Nestlé para asociar ambas marcas. Un objetivo principal de la versión 4.4 es hacer Android disponible en una gama aún más amplia de dispositivos, incluyendo aquellos con tamaños de memoria RAM de sólo 512 MB. Para ello, todos los componentes principales de Android han sido recortados para reducir sus requerimientos de memoria, y se ha creado una nueva API que permite adaptar el comportamiento de la aplicación en dispositivos con poca memoria. Más visibles son algunas nuevas características de la interfaz de usuario.

## ANDROID VS OTROS S.O

En el mundo podemos encontrar hasta 10 sistemas operativos diferentes. Bada, Symbian, Blackberry y los sistemas propietarios de los terminales de gama de entrada, conviven con los más extendidos Windows Phone, iOS y sobre todo Android.

A continuación les mostraremos una gráfica donde según el departamento de informática de la UNAN- León, android es el sistema operativo con mayor acceso a la red:



Es evidente que Android es el OS móvil número uno en el mundo, en parte porque ha sabido ganarse el favor de los usuarios versión tras versión y, para que negarlo, porque está presente por igual en los terminales más económicos y en los dispositivos de gama alta.

Muchas personas se encuentran ante un dilema al momento de elegir un smartphone cuando se trata de buscar información de los sistemas operativos de los equipos. La prestigiosa marca Apple, cuenta con su propio sistema operativo, el iOS, y como todo producto de Apple, ofrece calidad y confianza, pero ¿qué está pasando con Android? Hay quienes aseguran que es el mejor sistema operativo de la actualidad para teléfonos inteligentes, ofreciendo velocidad y practicidad. Los smartphones con Android e iOS tienen muchas diferencias, y cada uno tiene sus ventajas y desventajas.

Cada uno de estos sistemas operativos es bueno y recomendable, ya depende del comprador comprar lo que más se amolde a sus necesidades. Lo cierto es que tanto Android como iOS dominan ahora las listas de ventas.

Para muchos, el principal atractivo de Android es su capacidad de personalización. El usuario prácticamente puede moldear este sistema operativo como mejor le parezca (cosa contraria a iOS 5). Asimismo, posee atractivas herramientas para hacer de Android una herramienta que además de funcional le permitan verse bien.

## GOOGLE MAPS

*Google Maps* es un servicio de *Google* que ofrece imágenes vía satélite de todo el planeta, combinadas, en el caso de algunos países, con mapas de sus ciudades, lo que unido a sus posibilidades de programación abierta ha dado lugar a diversas utilidades ofrecidas desde numerosas páginas web.

Desde su lanzamiento en febrero de 2005, la aplicación cartográfica de *Google* ha conmocionado a la comunidad de desarrolladores. Si bien sus principios técnicos de base eran ya conocidos, incluso utilizados desde hacía tiempo, la aplicación de *Google* los combina de manera inteligente, y sobre todo ofrece una accesibilidad sin igual.

Varios aspectos de *Google Maps* son los responsables de su facilidad de uso por cualquier usuario: el sistema de deslizamiento de imagen, acoplado a la carga dinámica de nuevas imágenes; la adaptación del mapa al tamaño de ventana del navegador; la interfaz minimalista; la posibilidad de cambiar de tipo de mapa en un clic...

Como todas las demás aplicaciones *Google, Maps* descansa poderosamente sobre la utilización de JavaScript. La carga y el deslizamiento de imagen no podrían efectuarse sin este código.

Según el nivel de zoom, un mapa podría descomponerse en varias decenas de miles de cuadrados. Esta descomposición se realiza automáticamente por un script del lado servidor. Cuando el utilizador hace deslizar la imagen en un sentido, JavaScript calcula cuales son los cuadrados involucrados, envía al servidor una pregunta sobre las imágenes en cuestión, y las coloca alineadas junto a las otras.

Cada cuadrado es almacenado en un fichero cuyo nombre indica su longitud, su latitud, y el valor del zoom. Recuperar estas informaciones para todos los cuadrados a colocar, no es sino una cuestión de derivación de los datos conocidos para un solo cuadrado.

Todo ello necesita, por tanto, una sincronización perfecta entre lo que puede preguntar el código JavaScript, y lo que está efectivamente disponible en el servidor. El programa de recorte inicial de la imagen completa es por tanto extremadamente importante, porque es el responsable, para la precisión del corte y el rigor de su etiquetamiento, del buen desenvolvimiento de los acontecimientos desencadenados por las acciones del usuario.

Sin embargo no todo son flores para esta aplicación. *Google Maps*, aunque interesante para muchos usuarios y programadores, al decir de los especialistas no es una aplicación de gran perfección técnica, y se le han hecho críticas Como ejemplo de las cuales mencionaremos una con el título "Google Maps o la negación de la Cartografía".

*Google Maps* provee a los desarrolladores un API capaz de aprovechar los datos disponibles a través del servicio, en el seno de las propias aplicaciones. Y más interesante para el funcionamiento mismo del servicio, *Google* da acceso a *AjaXSLT*, un proyecto Open Source que propone una biblioteca JavaScript.

Con las API de Google Maps publicadas, numerosos sitios han aparecido proponiendo sus propias maneras de extraer los datos de los mapas de *Google* y presentarlos en la página a través de scripts específicos, proponiendo un servicio particular.

## HERRAMIENTAS DE DESARROLLO

### ECLIPSE

Eclipse es un programa informático compuesto por un conjunto de herramientas de programación de código abierto multiplataforma para



desarrollar lo que el proyecto llama "Aplicaciones de Cliente Enriquecido", opuesto a las aplicaciones "Cliente-liviano" basadas en navegadores. Esta plataforma, típicamente ha sido usada para desarrollar entornos de desarrollo integrados (del inglés IDE), como el IDE de Java llamado Java Development Toolkit (JDT) y el compilador (ECJ) que se entrega como parte de Eclipse (y que son usados también para desarrollar el mismo Eclipse). Sin embargo, también se puede usar para otros tipos de aplicaciones cliente, como BitTorrent o Azureus.

## **CARACTERISTICAS**

Eclipse dispone de un Editor de texto con resaltado de sintaxis. La compilación es en tiempo real. Tiene pruebas unitarias con JUnit, control de versiones con CVS, integración con Ant, asistentes (wizards) para creación de proyectos, clases, tests, etc., y refactorización.

Asimismo, a través de "plugins" libremente disponibles es posible añadir control de versiones con Subversion.4 e integración con Hibernate.

## **PLUGIN DE ECLIPSE**

Para integrar Android con Eclipse, necesitamos un Plugin, que dependiendo de la versión del IDE que tengamos, se instalará de una forma concreta. Para la versión de Eclipse 3.4 Ganymede (la última) Acceder al Menú "Help->Software updates". Desde aquí, podremos controlar todas las actualizaciones y plugins de Eclipse. Nosotros necesitaremos añadir una nueva dirección para que el sistema localice e instale el Plugin de Android.

## **SDK ANDROID**

El SDK (Software Development Kit ) de Android, incluye un conjunto de herramientas de desarrollo. Comprende un depurador de código, biblioteca, un simulador de documentación, ejemplos de código y tutoriales. Las plataformas de desarrollo soportadas incluyen Linux ( cualquier distribución moderna ), Mac OS X 10.4.9 o posterior, y Windows XP o posterior. La plataforma integral de desarrollo (IDE, Integrated Development Environment) soportada oficialmente es Eclipse junto con el complemento ADT (Android Development Tools plugin), aunque también puede utilizarse un editor de texto para escribir ficheros Java y XML y utilizar comandos en un terminal para crear y depurar aplicaciones.

Las Actualizaciones del SDK están coordinadas con el desarrollo general de Android. El SDK soporta también versiones antiguas de Android, por si los

programadores necesitan instalar aplicaciones en dispositivos ya obsoletos o más antiguos. Las herramientas de desarrollo son componentes descargables, de modo que una vez instalada la última versión, pueden instalarse versiones anteriores y hacer pruebas de compatibilidad, esta es una gran ventaja para esta plataforma.

Una aplicación Android está compuesta por un conjunto de ficheros empaquetados en formato *.apk* y guardada en el directorio */data/app* del sistema operativo Android (este directorio necesita permisos de superusuario, root, por razones de seguridad). Un paquete APK incluye ficheros *.dex* (ejecutables Dalvik, un código intermedio compilado), recursos, etc.

## PHP

Usamos php para permitirnos el acceso al servicio web de búsquedas que provee SIABUC y de esta forma crear una función dentro de las páginas web *.php* en los archivos que tiene el servicio y así poder utilizar los datos en nuestra aplicación ya que no nos permite acceder de forma directa a la base de datos.

## ¿QUE ES PHP?

**PHP** es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico.

Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera la página Web resultante. PHP ha evolucionado por lo que ahora incluye también una interfaz de línea de comandos que puede ser usada en aplicaciones gráficas independientes. Puede ser usado en la mayoría de los servidores web al igual que en casi todos los sistemas operativos y plataformas sin ningún costo.

Se considera uno de los lenguajes más flexibles, potentes y de alto rendimiento conocidos hasta el día de hoy. Lo que ha atraído el interés de múltiples sitios con gran demanda de tráfico como Facebook, para optar por PHP como tecnología de servidor.

## VISION GENERAL DE PHP

PHP puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. El lenguaje PHP se encuentra instalado en más de 20 millones de sitios web y en un millón de servidores. El enorme número de sitios en PHP ha visto reducida su cantidad a favor de otros nuevos lenguajes no tan poderosos desde agosto de 2005. El sitio web de Wikipedia está desarrollado en PHP. Es también el módulo Apache más popular entre las computadoras que utilizan Apache como servidor web.

El gran parecido que posee PHP con los lenguajes más comunes de programación estructurada, como C y Perl, permiten a la mayoría de los programadores crear aplicaciones complejas con una curva de aprendizaje muy corta. También les permite involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones.

Cuando el cliente hace una petición al servidor para que le envíe una página web, el servidor ejecuta el intérprete de PHP. Éste procesa el script solicitado que generará el contenido de manera dinámica (por ejemplo obteniendo información de una base de datos). El resultado es enviado por el intérprete al servidor, quien a su vez se lo envía al cliente.

Permite la conexión a diferentes tipos de servidores de bases de datos tanto SQL como NoSQL tales como MySQL, PostgreSQL, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird, SQLite o MongoDB.

También tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos, tales como Unix (y de ese tipo, como Linux o Mac OS X) y Microsoft Windows, y puede interactuar con los servidores de web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI.

Es una alternativa a las tecnologías de Microsoft ASP y ASP.NET (que utiliza C# y Visual Basic .NET como lenguajes), a ColdFusion de la empresa Adobe, a JSP/Java, CGI/Perl y a Node.js/Javascript. Aunque su creación y desarrollo se da en el ámbito de los sistemas libres, bajo la licencia GNU, existe además un entorno de desarrollo integrado comercial llamado Zend Studio. CodeGear (la división de lenguajes de programación de Borland) ha sacado al mercado un entorno de desarrollo integrado para PHP, denominado 'Delphi for PHP'. También existen al menos un par de módulos para Eclipse, uno de los entornos más populares.

## SQLITE ANDROID

SQLite es un sistema de gestión de bases de datos relacional compatible con ACID, contenida en una relativamente pequeña biblioteca escrita en C. SQLite es un proyecto de dominio público creado por D. Richard Hipp.

A diferencia de los sistemas de gestión de bases de datos cliente-servidor, el motor de SQLite no es un proceso independiente con el que el programa principal se comunica. En lugar de eso, la biblioteca SQLite se enlaza con el programa pasando a ser parte integral del mismo. El programa utiliza la funcionalidad de SQLite a través de llamadas simples a subrutinas y funciones. Esto reduce la latencia en el acceso a la base de datos, debido a que las llamadas a funciones son más eficientes que la comunicación entre procesos. El conjunto de la base de datos (definiciones, tablas, índices, y los propios datos), son guardados como un sólo fichero estándar en la máquina host. Este diseño simple se logra bloqueando todo el fichero de base de datos al principio de cada transacción.

En su versión 3, SQLite permite bases de datos de hasta 2 Terabytes de tamaño, y también permite la inclusión de campos tipo BLOB..

## CARACTERISTICAS

La biblioteca implementa la mayor parte del estándar SQL-92, incluyendo transacciones de base de datos atómicas, consistencia de base de datos, aislamiento, y durabilidad (ACID), triggers y la mayor parte de las consultas complejas.

SQLite usa un sistema de tipo inusual. En lugar de asignar un tipo a una columna como en la mayor parte de los sistemas de bases de datos SQL, los tipos se asignan a los valores individuales. Por ejemplo, se puede insertar un string en una columna de tipo entero (a pesar de que SQLite tratará en primera instancia de convertir la cadena en un entero). Algunos usuarios consideran esto como una innovación que hace que la base de datos sea mucho más útil, sobre todo al ser utilizada desde un lenguaje de scripting de tipos dinámicos. Otros usuarios lo ven como un gran inconveniente, ya que la técnica no es portable a otras bases de datos SQL. SQLite no trataba de transformar los datos al tipo de la columna hasta la versión 3.

Varios procesos o hilos pueden acceder a la misma base de datos sin problemas. Varios accesos de lectura pueden ser servidos en paralelo. Un acceso de escritura sólo puede ser servido si no se está sirviendo ningún otro acceso concurrentemente. En caso contrario, el acceso de escritura falla devolviendo un código de error (o puede automáticamente reintentarse hasta que expira un tiempo de expiración configurable). Esta situación de acceso concurrente podría cambiar cuando se está trabajando con tablas temporales.

Sin embargo, podría producirse un interbloqueo debido al multihilo. Este punto fue tratado en la versión 3.3.4, desarrollada el 11 de febrero de 2006.

Existe un programa independiente de nombre `sqlite` que puede ser utilizado para consultar y gestionar los ficheros de base de datos SQLite. También sirve como ejemplo para la escritura de aplicaciones utilizando la biblioteca SQLite.

## JSON

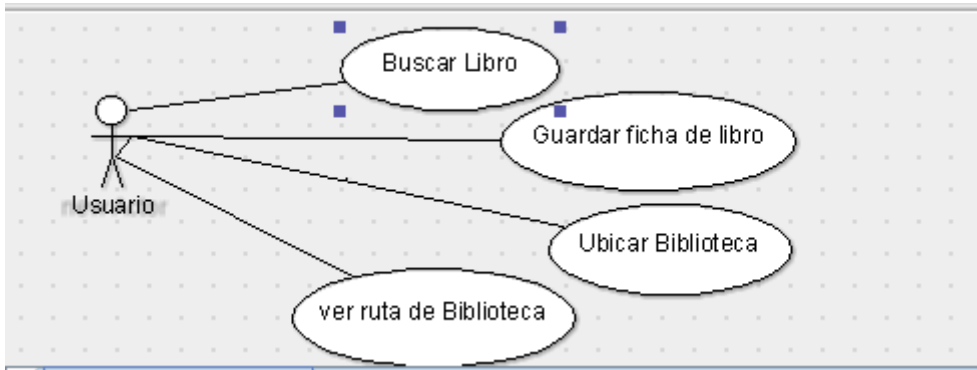
JSON Son las iniciales en inglés de JavaScript Object Notation. JSON es un subconjunto de la notación de objetos de Javascript, básicamente, es un formato para el intercambio de datos. Debido a su simplicidad, JSON se ha utilizado como un reemplazo del XML en AJAX. Una de las ventajas de JSON sobre XML, percibidas por los desarrolladores web, es la facilidad con que éste puede ser analizado usando el procedimiento `eval()` de Javascript.

El término JSON está altamente difundido entre los desarrolladores, sin embargo, es un término mal descrito ya que en realidad es sólo una parte de la definición del estándar ECMA-262 en que está basado Javascript. De ahí que ni Yahoo, ni Google emplean JSON, sino LJS.

Una de las cualidades intrínsecas de Javascript denominada LJS (Literal Javascript) facilita el flujo de datos e incluso de funciones, para la cual no requiere la función `eval()`, si son datos los que se transfieren como en el caso de XML. Todo lo referente a transferencia de datos en todos sus tipos, incluyendo arrays, booleans, integers, etc. no requieren de la función `eval()`, y es precisamente en eso en donde supera por mucho JavaScript al XML, si se utiliza el LJS y no la incorrecta definición de JSON.

## ANÁLISIS Y DISEÑO

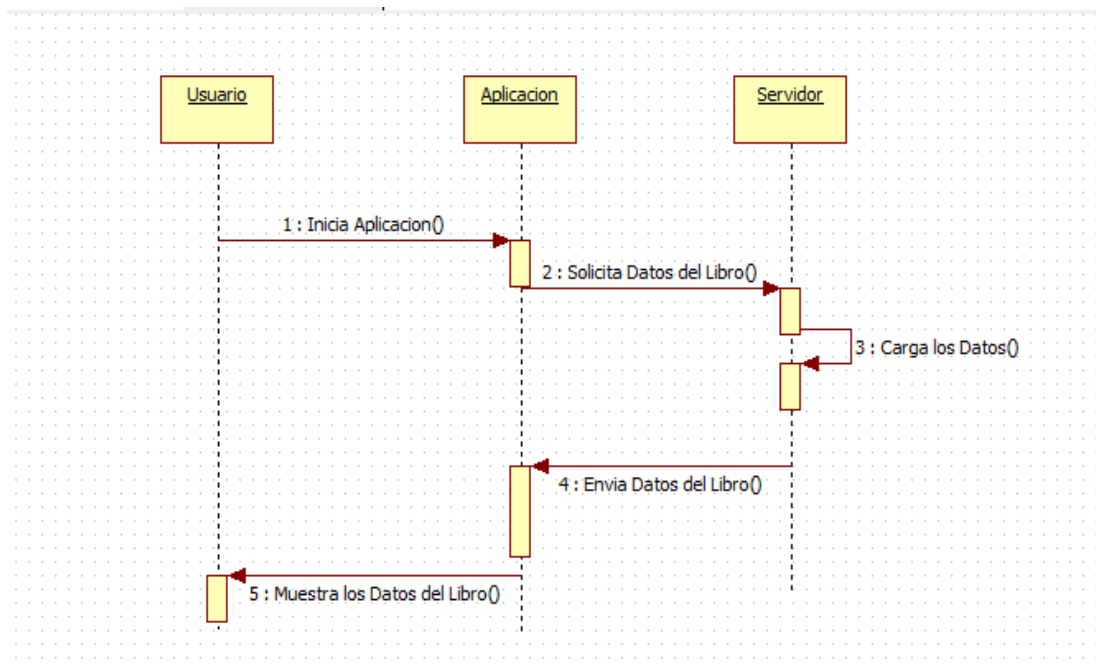
### DIAGRAMA DE CASOS DE USO (INTERFAZ USUARIO)



### DIAGRAMAS DE SECUENCIAS (Interfaz Usuario)

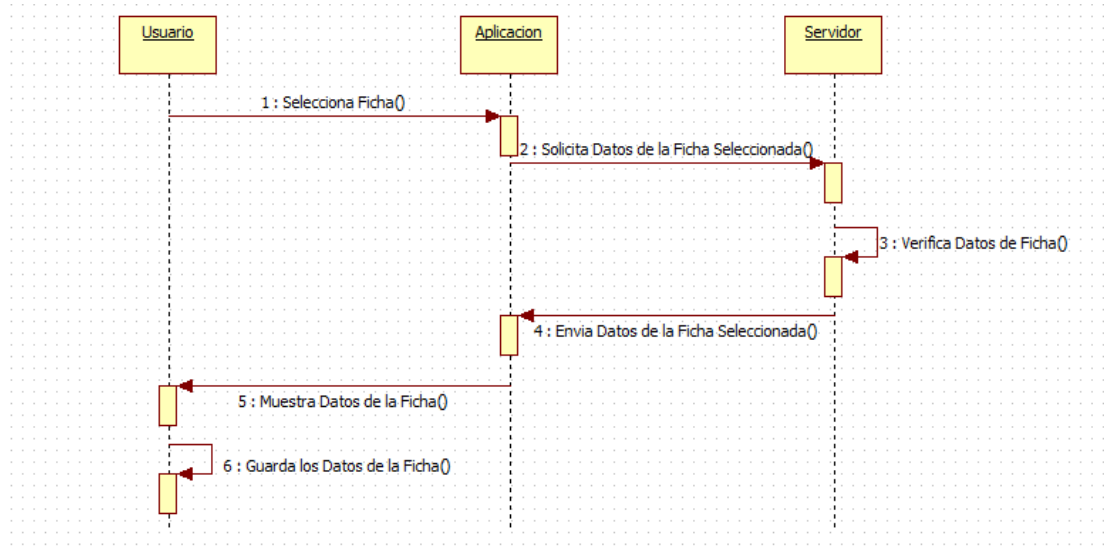
#### Buscar Libro

El usuario podrá visualizar en la búsqueda a través de la aplicación los detalles del libro que desee.



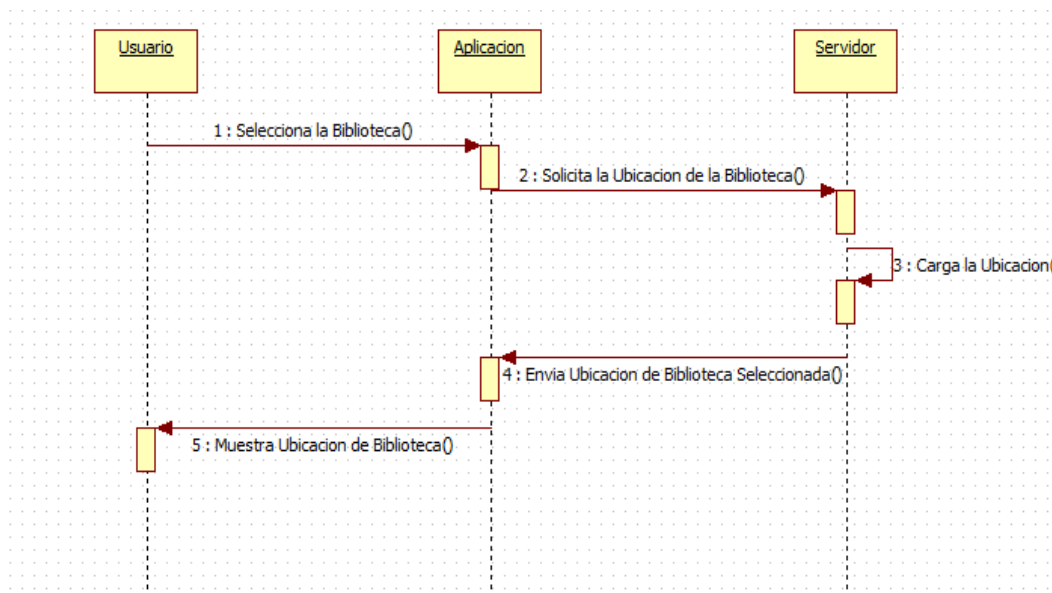
## Guardar Ficha de Libro

El usuario podrá guardar la ficha del libro que seleccionó.



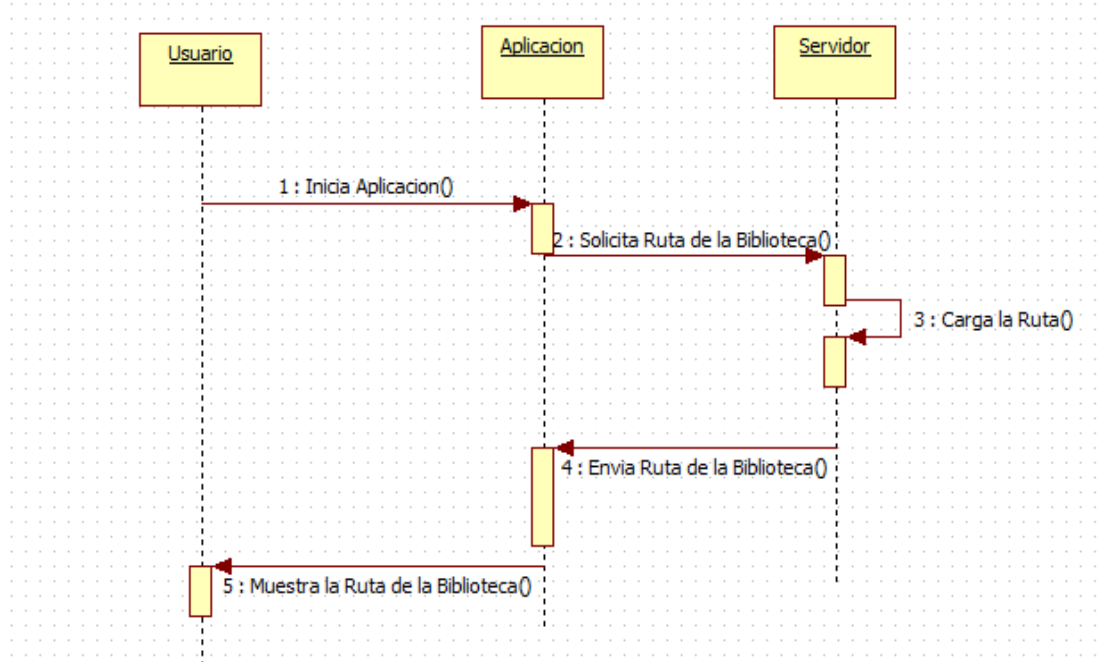
## Ubicar Biblioteca

El usuario ubicará la biblioteca seleccionada.



## Ver Ruta de Biblioteca

El usuario podrá visualizar la ruta de la biblioteca que desee.



## DISEÑO DE DATOS

### Base datos local SQLite

Fue necesario el almacenamiento de datos locales para el manejo de las fichas guardadas y para el registro de los nombres de bibliotecas.

### Tabla Ficha

Nombre	Descripción	Tipo
<b>Id</b>	Identifica la ficha	Int
<b>Título</b>	Nombre del libro	Text
<b>Autor</b>	Nombre del autor	Text
<b>Numero</b>	Numero de ficha	Text
<b>clasificacion</b>	Código de clasificación	Text
<b>publicacion</b>	Nombre de publicación	Text
<b>descripcion</b>	Descripción de ficha	Text
<b>tematica</b>	Tipo de temáticas del libro	Text

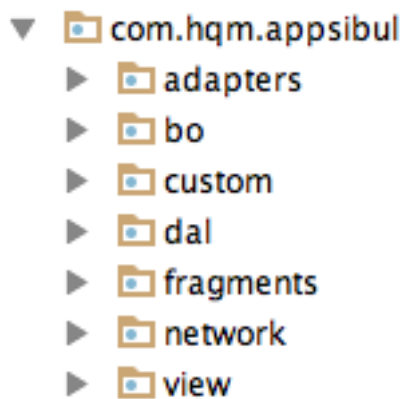






































## Tabla Biblioteca























Nombre	Descripción	Tipo
<b>Id</b>	Identifica la biblioteca	Int
<b>Nombre</b>	Nombre de la biblioteca	Text


















## DIAGRAMA DE ESTRUCTURA DE FICHEROS

Para la codificación de nuestra aplicación la hemos dividido en clases dentro de 7 carpetas las cuales contienen el código necesario para la funcionalidad satisfactoria de ésta.



- ▼  com.hqm.appsubul
  - ▼  adapters
    -   AdapterFichaList
    -   AdapterBuscar
    -   AdapterHelper
    -   AdapterLibrary
    -   AdapterMyFichas
    -   TabsPagerAdapter
  - ▼  bo
    -   BO\_Buscar
    -   BO\_disponibilidad
    -   BO\_Ficha\_Biblio
    -   BO\_Ficha\_Detail
    -   BO\_fichabiblio
    -   BO\_library
    -   BO\_material
    -   BO\_MyFicha
  - ▶  custom
  - ▶  dal
  - ▶  fragments
  - ▶  network
  - ▶  view

- ▼  com.hqm.appsibul
  - ▶  adapters
  - ▶  bo
  - ▼  custom
    - Ⓢ  CustomDialogClassInstallGooglePlayService
    - Ⓢ  CustomDialogNoInternet
    - Ⓢ  CustomDialogServerError
  - ▼  dal
    - Ⓢ  DAL\_ficha
  - ▼  fragments
    - Ⓢ  FragmentBibliotecas
    - Ⓢ  FragmentBuscar
    - Ⓢ  FragmentFichaBiblio
    - Ⓢ  FragmentFichaDetail
    - Ⓢ  FragmentMapBibliotecas
    - Ⓢ  FragmentMisFichas
    - Ⓢ  FragmentMyFichaDetail
    - Ⓢ  FTabBibliotecas
    - Ⓢ  FTabBuscar
    - Ⓢ  FTabMisFichas
  - ▶  network
  - ▶  view

- ▼  com.hqm.appsibul
  - ▶  adapters
  - ▶  bo
  - ▶  custom
  - ▶  dal
  - ▶  fragments
  - ▼  network
    -  Conexion
    -  DatabaseHandler
    -  DirectionUtil
    -  GPSUtil
    -  HandlerQuery
    -  JSONUtil
    -  NetUtil
  - ▼  view
    -  MainActivity
    -  SplashActivity

## CONCLUSIÓN

Hemos culminado nuestra tesis, llegando a las siguientes conclusiones:

Desarrollando una nueva forma de acceso al catálogo de bibliotecas de la UNAN-León, utilizando la plataforma android, brindamos a los estudiantes mayor oportunidad para las consultas en línea, también aprovechan sus dispositivos de manera positiva.

Al agregar el mapa y las rutas de las distintas bibliotecas, le brinda un mayor atractivo hacia estudiantes que no conocen dichas ubicaciones y necesitan orientación, de esta forma resulta ventajosa para ellos una aplicación de este tipo.

Otra herramienta importante como fue la creación de un almacenamiento local de fichas para su consulta offline, proveyendo al usuario datos necesarios para que tenga disponible en el momento de presentarse en una biblioteca.

Por lo tanto los objetivos marcados para este proyecto han sido satisfechos. El balance obtenido del mismo es muy positivo y esperamos que nuestra aplicación sea muy provechosa para nuestros compañeros en un futuro.

## RECOMENDACIONES

Culminado nuestro trabajo monográfico consideramos las siguientes recomendaciones para mejoras de nuestra aplicación:

- Poder realizar reservaciones de libros en línea.
- Ofrecer libros virtuales para su descarga.

En general recomendamos a todos los programadores o nuevos forjadores de la programación que dediquen cada día a fomentar el buen uso de la tecnología como un servicio que resuelve muchos problemas o que puede no hacerlo si hacemos mal uso de ella.

## BIBLIOGRAFIA

Sitio web del sistema de bibliotecas. URL <http://sibul.unanleon.edu.ni/>

Báez, Manuel. Borrego, Álvaro. Cordero, Jorge. Cruz, Luis. González, Miguel. Hernández, Francisco. Palomero, David. Rodríguez de Llera, José. Sanz, Daniel. Saucedo, Mariam. Torralbo, Pilar. Zapata, Álvaro. Introducción a Android, 1-76.

Gómez, Oliver Salvador. (2011). Manual de programación en android. 8-166

Curso Programación Android: Aprende a crear tus propias aplicaciones.

URL sgoliver.net. <http://www.sgoliver.net/blog/curso-de-programacion-android/>

Historia de android en la web. URL

[http://es.wikipedia.org/wiki/Anexo:Historial de versiones de Android,](http://es.wikipedia.org/wiki/Anexo:Historial_de_versiones_de_Android)

<http://www.monografias.com/trabajos101/sistema-operativo-android/sistema-operativo-android.shtml>

Página oficial de android. URL <http://developer.android.com/index.html>

Página oficial de Siabuc. URL [http://siabuc.ucol.mx/site/include/que\\_es.html](http://siabuc.ucol.mx/site/include/que_es.html)

Siabuc 9. <http://siabuc.ucol.mx/site/include/productos.html>

Página oficial de IIS. URL

<http://technet.microsoft.com/es-es/library/cc753433%28v=ws.10%29.aspx>

Ponce Suárez, Hugo César.pdf [Tutorías - Facultad de Telemática - Universidad de Colima.](#)

JSON. <https://es.wikipedia.org/wiki/JSON>

## ANEXOS

### ESPECIFICACIÓN DE REQUISITOS

#### 1 Introducción

Para el desarrollo de la aplicación móvil de consultas de catálogos de libros se requiere realizar una aplicación que cumpla con las necesidades analizadas, creando una opción de búsqueda más cómoda y agregando dos módulos como son guardar una ficha de un libro y ubicarse en la biblioteca deseada.

##### 1.1 Propósito

Definición de los requisitos software que tiene que cumplir la creación de la aplicación móvil para las consultas de los libros, almacenamiento de una ficha offline de un libro o muchos libros y la ubicación de las bibliotecas de la universidad. Para un usuario es muy importante usar su dispositivo móvil para cualquier necesidad que se le aparezca, de esta manera pretendemos ofrecer al usuario una mejor opción de vista y búsqueda de su libro.

##### 1.2 Alcance

Este proyecto llevará por nombre APLICACIÓN MÓVIL EN ANDROID PARA LAS CONSULTAS DE CATÁLOGOS DEL SISTEMA DE BIBLIOTECAS UNAN LEÓN.

Éste realizará las siguientes funciones:

##### USUARIO:

- 1) Buscar libro
- 2) Guardar ficha del libro
- 3) Ubicar una biblioteca
- 4) Ver ruta de biblioteca

##### 1.3 Definición, acrónimos y abreviaturas

Usuario: Persona que hace la consulta, guarda una ficha y elige la opción de ubicación de una biblioteca.

Offline: Fuera de línea.

App: Aplicación

Ficha: Datos catalográficas digital de un libro.



Acervo: Constituyen la parte sustantiva de la Biblioteca y debido a su importancia permiten calificarla como una de las más completas en su género.

## 2. Descripción General

### 2.1 Perspectiva del producto

Esta aplicación se realiza con el fin de proveer a los estudiantes otra opción aparte de la página web [SIBUL](#), que posee oficialmente nuestra universidad. Ésta hará búsquedas de catálogos de libros de las bibliotecas las cuales poseen fichas catalográficas que un usuario puede guardar localmente y consultarlas offline, también posee un mapa de ubicación de todas las bibliotecas y la opción de ubicación de rutas por GPS simuladas.

### 2.2 Funcionalidad del producto

Las funciones principales que la aplicación diseñada realizará son las ahora mencionadas:

- Buscar un libro
- Guardar una ficha catalográfica
- Ubicar una biblioteca
- Ver ruta de biblioteca

### 2.3 Características de los usuarios

Tipo de usuario	Estudiante
Formación	-----
Habilidades	Realizar búsquedas, guardar fichas catalográficas, ubicar una biblioteca
Actividades	-----

## 2.4 Restricciones

La aplicación deberá tener un diseño e implementación sencilla, creando una interfaz amigable para el usuario final. Esta aplicación es creada para la plataforma de android y funcionará con cualquier teléfono inteligente con la plataforma mencionada.

## 3 Requisitos

### 3.1 Requisitos Específicos

<b>Número de requisito</b>	RF1		
<b>Nombre de requisito</b>	Buscar Libro		
<b>Tipo</b>	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
<b>Prioridad del requisito</b>	<input checked="" type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/Opcional
<b>Descripción</b>	El usuario debe buscar el libro que desea, luego seguirá otra pantalla la cual muestra las bibliotecas donde se encuentra este libro y debemos elegirla para que se muestre la gama de libros coincidentes, elegir el que buscamos y de esta forma obtenemos los datos de la ficha del libro.		

<b>Número de requisito</b>	RF2		
<b>Nombre de requisito</b>	Guardar Ficha catalográficas		
<b>Tipo</b>	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
<b>Prioridad del requisito</b>	<input type="checkbox"/> Alta/Eencial	<input type="checkbox"/> Media/Deseado	<input checked="" type="checkbox"/> Baja/Opcional
<b>Descripción</b>	El usuario guarda la ficha del libro que consultó para obtenerla offline		

<b>Número de requisito</b>	RF3		
<b>Nombre de requisito</b>	Ubicar Bibliotecas		
<b>Tipo</b>	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
<b>Prioridad del requisito</b>	<input type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input checked="" type="checkbox"/> Baja/ Opcional
<b>Descripción</b>	Muestra la ubicación de la biblioteca con la opción de activar el GPS		
<b>Número de requisito</b>	RF4		
<b>Nombre de requisito</b>	Ver ruta de biblioteca		
<b>Tipo</b>	<input checked="" type="checkbox"/> Requisito		
<b>Prioridad del requisito</b>	<input type="checkbox"/> Alta/Esencial		
<b>Descripción</b>	Muestra la ubicación dela biblioteca con la opción de activar el GPS		

### 3.2 Requisitos Comunes de las Interfaces

- ❖ Entradas: Ninguna
- ❖ Salidas:

#### 3.1.1 Interfaces de usuarios

- Primeramente se muestran tres menús de la aplicación Buscar, Bibliotecas y Fichas
- Buscar se mostrará como pantalla principal con una caja de texto para buscar un libro por autor, por palabra si lo desea, luego
- Se muestra la siguiente pantalla donde están todas las bibliotecas en las cuales el libro está registrado.
- Una vez elegida la biblioteca, se muestra el catálogo de las posibles opciones de libros que buscamos.
- Se elige el libro y nos muestra al fin la ficha catalográfica de dicho libro o documento.
- En esta sección nos aparece la opción de guardar la ficha del libro.

- En el Menú Fichas se nos muestra todas las fichas guardadas en ese momento.
- En el menú Bibliotecas encontramos todas las bibliotecas las cuales poseen su ubicación en el mapa de Google y las cuales poseen la siguiente opción que es:
- Ubicación de ruta por GPS Ubicar una biblioteca donde el usuario posiblemente no conoce su ruta y que se dibuja la posible ruta a seguir.

### **3.1.2 Interfaces de Hardware**

La pc utilizada para el desarrollo de la app debe tomar en cuenta los siguientes requisitos:

- RAM: 2 gb
- Disco duro:250 gb
- Procesador mayor de 2.3Ghtz

### **3.1.3 Interfaces Software**

Son necesarias las siguientes herramientas para la realización de nuestra aplicación que posea una herramienta de desarrollo móvil:

- Computador con:
- Eclipse
- Android Studio

## **Requisitos funcionales**

### **RF1 Buscar Libro**

#### **1.1 Especificación**

##### **1.1.1 Introducción**

El usuario ingresa la búsqueda por palabra o autor y así se desplegarán las pantallas para seleccionar el libro que más le conviene.

##### **1.1.2 Entradas**

Datos de búsqueda

### 1.1.3 Procesos

Se validan los datos que el usuario ha ingresado y se extraerán los datos necesarios de la base de datos.

### 1.1.4 Salidas

Si el dato fue correcto proseguirá una pantalla donde se muestran las bibliotecas donde posiblemente el dato fue encontrado y nosotros seleccionamos la biblioteca y luego se muestran los libros con las coincidencias de los datos introducidos, debemos seleccionar el libro para que se nos muestre la ficha que tiene dicho libro.

## RF2 Guardar Ficha

### 1.2 Especificación

#### 1.2.1 Introducción

El usuario guardará la ficha del libro que desea tener en su teléfono y se mostrarán en el Menú Fichas.

#### 1.2.2 Entradas

-----

#### 1.2.3 Procesos

La ficha que se obtiene del servidor de Siabuc se guardará en una base de datos local del teléfono.

#### 1.2.4 Salidas

Se muestra un mensaje donde nos indica que la ficha ha sido guardada.

## RF3 Ubicar una biblioteca

### 1.3 Especificación

#### 1.3.1 Introducción

El usuario podrá ver la ubicación de la biblioteca que eligió en el Menú Bibliotecas desde google Maps.

#### 1.3.2 Entradas

-----

#### 1.3.3 Procesos

-----  
**1.3.4 Salidas**

Se muestra un punto donde la biblioteca está ubicada.

**RF4 Ver ruta de biblioteca**

**1.4 Especificación**

**1.4.1 Introducción**

El usuario debe activar el GPS de su android para que pueda dibujarse la ruta desde donde él está ubicado hasta la biblioteca que ha elegido.

**1.4.2 Entradas**

Activar GPS para permitir la ubicación desde google Maps

**1.4.3 Procesos**

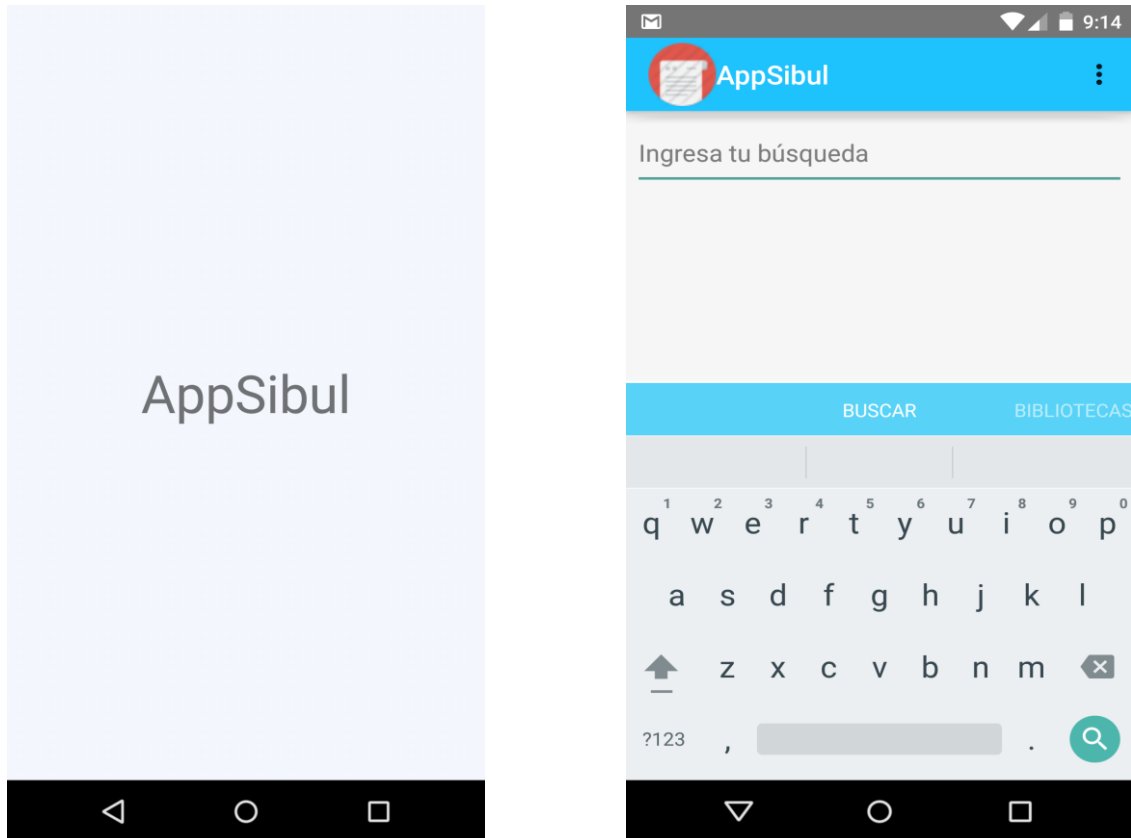
Se activa el GPS y se dibujará la ruta desde donde estás ubicado hasta la biblioteca seleccionada.

**1.4.4 Salidas**

Se visualizará la ruta de llegada de la biblioteca.

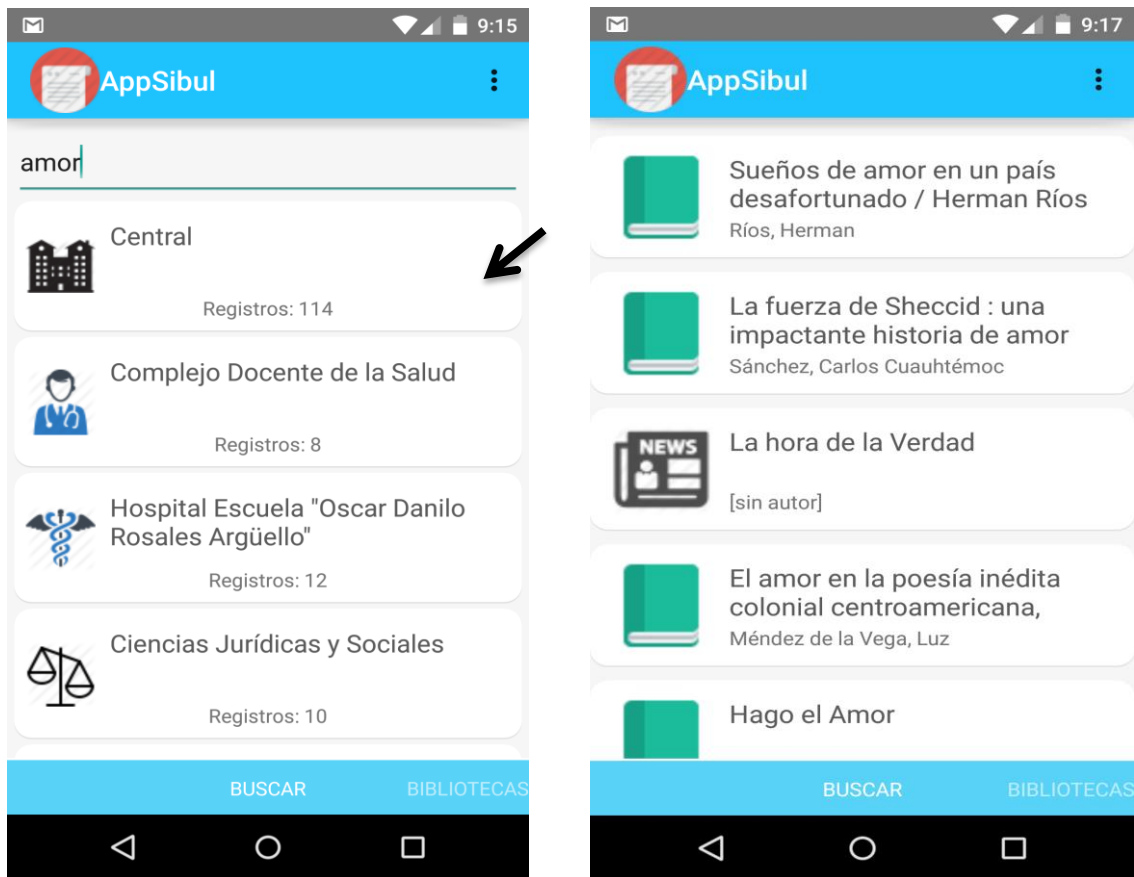
## FUNCIONAMIENTO DE LA APLICACION

A continuación mostraremos las imágenes de nuestra aplicación, mostrando su funcionamiento:



En la imagen izquierda, se muestra el inicio la aplicación con un splash como entrada a la pantalla principal.

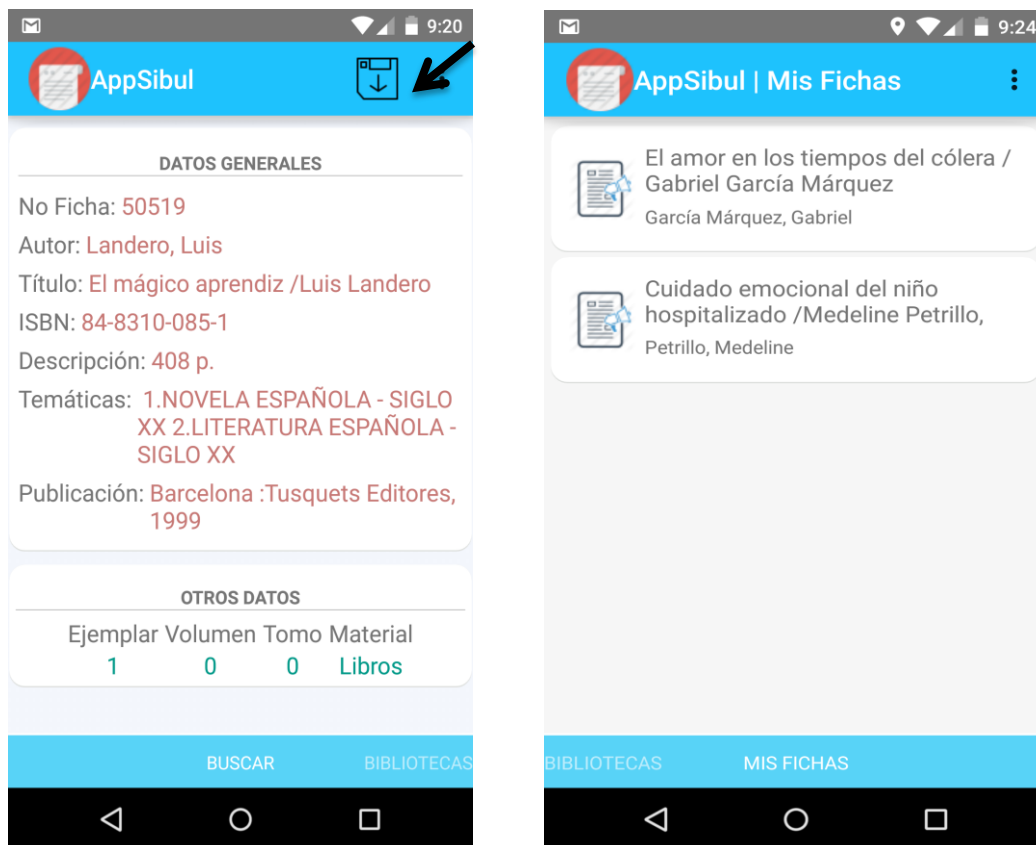
En la imagen derecha, inmediatamente se carga la pantalla de las consultas al catálogo de las bibliotecas, donde podemos escribir las palabras o palabra a buscar.



En la imagen izquierda se muestra el resultado de las bibliotecas relacionadas con el término buscado, elegimos la que deseamos

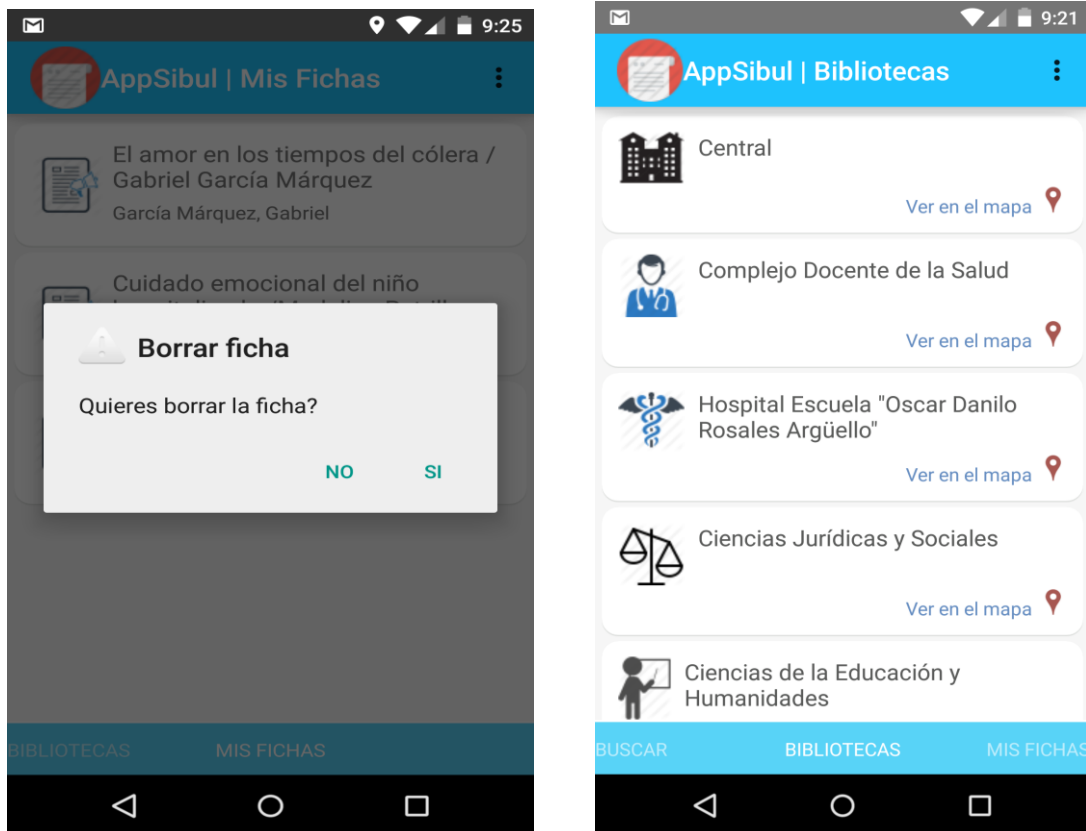
Y en la imagen derecha nos carga la colección de libros, revistas, etc, que posee dicha biblioteca y seleccionamos el libro que buscamos.





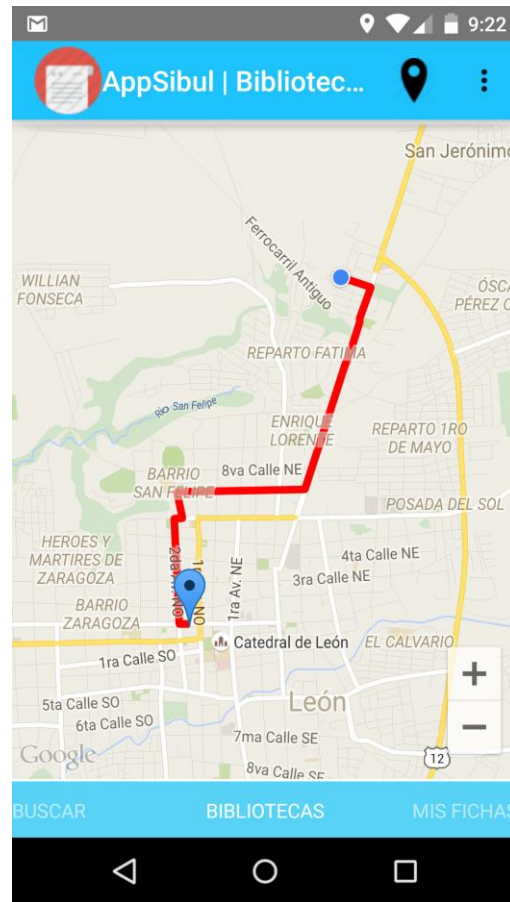
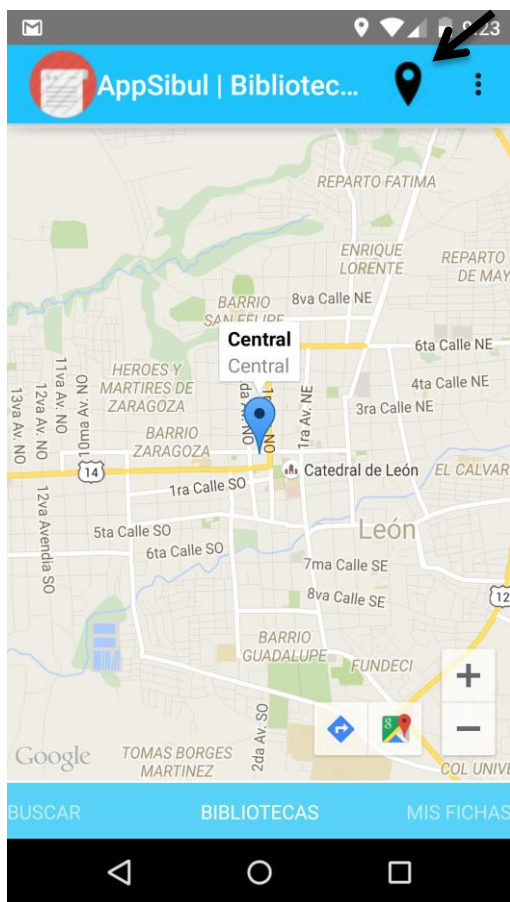
En la imagen izquierda al haber seleccionado un libro se nos carga la ficha catalógrafica de este libro con todos sus detalles, en esta imagen podemos observar el botón de **guardar** para almacenar la ficha localmente.

En la imagen derecha se nos muestra el menú **Mis fichas** donde podemos observar nuestras fichas guardadas.



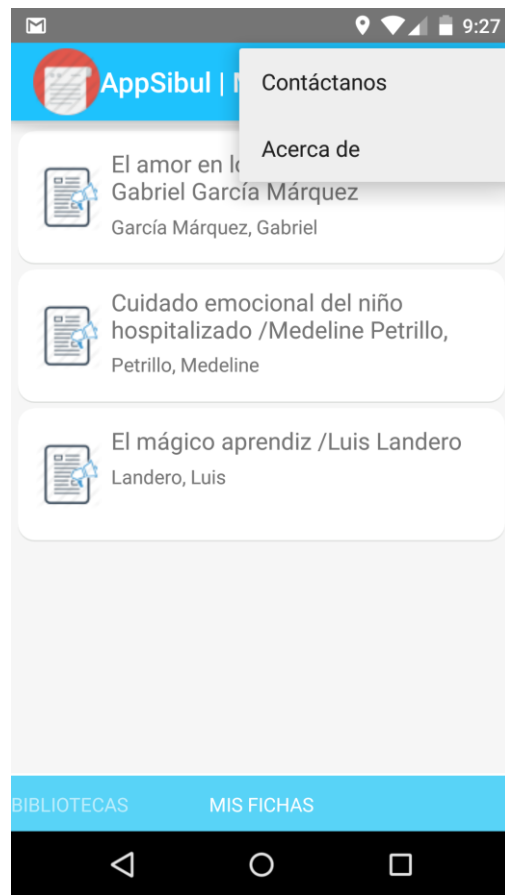
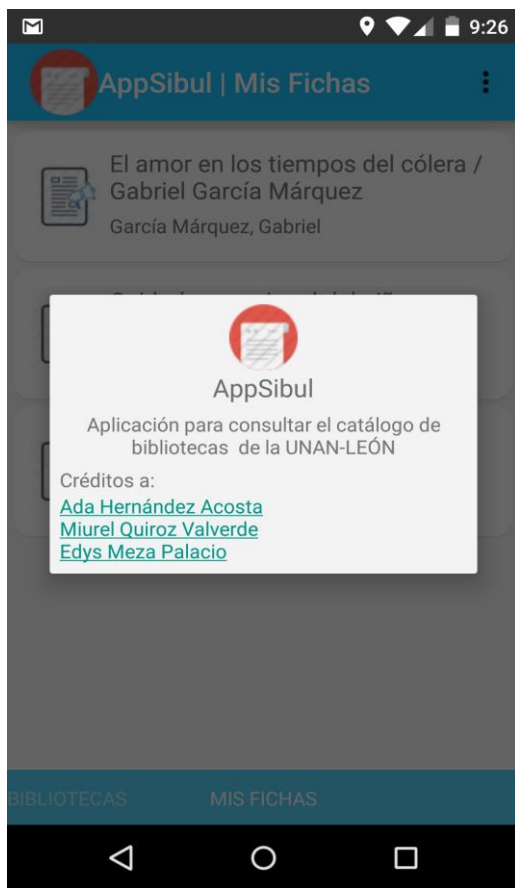
En la imagen izquierda se nos muestra la opción de **Borrar** una ficha si deseamos, damos Si para confirmar.

En la imagen derecha muestra el menú **Bibliotecas** donde se nos muestran todas las bibliotecas de nuestra universidad para que podamos ubicarlas en **Ver en el mapa**.



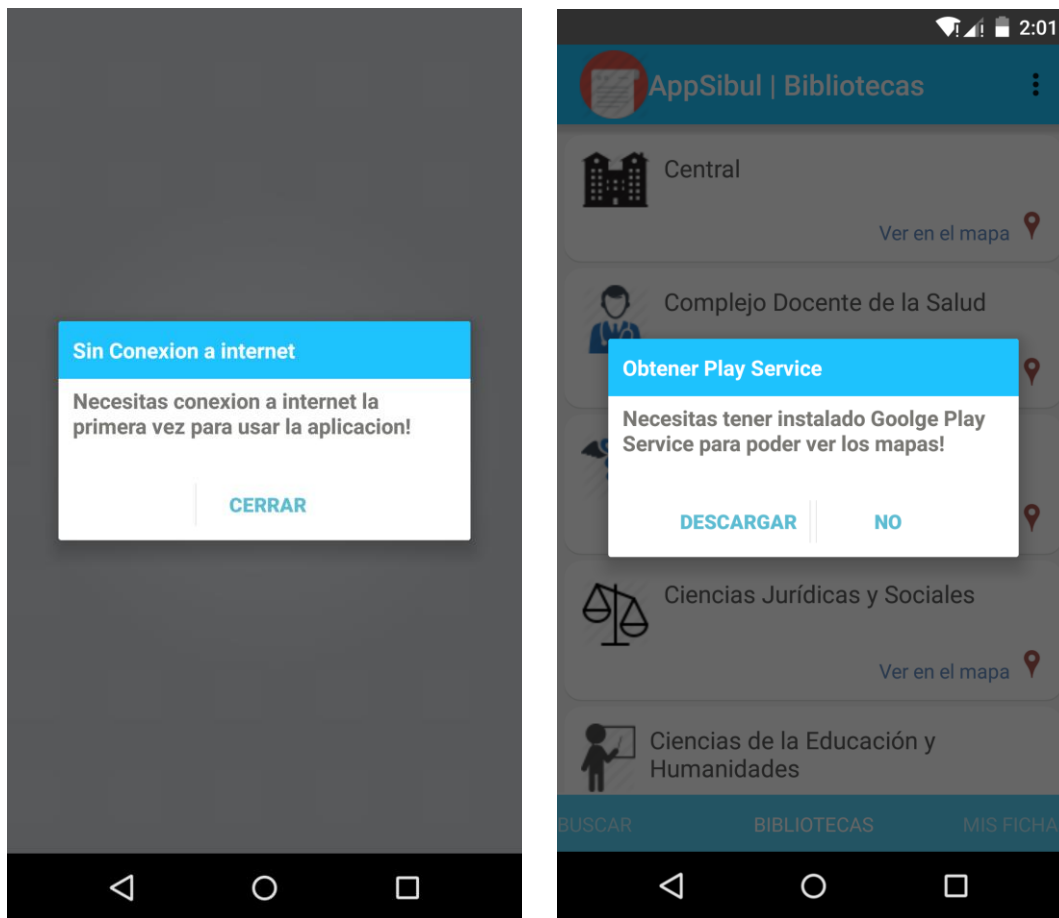
En la imagen izquierda se nos muestra la ubicación exacta de la biblioteca central en la ciudad de León, en la parte de arriba de la pantalla se muestra un botón de **¿Cómo llegar?** a esa biblioteca si usted no conoce.

En la pantalla derecha se nos muestra el camino que se traza en el mapa desde donde estás ubicado hasta la biblioteca destino.



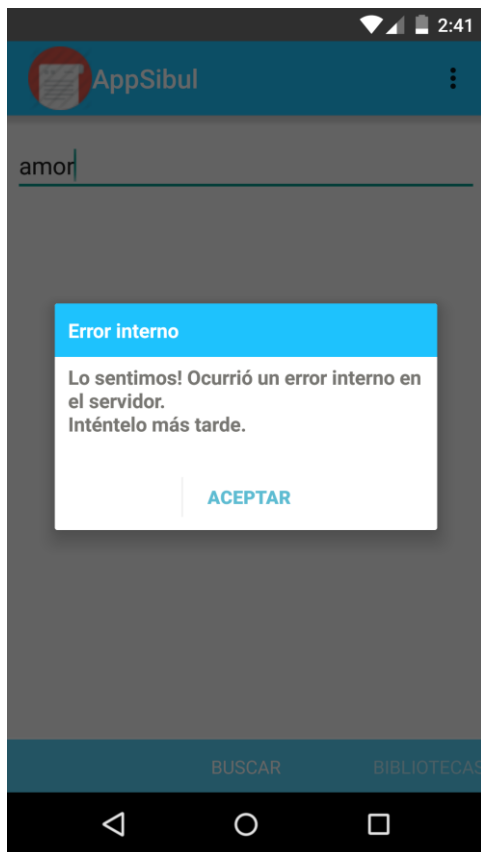
En la imagen izquierda se nos muestra la opción de dar créditos a las personas que desarrollaron esta aplicación.

En la imagen derecha se nos muestra los menús **Acerca de** la aplicación y **Contáctanos**.



En la imagen izquierda se nos muestra un mensaje de la aplicación antes de instalarse porque necesita ciertos complementos y debes tener internet, pero luego puedes usarla sin necesidad de ello para ver tus fichas guardadas localmente.

En la imagen derecha es otro mensaje que aparece porque no tiene tu teléfono los servicios de google para mostrarte la ubicación de las bibliotecas en el mapa.



En esta imagen se nos muestra un mensaje de error interno del servidor y por lo tanto no podrás realizar ninguna búsqueda pero puedes utilizar la aplicación para consultar tus fichas guardadas y la ubicación de las bibliotecas.