

**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***

TEMA:

Aplicación móvil bajo plataforma Android para la interacción y obtención de información de las áreas y especies del Jardín Botánico Ambiental UNAN-León a través de la utilización de Códigos QR y Mapeo Web.

Autores:

Br. Miguel José Jarquín Camacho

Br. Axel Rubén Flores Cornavaca

Tutores:

M.sc. Otón Jossemar Castillo Navas

M.sc. Denis Leopoldo Espinoza Hernández

León, Nicaragua. Noviembre 2017

¡A LA LIBERTAD POR LA UNIVERSIDAD!

RESUMEN

La Universidad Nacional Autónoma de Nicaragua - León (UNAN-León), creó el Jardín Botánico Ambiental (JBA) como una manera más de dar a conocer su quehacer en la docencia, la extensión y la investigación.

El JBA es un medio para fortalecer la cultura ambiental a escala nacional y regional, acerca la ciencia a las personas y contribuye directamente con la conservación de las especies amenazadas del Bosque Seco Tropical.

Actualmente no cuenta con ningún tipo de tecnología dentro del recorrido, solamente con un personal limitado de guías turísticos, los cuales si se hacen presente uno o varios turistas mientras los guías están atendiendo a un grupo cada uno, se dificulta el trabajo impartido por eso para mejorar la situación en la que los turistas no pueden adquirir los conocimientos de un guía, se implementara un proyecto orientado a ofrecer herramientas innovadoras en el recorrido. Lo anterior consiste en añadir capas de información sobre una especie real con la cual se pueda identificar de una manera más fácil e interactiva. Para hacer esto posible, debemos tomar en cuenta que se usaron códigos QR, los cuales nos permitieron conocer la exacta información de la especie de interés, haciendo uso de un dispositivo Android delante del código QR, también el uso de Mapeo Web en la aplicación mostrara la ubicación exacta de los lugares más relevantes del Jardín Botánico, lugar de interés y orientación en el área.

DEDICATORIA

A nuestros Padres y Familiares, por su apoyo incondicional a lo largo de nuestros años de estudio.

A los Docentes y Estudiantes del Departamento de Computación de la UNAN-León, a los que está dirigido este trabajo para ayudarles en el desarrollo de sus conocimientos.

AGRADECIMIENTO

Agradecemos a Dios por darnos fortaleza y su apoyo para seguir adelante en los momentos difíciles de la carrera.

A nuestros Padres, que son pilares muy importantes en nuestras vidas, ya que sin ellos no podríamos haber alcanzado nuestras metas.

A los Tutores, por sus indicaciones acertadas a la hora de realizar este trabajo.

A los Docentes del Departamento de Computación UNAN-León que durante nuestra formación profesional por habernos brindado sus conocimientos para ser cada día mejores.

A nuestros hermanos y familiares que apreciamos tanto.

Contenido

1. INTRODUCCIÓN	7
2. ANTECEDENTES	8
3. PLANTEAMIENTO DEL PROBLEMA	9
4. JUSTIFICACIÓN	10
4.1. Originalidad	10
4.2. Alcance	10
4.3. Producto	11
4.4. Impacto	11
5. OBJETIVOS	12
5.1. Objetivo General	12
5.2. Objetivos Específicos	12
6. MARCO TEORICO	13
6.1. Jardín Botánico Ambiental UNAN-León	13
6.2. Aplicaciones móviles	15
6.2.1. Aplicaciones Móviles en el Turismo Nicaragüense	17
6.3. Sistema Operativo Android	17
6.3.1. Android Studio	18
6.4. Códigos QR	19
6.4.1. ¿Cómo funcionan los códigos QR?	20
6.4.2. Aplicación y uso de los códigos QR	21
6.4.3. Mobile Vision API	21
6.5. Mapeo Web	25
6.5.1. GIS Web en la nube	26
6.5.2. Mapbox	26
6.5.3. ¿Qué tiene Mapbox que no tienen los demás?	27
7. DISEÑO METODOLÓGICO	29
7.1. Descripción General del Proyecto	29
7.2. Fases del proyecto	29
7.2.1. Fase 1: Selección del campo de aplicación y herramientas utilizadas para la aplicación móvil	29
7.2.2. Fase 2: Implementación de Lector de Códigos QR y Diseño de Base de Datos	30
7.2.3. Fase 3: Implementación de Mapeo Web con Mapbox Android SDK y Marcadores	31
7.2.4. Fase 4: Diseño de Interfaz Gráfica	32

7.3. Requerimientos de la aplicación JBA - UNAN León	32
7.3.1. Recursos Hardware	32
7.3.2. Recursos Software	33
8. RESULTADOS	34
8.1. Mostrar la información asociada a las especies del Jardín Botánico Ambiental UNAN-León a través de la lectura de códigos QR	34
8.2. Implementación el mapeo web desde la aplicación para acceder a los recursos asociados a las áreas del Jardín Botánico Ambiental UNAN-León	35
8.3. Proporción de datos históricos y turísticos acerca del Jardín Botánico Ambiental UNAN-León	36
9. CONCLUSION	37
10. RECOMENDACIONES	38
11. REFERENCIAS	39
ANEXOS	40
Anexo 1: Creación de Lector QR usando Mobile Vision API (GOOGLE) y Base de Datos	40
Anexo 2: Implementación de Mapeo Web con Mapbox y Marcadores	45
Anexo 3: Diseño de Interfaz Gráfica	49

INDICE DE FIGURAS

Figura 1: Estructura de un Código QR	20
Figura 2: barcode multiprocessor	24
Figura 3: Diferencia entre Mapbox Android SDK y Google Maps API.....	28
Figura 4: Funcionamiento Básico de la Aplicación	29
Figura 5: Lector de Códigos QR Figura 6: Datos representados en la interfaz	34
Figura 7: Marcadores en Mapa del Jardín Botánico.....	35
Figura 8: Datos Históricos e Información	36
Figura 9: Petición de Permisos de Cámara del Sistema	41
Figura 10: Creación de cuenta e inicio de sección en Mapbox	45
Figura 11: Generación de Llave en Mapbox.....	45
Figura 12: Representación del Jardín Botánico en Mapbox	47
Figura 13: Icono de la aplicación móvil.....	49
Figura 14: Splash	51
Figura 15: Menú de la aplicación.....	53
Figura 16: Funcionamiento de AppLayout	54
Figura 17: Interfaz de información de especies (MainActivity.java)	55
Figura 18: Ventanas emergentes caso de carácter y entero no en base de datos respectivamente...56	



1. INTRODUCCIÓN

En la actualidad en internet se encuentra un sin número de aplicaciones móviles que sirven para la consulta de lugares turísticos, algunas de estas son desarrolladas con un eje general del turismo nicaragüense, por lo tanto, no contienen información completa del ecoturismo del país.

El propósito del presente proyecto es la implementación de una herramienta tecnológica que sirva como una guía turística y pueda ser implementada en plataformas de dispositivos móviles Android, esta permitirá tener indagación del Jardín Botánico Ambiental UNAN-León para brindar información única y precisa a las diversas áreas y especies que contiene.



2. ANTECEDENTES

En un principio, los dispositivos móviles fueron considerados como un artículo de lujo al que solo grandes empresarios o directivos podían tener acceso; sin embargo; con el paso del tiempo este paradigma se rompió y los dispositivos móviles se han convertido en una de las principales herramientas de entretenimiento, comunicación y productividad para la población en general.

Gracias al avance de esta tecnología y gracias a que los dispositivos móviles disponen de mayor capacidad de procesamiento e incluyen cámaras digitales, sensores de última generación y sistemas de localización global, ha sido posible desarrollar aplicaciones más precisas y asequibles. Actualmente, es habitual encontrar aplicaciones que funcionan en teléfonos móviles y que son capaces de mezclar lugares de interés con información digital relevante, visualizar contenidos adicionales en museos, tales como reconstrucciones, y una gran cantidad de ejemplos:

En el desarrollo de lector de códigos QR están las aplicaciones como “Jardín Botánico da Madeira” que utiliza la interactiva de lector QR para algunos sitios del Jardín en Portugal. [1]

También un Sistema para la obtención de información de los espacios y monumentos de la Catedral de León a través de dispositivos móviles. UNAN – León, tesis que se desarrolló en 2015. [2]

En el desarrollo de mapeo web se encuentran aplicaciones como “COMPDES 2016” que utiliza google maps para mostrar la localización de los edificios donde se efectuaron ponencias del congreso. [3]



3. PLANTEAMIENTO DEL PROBLEMA

La poca información que existe del Jardín Botánico como punto turístico de la ciudad de León, afecta de manera directa a las personas que desean invertir en este sector, ocasionando un serio problema al desarrollo ecoturístico.

La carencia de herramientas tecnológicas en el Jardín Botánico, hace poco atractiva turísticamente en la ciudad de León.

Pregunta General:

- ¿Cómo se puede tener una aplicación móvil que permita a los turistas que visiten el Jardín Botánico, obtener información multimedia de las diferentes áreas y especies a través de la utilización de códigos QR y mapeo web?

Pregunta específica:

- ¿Qué mecanismo se ha de utilizar para almacenar la información de cada recurso asociado a las áreas y especies del Jardín Botánico?



4. JUSTIFICACIÓN

Teniendo en cuenta que el turismo representa un potencial enorme de desarrollo y progreso para el futuro del Jardín Botánico, se plantea como proyecto de tesis una aplicación móvil que permita brindar Información con la facilidad de saber de antemano los servicios que este puede brindar y las actividades que el turista puede realizar en el Jardín Botánico.

El desarrollo de este proyecto se basa en la utilidad que representa hoy en día el uso de las aplicaciones móviles y como esta puede aportar en el desarrollo del turismo.

4.1. Originalidad

Anteriormente no se ha desarrollado ninguna aplicación móvil que permita resolver la necesidad de informar a los turistas locales y extranjeros sobre el Jardín Botánico, por lo que este trabajo será de gran ayuda para adquirir información que sea de su interés y le permita visualizar las distintas opciones para obtener la información de la manera deseada.

4.2. Alcance

A través de la aplicación los turistas podrán interactuar mediante el uso del lector de códigos QR, generando la genuina información de las distintas especies de flora haciendo que esta información se “consume” y se difunda de forma viral. El turista también podrá tener orientación durante el recorrido mediante el uso del mapa que contiene la aplicación observado así los puntos de interés en el jardín. La aplicación también contendrá información relevante sobre la historia del jardín botánico.



4.3. Producto

El resultado de este proyecto es una aplicación móvil para el Sistema Operativo Android sencilla al usuario, que incorpore todas las funcionalidades que se requieren para brindarle un recorrido satisfactorio y placentero. Además, les permita adquirir la información de manera interactiva.

4.4. Impacto

Con la aplicación JBA – UNAN León se contribuirá a incrementar la cantidad de turistas que se visitan el jardín botánico a la vez que ayuda a mejorar la experiencia de dicha visita y así tener un mayor impacto como atractivo turístico en la ciudad de León.



5. OBJETIVOS

5.1. Objetivo General

Desarrollar una aplicación móvil en Android 4.3 capaz de proporcionar información multimedia de las diferentes áreas y especies del Jardín Botánico Ambiental UNAN-León en el período de Junio – Noviembre del 2017.

5.2. Objetivos Específicos

- Mostrar la información asociada a las especies del Jardín Botánico Ambiental UNAN-León a través de la lectura de códigos QR.
- Implementar el mapeo web desde la aplicación para acceder a los recursos asociados a las áreas del Jardín Botánico Ambiental UNAN-León.
- Proporcionar a través de la aplicación datos históricos y turísticos acerca del Jardín Botánico Ambiental UNAN-León.



6. MARCO TEORICO

6.1. Jardín Botánico Ambiental UNAN-León

La primera idea acerca del Jardín Botánico Ambiental, surgió en el año 2004, como parte del Proyecto: “Desarrollando Capacidades y Compartiendo Tecnologías para la Conservación de la Biodiversidad en Centroamérica” el que reunía a los herbarios centroamericanos desde el año 2003.

El precursor de esta idea fue el Dr. Ricardo Rueda, Botánico Nicaragüense, quien junto con un grupo de profesores del Departamento de Biología de la UNAN-León, iniciaron su formulación en el año 2006, acompañados por un equipo técnico del INBio, Costa Rica.

Como parte de los esfuerzos de búsqueda de financiamiento para la iniciativa, se presentó el proyecto a la cooperación Noruega, quien se interesó en la idea y aprobó el financiamiento para una fase inicial, la que se inició en el año 2008, bajo el rectorado del Dr. Rigoberto Sampson Granera, quien le dio un empuje importante a esta iniciativa, luego de haber puesto las bases desde 1997 en su calidad de Alcalde Municipal al haber concedido a la UNAN-León el terreno en donde se establecería el JBA.

Es debido a ese apoyo e interés que manifestó este rector amante de la naturaleza, que se concluyó lo que para la Universidad Nacional Autónoma de Nicaragua - León y los profesores del Departamento de Biología era un sueño, es por eso que se decidió dedicar este esfuerzo al Dr. Rigoberto Sampson "In Memoriam".

El 3 de junio del año 2009 con la presencia de autoridades Municipales, Gobierno de Noruega, y la UNAN-León, se colocó la primera piedra, del primer Jardín Botánico de Nicaragua, una obra maestra en pro de la Conservación de los Recursos Naturales del país.



En la actualidad el JBA es un sitio que integra en un solo espacio actividades de docencia, investigación y extensión, quehaceres primordiales en la UNAN-León.

Para la UNAN-León este proyecto representa una oportunidad de reafirmar a la comunidad nacional e internacional su compromiso con el ambiente, con la formación profesional integral de sus estudiantes y particularmente con la comunidad nacional que requiere espacios de recreación y aprendizaje como este.

Este es un museo viviente, que alberga colecciones vivas de especies representativas de los ecosistemas nicaragüenses, como el Bosque Húmedo, Bosque de Transición y el Bosque Seco, el ecosistema más deteriorado de la región.

El JBA se establece con el objetivo de desarrollar un área de importancia ecológica que contribuya al desarrollo del conocimiento y la labor docente, así como el manejo de aspectos sociales y o ambientales, contribuyendo al mejoramiento de la calidad de vida de la población de la ciudad de León, a través de componentes educativos, científicos, recreativos y sociales.

El JBA se encuentra en una propiedad de 12 hectáreas a 2 Km al oeste del centro urbano de León, rodeada por los ríos Los Aposentos y El Ojoche.

En particular, este jardín, que ya ha sido inscrito como parte de la red internacional de jardines botánico, tiene el objetivo, desde el punto de vista botánico, de ser el sitio en que se conserve la diversidad botánica de las plantas del Bosque Seco de Mesoamérica, uno de los ecosistemas más degradados de la región, y más amenazados por el cambio climático. El Bosque Seco es el hábitat terrestre que está en mayor peligro en Centroamérica, actualmente reducido a menos del 1% de su extensión original, y completamente fragmentado.

Además de plantas del Bosque Seco, el visitante encuentra en el JBA especies de plantas del Bosque de Transición y el Bosque Húmedo Nicaragüense, y un jardín de mariposas, todo esto con el objeto de acercar a la comunidad nicaragüense a su riqueza natural. También se incluye un área demostrativa de



prácticas agrícolas y ambientales sostenibles, lo que, en conjunto con las áreas de jardines, permite dar la visitante una visión integral de las relaciones que en doble vía se dan entre las personas y la naturaleza.

El JBA ha sido diseñado con la idea de atender necesidades educativas y recreativas del Occidente de Nicaragua y de la población del país en general, tanto familias como estudiantes de primaria y secundaria, así como específicamente estudiantes de diferentes carreras de la UNAN León que encuentran en el JBA un laboratorio para sus prácticas en las carreras de Agroecología, Biología, Turismo y Educación. También el JBA atenderá el turismo internacional que visita la zona, por lo que se cuenta con rotulación en español e inglés.

Los visitantes, mediante recorridos guiados o autoguiados podrán conocer y aprender sobre la riqueza natural de la región, así como sobre prácticas agropecuarias y ambientales sostenibles relacionadas con uso de químicos, manejo del agua y el uso de energías limpias.

En el aspecto botánico, el JBA tendrá un carácter Mesoamericano, centrado en la representación y conservación de las especies de plantas del Bosque Seco y de los principales ecosistemas de León. También estarán representados el Bosque de Transición, Bosque de Galería y el Bosque Húmedo de Nicaragua. [4]

6.2. Aplicaciones móviles

Una aplicación móvil es un programa informático diseñado para ejecutarse en los teléfonos inteligentes, computadoras Tablet y otros dispositivos móviles. Las Aplicaciones suelen estar disponibles a través de las plataformas de distribución de aplicaciones, que comenzaron a aparecer en 2008 y por lo general son operados por el propietario del sistema operativo para móviles, como el de Apple App Store, Google Play, Windows Store



Por lo general, se descargan desde la plataforma de un dispositivo de destino, como un iPhone, BlackBerry, Android, o Windows Phone, pero a veces pueden ser descargados a las computadoras portátiles o computadoras de escritorio. Para aplicaciones con un precio, en general, un porcentaje, el 20-30%, va al proveedor de distribución (como iTunes), y el resto va al productor de la aplicación. (Vyas, 2014)

La misma aplicación, por tanto, puede costar al usuario promedio un precio diferente dependiendo si utilizan iPhone, Android o dispositivos BlackBerry 10. El término "app" es una abreviación del término "software de aplicación". Se ha vuelto muy popular y en 2010 fue catalogado como " Palabra del Año "por la Sociedad Americana del Dialecto. (Sáenz, 2011)

Las aplicaciones móviles se ofrecieron originalmente para la productividad general y la recuperación de información, incluido el correo electrónico, calendario, contactos y mercado de valores y la información meteorológica. Sin embargo, la demanda pública y la disponibilidad de herramientas de desarrollo impulsaron la rápida expansión en otras categorías, como juegos móviles, automatización de fábricas, GPS y servicios basados en la ubicación, la banca, seguimiento de pedidos, compra de entradas y recientemente aplicaciones médicas móviles. (QR, 2013)

La explosión en el número y variedad de aplicaciones hechas descubrieron un nuevo reto, que a su vez condujo a la creación de una amplia gama de revisiones, recomendaciones, y las fuentes de información, incluyendo blogs, revistas y servicios de revisión de aplicaciones de en línea dedicados. La popularidad de las aplicaciones móviles ha seguido aumentando, ya que su uso es cada vez más frecuente entre los usuarios de telefonía móvil.

Las aplicaciones móviles se presentan en dos formatos distintos: aplicaciones nativas y aplicaciones web para móviles. Debido a las diferencias en



su tecnología subyacente, cada método tiene ventajas y desventajas inherentes. (IBM, 2012)

6.2.1. Aplicaciones Móviles en el Turismo Nicaragüense

El desarrollo de aplicaciones para teléfonos móviles ha cobrado relevancia en los últimos años en Nicaragua, esto ha sido tomado por diversas compañías con el objetivo de promover las bellezas naturales y sitios turísticos del país de una manera ágil a nivel nacional e internacional.

El Instituto Nicaragüense de Turismo (INTUR) desarrolló en el 2016 la plataforma “Nicaragua” en la que los usuarios de manera gratuita encuentran tres grandes componentes: Atractivos, Directorios y Geolocalización. La aplicación móvil ‘Nicaragua’ es parte de la estrategia que viene implementando para la promoción del país como destino turístico. [5]

La Cámara Nacional de Turismo de Nicaragua (Canatur) lanzó en abril del 2017 la aplicación móvil “Ruta Feliz”, que promueve unos 20 destinos turísticos del país y 700 establecimientos. [6]

6.3. Sistema Operativo Android

Android es un sistema operativo de código abierto basado en el sistema operativo Linux para dispositivos móviles como teléfonos inteligentes, computadoras y tabletas. Android fue desarrollado por Open Handset Alliance, liderado por google y otras compañías. Android ofrece un enfoque unificado para el desarrollo de aplicaciones para dispositivos móviles que significa que los desarrolladores necesitan sólo desarrollar para Android, y sus aplicaciones deben ser capaces de funcionar en diferentes dispositivos impulsados por Android.



La primera versión beta del kit de desarrollo de software de Android (SDK) fue lanzado por Google en 2007 donde como la primera versión comercial, Android 1.0, fue lanzado en septiembre de 2008. (Menéndez, 2012)

El 21 de marzo de 2017, Google lanzó la primera versión preliminar para desarrolladores de Android "O", Google está programado para lanzar un total de cuatro previas para desarrolladores; La versión final de Android 8.0 Oreo fue liberada al público el 21 de agosto de 2017, para los dispositivos Nexus y Pixel compatibles, la cual incluye nuevas mejoras que aseguran la estabilidad del sistema.

El código fuente de Android está disponible bajo licencias de software libre y de código abierto. Google publica la mayoría del código bajo la licencia apache versión 2.0 y el resto, los cambios del kernel de Linux, bajo la GNU (General Public License) versión 2.

6.3.1. Android Studio

Android Studio es un entorno de desarrollo integrado (IDE) para el desarrollo en la plataforma Android. Se anunció el 16 de mayo de 2013 a la conferencia Google I/O de Google de Product Manager, Ellie Powers. Android Studio está disponible libremente bajo la licencia Apache 2. Android estudio estaba en fase de acceso primera vista previa partir de la versión 0.1 en mayo de 2013, luego entró en fase beta a partir de la versión 0.8, que fue lanzado en junio de 2014. La primera versión estable fue lanzada en diciembre de 2014, a partir de la versión 1.0. Basado en el software IDEA IntelliJ JetBrains, el estudio se ha diseñado específicamente para el desarrollo de Android. Está disponible para su descarga en Windows, Mac OS X y Linux, y sustituida Eclipse como IDE principal de Google para desarrollo de aplicaciones de Android nativo.



Características:

- Renderización en tiempo real.
- Consola de desarrollador: consejos de optimización, ayuda para la traducción, estadísticas de uso.
- Soporte para construcción basada en gradle.
- Refactorización específica de Android y arreglos rápidos.
- Herramientas lint para detectar problemas de rendimiento, usabilidad, compatibilidad de versiones, y otros problemas.
- Plantillas para crear diseños comunes de Android y otros componentes.
- Soporte para programar aplicaciones para Android Wear.

6.4. Códigos QR

Un código QR según sus siglas en inglés Quick Response (Respuesta Rápida) es un código de barras utilizado para contener información en una matriz de puntos bidimensional tanto horizontal como vertical que contiene información codificada en ella, a diferencia de los tradicionales códigos de barras, los cuales solo pueden almacenar información en una sola dimensión horizontal.

Dentro de las características técnicas de estos códigos es que pueden contener mucha más información que los antiguos códigos de barras. Concretamente, pueden contener hasta:

- 4.296 caracteres alfanuméricos
- 7.089 caracteres numéricos
- 2.953 caracteres codificados como byte
- 1.817 caracteres japoneses (Kanji).

En la actualidad estos códigos se han convertido en una excelente herramienta del Marketing móvil.



6.4.1. ¿Cómo funcionan los códigos QR?

Un código QR se distingue por ser un cuadrado con tres pequeños cuadros en tres esquinas y gráficamente se puede identificar varias regiones adicionales.

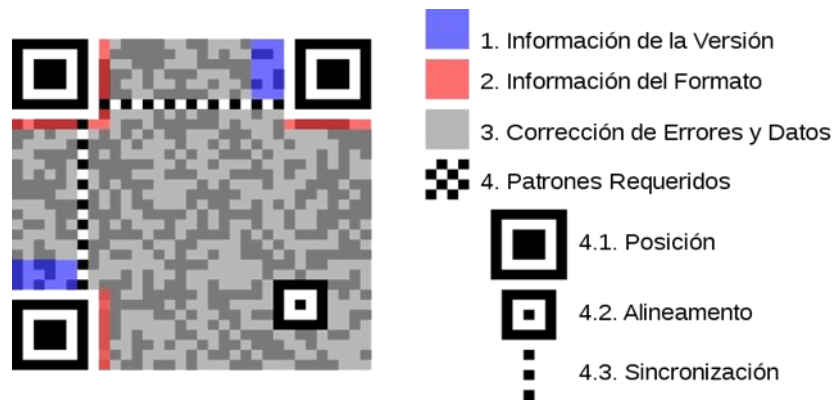


Figura 1: Estructura de un Código QR

Para poder decodificarlo es necesario el uso de un teléfono móvil con cámara y de un programa que sea capaz de traducir este tipo de lenguaje. Estos códigos pueden ser visualizados desde cualquier ángulo sin perder la información original contenida en ellos.

Teniendo en cuenta la diversidad de marcas y modelos de equipos celulares existentes en el mercado, el usuario deberá identificar el lector de código QR más apropiado de acuerdo a su Sistema Operativo.

- ANDROID
- IPHONE
- BLACKBERRY
- SYMBIAN
- Otros



Para obtener un lector de código QR, se recomienda dentro del “Market place” del terminal, buscar por las palabras en inglés “Reader QR”, con lo que obtendremos distintas posibilidades de descargas de lectores de código.

6.4.2. Aplicación y uso de los códigos QR

Los códigos QR conectan el mundo real con el mundo digital, estos pueden contener información básica desde texto, un número telefónico, un mensaje de texto - SMS, una tarjeta de presentación - Vcard, hasta contenidos más complejos como direcciones URL. [7]

Algunas de las aplicaciones prácticas que se le pueden dar a este tipo de códigos son entre otras:

- Educación.
- Artículos en revistas.
- Vallas.
- Publicidad.
- Ventas de productos.
- Libros o enciclopedias.
- Manuales de instrucciones.
- Cupones virtuales.
- Geo posicionamiento.
- Escarapelas en eventos.

6.4.3. Mobile Vision API

La API Mobile Vision proporciona un marco para encontrar objetos en fotos y videos. El marco incluye detectores, que ubican y describen objetos visuales en



imágenes o marcos de video, y una API impulsada por eventos que rastrea la posición de esos objetos en video. [8]

Actualmente, Mobile Vision API incluye cara, código de barras y detectores de texto, que se pueden aplicar por separado o en conjunto.

El paquete de visión incluye un marco de funcionalidad base común y subpaquetes para implementaciones de detectores específicos:

- Funcionalidad común: **com.google.android.gms.vision**
- Detector de rostro: **com.google.android.gms.vision.face**
- Detector de código de barras: **com.google.android.gms.vision.barcode**
- Detector de texto: **com.google.android.gms.vision.text**

Descripción general de API de código de barras en Android.

Barcode API detecta los códigos de barras en tiempo real, en el dispositivo, en cualquier orientación. También puede detectar múltiples códigos de barras a la vez.

Lee los siguientes formatos de código de barras:

- 1D barcodes: **EAN-13, EAN-8, UPC-A, UPC-E, Code-39, Code-93, Code-128, ITF, Codabar**
- 2D barcodes: **QR Code, Data Matrix, PDF-417, AZTEC**

Analiza automáticamente los códigos QR, Data Matrix, PDF-417 y los valores de Aztec, para los siguientes formatos admitidos:

- URL
- Información de contacto (VCARD, etc.)
- Evento de calendario



- Email
- Teléfono
- SMS
- ISBN
- Wifi
- Geo-ubicación (latitud y longitud)
- Licencia de conducir / identificación de AAMVA

Crear un detector de código de barras

Crea un detector de código de barras para rastrear códigos de barras:

```
BarcodeDetector barcodeDetector = new BarcodeDetector.Builder(context).build ();
```

Por defecto, el detector de códigos de barras buscará todos los formatos de códigos de barras admitidos.

Alternativamente, puede especificar un subconjunto de formatos para detectar mediante el método `BarcodeDetector.Builder.setFormats` . Reducir la cantidad de formatos para detectar hará que el detector de códigos de barras sea más rápido. [9]

Crear un multiprocesador para administrar los códigos de barras detectados

Debe proporcionar un `TrackerFactory` para crear una nueva instancia de `Tracker` para cada código de barras. Para este ejemplo, `BarcodeTrackerFactory` crea un `BarcodeGraphic` que se superpone al cuadro delimitador y al valor de cada resultado del código de barras:

```
BarcodeTrackerFactory barcodeFactory = new BarcodeTrackerFactory(mGraphicOverlay);  
barcodeDetector.setProcessor( new MultiProcessor.Builder<>(barcodeFactory).build())
```




Conceptualmente, esto crea una porción de la tubería de procesamiento del detector que se ve así:

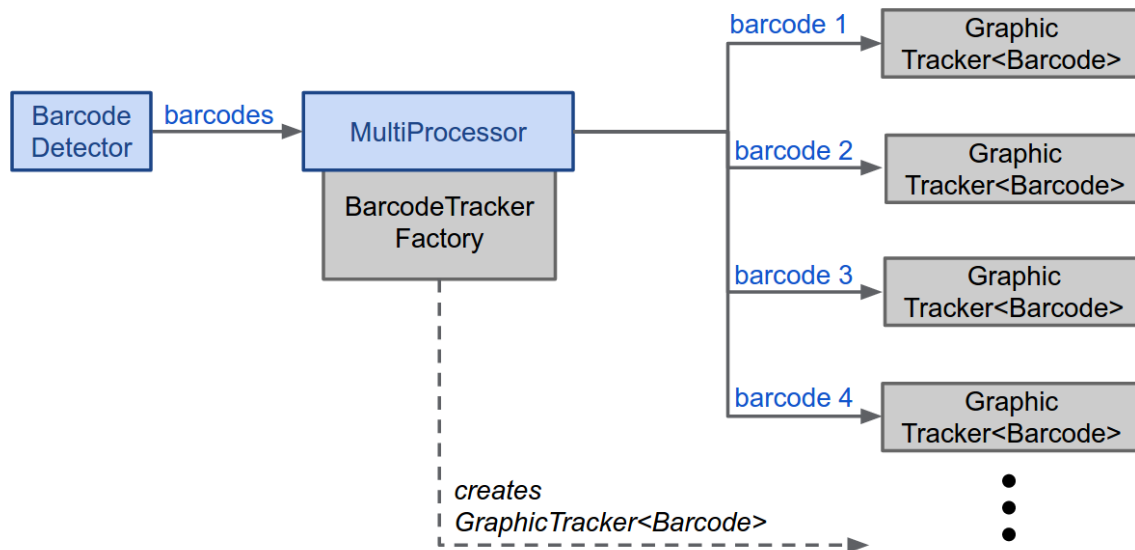


Figura 2: barcode multiprocessor

Esta parte de la tubería funciona de la siguiente manera:

1. El detector de código de barras detecta los códigos de barras y crea una colección de instancias de código de barras.
2. Una instancia de multiprocesador realiza un seguimiento de cada código de barras que está activo actualmente. Utiliza una fábrica para crear una nueva instancia de rastreador gráfico por código de barras.
3. Como los códigos de barras se rastrean en los marcos de video, el multiprocesador envía actualizaciones a las instancias correspondientes del rastreador de códigos de barras. En este caso, las actualizaciones del código de barras son simplemente la posición y el tamaño del código de barras dentro del marco de video.



6.5. Mapeo Web

El mapeo web es el proceso de usar mapas entregados por los sistemas de información geográfica (GIS). Se sirve y se consume un mapa web en la World Wide Web, por lo que el mapeo web es más que una cartografía web, es un servicio mediante el cual los consumidores pueden elegir lo que mostrará el mapa. Web GIS enfatiza aspectos de procesamiento de geodatos más involucrados con aspectos de diseño tales como adquisición de datos y arquitectura de software de servidor, como almacenamiento de datos y algoritmos, que los informes del usuario final.

Los términos web SIG y mapeo web siguen siendo sinónimos. Web GIS utiliza mapas web, y los usuarios finales que están mapeando en la web están ganando capacidades analíticas. El término servicios basados en la ubicación se refiere a la cartografía web de bienes y servicios de consumo. El mapeo web generalmente involucra un navegador web u otro agente de usuario capaz de interacciones cliente-servidor. Las cuestiones de calidad, usabilidad, beneficios sociales y restricciones legales están impulsando su evolución.

El advenimiento del mapeo web se puede considerar como una nueva tendencia importante en la cartografía. Hasta hace poco, la cartografía estaba restringida a unas pocas empresas, institutos y agencias de cartografía, que requerían hardware y software relativamente caros y complejos, así como cartógrafos especializados e ingenieros en geomática.

El mapeo web ha traído muchos conjuntos de datos geográficos, incluidos los gratuitos generados por OpenStreetMap y los conjuntos de datos propiedad de Navteq, Google, Waze y otros. También se ha concebido e implementado una gama de software libre para generar mapas junto con herramientas propietarias como ArcGIS. Como resultado, la barrera de entrada para servir mapas en la web se ha reducido. [10]



6.5.1. GIS Web en la nube

Varias compañías ahora ofrecen mapeo web como un software basado en la nube como un servicio. Estos proveedores de servicios permiten a los usuarios crear y compartir mapas cargando datos en sus servidores (almacenamiento en la nube). Los mapas se crean utilizando un editor en el navegador o escribiendo scripts que aprovechan las API de los proveedores de servicios. [10]

Principales empresas de GIS web:

- ArcGIS Online - Mapeo sin límites.
- Mango - GIS web hecho fácil.
- Mapbox: la plataforma de ubicación para desarrolladores y diseñadores.
- BatchGeo - Crea un mapa interactivo a partir de tus datos.

6.5.2. Mapbox

Mapbox es un gran proveedor de mapas en línea personalizados para sitios web y aplicaciones como Foursquare, Lonely Planet, Evernote, Financial Times, The Weather Channel y Snapchat. Desde 2010, ha ampliado rápidamente el nicho de mapas personalizados, como respuesta a la opción limitada que ofrecen los proveedores de mapas como Google Maps. Mapbox es el creador de, o un colaborador significativo de algunas bibliotecas y aplicaciones de mapeo de código abierto, incluida la especificación MBTiles, el IDE de la cartografía TileMill, el Folleto La biblioteca de JavaScript y el lenguaje de estilo de mapas de CartoCSS y el analizador.



6.5.3. ¿Qué tiene Mapbox que no tienen los demás?

- Todo es 100% personalizable. Desde cualquier color hasta mostrar/ocultar capas en el mapa. Hay una serie de estilos disponibles por defecto (Dark, Light, etc.) pero también tienes la posibilidad de diseñar/crear tus propios estilos por completo a través de la herramienta Mapbox Studio. Recientemente se lanzó Cartogram, que te genera un estilo automáticamente a partir de una imagen. Los estilos se integran en las aplicaciones añadiendo una sola línea de código. Como puedes imaginar, todo ello te permite generar mapas acordes con el branding de tu app y así ofrecer una mejor experiencia a tus usuarios.
- Open source. Todo el código está abierto y basado en estándares abiertos. Mapbox dispone de más de 500 repositorios en Github. Esto te permite ver las funcionalidades que están en desarrollo, reportar cualquier tipo de problema que puedas tener o incluso contribuir mandando PRs. Mapbox es una plataforma hecha por desarrolladores para desarrolladores.
- No importa para qué plataforma desarrolles porque hay SDKs disponibles para prácticamente todas. Entre las cuales están Android, iOS, Web, Qt, Unity y MacOS. Además, todas las funcionalidades están accesibles desde cada uno de ellos, ya que comparten una misma base.
- In-app. Todo ocurre en tu app. No hay necesidad de tener que saltar a otra aplicación para nada (Mapbox no tiene apps para usuarios así que no tiene interés en competir con sus desarrolladores). Esto es especialmente útil si estás desarrollando una app de navegación, ya que, a diferencia de otros SDKs, puedes proveer una experiencia de navegación completa, sin perder a tus usuarios en ningún momento. Otro ejemplo es la posibilidad de guardar mapas offline sin salir de tu app.
- El Core está basado en mapas vectoriales implementados en C++, de forma que únicamente se mandan los datos necesarios al dispositivo y se



interpretan en tiempo real, lo que se traduce en mapas muy rápidos y en visualizaciones mucho más ligeras.

- Tanto el SDK de Android como el de iOS tienen una API similar a la de Google Maps y MapKit respectivamente. Por lo que todo el conocimiento que ya tuvieras te seguirá siendo útil y además migrar a Mapbox no te costará nada. Para hacer el cambio basta con hacer un Reemplazar todos a través del IDE y listo.

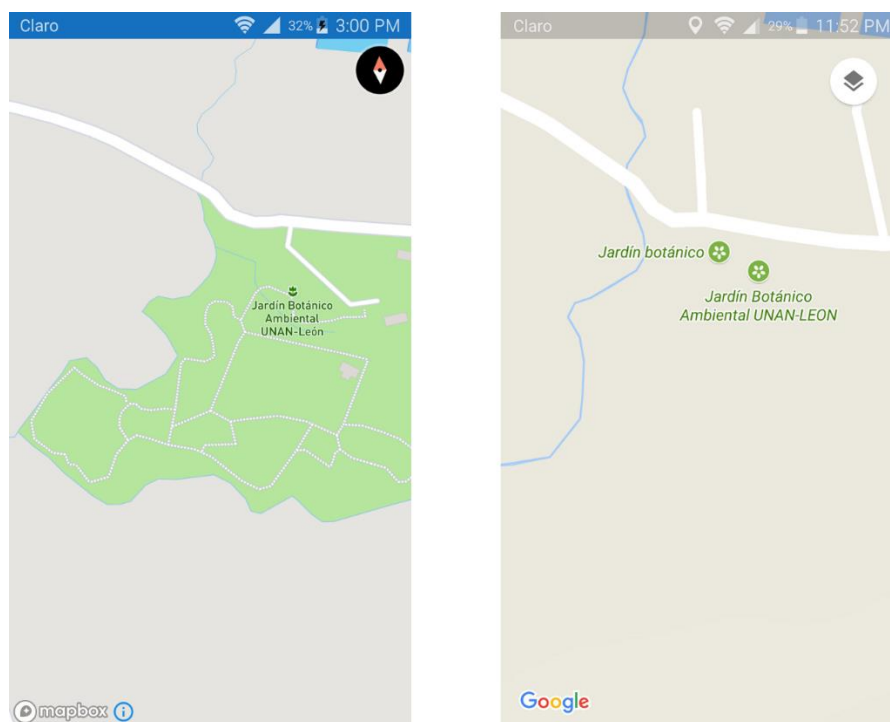


Figura 3: Diferencia entre Mapbox Android SDK y Google Maps API

En conclusión, Mapbox trae muchas funcionalidades que no nos ofrecen otros frameworks. Y esto tan solo es una pequeña muestra de lo que podemos hacer, ya que aparte del SDK de mapas, tenemos disponibles el Mapbox Services SDK (APIs de Geocoding, Map Matching, Directions, etc.), el Navigation SDK o los Mapbox Plugins.



7. DISEÑO METODOLÓGICO

7.1. Descripción General del Proyecto

Se desarrolló una aplicación móvil que permita a los turistas que visitan el Jardín Botánico tener mayor interacción a la información haciendo uso de teléfonos inteligente. Con el cual la aplicación deberá solicitar información de interés para el visitante desde un código QR para luego acceder a la base de datos el cual tendrá toda la información que le pueda ser de utilidad.



Figura 4: Funcionamiento Básico de la Aplicación

7.2. Fases del proyecto

7.2.1. Fase 1: Selección del campo de aplicación y herramientas utilizadas para la aplicación móvil

En la actualidad las aplicaciones móviles se emplean en muchos campos, se opta en el turismo, porque es posible mostrar a los viajeros las diversas especies de flora y biomas que se desarrollan en la región, aumentando el interés por aprender a través de la interacción y la visualización de la información de estos. Esta tecnología servirá de apoyo al desarrollo de los conocimientos del turista.



El fin es hacer una contribución de apoyo para brindar información a los turistas que visiten el Jardín Botánico Ambiental UNAN-León.

La plataforma de trabajo utilizada es el IDE Android Studio que proporciona las herramientas más rápidas para crear aplicaciones en todas las clases de dispositivos Android de forma gratuita.

7.2.2. Fase 2: Implementación de Lector de Códigos QR y Diseño de Base de Datos

En esta fase se desarrolló un lector de códigos QR capaz de acceder a la cámara del dispositivo y notifica cuando hay un código QR dentro de la vista previa, implementado con base a las librerías de Mobile Vision API.

Se programó un SQLiteHelper, este posee un onCreate, en este apartado tenemos la creación de nuestra única tabla llamada **db_jba** que contendrá toda nuestra información de las plantas (Campos), URL local de las imágenes, en la misma función insertamos los datos a nuestra base de datos, con un Identificador **PK** (Primary Key), URL Imagen, Nombre común, Nombre Científico, Especie.



7.2.3. Fase 3: Implementación de Mapeo Web con Mapbox Android SDK y Marcadores

En esta fase se desarrolló un nuevo proyecto en Android Studio capaz de mostrar la geolocalización del Jardín Botánico tomando en cuenta requisitos para utilizar Mapbox en un proyecto Android Studio que son los siguientes: [11]

- **Una cuenta de Mapbox y un tokens de acceso.**
- **Android Studio.**
- **Mapbox Android SDK.**
- **Un dispositivo Android (físico o virtual).**

Los Marcadores se usan para designar puntos de interés colocándolos en cualquier parte de un mapa, luego le damos a cada uno un título, una descripción o traducción y un pin totalmente único con opciones de tamaño, color y símbolo. Cuando las personas miran el mapa, verán los pines personalizados y podrán pasar el cursor sobre cada uno para ver su título y descripción en el área del Jardín Botánico.



7.2.4. Fase 4: Diseño de Interfaz Gráfica

La interfaz de la aplicación está compuesta por Actividades y mediante el uso de ilustraciones graficas que nos ayudó a la creación del menú de la aplicación.

| **Nota:** Para mayor información **Ver Anexos.**

7.3. Requerimientos de la aplicación JBA - UNAN León

Para que la aplicación se ejecute en el dispositivo es necesario los siguientes requisitos mínimos:

- ✓ Almacenamiento: 30 MB de espacio libre
- ✓ Memoria RAM: 1 GB
- ✓ Resolución de Cámara: 6 MP
- ✓ Sistema Operativo Android: Lollipop (5.1)

7.3.1. Recursos Hardware

Equipo hardware donde se desarrolló el proyecto y dispositivos móviles Android donde se ensayó la aplicación.

Ordenador				
Equipo	CPU	RAM	Almacenamiento	S.O.
DELL Inspiron 5459	Intel Core i5 inside 2.40 GHz	4.00 GB	1 TB	Windows 10 Home Single Language 64 bits



Dispositivos móviles y emulador Android				
Dispositivos	S.O.	RAM	Almacenamiento	Cámara
Samsung Galaxy S5 SM-G900P	6.0.1 (Marshmallow)	2 GB	16 GB	16 MP
LG G3 VS985 4G	6.0.1 (Marshmallow)	2.8 GB	32 GB	13 MP
Genymotion Samsung Galaxy S6 2110	5.1.0 (Lollipop)	3 GB	4.84 GB	-

7.3.2. Recursos Software

Lista de softwares utilizados en el proyecto:

- Android Studio 3.0
- Adobe Illustrator CC
- Adobe Photoshop CC
- Genymotion 2.11.0



8. RESULTADOS

8.1. Mostrar la información asociada a las especies del Jardín Botánico Ambiental UNAN-León a través de la lectura de códigos QR

Mediante la implementación de Mobile Visión API y SQLite en el proyecto obtuvimos un lector capaz de leer el código QR con identificador numérico a las distintas especies de flora que existen en el jardín botánico.



Figura 5: Lector de Códigos QR



Figura 6: Datos representados en la interfaz



8.2. Implementación el mapeo web desde la aplicación para acceder a los recursos asociados a las áreas del Jardín Botánico Ambiental UNAN-León

Con el uso de Mapbox Android SDK y las coordenadas exactas del jardín botánico obtuvimos un mapa completo del área y con el uso de marcadores e ilustraciones proporcionamos visualmente los puntos más relevantes en el Jardín Botánico.



Figura 7: Marcadores en Mapa del Jardín Botánico



8.3. Proporción de datos históricos y turísticos acerca del Jardín Botánico Ambiental UNAN-León

Se creó una actividad capaz de informar a los turistas datos sobre la historia del jardín botánico. También información acerca de las funciones que contiene la aplicación.



Figura 8: Datos Históricos e Información

| **Nota:** Para mayor información Ver **Anexos**.



9. CONCLUSION

Se desarrolló una aplicación móvil Android capaz de proporcionar información específica según el interés de cada usuario y para facilitarle el recorrido por las instalaciones del Jardín Botánico haciendo más entretenida la visita mediante la utilización de dispositivos móviles.

Se implementó tecnología QR, mediante el uso de Mobile Vision API (Google), facilitando la manera a los usuarios de informarse acerca del lugar en el que fue implementado. También se implementó el mapeo web de Mapbox Android SDK, para acceder a los recursos asociados a las áreas.

De este modo la aplicación JBA – UNAN León, proporciona datos turísticos más fiables y concretos facilitando información acerca del Jardín Botánico.



10. RECOMENDACIONES

Concluido este proyecto, se sugieren las siguientes recomendaciones para el seguimiento y uso de la aplicación realizada:

- Se deberá capacitar al personal o administrador del sitio para el correcto uso y manejo de la aplicación móvil, además establecer en el local un espacio que preste las condiciones necesarias para la instalación de códigos QR que serán utilizados para la puesta en marcha del proyecto.
- La aplicación **JBA – UNAN León** puede ampliarse, continuando el enfoque en el campo turístico, para que abarque otras herramientas de la misma.
- La aplicación puede migrar a la plataforma iOS para tener soporte en este tipo de dispositivos móviles. Se deja abierta la posibilidad de utilizar un Framework multiplataforma.
- Para la instalación de la aplicación **JBA – UNAN León** se recomienda disponer de un dispositivo móvil Android que cumpla con los requerimientos hardware y software antes mencionados, con el fin de tener una correcta funcionalidad e interacción con la aplicación.
- Se deberá desarrollar una aplicación web para la administración de datos de la aplicación móvil y pueda actualizar correctamente la nueva información.



11. REFERENCIAS

- [1] Jardín Botánico da Madeira. Obtenido del sitio: https://play.google.com/store/apps/details?id=com.Madeira_BG&hl=en
- [2] Danny Torres & Sara Mendoza. (2015) Sistema para la obtención de información de los espacios y monumentos de la Catedral de León a través de dispositivos móviles. UNAN – León
- [3] Congreso COMPDES. Obtenido del sitio: <https://play.google.com/store/apps/details?id=com.just0rick.planb&hl=en>
- [4] Jardín Botánico Ambiental UNAN-LEÓN. Obtenido del sitio: <http://jba.unanleon.edu.ni/Jardin%20Botanico%20Ambiental.html>
- Aplicaciones Móviles. Miguel Jiménez & Edgar García (2015). APLICACIÓN MÓVIL CELULAR PARA INCENTIVAR EL TURISMO URBANO EN GUAYAQUIL. UNIVERSIDAD POLITÉCNICA SALESIANA SEDE GUAYAQUIL.
- [5] Aplicaciones Móviles Turismo Nicaragüense. Obtenido de los sitios: <https://www.el19digital.com/articulos/ver/titulo:42156-intur-lanza-novedosa-aplicacion-para-promover-el-turismo>
- [6] <http://www.elnuevodiario.com.ni/economia/424451-presentan-aplicacion-movil-promover-turismo-nicara/>
- [7] Códigos QR. Diputación de Cádiz. Servicio de Sociedad de la Información. Manual de Códigos QR.
- [8] Mobile Vision API. Obtenido de los sitios de developer google: <https://developers.google.com/vision/introduction>
- [9] <https://developers.google.com/vision/android/barcodes-overview>
- [10] Mapeo Web y GIS en la nube. Obtenido del sitio: https://en.wikipedia.org/wiki/Web_mapping
- [11] Mapbox Android SDK. Obtenido del sitio: <https://www.mapbox.com/help/first-steps-android-sdk/>



ANEXOS

Anexo 1: Creación de Lector QR usando Mobile Vision API (GOOGLE) y Base de Datos

Esta aplicación implementa un Android View donde se enciende la cámara y se muestra en tiempo real lo que tenemos de frente a ella, listo para ser escaneado, regresándonos una consulta de nuestra base de datos, sea válido o no el resultado.

La API Mobile Vision trae consigo clases que controlan la cámara de manera interna y autofocus para facilitar el uso de la cámara y lectura de los códigos QR.

También puede usar esto para fines de Realidad Aumentada, ya que obtiene coordenadas de puntos de control QR al decodificar.

Permisos de Cámara del Dispositivo

Necesitamos agregar al **Manifest.xml** el código que nos permitirá usar la cámara dándole permisos para encenderla.

```
<uses-permission android:name="android.permission.CAMERA" />
```



Aplicación móvil bajo plataforma Android para la interacción y obtención de información de las áreas y especies del Jardín Botánico Ambiental UNAN-León a través de la utilización de códigos QR y Mapeo Web.

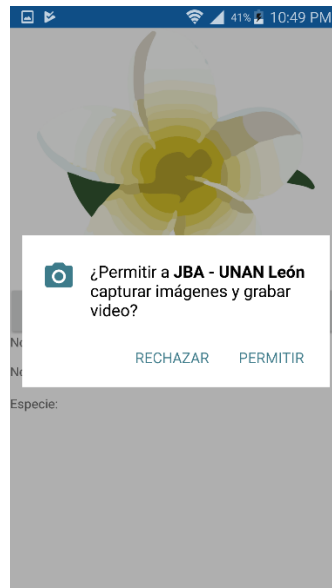


Figura 9: Petición de Permisos de Cámara del Sistema

Implementación de Código XML Lector QR

Creemos una actividad con el nombre de **Activity_scan.xml**, nuestra actividad está compuesta únicamente por:

RelativeLayout

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/activity_scan"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.example.user.mapay2.ScanActivity">
```



SurfaceView

```
<SurfaceView
    android:id="@+id/cameraView"
    android:layout_width="match_parent"
    android:layout_height="match_parent" />
```

ImageView

```
<ImageView
    android:id="@+id/imageView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:layout_centerHorizontal="true"
    android:layout_marginBottom="12dp"
    app:srcCompat="?android:attr/listChoiceIndicatorSingle" />
```

Creamos una Activity llamada ScanActivity.java donde se implementen los Listener, métodos requeridos y objetos necesarios.

Agregar Dependencias al Proyecto

Nuestro proyecto necesita de ciertas dependencias para el correcto funcionamiento.

```
dependencies {
    compile fileTree(dir: 'libs', include: ['*.jar'])
    androidTestCompile('com.android.support.test.espresso:espresso-core:2.2.2', {
        exclude group: 'com.android.support', module: 'support-annotations'
    })
    //Librerias MapBox
    compile('com.mapbox.mapboxsdk:mapbox-android-sdk:5.1.3@aar') {
        transitive = true
    }
    compile 'com.android.support:appcompat-v7:25.0.1'
    compile 'com.android.support.constraint:constraint-layout:1.0.2'
    compile 'com.google.android.gms:play-services:10.0.1'
    testCompile 'junit:junit:4.12'
```



Script SQLiteHelper

Nuestra aplicación necesita de una base de datos, para eso necesitamos crear **SQLiteHelper.java**

Nuestra **SQLiteHelper** contiene tres métodos:

Método SQLiteHelper

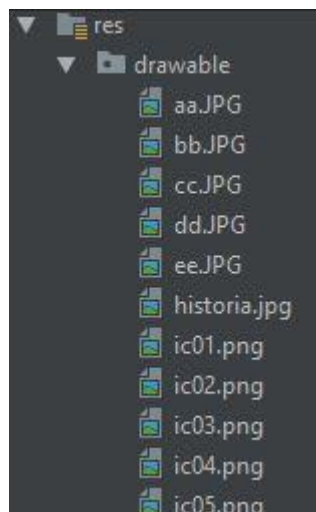
```
public SQLiteHelper(Context context, String nombre, SQLiteDatabase.CursorFactory factory, int version) {  
    super(context, nombre, factory, version);  
}
```

Método onCreate

El método onCreate, tiene la consulta de creación de tabla, campos utilizados en la aplicación, y a su vez la inserción de datos.

Al usar la consulta se crea una tabla con el nombre de **db_jba**, posee un campo Id de tipo **Integer** como llave primaria, **imagen** de tipo texto (Almacena la URL de la Imagen), **nombre** de tipo texto, **científico** tipo texto, **especie** tipo texto.

```
@Override  
public void onCreate(SQLiteDatabase db) {  
  
    db.execSQL("Create table db_jba (Id integer primary key,imagen text,nombre text,cientifico text,especie text)");  
    db.execSQL("insert into db_jba values(1,'drawable/aa','Barleria','Cristata','ACANTHACEAE')");  
}
```





Aplicación móvil bajo plataforma Android para la interacción y obtención de información de las áreas y especies del Jardín Botánico Ambiental UNAN-León a través de la utilización de códigos QR y Mapeo Web.

Método OnUpgrade

```
@Override  
public void onUpgrade(SQLiteDatabase db, int version1, int version2) {  
    db.execSQL("drop table if exists db_jba");  
}
```



Aplicación móvil bajo plataforma Android para la interacción y obtención de información de las áreas y especies del Jardín Botánico Ambiental UNAN-León a través de la utilización de códigos QR y Mapeo Web.

Anexo 2: Implementación de Mapeo Web con Mapbox y Marcadores

Creación de Cuenta Mapbox y Llave

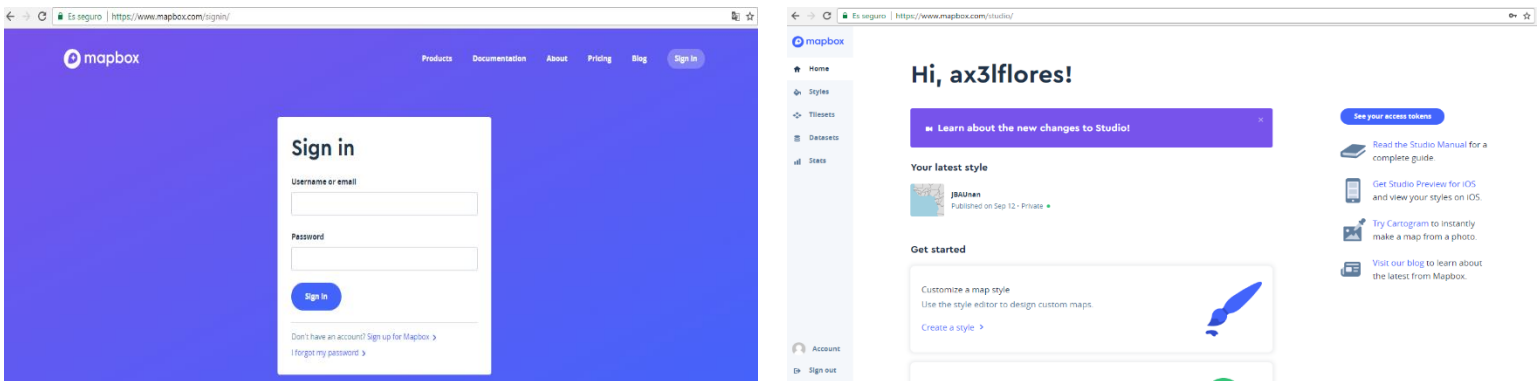


Figura 10: Creación de cuenta e inicio de sección en Mapbox

Para utilizar la API de Mapeo Web de Mapbox es necesario generar una llave, un **Access_token** desde el sitio web una vez dado de alta en el sitio. Las llaves pueden ser renovadas una vez expirada el periodo de validez para seguir usando la API.

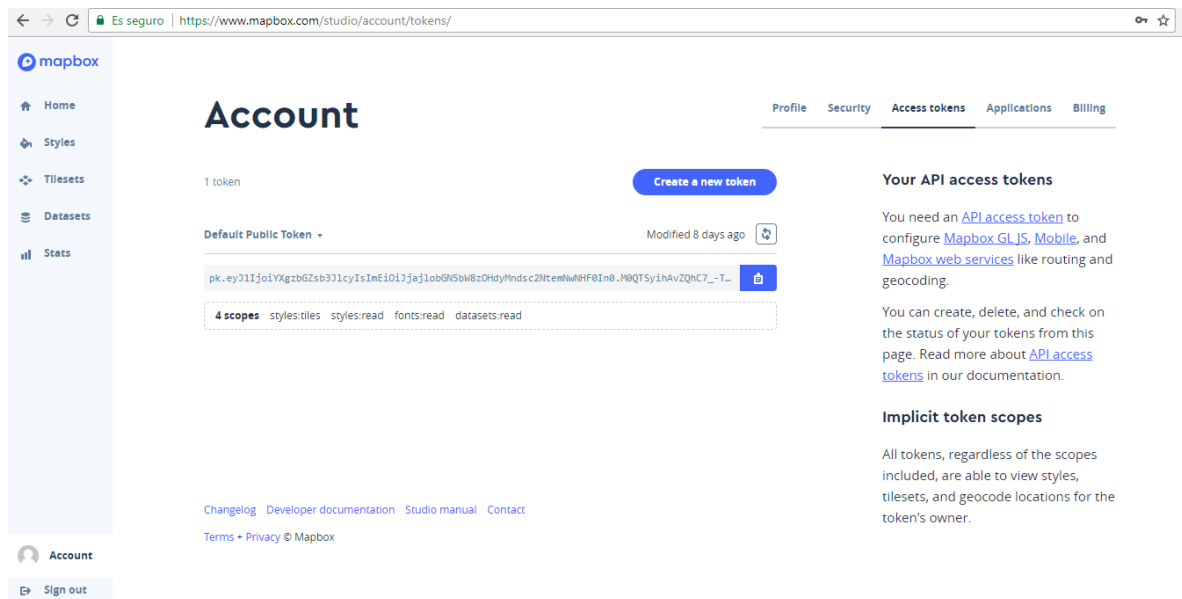


Figura 11: Generación de Llave en Mapbox



```
<string name="access_token">pk.eyJ1IjoiYXgzbGZsb3JlcyIsImEiOiJjajlobGN5bW8zOHdyMndsc2NtemNwNHFOIn0.M0QTSyihAvZQhC7_-TsrAQ</string>
```

Creamos un Empty Activity, el nombre del layout será **activity_jba_map_box.xml**, el nombre de la **Activity** es **Jba_MapBox**.

Antes de empezar con la utilización de la API Mapbox Android SDK, se necesitan hacer modificaciones a los diferentes archivos en el proyecto que se utilizara.

Build.gradle

```
compile('com.mapbox.mapboxsdk:mapbox-android-sdk:5.1.3@aar') {  
    transitive = true  
}
```

AndroidManifest.xml

```
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />  
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
```

Activity_jba_map_box.xml

El layout del mapa contiene un RelativeLayout

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
    xmlns:mapbox="http://schemas.android.com/apk/res-auto"  
    xmlns:tools="http://schemas.android.com/tools"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    tools:context=".Jba_MapBox">
```



Incorporando una MapView dentro del RelativeLayout, el MapView posee propiedades en las cuales indicaremos: **Longitud, Latitud, Zoom del Mapa, URL del estilo del mapa.**

```
<com.mapbox.mapboxsdk.maps.MapView
    android:id="@+id/mapview"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    mapbox:mapbox_cameraTargetLat="12.434100"
    mapbox:mapbox_cameraTargetLng="-86.9141999"
    mapbox:mapbox_cameraTilt="20"
    mapbox:mapbox_cameraZoom="16"
    mapbox:mapbox_styleUrl="mapbox://styles/mapbox/streets-v10">
</com.mapbox.mapboxsdk.maps.MapView>
```



Figura 12: Representación del Jardín Botánico en Mapbox



Jba_map_box.java

Clases utilizadas

```
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;

import com.mapbox.mapboxsdk.Mapbox;
import com.mapbox.mapboxsdk.maps.MapView;
import com.mapbox.mapboxsdk.maps.MapboxMap;
import com.mapbox.mapboxsdk.maps.OnMapReadyCallback;
```

Iconos

Se necesita establecer el recurso que se utilizaran los marcadores como icono.

```
Icon i24 = iconFactory.fromResource(R.drawable.ic24);
```

Marcadores

En las opciones de marcadores se agrega la posición: Longitud, Latitud. Los iconos previamente declarados, título del Marcador y el Snippet.

```
mapboxMap.addMarker(new MarkerViewOptions()
    .position(new LatLng( latitude: 12.43480, longitude: -86.91245))
    .icon(i01)
    .title("Parqueo los Agüisotes")
    .snippet("Los Agüisotes parking lot"));
```



Anexo 3: Diseño de Interfaz Gráfica

Logotipo

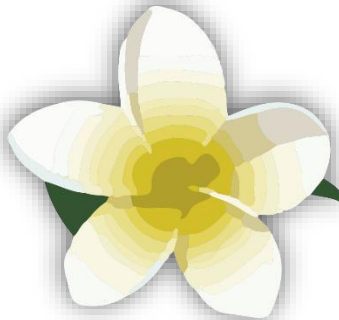


Figura 13: Icono de la aplicación móvil

La interfaz de la aplicación está compuesta por Activity, cada una con su respectiva clase.

- Interfaz Splash: Únicamente está diseñada por una drawable.
- Interfaz Principal: La interfaz principal contiene cinco ImageButton dentro de una relative layout con 5dp de margen en los cuatros puntos right, left, top, bottom, cada ImageButton tiene su TextView que nos indica a que Activity nos mandara al momento de llamada del evento OnClic (Historia, Lector QR, Mapa, Información).
- Interfaz Previa Lector QR: La interfaz previa al lector está diseñada con un RelativeLayout, lo primero que nos muestra esta Activity es un ImageView en este se muestra por defecto una imagen antes de escanear cualquier QR, cuando se escanea un QR mostrará como resultado la Planta relacionada con el mismo, Botón para lanzar la Activity del Lector QR, en total se



insertaron seis TextView, Tres de ellos son estáticos que nos indican la información que se nos presentara a la hora de escanear el QR, los otros tres mostraran la consulta que regrese la Activity del Lector QR.

- Interfaz Lector QR: La actividad Lector QR es lanzada desde la Interfaz Previa al Lector, únicamente posee una SurfaceView que nos mostrará lo que estemos viendo a través de la cámara en este caso para poder escanear el Código QR, un ImageView que contendrá el logo de la Universidad.

- Interfaz Historia: La Interfaz Historia está diseñada con una UI (User Interface o Interfaz de Usuario) moderna con un Collapsing Toolbar (Barra de herramientas con efecto de colapso), el Collapsing Toolbar contiene un ImageView, incorporando un NestedScrollView que posee un RelativeLayout y un ScrollView que nos mostrará la información necesaria a mostrar dentro de la aplicación.

- Interfaz de Información: La interfaz de información de igual manera que Interfaz Historia está diseñada con una UI Moderna, incorporando todos los ítems similares:
 - NestedScrollView
 - RelativeLayout
 - ScrollView

- Interfaz del Mapa: La interfaz del Mapa únicamente contiene un MapView que dentro de sus propiedades están definidas las coordenadas del mapa a Mostrar, el tamaño del Zoom definido.



Splash.xml

Esta actividad es lanzada al abrir la aplicación, como una pantalla de inicio esperando que se cargue correctamente la base de datos.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/splash"
    android:orientation="vertical">
</LinearLayout>
```

Splash.xml únicamente está compuesto por un linear layout con un background llamado Splash que es la imagen que se presenta al Inicio.



Figura 14: Splash

El tiempo de duración del Splash está indicado dentro de **SplashScreen.java**, el tiempo destinado a mostrar el Splash es de 3000 milisegundos. Cuando se cumplen la cantidad de milisegundos se invocará la actividad principal **MainActivityMenu.java**.



```
Thread timerThread = run() - {  
    try{  
        sleep( millis: 3000);  
    }catch(InterruptedExcepcion e){  
        e.printStackTrace();  
    }finally{  
        Intent intent = new Intent( packageContext: SplashScreen.this,MainActivityMenu.class);  
        startActivity(intent);  
    }  
}
```

ActivityMain.xml

En esta actividad estará representado todas las opciones que podremos elegir dentro de la aplicación.

```
<RelativeLayout  
    android:layout_width="324dp"  
    android:layout_height="468dp"  
    tools:layout_editor_absoluteX="16dp"  
    tools:layout_editor_absoluteY="61dp">  
    <ImageButton...>  
    <ImageButton...>  
    <ImageButton...>  
    <ImageButton...>  
    <TextView...>  
    <TextView...>  
    <TextView...>  
    <TextView...>  
</RelativeLayout>
```



Aplicación móvil bajo plataforma Android para la interacción y obtención de información de las áreas y especies del Jardín Botánico Ambiental UNAN-León a través de la utilización de códigos QR y Mapeo Web.



Figura 15: Menú de la aplicación

Historia.xml – Info_app.xml

Las dos actividades: Historia e Info_app tienen programado un AppLayout que se colapsa de la siguiente manera, según el desplazamiento dentro de la actividad.



Activity_main_2.xml

Esta actividad es la que nos permitirá acceder a la información necesaria una vez recibida cuando se escanee el código QR válido.

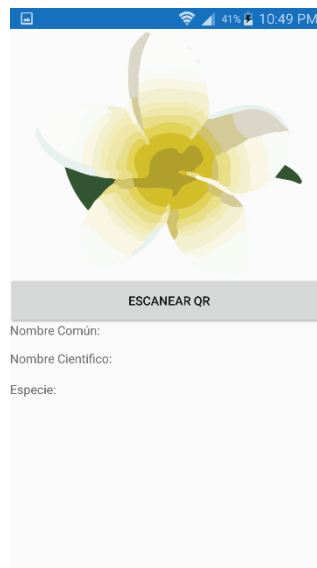


Figura 17: Interfaz de información de especies (MainActivity.java)

MainActivity.java

Un Toast proporciona comentarios simples sobre una operación en una pequeña ventana emergente. Solo llena la cantidad de espacio requerido para el mensaje y la actividad actual permanece visible e interactiva. Toast desaparecen automáticamente después de un tiempo de espera.



```
} else{
    txt_1.setText("");
    txt_2.setText("");
    txt_3.setText("");
    imagen.setImageResource(R.drawable.preset);

    Toast.makeText(getApplicationContext(), text: "La Planta no se encuentra registrada.", Toast.LENGTH_LONG).show();
    db.close();
}
}
catch(Exception ex){
    ex.printStackTrace();
    txt_1.setText("");
    txt_2.setText("");
    txt_3.setText("");
    imagen.setImageResource(R.drawable.preset);
    Toast.makeText(getApplicationContext(), text: " El Código QR no es valido", Toast.LENGTH_LONG).show();
}
}
```

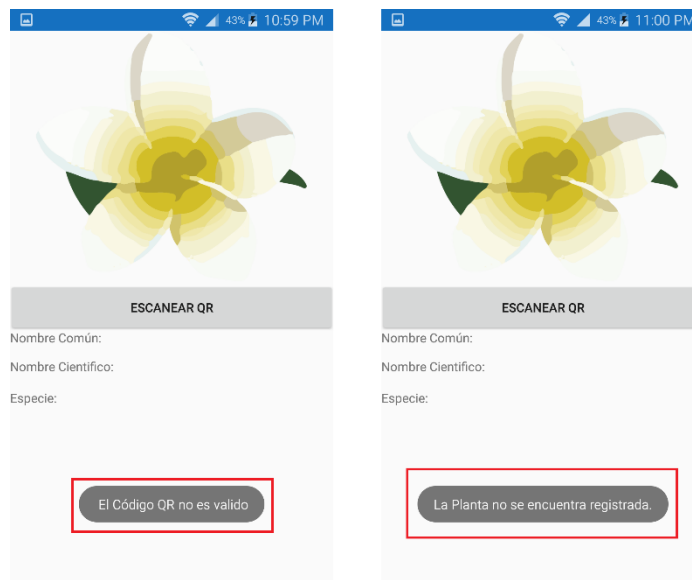


Figura 18: Ventanas emergentes caso de carácter y entero no en base de datos respectivamente