



**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA**

**CARRERA DE INGENIERÍA ELECTRÓNICA, REDES Y  
COMUNICACIÓN DE DATOS**

**TESIS PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO  
EN ELECTRÓNICA, REDES Y COMUNICACIÓN DE DATOS.**

**AUTORA: BETSABÉ INÉS SIMBAÑA JAYA**

**TEMA: DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN  
BASADA EN REALIDAD AUMENTADA PARA LOS EDIFICIOS  
PATRIMONIALES DEL CENTRO HISTÓRICO DE QUITO**

**DIRECTOR: MSC. ING. ABG DARWIN ALULEMA**

**CO DIRECTOR: ING. CHRISTIAN VEGA**

**SANGOLQUÍ, ENERO DE 2015**

**UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE**  
**INGENIERÍA ELECTRÓNICA EN REDES Y COMUNICACIÓN**  
**DE DATOS**

**DECLARACIÓN DE RESPONSABILIDAD**  
**BETSABÉ INÉS SIMBAÑA JAYA**

**DECLARO QUE:**

El proyecto de grado denominado “DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN BASADA EN REALIDAD AUMENTADA PARA LOS EDIFICIOS PATRIMONIALES DEL CENTRO HISTÓRICO DE QUITO”, ha sido desarrollado con base a una investigación exhaustiva, respetando derechos intelectuales de terceros, conforme las citas que constan el pie de las páginas correspondiente, cuyas fuentes se incorporan en la bibliografía.

Consecuentemente este trabajo es mi autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance científico del proyecto de grado en mención.

Sangolquí, Enero de 2015

---

Betsabé Inés Simbaña Jaya

**UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE**

**INGENIERÍA ELECTRÓNICA EN REDES Y COMUNICACIÓN  
DE DATOS**

**CERTIFICADO**

Msc. Ing. Abg Darwin Alulema

Ing. Christian Vega

**CERTIFICAN**

Que el trabajo titulado “DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN BASADA EN REALIDAD AUMENTADA PARA EDIFICIOS PATRIMONIALES DEL CENTRO HISTÓRICO DE QUITO” realizado por Betsabé Inés Simbaña Jaya, ha sido guiado y revisado periódicamente y cumple normas estatutarias establecidas por la ESPE, en el Reglamento de Estudiantes de la Universidad de las Fuerzas Armadas - ESPE.

Debido a que se trata de un trabajo de investigación recomiendan su publicación. El mencionado trabajo consta de un documento empastado y un disco compacto el cual contiene los archivos en formato portátil de Acrobat (PDF). Autorizan a Betsabé Simbaña Jaya que lo entregue al Dr. Nikolai Espinosa, en su calidad de Director de la Carrera.

Sangolquí, Enero de 2015

---

Msc. Ing. Abg Darwin Alulema  
DIRECTOR

---

Ing. Christian Vega  
CODIRECTOR

**UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE****INGENIERÍA ELECTRÓNICA EN REDES Y COMUNICACIÓN  
DE DATOS****AUTORIZACIÓN**

Yo, Betsabé Inés Simbaña Jaya

Autorizo a la Universidad de las Fuerzas Armadas - ESPE la publicación, en la biblioteca virtual de la Institución del trabajo “DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN BASADA EN REALIDAD AUMENTADA PARA LOS EDIFICIOS PATRIMONIALES DEL CENTRO HISTÓRICO DE QUITO”, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y autoría.

Sangolquí, Enero de 2015

---

Betsabé Inés Simbaña Jaya

## **DEDICATORIA**

Dedico esta tesis en su totalidad a Dios Todopoderoso que ha sido mi ayuda, guía y quien ha puesto a las personas y situaciones correctas en el su tiempo perfecto.

A mis padres que siempre tienen palabras de aliento y apoyo, que nunca dejaron que me dé por vencida, que cuando me equivoque me corrigieron y que siempre estarán de mi lado.

## **AGRADECIMIENTO**

El desarrollo de esta tesis ha sido un gran reto para mí, no lo podría haber realizado yo sola por eso agradezco de gran manera a Dios que con su ayuda logré terminar y cumplir esta meta.

Agradezco a mis padres que siempre estuvieron pendientes de mi avance, dándome ánimos y orando por mí.

Agradezco a mi director de tesis, Ing. Darwin Alulema, que me ayudo, guío y corrigió en este proceso.

Agradezco a mis amigos y amigas, que me quieren por como soy y siempre me dieron palabras de aliento.

A todos ustedes les agradezco y le pido a Dios que les dé el triple de bendiciones de lo bueno que ustedes me desean. Los quiero mucho a todos.

## TABLA DE CONTENIDO

<b>DECLARACIÓN DE RESPONSABILIDAD .....</b>	<b>I</b>
<b>CERTIFICADO .....</b>	<b>II</b>
<b>AUTORIZACIÓN .....</b>	<b>III</b>
<b>DEDICATORIA.....</b>	<b>IV</b>
<b>AGRADECIMIENTO.....</b>	<b>V</b>
<b>TABLA DE CONTENIDO .....</b>	<b>VI</b>
<b>LISTADO DE CUADROS .....</b>	<b>IX</b>
<b>LISTADO DE FIGURAS .....</b>	<b>X</b>
<b>RESUMEN EJECUTIVO.....</b>	<b>XII</b>
<b>EXECUTIVE ABSTRACT.....</b>	<b>XIII</b>
<b>CAPÍTULO 1 .....</b>	<b>1</b>
<b>DESCRIPCIÓN GENERAL DEL PROYECTO.....</b>	<b>1</b>
<b>1.1 PRESENTACIÓN.....</b>	<b>1</b>
<b>1.2 ANTECEDENTES.....</b>	<b>2</b>
<b>1.3 OBJETIVOS .....</b>	<b>4</b>
<b>1.3.1 OBJETIVO GENERAL .....</b>	<b>4</b>
<b>1.3.2 OBJETIVOS ESPECÍFICOS .....</b>	<b>4</b>
<b>CAPÍTULO 2 .....</b>	<b>5</b>
<b>ESTADO DEL ARTE.....</b>	<b>5</b>
<b>2.1 REALIDAD AUMENTADA.....</b>	<b>5</b>
<b>2.1.1 DEFINICIÓN.....</b>	<b>5</b>
<b>2.1.2 CARACTERÍSTICAS.....</b>	<b>6</b>
<b>2.1.3 SUBSISTEMAS DE FUNCIONAMIENTO .....</b>	<b>6</b>
<b>2.1.3.1 VISUALIZACIÓN.....</b>	<b>7</b>
<b>2.1.3.2 UBICACIÓN DE OBJETOS VIRTUALES .....</b>	<b>8</b>
<b>2.1.3.3 MÉTODO DE INTERACCIÓN.....</b>	<b>8</b>
<b>2.1.4 DIFERENCIAS CON LA REALIDAD VIRTUAL.....</b>	<b>9</b>
<b>2.1.5 TIPOS DE REALIDAD AUMENTADA.....</b>	<b>9</b>
<b>2.1.5.1 REALIDAD AUMENTADA BASADA EN MARCADORES .....</b>	<b>9</b>
<b>2.1.5.2 REALIDAD AUMENTADA BASADA EN POSICIÓN .....</b>	<b>10</b>
<b>2.1.6 ACCESO A LA REALIDAD AUMENTADA.....</b>	<b>11</b>

2.1.7	APLICACIONES DE LA REALIDAD AUMENTADA .....	12
2.1.8	AVANCES E INNOVACIÓN.....	13
2.2	ANDROID .....	17
2.2.1	DEFINICIÓN.....	17
2.2.2	CARACTERÍSTICAS.....	17
2.2.3	ARQUITECTURA Y HARDWARE .....	18
2.2.4	VERSIONES .....	22
2.2.5	DISEÑO Y DESARROLLO .....	25
2.2.5.1	ECLIPSE.....	25
2.2.5.2	ANDROID SDK .....	25
2.2.5.3	ANDROID VIRTUAL DEVICE .....	27
2.2.5.4	ANDROID EMULATOR.....	27
2.2.6	SENSORES .....	28
2.2.6.1	SENSORES DE MOVIMIENTO .....	28
2.2.6.2	SENSORES DE POSICIÓN.....	29
2.2.6.3	SENSORES DE CONDICIONES AMBIENTALES.....	29
2.2.7	FUNDAMENTOS DE LAS APLICACIONES .....	30
2.2.7.1	COMPONENTES DE UNA APLICACIÓN .....	30
2.2.7.2	TIPOS DE APLICACIONES .....	31
2.3	SDK PARA REALIDAD AUMENTADA .....	32
2.3.1.	VUFORIA .....	32
2.3.1.1	CARACTERÍSTICAS .....	33
2.3.1.2	ARQUITECTURA.....	33
2.3.2.	OPEN CV .....	33
2.3.2.1	CARACTERÍSTICAS .....	34
2.3.2.2	ARQUITECTURA.....	34
2.3.3.	WIKITUDE.....	35
2.3.3.1	CARACTERÍSTICAS .....	36
2.3.3.2	ARQUITECTURA.....	36
	<b>CAPÍTULO 3 .....</b>	<b>38</b>
	<b>DISEÑO E IMPLEMENTACIÓN DE LA APLICACIÓN.....</b>	<b>38</b>
3.1	ESPECIFICACIÓN DE REQUERIMIENTOS.....	38
3.2	ELECCIÓN DE SDK DE REALIDAD AUMENTADA.....	39
3.3	DESCRIPCIÓN DE OPCIONES DE USO .....	41

3.4	DISEÑO DE LA APLICACIÓN.....	47
3.4.1	ARQUITECTURA .....	47
3.4.2	ELEMENTOS .....	48
3.4.3	DIAGRAMA DE CLASES .....	49
3.4.4	DIAGRAMA DE SECUENCIA.....	51
3.5	IMPLEMENTACIÓN.....	54
3.5.1	NOMBRE DE LA APLICACIÓN .....	54
3.5.2	DISEÑO DE INTERFACES .....	55
3.5.3	MENÚ INICIO .....	58
3.5.4	VISTA EN REALIDAD AUMENTADA .....	59
3.5.5	MENÚ AYUDA.....	63
3.5.6	SERVICIO WEB .....	68
3.5.7	BASE DE DATOS.....	69
3.6	DURACIÓN DEL PROCESO DE DISEÑO E IMPLEMENTACIÓN.....	70
	<b>CAPÍTULO 4 .....</b>	<b>72</b>
	<b>FUNCIONAMIENTO Y RESULTADOS DE LA APLICACIÓN.....</b>	<b>72</b>
4.1	FUNCIONAMIENTO DE LA APLICACIÓN.....	72
4.2	DISPOSITIVOS Y VERSIONES.....	79
4.3	USABILIDAD Y PRECISIÓN.....	85
4.4	MEMORIA.....	87
4.5	CONSUMO DE DATOS MÓVILES Y WI-FI.....	89
	<b>CAPÍTULO 5 .....</b>	<b>92</b>
	<b>CONCLUSIONES Y RECOMENDACIONES.....</b>	<b>92</b>
5.1	CONCLUSIONES .....	92
5.2	RECOMENDACIONES .....	94
	<b>BIBLIOGRAFÍA.....</b>	<b>96</b>
	<b>ANEXO A .....</b>	<b>100</b>
	<b>ANEXO B .....</b>	<b>110</b>
	<b>ANEXO C .....</b>	<b>113</b>
	<b>ANEXO D .....</b>	<b>121</b>
	<b>ANEXO E .....</b>	<b>125</b>
	<b>ANEXO F.....</b>	<b>130</b>
	<b>ANEXO G.....</b>	<b>131</b>

## LISTADO DE CUADROS

Cuadro 1 Diferencias entre Realidad Aumentada y Realidad Virtual.....	9
Cuadro 2 Características de Android .....	18
Cuadro 3 Versión Android vs Versión Kernel Linux .....	19
Cuadro 4 Paquetes de Android SDK .....	27
Cuadro 5 Comparación entre SDKs.....	40
Cuadro 6 Descripción de Función: Mostrar Menú Inicio .....	42
Cuadro 7 Descripción de Función: Generar Realidad Aumentada .....	42
Cuadro 8 Descripción de Función: Mostrar Información Virtual.....	43
Cuadro 9 Descripción Función: Captura de Pantalla .....	43
Cuadro 10 Descripción de Función: Rango .....	44
Cuadro 11 Descripción de Función: Ayuda.....	44
Cuadro 12 Lugares Disponibles para el Usuario.....	65
Cuadro 13 Duración del Proceso de Diseño e Implementación.....	70
Cuadro 14 Prueba Caso de Uso: Menú Inicio.....	73
Cuadro 15 Prueba Caso de Uso: Vista en Realidad Aumentada.....	74
Cuadro 16 Prueba Caso de Uso: Detalles de POIs .....	75
Cuadro 17 Prueba Caso de Uso: Captura de Pantalla. ....	76
Cuadro 18 Prueba Caso de Uso: Rango .....	77
Cuadro 19 Prueba Caso de Uso: Ayuda .....	78
Cuadro 20 Prueba a Servicio Web.....	79
Cuadro 21 Características de Dispositivos de Prueba.....	79
Cuadro 22 Resultados de las funciones en los Dispositivos.....	80
Cuadro 23 Precisión y Usabilidad.....	85

## LISTADO DE FIGURAS

Figura 1 Realidad Aumentada .....	5
Figura 2 Ejemplificación Video See-Trough (VST).....	6
Figura 3 Método de Visión Directo.....	7
Figura 4 Método de Visión Indirecto .....	7
Figura 5 Método de Visión con Monitores Externos.....	8
Figura 6 Tipos de Ubicación .....	8
Figura 7 Ejemplo de Marcador.....	10
Figura 8 Ejemplo de Realidad Aumentada basada en Posición .....	11
Figura 9 Usos de la Realidad Aumentada .....	13
Figura 10 Aplicación LEGO Fusion.....	14
Figura 11 SmartGlasses Epson Moverio BT-200.....	15
Figura 12 Google Glass .....	15
Figura 13 Aplicación de Realidad Aumentada de McDonald's .....	16
Figura 14 Stringer Video juego de Realidad Aumentada .....	16
Figura 15 Arquitectura de Android .....	21
Figura 16 Porcentaje de las Versiones de Plataformas Junio 2014.....	25
Figura 17 Android SDK Manager .....	26
Figura 18 Android Emulator .....	28
Figura 19 Logo de Vuforia .....	32
Figura 20 Logo de OpenCV .....	34
Figura 21 Logo de Wikitude .....	35
Figura 22 Plan de Desarrollo .....	38
Figura 23 Diagrama de Casos de Uso.....	46
Figura 24 Proceso de Consulta de Información .....	47
Figura 25 Arquitectura Física .....	49
Figura 26 Diagrama de Clases de Quito Visión+ .....	50
Figura 27 Diagrama de Secuencia - Vista en Realidad Aumentada .....	51
Figura 28 Diagrama de Secuencia - Mostrar Detalles de POI .....	52
Figura 29 Diagrama de Secuencia - Rango.....	52
Figura 30 Diagrama de Secuencia - Captura de Pantalla.....	53
Figura 31 Diagrama de Secuencia: Ayuda .....	53
Figura 32 Logotipo Quito Visión.....	54
Figura 33 Pantalla de Inicio Quito Visión+ .....	55
Figura 34 Interfaz: Menú Inicio.....	56
Figura 35 Interfaz: Ayuda.....	56
Figura 36 Interfaz: Lugares Disponibles .....	57

Figura 37 Interfaz: Sobre Quito Visión .....	57
Figura 38 Interfaz: Vista en Realidad Aumentada.....	58
Figura 39 Pantalla de Menú Inicio.....	59
Figura 40 Imagen Launcher de Wikitude .....	60
Figura 41 Interfaz Realidad Aumentada .....	60
Figura 42 Panel Detalles.....	61
Figura 43 Panel Rango .....	62
Figura 44 Compartir Captura .....	62
Figura 45 Barra de Status.....	63
Figura 46 Menú Ayuda.....	64
Figura 47 Lugares Disponibles .....	67
Figura 48 Pantalla de Ajustes GPS.....	67
Figura 49 Mensaje de la Desarrolladora .....	68
Figura 50 Respuesta del Servicio Web.....	69
Figura 51 Información de Base de Datos en PHPMyAdmin .....	70
Figura 52 Mensaje de Error en Xperia Play .....	81
Figura 53 Menú Inicio en Samsung S3 Mini .....	82
Figura 54 Vista en Realidad Aumentada en Samsung Galaxy S5.....	82
Figura 55 Detalles de POI en Samsung S3 Mini.....	83
Figura 56 Detalles de POI Samsung Galaxy S5 .....	83
Figura 57 Captura de Pantalla en Samsung S3 Mini .....	84
Figura 58 Rango en Samsung S3 Mini .....	84
Figura 59 Menú Ayuda en Samsung Galaxy S5 .....	85
Figura 60 Resultado de 3CToolbox en Samsung S5 .....	87
Figura 61 Resultado de 3CToolbox en Samsung S3 Mini .....	88
Figura 62 Resultado de 3CToolbox en Xperia Play .....	88
Figura 63 Uso de la Memoria RAM.....	88
Figura 64 Uso de Datos obtenidos de Samsung S3 Mini.....	89
Figura 65 Uso de Datos Obtenidos de Samsung S5 .....	90
Figura 66 Uso de Datos Móviles .....	90
Figura 67 Uso de Wi-Fi obtenidos de Samsung S3 Mini.....	91
Figura 68 Uso de Wi-Fi obtenidos de Samsung S5 .....	91

## RESUMEN EJECUTIVO

El presente proyecto desarrolló una aplicación de realidad aumentada para dispositivos móviles con sistema operativo Android. “Quito Visión+”, el nombre de la aplicación, busca mostrar al usuario la información de mayor relevancia de los edificios patrimoniales del centro histórico de Quito y de esa manera ayudar a promover el turismo en la ciudad capital. El usuario podrá observar los puntos de interés presentados, en realidad aumentada mediante el enfoque de la cámara, en la pantalla de su teléfono, así también cada POI cuenta con un panel de detalles informativos. El documento consta con cuatro capítulos, en donde se realiza la investigación de temas como: realidad aumentada, Android y servicios web, también consta el análisis para la elección de un SDK, que en este caso fue Wikitude, se incluye el diseño e implementación de la aplicación en sí, para ello se utilizaron los lenguajes de programación Java, JavaScript y HTML. Finalmente, se muestra las pruebas de funcionamiento realizadas en tres dispositivos físicos de diferentes características y versiones de sistema operativo. El último capítulo, es en donde se puede encontrar las conclusiones y recomendaciones para el desarrollo de este tipo de aplicaciones de geolocalización. Además se implementó un servicio web, que se encuentra publicado en un web hosting, que sirve de puente entre la base de datos que contienen los POIs y la aplicación. Se puede encontrar también en los Anexos, el Manual de Usuario y varios tutoriales para quienes quieran mejorar a “Quito Visión+”.

Palabras Claves:

- Realidad Aumentada
- Servicio Web
- Java
- MySQL
- Android

## EXECUTIVE ABSTRACT

This project developed an augmented reality application for mobile devices with Android operating system. “Quito Visión+”, the name of the application, aims to show the user the most important information of the heritage buildings in the historic center of Quito and thus help to promote tourism in the capital city. The user will be able to watch points of interest presented in augmented reality through the camera focus, in the screen of their device, also each POI has a panel of informative details. This document include four chapters where the research topics are: augmented reality, Android and web services, also it shows the analysis for choosing an SDK, which in this case was Wikitude, it includes the application’s design and implementation, for it languages like Java, JavaScript and HTML were used. Finally, is shown the performing tests made in three physical devices with different features and operating system versions. The final chapter is where the conclusions and recommendations can be found, for the development of application based on geo-localization. In addition, a web service was implemented, which is published on a web hosting, it serves as a bridge between the database containing de POIs information and the application itself. In the Annexes, the User Manual and some tutorial can be found for whoever wants to improve “Quito Visión+”.

### Keywords

- Augmented Reality.
- Web Service
- Java
- MySQL
- Android

## **CAPÍTULO 1**

### **DESCRIPCIÓN GENERAL DEL PROYECTO**

#### **1.1 PRESENTACIÓN**

Quito cuenta con varios atractivos históricos y culturales que fomentan el turismo y también son motivo de estudio, lastimosamente los quiteños no se involucran con el conocimiento sobre las principales edificaciones del centro histórico de la ciudad.

La visión del actual gobierno pretende impulsar el turismo en el país de diferentes formas, una de ellas con la campaña All You Need Is Ecuador. Con la tecnología actual se puede realizar el impulso del turismo, no solo en las redes sociales sino también en el desarrollo de aplicaciones de información e inclusión que permitan al turista informarse de manera dinámica sobre el lugar en donde se encuentra.

Siendo Quito la capital del Ecuador donde lo clásico y tradicional se mezcla con la modernidad e innovación, el objetivo es que Quito se convierta en una Smart City cada vez más enfocada al futuro. A partir del año 2013, el concepto hacer de Quito una Smart City ha venido tomando tanta fuerza que ha impulsado la realización de conferencias y foros sobre el tema, por ejemplo: la Smart City Expo World Congress que se llevó a cabo en el mes de Mayo del año ya mencionado. En donde se habló netamente de la utilización aplicaciones que ayuden a las personas en tiempo real. Así también de aplicaciones que con la geolocalización permitan llegar a un destino con mayor rapidez y menos tráfico. Con las Smart Cities, Quito y sus habitantes tendrían la solución a problemas del diario vivir al alcance de una computadora, teléfono inteligente o Tablet.

La realidad aumentada es una opción para que tanto los turistas nacionales como internacionales se interesen en el valor histórico del Ecuador en una manera innovadora y didáctica.

La aplicación impulsa el turismo en la ciudad de Quito, mediante la presentación en realidad aumentada de la información relevante de las edificaciones del centro histórico de Quito.

Debido a que Quito cuenta con varias edificaciones culturales e históricas, la aplicación presenta información de iglesias, edificios administrativos o museos.

El usuario debe enfocar con su teléfono móvil hacia la edificación en la que esté interesado. Y la aplicación muestra información como: nombre, historia, contactos, horarios de visita, etc. La aplicación es exclusiva para dispositivos con sistema operativo Android y el tipo de reconocimiento de realidad aumentada que utiliza es por geolocalización.

## **1.2 ANTECEDENTES**

La realidad aumentada es la manera de agregar a una realidad física, una realidad virtual en un mismo espacio y a tiempo real. Creando un ambiente mixto, en donde los sentidos del ser humano perciben un entorno combinado sin sustituir la realidad física.

“La principal diferencia con la realidad virtual, en un concepto mucho más extendido, es que no sustituye la realidad física, sino que superpone la información digital en tiempo real al mundo real. Así pues la realidad aumentada extrae información del entorno convirtiendo lo digital en real.” (Minetur, 2013)

Para acceder a la realidad aumentada se necesita de un dispositivo tecnológico que disponga de una cámara que permita la interacción con el usuario, como por ejemplo: smartphones, tabletas, computadoras, etc.

La realidad aumentada está dirigida a múltiples actividades de la vida cotidiana de los seres humanos, tales como: educación, marketing, entretenimiento, ocio, salud, etc. En el mercado de aplicaciones para dispositivos Android se puede encontrar programas enfocados a la educación para niños y adultos en diferentes temas. El ocio y entretenimiento de igual manera son los más populares al momento de hablar de realidad aumentada debido a los videos juegos.

Los tipos de Realidad Aumentada son específicamente por reconocimiento y por geolocalización. Para ello hay varios programas que permiten la creación de estas aplicaciones.

En Ecuador existen empresas que están usando la realidad aumentada en la publicidad y en el sector inmobiliario, que se encargan de realizar animaciones digitales y modelado en 3D. Camaleón Diseño Visual es una empresa quiteña que realiza aplicaciones no solo para teléfonos móviles sino también para computadoras; sus aplicaciones están desarrolladas para clientes en varios ámbitos como son el arte, educación, medicina y publicidad. En Guayaquil, la empresa PlanBellow, trabaja en el marketing interactivo con el servicio de realidad aumentada, entre sus clientes se encuentran: Nestle, Sony, Diario El Universo y más.

Empresas líderes en el Ecuador utilizan la realidad aumentada para ofrecer sus servicios como lo es el diario HOY. “Diario HOY ofrece desde esta edición esta posibilidad en sus contenidos editoriales: esta es la herramienta más rápida para pasar del papel al video.” (Diario HOY, 2013)

La realidad aumentada en el Ecuador se encuentra enfocada principalmente al marketing dejando de lado al turismo y un tanto la educación. Sectores en los cuales las aplicaciones para teléfonos móviles tendrán una gran acogida.

### **1.3 OBJETIVOS**

#### **1.3.1 OBJETIVO GENERAL**

Desarrollar una aplicación basada en realidad aumentada para los edificios patrimoniales del centro histórico de Quito.

#### **1.3.2 OBJETIVOS ESPECÍFICOS**

- Investigar sobre realidad aumentada y lenguajes de desarrollo de aplicaciones para Android.
- Obtener los principales datos e información de los edificios patrimoniales del centro de Quito.
- Diseñar la aplicación basada en realidad aumentada para teléfonos móviles con tecnología Android.
- Realizar pruebas de funcionamiento y corrección de errores.

## CAPÍTULO 2

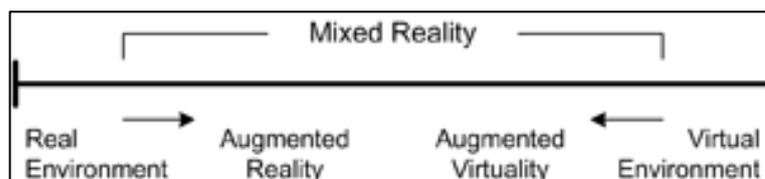
### ESTADO DEL ARTE

#### 2.1 REALIDAD AUMENTADA

##### 2.1.1 DEFINICIÓN

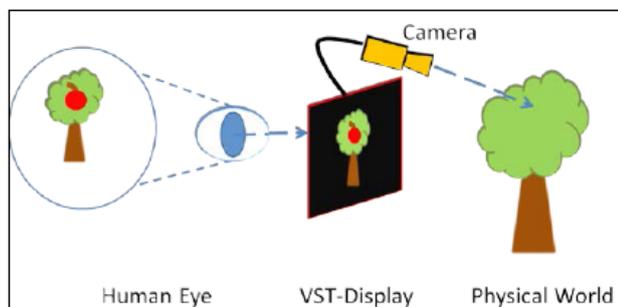
La realidad aumentada es una nueva manera de interactuar con el mundo físico (o real). Esta crea una versión modificada de la realidad que se observa, enriquecida con la información digital (o virtual) en la pantalla de la computadora de escritorio o de un teléfono móvil. (Grubert & Dr Grasset, 2013)

Mediante la combinación de los elementos virtuales con los reales permite la creación una realidad mixta en tiempo real, la cual permite al usuario interactuar de manera dinámica e interactiva con su entorno. (Véase Figura 1)



**Figura 1 Realidad Aumentada**  
(Drgoldie, 2009)

Este tipo de realidad permite el reconocimiento de patrones o puntos claves para iniciar por medio del software la adición de una capa de información digital utilizando la tecnología *Video See-Through (VST)*, en la cual, se busca visualizar en entorno real mediante una pantalla y a su vez visualizar el contenido adicional.



**Figura 2 Ejemplificación Video See-Trough (VST)**  
(Grubert & Dr Grasset, 2013)

En la Figura 2 muestra que al coexistir la realidad física con la virtual, la información que se adiciona podrían ser imágenes, sonido, texto o video. De esa manera se cumple con uno de sus objetivos de mejorar la percepción del usuario ante la realidad y permitirle interactuar con elementos no presentes en lo real.

La realidad aumentada esta posesionándose cada vez más en el mundo de los teléfonos celulares, debido a que compañías invierten en el desarrollo de aplicaciones que usen este tipo de realidad. En los campos que la realidad aumentada se encuentra incursionando son en la medicina, educación, turismo y principalmente en el entretenimiento.

### 2.1.2 CARACTERÍSTICAS

Se puede considerar como realidad aumentada cuando cumple con tres características esenciales:

- Combinar el mundo real y el mundo virtual.
- Interactuar en tiempo real.
- Registrado en 3D.

### 2.1.3 SUBSISTEMAS DE FUNCIONAMIENTO

El funcionamiento de las aplicaciones de realidad aumentada consta de una salida a la que se le llama Visualización, un registro que es la ubicación

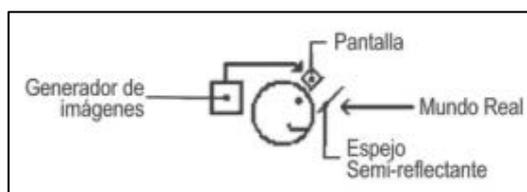
de los objetos y una entrada que es el método de interacción que se utiliza. A continuación se detalla cada subsistema de funcionamiento:

### 2.1.3.1 VISUALIZACIÓN

En la realidad aumentada es necesario el uso de dispositivos ya sean cascos, gafas o teléfonos móviles para poder mostrar las imágenes. La combinación de real y virtual con sus métodos son los siguientes:

- Método de Visión Directo

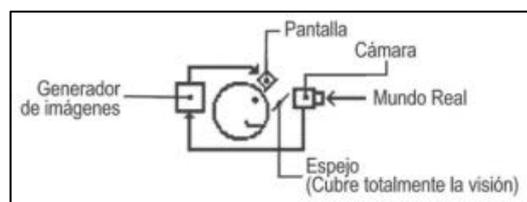
Este método utiliza un espejo semireflectante para observar el mundo real y proyectar las imágenes virtuales. (Véase Figura 3)



**Figura 3 Método de Visión Directo**  
(Ochoa, 2013)

- Método de Visión Indirectos

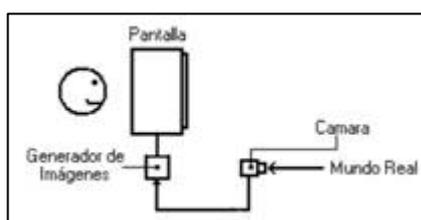
En este método la visión de la persona queda totalmente cubierta y las imágenes procesadas del mundo real con el virtual se proyectan. (Véase Figura 4)



**Figura 4 Método de Visión Indirecto**  
(Ochoa, 2013)

- Método de Visión con Monitores Externos

Se asemeja al Método de Visión Indirecta con la diferencia que se utiliza dispositivos para receptor la imagen del mundo real. Este método es común en los videos juegos. (Véase Figura 5)



**Figura 5 Método de Visión con Monitores Externos**  
(Ochoa, 2013)

### 2.1.3.2 UBICACIÓN DE OBJETOS VIRTUALES

Es el registro de los objetos virtuales de cualquier clase con el mundo real en una posición específica, así el usuario aunque se mueva podrá ver la información digital en la misma ubicación.

Los sensores son los que captan la posición de los objetos en el mundo real. En la Figura 6 se muestran los diferentes tipos de registro son los siguientes:



**Figura 6 Tipos de Ubicación**  
(Ochoa, 2013)

### 2.1.3.3 MÉTODO DE INTERACCIÓN

Es el método escogido para manipular o modificar los objetos y así permitir la comunicación entre usuario – Realidad Aumentada, puede ser el

método de marcadores o el método sin marcadores, donde se utiliza GPS, brújulas, etc.

#### **2.1.4 DIFERENCIAS CON LA REALIDAD VIRTUAL**

Es necesario dar a conocer las diferencias de cada tipo de realidad debido a que al ser temas de interés muchas veces se los llega a confundir. A continuación el cuadro comparativo:

#### **Cuadro 1**

#### **Diferencias entre Realidad Aumentada y Realidad Virtual**

<b>REALIDAD VIRTUAL</b>	<b>REALIDAD AUMENTADA</b>
<b>DIFERENCIAS</b>	
Sustituye de manera completa a la realidad física.	No sustituye la realidad física, sobreimprime información virtual sobre ella.
Sistema informático que genera en tiempo real representaciones del mundo físico.	Sistema informático que genera una visión directa o indirecta de un entorno físico en el mundo real.
Introduce al usuario en un ambiente informático digital.	No aleja al usuario de la realidad y permite la interacción con el entorno digital.

#### **2.1.5 TIPOS DE REALIDAD AUMENTADA**

Se establecen claramente dos tipos de realidad aumentada, basada en marcadores o imágenes y la basada en posicionamiento. El detalle de cada tipo a continuación:

##### **2.1.5.1 REALIDAD AUMENTADA BASADA EN MARCADORES**

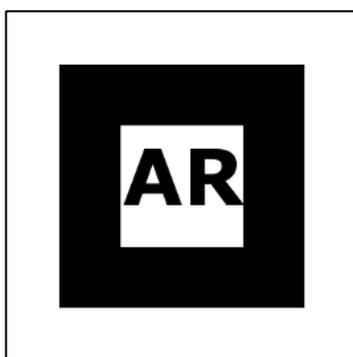
Este tipo de realidad utiliza símbolos impresos, a los que se les conoce como marcadores y también puede utilizar imágenes. La cámara del dispositivo móvil enfoca estos marcadores en un ambiente real y el usuario

podrá ver la superposición de cualquier información virtual, comúnmente de imágenes en 3D o vídeos.

Los marcadores se caracterizan por ser de color negro con blanco y contienen un diseño relacionado con la información virtual. Para acceder al contenido virtual no se necesita una conexión a internet tan solo el software apropiado.

Los códigos QR pueden ingresar al grupo de realidad aumentada basada en marcadores ya que la información o acción para la que fueron creados está codificada en el propio patrón y son interpretadas por un lector QR.

A continuación en la Figura 7 un ejemplo de reconocimiento basado en marcadores:



**Figura 7 Ejemplo de Marcador**  
(AumentaME, 2011)

#### **2.1.5.2 REALIDAD AUMENTADA BASADA EN POSICIÓN**

Este tipo de realidad aumentada se viene desarrollando desde el 2009 y se los conoce como navegadores de realidad aumentada. En este caso se trabaja con la cámara del teléfono móvil o Tablet, sensor de posicionamiento, brújula y acelerómetro.

Al trabajar en conjunto ubican la posición, altura y dirección del dispositivo mediante la cámara, con esos datos el software de realidad

aumentada puede mostrar la información virtual al usuario a través de la pantalla. (Véase Figura 8)

Para acceder a la información virtual el dispositivo necesariamente debe estar conectado a internet.



**Figura 8 Ejemplo de Realidad Aumentada basada en Posición**  
(WION, 2013)

### 2.1.6 ACCESO A LA REALIDAD AUMENTADA

La realidad aumentada está disponible para todas las personas que posean los siguientes elementos:

- **Cámara:** Necesaria para la activación de la aplicación de Realidad Aumentada. Enfoca ya sea los puntos de interés (POI) o los marcadores impresos.
- **Información Virtual:** Lo que el usuario ve superpuesto a la realidad física.
- **Pantalla:** Necesaria para poder visualizar la información virtual.

- **Aplicación de Realidad Aumentada:** Software necesario para interpretar la aplicación y la reproduzca en la pantalla.

### 2.1.7 APLICACIONES DE LA REALIDAD AUMENTADA

A manera que la realidad aumentada va estableciéndose cada vez más en el ámbito tecnológico, sus usos se enfocan diferentes campos de la vida cotidiana o científicos.

Con las aplicaciones desarrolladas se pretende ayudar al usuario con la ampliación y mejor entendimiento de la realidad física, entre los campos donde más se utiliza la realidad virtual se encuentran los siguientes:

- **Educación:** Existen las aplicaciones que ayudan a los niños a conocer de mejor manera las asignaturas tales como: ciencias naturales, matemáticas, inglés o informática. La realidad aumentada genera una manera didáctica y a la vez entretenida del aprendizaje.
- **Marketing:** Actualmente las empresas utilizan la realidad aumentada para poder presentar de una mejor manera sus productos mediante el uso de catálogos interactivos o campañas publicitarias.
- **Entretenimiento:** Este ámbito es uno de los más grandes en el uso de realidad aumentada, los video juegos en donde se pueda interactuar con el entorno llaman cada vez la atención. A su vez el número de desarrolladores de juegos de realidad aumentada va creciendo.
- **Turismo:** Este campo es nuevo y demandante, los turistas prefieren utilizar aplicaciones que les ayude a encontrar puntos de interés con el uso de su teléfono inteligente. Mediante el uso de guías virtuales los usuarios reciben información sobre museos, centros de ocio, hoteles, restaurantes, etc.

En la Figura 9 un resumen de varios de los usos que ofrece la realidad aumentada:



**Figura 9 Usos de la Realidad Aumentada**  
(Del Alcazar Ponce, 2014)

### 2.1.8 AVANCES E INNOVACIÓN

Debido a los usos de la realidad aumentada en ámbitos como la educación, entretenimiento, ocio, turismo y más, las aplicaciones no solo para teléfonos móviles sino también para tablets, televisores inteligentes o gafas han ido creciendo e innovándose. A continuación ejemplos de avances de esta tecnología:

- La empresa que fabrica los famosos ladrillos, LEGO, lanzó una nueva línea comercial llamada Fusion, la cual incluye sets que se integran con la realidad aumentada. (Véase Figura 10)

Cada pieza tiene un marcador de que al momento de ser escaneadas con la Tablet muestra personajes virtuales que interactúan con los bloques. Se podrá guardar y compartir las creaciones, así como también crear ambientes como pueblos mitológicos o grandes ciudades.



**Figura 10 Aplicación LEGO Fusion**  
(LEGO, 2014)

- Los SmartGlasses son sin duda una muestra de la innovación de la realidad aumentada, empresas como Google y Epson han desarrollado gafas de visión binocular con el sistema operativo Android.

Por parte de Epson se cuenta las con las gafas Moverio BT-200. (Véase Figura 11) Las Moverio BT-200 ya están generando todo un nuevo terreno de opciones, permitiendo a los desarrolladores Android llevar las aplicaciones a un nuevo nivel.

Las SmartGlasses de Epson serán un buen compañero para mejorar eficiencia y productividad en entornos industriales, de mantenimiento, formativos, servicios y cualquier entorno donde la interactividad permita ampliar información útil de forma sencilla. (SOCIALetic, 2014)



**Figura 11 SmartGlasses Epson Moverio BT-200**  
(The Geek Church, 2014)

Por parte de Google un referente muy clamado, son las Google Glass que combinan la tecnología con un diseño moderno. (Véase Figura 12) Las gafas inteligentes de Google prometen revolucionar la forma de ver el mundo.

Los usuarios podrán visualizar información ante sus ojos y navegar a través de internet así como usar diferentes aplicaciones multimedia mientras realizan las tareas de su día a día. Google Glass ya ha demostrado su potencial en medicina, educación o industria. (Climent, 2014)



**Figura 12 Google Glass**  
(Planet Numbers, 2014)

- La realidad aumentada en el marketing es de gran ayuda por ello McDonald's con motivo al mundial de fútbol 2014 presentó una aplicación que permite a los clientes hacer goles virtuales. (Véase Figura 13)

Los envases de las papas fritas son los arcos y los goles realizados ingresan a un ranking mundial. Adicional, la aplicación muestra otras

características como pintarse la cara con los colores de las banderas de los países participantes.



**Figura 13 Aplicación de Realidad Aumentada de McDonald's**  
(Nieto, 2014)

- El video juego Stringer de Ben Sainsbury se basa en la realidad aumentada para poder entrenar a los periodistas de guerra en áreas de conflicto, este video juego funcionará con las SmartGlasses Oculus Rift mediante la simulación de escenas los periodistas sabrán cómo reaccionar ante las situaciones tal como lo muestra la Figura 14.



**Figura 14 Stringer Video juego de Realidad Aumentada**  
(Kotzer, 2014)

## **2.2 ANDROID**

### **2.2.1 DEFINICIÓN**

Android es un sistema operativo open source basado en el Kernel de Linux y en el lenguaje de programación Java de Oracle para crear aplicaciones y contenido libre para usuarios de tanto para teléfonos celulares como de tabletas, SmartTVs y más dispositivos inteligentes conectados a internet.

Android fue creado por Andy Rubin en el 2003 tras pasar por una serie de intentos por darse a conocer en el mercado de las telecomunicaciones. En el 2005, Google compró Android Inc., creando el nuevo departamento de Plataformas Móviles de Google y posesionó a Rubin como Vicepresidente de Ingeniería de Google.

Dos años más tarde, 5 de Noviembre del 2007 Google hace la presentación oficial de Android como sistema operativo y en el 2008 se lanzó al mercado el HTC Dream con la primera versión. A partir de ese entonces Android ha comenzado a posicionarse en el mercado superando a Nokia Symbian y Microsoft Windows Mobile.

Android está disponible gratuitamente para el uso comercial, gracias al hecho de ser open source permite que más desarrolladores lo prefieran. Los desarrolladores de Android pueden desarrollar no sólo para los teléfonos inteligentes, sino también para dispositivos electrónicos nuevos que son compatibles para conectar a la red y por lo tanto posibles de conectar al Android Market. (Jackson, 2011)

### **2.2.2 CARACTERÍSTICAS**

Conforme avanza el tiempo las versiones de Android van mejorando en diferentes aspectos o incorporando nuevas tecnologías. Entre las principales características del sistema operativo Android se destaca las siguientes:

## Cuadro 2

### Características de Android

CARACTERÍSTICAS	DETALLE
<b>Framework de Aplicaciones</b>	Permite el acceso total a los códigos dando la posibilidad a que estos sean eliminados, modificados o reutilizados.
<b>SQLite</b>	Una base de datos para el almacenamiento estructurado.
<b>Conectividad</b>	Soporta tecnologías como CDMA, UMTS, Bluetooth, EV-DO, IDEN, GSM/EDGE, Wi-Fi, LTE y WiMax. (Depende del terminal)
<b>Navegador Web Integrado</b>	Basado en el motor open source WebKit emparejado con JavaScript de Google Chrome.
<b>Máquina Virtual Dalvik</b>	Diseñada para que el bytecode de Java sea compilado en un ejecutable Dalvik.
<b>Multimedia</b>	Soporte de varios medios con formatos comunes de audio, imágenes y video: MPEG4, MP3, H.264, AAC, JPG, AMR, PNG, GIF, etc.
<b>Mensajería</b>	SMS y MMS
<b>Tethering</b>	Permite que el terminal se convierta en un Access Point y a un PC usar la conexión de datos móviles.
<b>Pantalla Multi-Táctil</b>	Soporte nativo.
<b>Sensores</b>	Cámaras, GPS, aceleración, Wi-Fi, proximidad, etc.

### 2.2.3 ARQUITECTURA Y HARDWARE

Android ha dispuesto una arquitectura que se compone de 5 capas. Cada capa cumple con funciones específicas y ofrecen servicios diferentes a las demás. A continuación se detalla cada una de ellas:

#### 1. Capa de Linux Kernel

Android tomó el código de Linux Kernel y lo ha modificado para que este funcione en un entorno embebido o integrado. Por lo tanto, no tiene todas las características de una distribución tradicional de Linux. (Dubey & Misra, 2013)

Esta es una capa de abstracción para el hardware de Android, para las primeras versiones utiliza la versión del Kernel Linux 2.6 pero a partir de la

versión Ice Cream Sandwich Android utiliza la versión 3.0.x. El núcleo de Linux contiene los drivers, la pila de red y la gestión de memoria y procesos para que cualquier dispositivo pueda ser utilizado en conjunto con el software.

### Cuadro 3

#### Versión Android vs Versión Kernel Linux

Versión de Android	Versión de Linux Kernel
1.0	2.6.25
1.5 (Cupcake)	2.6.27
1.6 (Donut)	2.6.29
2.2 (Froyo)	2.6.32
2.3 (Gingerbread)	2.6.35
3.0 (Honeycomb)	2.6.36
4.0.x (Ice Cream Sandwich)	3.0.1
4.2/4.3 (Jelly Bean)	3.4.y
4.4 (KitKat)	3.10.y
5.0 (Lollipop)	3.16

## 2. Capa del Entorno de Ejecución Android

Capa que se encuentra al mismo nivel de la capa de librerías, cuya función es ser el motor que pone en marcha las aplicaciones. Al entorno de ejecución lo constituyen:

- **Librerías de Núcleo:** Librerías con multitud de clases Java.
- **Máquina Virtual Dalvik:** Dalvik VM es un intérprete que solo ejecuta archivos con formato *.dex* (*Dalvik Executable*). Este formato está optimizado para el almacenamiento eficiente de la memoria lo cual consigue delegando en el Kernel la gestión de hilos (*multithreading*), de memoria y procesos. (Universidad Carlos III De Madrid, 2012)

Máquina virtual para dispositivos con poca memoria y poca capacidad de proceso. En Android cada proceso tiene su propia instancia.

### 3. Capa de Librerías

Capa en donde se encuentran todas las librerías que utiliza Android, las cuales están escritas en el lenguaje C/C++ y son las que le dan a Android sus características. Trabajan en conjunto con las Librerías de Núcleo del entorno de ejecución. Entre las librerías de importancia se detallan las siguientes:

- **Libc:** Contiene las cabeceras y funciones del lenguaje C.
- **Surface Manager:** Realiza la gestión de las ventanas activas en cada momento y proporciona facilidad en la navegación.
- **SGL:** Librería que maneja los gráficos 2D.
- **OpenGL ES:** Librería que maneja los gráficos 3D.
- **Media Libraries:** Proporciona los códecs necesarios para el contenido multimedia (audio, imagen y video).
- **FreeType:** Facilita el trabajo con distintos tipos de fuentes.
- **SSL:** Protocolo para establecer conexiones seguras.
- **SQLite:** Para la creación de bases de datos.
- **WebKit:** Núcleo del navegador incluido por defecto.

### 4. Application Framework

Capa relacionada directamente con el desarrollo y manejo de las aplicaciones de Android. De igual manera permite compartir, publicar y reutilizar los servicios y datos con otras aplicaciones.

Las aplicaciones utilizan un mismo conjunto de APIs y el mismo Framework. Las APIs de importancia se encuentran las siguientes:

- **Activity Manager:** API que gestiona el ciclo de vida de las aplicaciones.
- **Views System:** Permite la construcción de interfaces de usuario para actividades mediante herramientas como botones, listas, etc.
- **Notification Manager:** Realiza el aviso de cualquier evento que esté sucediendo en el teléfono móvil a cualquier momento.
- **Location Manager:** Permite manejar la ubicación con el uso de GPS.
- **Telephone Manager:** API de las funcionalidades del teléfono como llamadas, mensajes, etc.

## 5. Capa de Aplicaciones

Este nivel contiene, tanto las aplicaciones incluidas por defecto de Android como aquellas que el usuario vaya añadiendo posteriormente, ya sean de terceras empresas o de su propio desarrollo. Todas estas aplicaciones utilizan los servicios, las API y librerías de los niveles anteriores. (Universidad Carlos III De Madrid, 2012)

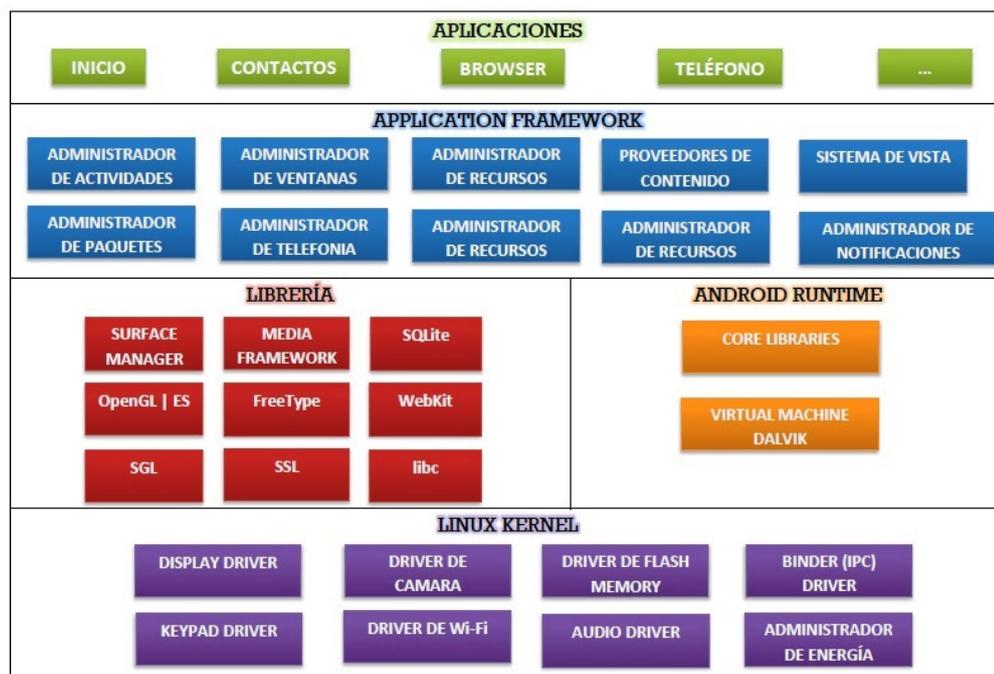


Figura 15 Arquitectura de Android

## 2.2.4 VERSIONES

La cronología de las versiones que Android ha ido desarrollando muestra una mejora en cuanto a la eliminación y corrección de bugs. Así también diferentes características y funcionalidades en cada una de ellas.

En el año 2008, Android Inc., da a conocer la primera versión 1.0 y en el año 2009 se lanza la versión 1.1 la cual fue la primera en ser integrada en un smartphone en España. A partir de la versión 1.5 cada una lleva un nombre clave y en orden alfabético.

A continuación, el detalle de las características de todas las versiones que Android ha desarrollado hasta lo que va del año 2014:

1. Versión 1.0 : Android Apple Pie
  - Soporte de resolución para cámara.
  - Navegador web para páginas en formato HTML y XHTML.
  - Sincronización Gmail, Calendario y Contactos.
  - Soporte para POP3, IMAP4 y SMTP.
2. Android 1.1: Banana Bread
  - Mejora en Google Maps.
  - Permite envío de archivos adjuntos en mensajes.
  - Resolución de fallos de la versión 1.0.
3. Versión 1.5: Cupcake
  - Basada en el Kernel de Linux 2.6.27.
  - Soporte teclados con texto predilecto y diccionario personalizado.
  - Se incorpora velocidad en la cámara.
  - Soporte para sonido estéreo Bluetooth.
4. Versión 1.6: Donut
  - Basado en el Kernel de Linux 2.6.29.
  - Soporte para tecnologías CDMA, VPN y 802.1x.
  - Sistema operativo en múltiples resoluciones de pantalla.

- Inclusión de Quick Search Box con autocompletado y aprendizaje.
5. Android 2.0/2.1: Eclair
    - Navegador con soporte HTML 5.
    - Incorporación para flash de cámara.
    - Incorporación Bluetooth 2.1.
    - Soporte Microsoft Exchange.
    - Cambio en API y comportamiento de Framework.
  6. Android 2.2: Froyo
    - Mejoras en la velocidad, memoria y rendimiento.
    - Incorporación JavaScript V8.
    - Soporte para el servicio Android Cloud to Device Messaging.
    - Funcionalidad de anclaje de red por USB y Wi-Fi hotspot.
    - Discado por voz.
  7. Android 2.3: GingerBread
    - Basado en el Kernel de Linux 2.6.35.
    - Mejora en copiar y pegar tras la selección del texto.
    - Soporte de multitáctil para el teclado.
    - Soporte para cámara frontal.
    - Apoyo a la tecnología NFC.
  8. Android 3.0: Honeycomb
    - Versión únicamente para Tablets.
    - Basada en el Kernel de Linux 2.6.36.
    - Agrega barra de sistema con accesos rápidos.
    - Multitarea simplificada.
    - Soporte Flash y Divx e integra Dolphin.
  9. Android 4.0: Ice Cream Sandwich
    - Multiplataforma.
    - Basado en el Kernel de Linux 3.0.1.
    - Mejora en el sistema de corrección del teclado.
    - Posibilidad de crear carpetas en la pantalla de inicio.

- Soporte de tecnología NFC.
- Opciones de desbloqueo por contraseña, por patrón y por reconocimiento facial.
- Gestor de uso de datos.
- Sincronización automática con los marcadores de Chrome.
- Cámara sin retardo en el obturador.
- Incorporación Wi-Fi Direct.

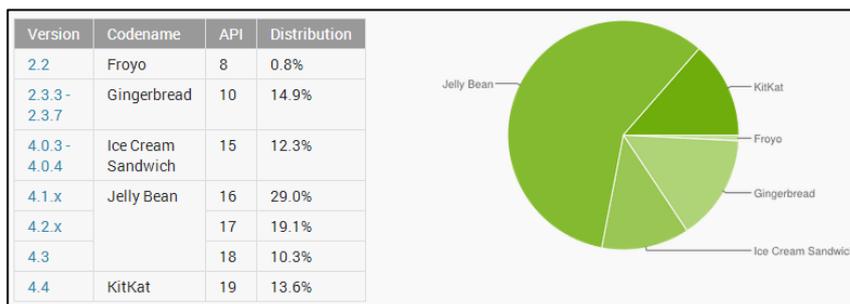
#### 10. Android 4.1/ 4.2/ 4.3: Jelly Bean

- Basado en el Kernel de Linux 3.0.31.
- Anticipación táctil triple buffer.
- Latencia vsync extendida.
- Mejora de velocidad en 60 fps.
- Soporte para Bluetooth de baja energía.
- Soporte para 5 idiomas más.
- Wi-Fi en segundo plano.
- Mejoras en la seguridad.
- Incorporación llamada por gestos.
- Integración de Google Cloud Sprint.

#### 11. Android 4.4: KitKat

- Compatibilidad MAP de Bluetooth.
- Compatibilidad con Chromecast.
- Cambio de pantalla de inicio.
- Actualización de la aplicación de Correo.
- Reproducción de audio de baja energía.
- Podómetro integrado.
- Pagos NFC.
- Integración HTTP *Live Streaming*.
- Incorpora GLES2.0.
- Conexión por infrarrojos (IR Blasters).

La Figura 16 muestra una estadística las diferentes versiones de Android distribuidas en el mundo. Estos datos se los obtiene cada 7 días y los datos menores a 0.1% no se muestran.



**Figura 16 Porcentaje de las Versiones de Plataformas Junio 2014**  
(Android Developers, 2014)

## 2.2.5 DISEÑO Y DESARROLLO

Android se encuentra entre las plataformas preferidas para realizar aplicaciones para teléfonos celulares y tablets. Para ello se utiliza un lenguaje de programación conocido como lo es Java y las herramientas de desarrollo provistas por Android (SDK, *Software Development Kit*).

### 2.2.5.1 ECLIPSE

Eclipse es un IDE de código abierto principalmente para Java que permite programar de manera fácil y también permite la instalación de otros lenguajes mediante los plugins.

Eclipse se lo puede descargar de manera gratuita en la página oficial (<https://www.eclipse.org/>). Está disponible para Windows, Linux y Mac OS. Adicional necesita del JDK (*Java Development Kit*).

### 2.2.5.2 ANDROID SDK

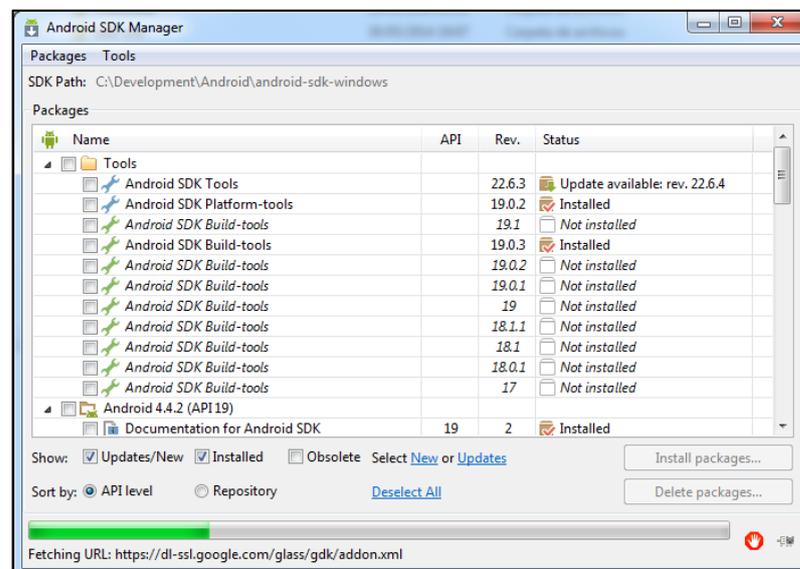
El Android SDK provee las librerías de los API y las herramientas de desarrollo necesarias para crear, probar y depurar las aplicaciones para Android. (Android Developers, 2012).

Android da la facilidad a los desarrolladores de poder utilizar el ADT Bundle, que contiene los componentes importantes y necesarios para la creación de las aplicaciones. Los componentes del ADT Bundle son los siguientes:

- Eclipse + ADT plugin
- Android SDK Tools
- Android Platform-tools
- La última versión de la plataforma de Android
- Emulador Android

El ADT Bundle con todos sus componentes se encuentran disponibles tanto para Windows como para la Mac OS y Linux.

El SDK de Android se compone por paquetes que se descargan mediante el Android SDK Manager, el cual ayuda a mantener actualizado las librerías y herramientas. (Véase Figura 17)



**Figura 17 Android SDK Manager**  
(Android Developers, 2014)

Entre los paquetes importantes incluidos en el Android SDK son los siguientes con su descripción:

## Cuadro 4

### Paquetes de Android SDK

PAQUETE	DESCRIPCIÓN
<b>SDK Tools</b>	Contiene las herramientas para la depuración y pruebas de las aplicaciones.
<b>SDK Plataform-tools</b>	Contiene las herramientas para el desarrollo en cada tipo de plataforma. Cada herramienta soporta las actualizaciones de Android.
<b>Documentación SDK Plataform</b>	Documentos actualizados de las APIs disponibles <i>offline</i> . Librerías disponibles para cada versión Android.
<b>System Images</b>	Imágenes como ARM y x86 para que el emulador pueda funcionar.
<b>Google Play Licensing</b>	Provee librerías estáticas que permiten la verificación de licencias para la distribución de la aplicación.

#### 2.2.5.3 ANDROID VIRTUAL DEVICE

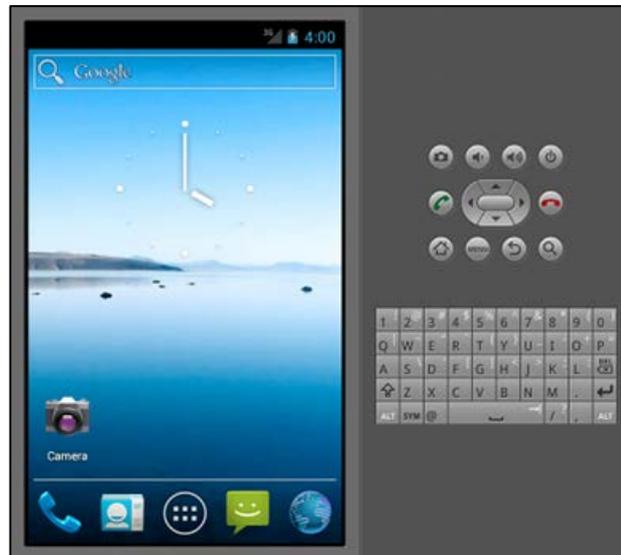
Entre los servicios que posee el SDK está el *Android Virtual Device* que permite emular el hardware y software un dispositivo actual mediante el *Android Emulator*.

Con la ayuda del AVD Manager se puede crear dispositivos con las características reales de hardware y la versión de Android que se requiera. Adicionalmente se puede escoger el tipo de skin con las dimensiones y apariencia que desee el desarrollador.

#### 2.2.5.4 ANDROID EMULATOR

El *Android Emulator* permite realizar pruebas de las aplicaciones sin necesidad de utilizar un dispositivo físico.

El Android Emulator simula todas las características típicas del hardware y software, excepto las llamadas telefónicas. Permite la navegación y control de teclas. También provee una pantalla en donde se presenta la aplicación. (Android Developers, 2012).



**Figura 18 Android Emulator**  
(Android Developers, 2014)

## 2.2.6 SENSORES

Android dispone de 3 grupos de sensores: de movimiento, orientación y de condiciones ambientales. Estos sensores son capaces de entregar datos en bruto con alta precisión y exactitud. (Android Developers, 2014)

Mediante la utilización del el framework se tiene acceso a sensores basados en el hardware, los cuales son componentes físicos del dispositivo, y a sensores basados en el software, los cuales se pueden emular de los sensores físicos obteniendo los datos de los sensores basados en hardware y son comúnmente llamados sensores virtuales. La descripción de cada grupo de sensores que soporta Android, se muestra a continuación.

### 2.2.6.1 SENSORES DE MOVIMIENTO

Los sensores de movimiento manejan la aceleración y la rotación en los tres ejes, como por ejemplo: acelerómetro, sensor de gravedad, giroscopio y sensor rotacional.

El acelerómetro y giroscopio son únicamente basados en hardware mientras que el sensor gravitacional, el sensor de aceleración lineal y sensor rotacional pueden ser basados tanto en hardware o software.

Este tipo de sensores permite el monitoreo del movimiento generado por la reflexión de entrada directa del usuario, en donde se enfoca en el cuadro de referencia de una aplicación, por ejemplo: Un juego donde el usuario controle un balón. También este tipo de sensores permite el censo mediante la reflexión del movimiento físico del entorno en donde el marco de referencia es el mundo en sí.

#### **2.2.6.2 SENSORES DE POSICIÓN**

Los sensores de posición permiten censar la localización física del dispositivo. Android soporta dos tipos de sensores de posición: de Orientación y el sensor de Campo Geomagnético. Cada plataforma de Android provee el sensor de Proximidad.

El sensor de campo geomagnético y el de proximidad son basados en el hardware mientras que el sensor de orientación es basado en el software el cual adquiere sus datos del acelerómetro y del sensor de campo geomagnético.

Este tipo de sensores permite ubicar generalmente la posición física del dispositivo en un cuadro de referencia mundial pero también permite ubicar el dispositivo en un cuadro de referencia respecto a la aplicación.

#### **2.2.6.3 SENSORES DE CONDICIONES AMBIENTALES**

Los sensores de condiciones ambientales se encargan de censar los parámetros del medio ambiente, como por ejemplo: temperatura, presión, iluminación o humedad. Entre estos tipos de sensores esta los barómetros, fotómetros y termómetros.

Los sensores de condiciones ambientales son basados en el hardware por ende están disponibles si son incorporados en el dispositivo con excepción del sensor de luz ya que se los fabricantes lo utilizan para controlar el brillo de la pantalla.

## **2.2.7 FUNDAMENTOS DE LAS APLICACIONES**

Las aplicaciones de Android son escritas en el lenguaje de programación Java y el Android SDK compila el código en un *Android Package* (APK), el cual es un archivo que contiene todos los componentes de la aplicación y es el que se instala en los dispositivos Android.

Una vez instaladas, las aplicaciones se caracterizan por:

- Ejecutarse en su propio proceso Linux.
- Poseen su propio entorno de seguridad.
- Cada proceso posee su propia máquina virtual.
- A cada aplicación se le asigna su user ID de Linux.

El sistema Android implementa el principio de privilegios mínimos. El cual es, cada aplicación, por defecto, tiene acceso solo a los componentes que requiera para realizar un trabajo y nada más. (Android Developers, 2014)

Compartir los datos entre aplicaciones está permitido siempre y cuando las dos aplicaciones posean los permisos para acceder a los archivos y compartan el user ID ayudando así a conservar el procesamiento. Con ese permiso se tiene acceso a los contactos del usuario, SMS, memoria SD, cámara y más.

### **2.2.7.1 COMPONENTES DE UNA APLICACIÓN**

A los componentes de una aplicación se los construye en bloques, cada componente es diferente y solo se accederá a este si la aplicación lo

necesita. Cada componente puede ser individual o dependerá de otros para trabajar. Así se podrá establecer el comportamiento futuro de la aplicación.

Existen 4 tipos de componentes con diferente propósito y ciclo de vida, a continuación se los detalla:

- **Actividades:** Componente que provee una sola pantalla con la cual los usuarios pueden utilizar para realizar alguna acción. Cada actividad tendrá su propia pantalla en la cual se encuentra la interfaz del usuario. (Android Developers, 2014).
- **Servicios:** Componente que permite ejecutar operaciones de larga duración o realiza trabajos de procesos remotos. No proporciona una interfaz de usuario y se ejecuta en segundo plano.
- **Proveedores de Contenido:** Componente que gestiona un conjunto de datos compartidos de una aplicación. Los datos se pueden almacenar en el sistema de archivos, base de datos SQLite, en la red, etc. Los *Content Providers* permiten que las aplicaciones consulten o modificar datos. (Alejandro Alcalde, 2013)
- **Receptor de Notificaciones:** Componente que responde a la emisión de un mensaje en el sistema, no muestra interfaz de usuario y puede crear una notificación en la barra de estado. Un receptor de notificaciones es un Gateway a otro componente. Ejemplo: Notificación de batería baja.

**Intents**, son los mensajes de solicitud de una acción a un componente y permiten la comunicación entre componentes individuales en tiempo de ejecución. Cada intent es explícito y define una acción a realizar.

### 2.2.7.2 TIPOS DE APLICACIONES

Las aplicaciones de Android se las puede categorizar en 4 distintos tipos, a continuación su detalle:

- **De Primer Plano o Foreground:** Son aquellas que solo se utilizan cuando el usuario la está usando, si se cierra deja de actuar.
- **De fondo o Background:** Aquellas aplicaciones que interactúan limitadamente con el usuario y esperan un evento para ser activadas.
- **Intermitentes o Intermittent:** Aplicaciones interactivas, la mayoría de su trabajo es de fondo y cuando sea necesario será de primer plano.
- **Widget y Live Wallpapers:** Aplicaciones especialmente diseñadas para la pantalla de inicio del dispositivo.

### 2.3 SDK PARA REALIDAD AUMENTADA

Existen una gran variedad de SDK encargados del desarrollo de aplicaciones de realidad aumentada, entre los más importantes por sus beneficios y características están los siguientes:

#### 2.3.1. VUFORIA

Vuforia es un SDK de la empresa Qualcomm que está especialmente diseñada para aplicaciones de realidad aumentada compatibles con Android, iOS y con el framework de desarrollo Unity 3D.

La plataforma de Vuforia usa una eficiencia superior, estable y técnica de computación basada en reconocimiento de imágenes y ofrece el amplio set de características y capacidades para los desarrolladores. (Qualcomm, 2014)



**Figura 19 Logo de Vuforia**  
(TechENT, 2014)

### 2.3.1.1 CARACTERÍSTICAS

Las aplicaciones desarrolladas con Vuforia tendrán características como las siguientes:

- Rapidez en la detección de targets.
- Reconocimiento de texto. Incluyendo texto alfanumérico.
- Reconocimiento de cilindros.
- Tracking robusto y simultáneo de hasta 5 targets.
- Optimización en la calidad con gráficos más realísticos.

### 2.3.1.2 ARQUITECTURA

Una aplicación de Vuforia para realidad aumentada se compone de los siguientes componentes del núcleo:

- **Convertidor de Imagen:** Convierte el formato de la cámara en un formato acorde con OpenGL ES y también acorde para realizar el Tracking.
- **Tracker:** Contiene los algoritmos que permiten detectar y seguir los objetos reales en los cuadros de video de la cámara. Los algoritmos dependen de la imagen que la cámara provea para poder localizar nuevos targets o marcos.
- **Base de Datos de Dispositivos y de Nube:** Bases de datos que contienen un archivo de configuración XML que permite al desarrollador establecer características específicas del tracking en diferentes dispositivos.

### 2.3.2. OPEN CV

OpenCV es una biblioteca de visión artificial desarrollada en el año 1999, liberada bajo la licencia BSD y es libre para usos académicos y comerciales. Es utilizado para crear aplicaciones donde la seguridad y el control de

procesos son primordiales. Open CV aprovecha las capacidades de los procesadores multi-núcleo para permitir una programación optimizada en lenguajes como C o C++. OpenCV es popular en el mundo de los desarrolladores ya que 47 mil personas son usuarios de sus beneficios y se estima que el número de descargas es de 7 millones.



**Figura 20 Logo de OpenCV**  
(Di Nardo, 2013)

### **2.3.2.1 CARACTERÍSTICAS**

OpenCV cuenta con características que ayudan al programador a crear aplicaciones multiplataforma y en tiempo real, a continuación las más importantes:

- Soporta interfaces como C, C++, Python y Java
- Soporta plataformas como Windows, Linux, Mac OS, iOS y Android.
- Diseñado para eficiencia computacional y en tiempo real.
- Tiene funciones de captura y presentación de imágenes.
- Utiliza operaciones con matrices para el procesamiento de imágenes.
- Cross-Platform

### **2.3.2.2 ARQUITECTURA**

En la versión 2.2 de Open CV los creadores subdividieron las funcionalidades y obteniendo así una arquitectura de diseño que muestra así

una relación entre las librerías de imágenes y video. (Momma & Minagawa, 2011) Las funcionalidades de los módulos principales son las siguientes:

- **Core:** Es la definición de las clases y templates.
- **Imgproc:** Es el procesado de imagen.
- **Highgui:** Permite la E/S de las imágenes y videos.
- **MI:** Es la máquina que permite el conocimiento.
- **Calib3d:** Calibración de imagen y reconstrucción 3D.
- **Features2d:** Características de las imágenes 2D.
- **Objdetect:** Detección de objetos.
- **Video:** Análisis de los videos.

### 2.3.3. WIKITUDE

Es un SDK para desarrollar aplicaciones de realidad aumentada lanzado al mercado en el año 2008. Con Wikitude es posible el Geo-AR, reconocimiento de imágenes, tracking y modelado 3D. Con esta tecnología se puede crear grandes y emocionantes experiencias con realidad aumentada. (Wikitude, 2014)



**Figura 21 Logo de Wikitude**  
(Business Wire, 2013)

La forma de trabajo de Wikitude permite una interesante, innovadora e informativa plataforma de realidad aumentada fácil de desarrollar y divertida para los usuarios de dispositivos móviles.

La aplicación Wikitude World Browser fue elegida como el mejor explorador de Realidad Aumentada en el año 2010, en el año 2011 obtuvo el premio a Mejor Herramienta de Desarrollo de Realidad Aumentada y en el año 2012 recibió el premio a Mejor Aplicación de Realidad Aumentada, además consta con más de 1500 aplicaciones desarrolladas con el SDK Wikitude en el mercado de aplicaciones de los dos principales sistemas operativos celulares, también empresas como EPSON, SONY, LG y Samsung utilizan los servicios de desarrollo de Wikitude.

### 2.3.3.1 CARACTERÍSTICAS

El SDK de Wikitude posee librerías que son fáciles de usar, así los desarrolladores pueden crear sus aplicaciones ya que consta con las siguientes características:

- Programación en lenguajes HTML5, JavaScript y CSS.
- Reduce el ciclo de vida de las aplicaciones.
- Posee un motor para los gráficos en 3D.
- Multiplataforma: iOS, Android, BlackBerry y Windows Phone.
- Interface gráfica intuitiva

### 2.3.3.2 ARQUITECTURA

Una aplicación que trabaje con Wikitude consta de los siguientes elementos para poder mostrar su interfaz de realidad aumentada:

- **ArchitectView:** Es un framework para crear experiencias de realidad aumentada que utiliza lenguajes de programación para páginas web como HTML, CSS y JavaScript.

- **World:** Estos “mundos” son la experiencia en realidad aumentada que el usuario puede ver, puede ser por geolocalización o reconocimiento de imágenes. El mundo está implementado en HTML y CSS.
- **Multimedia:** Wikitude permite agregar audio o video a los “mundos”.
- **GPS y Redes Móviles:** Dos elementos importantes para desarrolladores de POIs Browsers.

## CAPÍTULO 3

### DISEÑO E IMPLEMENTACIÓN DE LA APLICACIÓN

En este capítulo se muestra el proceso para diseñar e implementar la aplicación de realidad aumentada cuyo nombre será “QUITO VISIÓN+”.

En la Figura 22 se muestra el plan de desarrollo que consta de 6 fases: Especificación de Requerimientos, Elección de SDK de Realidad Aumentada, Descripción de Opciones de Uso, Diseño e Implementación. Las pruebas realizadas serán detalladas en el Capítulo 4 del documento.



Figura 22 Plan de Desarrollo

#### 3.1 ESPECIFICACIÓN DE REQUERIMIENTOS

Esta fase es la primera del plan de desarrollo de la aplicación, aquí se esclarecerán los requerimientos funcionales de la aplicación, es decir, lo que la aplicación realizará. Las funciones serán 6, las cuales se detallan a continuación:

- **Menú Inicial:** Es la primera interacción que el usuario tendrá con la aplicación, en donde tendrá que presionar el botón de Vista de Realidad Aumentada o la opción de Ayuda.
- **Generar la Realidad Aumentada:** Función que es la esencia de la aplicación, de alta prioridad y será esta la genere en la pantalla del dispositivo del usuario la vista de la información virtual.
- **Mostrar Información Virtual:** Función de prioridad alta que se encarga de sobreponer la información en un panel una vez que el punto de interés (POI) haya sido detectado y seleccionado.
- **Captura de Pantalla:** Es una función opcional que le permite al usuario realizar una foto de lo que observa y si desea compartir la foto en redes sociales, correo electrónico o con otro dispositivo móvil.
- **Rango:** Función opcional que le da al usuario la opción de limitar la visibilidad de POIs de acuerdo a un rango de interés.
- **Ayuda:** Función que permite al usuario acceder a una lista de los lugares a visitar, direccionamiento para la activación del GPS y una breve descripción de la aplicación.

### 3.2 ELECCIÓN DE SDK DE REALIDAD AUMENTADA

La elección de un adecuado *Software Development Kit* (SDK) es la segunda etapa de este proceso, por ello previamente en el Capítulo 2 sección 2.3 se detalla las características y arquitecturas de 3 ambientes de desarrollo de realidad aumentada.

El Cuadro 5 presenta una comparación de características esenciales que cumplieron con los requerimientos de la aplicación desarrollada, los SDK a estudiar son Vuforia, OpenCV y Wikitude:

**Cuadro 5**  
**Comparación entre SDKs**

<b>CUADRO COMPARATIVO</b>						
<b>SDK</b>	<b>RECONOCIMIENTO</b>	<b>TRACKING</b>	<b>PLATAFORMAS</b>	<b>GPS</b>	<b>LICENCIA</b>	<b>DOCUMENTACIÓN</b>
<b>VUFORIA</b>	Imágenes 2D, Marcadores	Marcadores, NFT, Texto	iOS, Android	No	Gratis	Extensa con ejemplos
<b>WIKITUDE</b>	Imágenes 2D, Marcadores, 3D, Geolocalización	NFT, GPS, Marcadores	Android, iOS, Blackberry, Epson Moverio, Google Glass	Si	Pagada \$0-2490	Básica con ejemplos
<b>OPENCV</b>	Imágenes 2D, 3D	Marcadores	Windows, Linux, Mac OS, iOS, Android	No	Gratis y Pagada	Básica con ejemplos

Con el análisis de todas las características de los SDKs presentados, se procedió a elegir Wikitude ya que este cumple con los requerimientos para poder realizar una aplicación basada en geolocalización óptima.

Los requerimientos de la aplicación son:

- Tracking mediante GPS.
- Disponible para dispositivos Android a partir de la versión 2.3.3.
- Experiencia de Realidad Aumentada basada en geolocalización.
- POI Browser.
- Manejo de cámara principal.
- Presentación de detalles.
- Uso de brújula y GPS.
- Uso de Wi-Fi y Datos Móviles.
- Interfaces interactivas con el usuario.
- Presentación de POIs agradable.

### **3.3 DESCRIPCIÓN DE OPCIONES DE USO**

Esta es la tercera etapa en el plan de desarrollo, en la misma que se muestra una descripción general de los usos de las funciones establecidas en los requerimientos que tendrá la aplicación.

A continuación, se muestran cuadros de las 6 funciones de la aplicación, en donde se detalla la actividad que realiza, condiciones previas para su uso, a quien se dirige la actividad, prioridad, importancia y requisitos de hardware y software:

- **Mostrar Menú Inicio**

### Cuadro 6

#### Descripción de Función: Mostrar Menú Inicio

<b>MOSTRAR MENÚ INICIO</b>	
<b>Descripción</b>	La primera pantalla, en donde se encuentra un botón que dirige al usuario a la vista de los puntos de interés en realidad aumentada y una opción de ayuda. 1. Quito en Realidad Aumentada. 2. “?” Ayuda.
<b>Dirigida a</b>	Usuario de Aplicación
<b>Condiciones previas</b>	✓ Haber activado la aplicación desde el menú de aplicaciones en el dispositivo móvil.
<b>Requisitos</b>	✓ Dispositivo Android, mínimo con la versión 2.3.3 Gingerbread.
	✓ Tener instalada la aplicación QUITO VISIÓN+.
	✓ Versión mínima OPENGL 2.0
<b>Prioridad</b>	Alta e importante.

- **Generar la Realidad Aumentada**

### Cuadro 7

#### Descripción de Función: Generar Realidad Aumentada

<b>GENERAR LA REALIDAD AUMENTADA</b>	
<b>Descripción</b>	La función muestra en la pantalla del usuario la vista en realidad aumentada con puntos de interés de acuerdo a la ubicación del usuario.
<b>Dirigida a</b>	Usuario de Aplicación
<b>Condiciones previas</b>	✓ Haber escogido presionado en el botón del menú inicial: Vista en Realidad Aumentada.
<b>Requisitos</b>	✓ Dispositivo Android, mínimo con la versión 2.3.3 Gingerbread.
	✓ Tener instalada la aplicación QUITO VISIÓN+.
	✓ Versión mínima OPENGL 2.0
	✓ Cámara del dispositivo móvil en buen estado.
	✓ Conexión a internet, GPS y Datos móviles activados.
<b>Prioridad</b>	Alta e importante.

- **Mostrar información virtual**

### Cuadro 8

#### Descripción de Función: Mostrar Información Virtual

<b>MOSTRAR INFORMACIÓN VIRTUAL</b>	
<b>Descripción</b>	Función que se encarga de sobreponer la información de los puntos de interés una vez que hayan sido seleccionados mediante un clic del usuario.
<b>Dirigida a</b>	Usuario de Aplicación
<b>Condiciones previas</b>	<ul style="list-style-type: none"> <li>✓ Haber escogido la opción uno del menú inicial: Vista en Realidad Aumentada.</li> <li>✓ Administrador de la aplicación haya creado los POIs e información virtual.</li> <li>✓ Haber seleccionado un POI.</li> </ul>
<b>Requisitos</b>	<ul style="list-style-type: none"> <li>✓ Dispositivo Android desde la versión 3.0 Honeycomb.</li> <li>✓ Versión mínima OPENGL 2.0</li> <li>✓ Tener instalada la aplicación QUITO VISIÓN+.</li> <li>✓ Cámara del dispositivo móvil en buen estado.</li> <li>✓ Conexión a internet, GPS y Datos móviles activados.</li> </ul>
<b>Prioridad</b>	Alta e importante.

- **Captura de Pantalla**

### Cuadro 9

#### Descripción Función: Captura de Pantalla

<b>CAPTURA DE PANTALLA</b>	
<b>Descripción</b>	Función que permite al usuario hacer una foto y compartirla en redes sociales, correo u otro dispositivo.
<b>Dirigida a</b>	Usuario de Aplicación
<b>Condiciones previas</b>	<ul style="list-style-type: none"> <li>✓ Haber escogido la opción uno del menú inicial: Quito en Realidad Aumentada.</li> <li>✓ Presionar el botón CAPTURA.</li> </ul>

Continúa →

<b>Requisitos</b>	<ul style="list-style-type: none"> <li>✓ Tener instalada la aplicación QUITO VISIÓN+.</li> <li>✓ Cámara del dispositivo móvil en buen estado.</li> <li>✓ Conexión a internet, GPS y Datos móviles activados.</li> <li>✓ Dispositivo Android desde la versión 3.0 Honeycomb.</li> </ul>
<b>Prioridad</b>	Baja y opcional.

- **Rango**

#### Cuadro 10

##### Descripción de Función: Rango

<b>RANGO DE INTERÉS</b>	
<b>Descripción</b>	Función que permite al usuario escoger el número de POIs de acuerdo a un rango.
<b>Dirigida a</b>	Usuario de Aplicación
<b>Condiciones previas</b>	<ul style="list-style-type: none"> <li>✓ Haber escogido la opción uno del menú inicial: Quito en Realidad Aumentada.</li> <li>✓ Presionar el botón RANGO.</li> </ul>
<b>Requisitos</b>	<ul style="list-style-type: none"> <li>✓ Cámara del dispositivo móvil en buen estado.</li> <li>✓ Conexión a internet, GPS y Datos móviles.</li> <li>✓ Dispositivo Android desde la versión 3.0 Honeycomb.</li> </ul>
<b>Prioridad</b>	Baja y opcional.

- **Ayuda**

#### Cuadro 11

##### Descripción de Función: Ayuda

<b>AYUDA</b>	
<b>Descripción</b>	Función que permite al usuario acceder a una pantalla que le permitirá conocer los lugares a visitar, activar el GPS y una breve descripción de la aplicación.
<b>Dirigida a</b>	Usuario de Aplicación
<b>Condiciones previas</b>	<ul style="list-style-type: none"> <li>✓ Haber escogido el icono de ayuda del menú inicial.</li> </ul>

Continua →

<b>Requisitos</b>	✓ Tener instalada la aplicación QUITO VISIÓN+.
<b>Prioridad</b>	Baja y opcional.

Cada función que cumple la aplicación fue diseñada con el objetivo de facilitar el uso por parte del usuario y de igual manera que no sea aburrido acceder a la aplicación.

Con la ayuda de los cuadros y sus descripciones se procedió a realizar un diagrama de casos de uso para establecer de manera general el comportamiento de la aplicación frente al usuario.

En la figura 23 se presenta el diagrama de usos que se obtuvo después del análisis de los cuadros. Dicha figura se encuentra en orientación horizontal para mejor visualización:

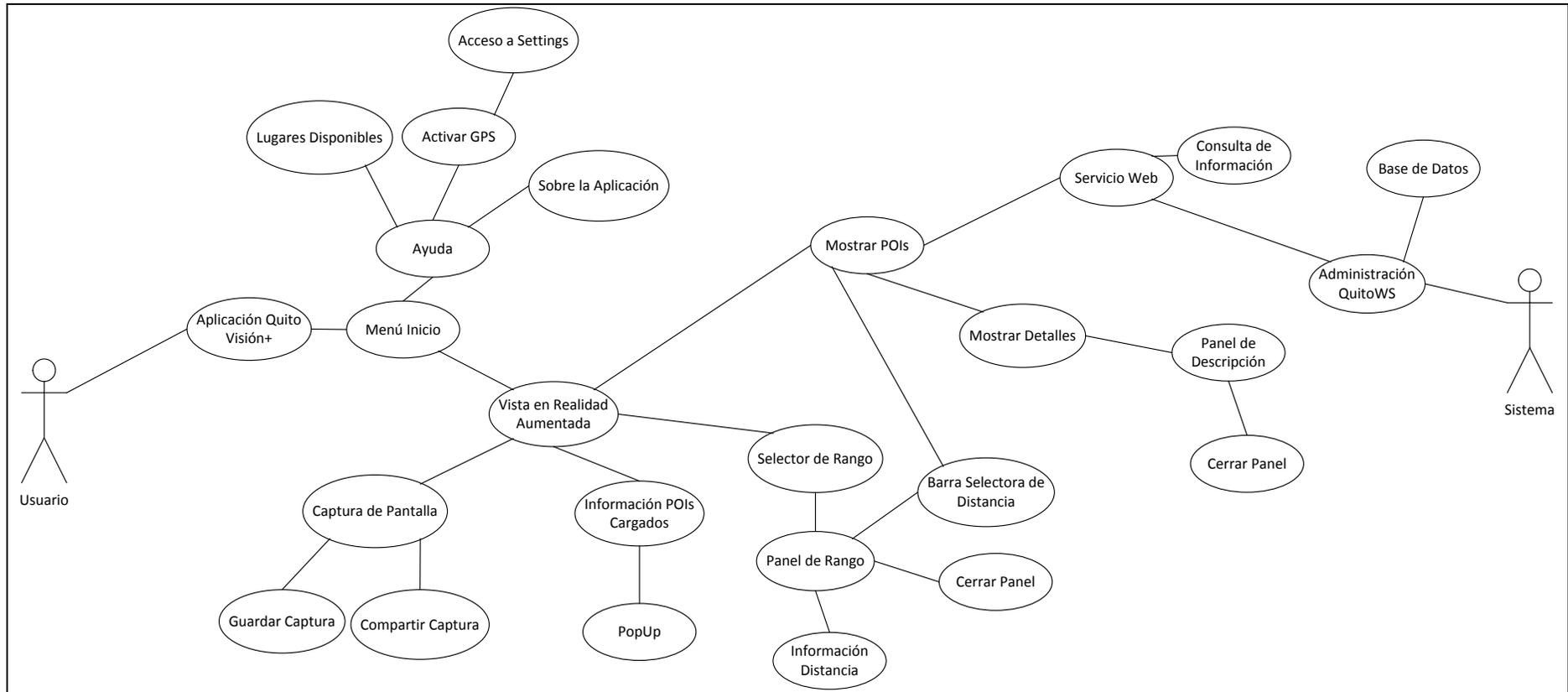


Figura 23 Diagrama de Casos de Uso

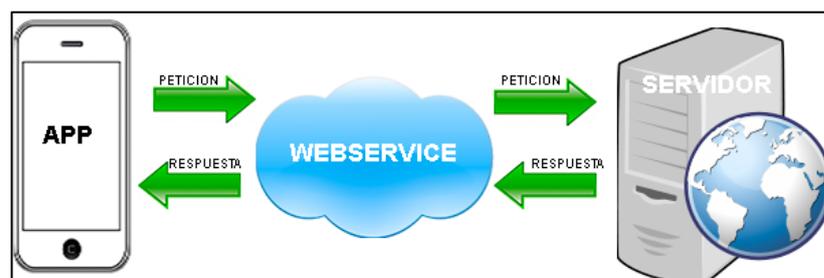
### 3.4 DISEÑO DE LA APLICACIÓN

El tercer punto del plan de desarrollo es el diseño de la aplicación, en donde se utilizará la ayuda de diagramas de clase y secuencia para establecer la arquitectura, componentes y comportamiento de “Quito Visión+”.

#### 3.4.1 ARQUITECTURA

A continuación una breve descripción de la arquitectura de la aplicación que fue diseñada para reducir tiempos de carga y procesos:

- “Quito Visión+” es cliente de un Servicio Web de nombre Quito Visión WS y se puede acceder a este por Internet.
- La aplicación tiene implementada una vista propia del SDK Wikitude, que es en donde se muestra la realidad aumentada como layout.
- Los *Event Listeners* son los controladores que interpretan las acciones del usuario cuando toque la interfaz y así mismo son los que manejan la brújula y el GPS.
- El servicio web cumple la función de realizar las consultas a la base de datos. Estos están alojados en un web hosting y para así ser accesibles desde Internet.
- La base de datos tiene la información como nombre, descripción, latitud, longitud y altitud de los puntos de interés para ser mostrados al usuario.



**Figura 24 Proceso de Consulta de Información**  
(Abellán, 2014)

### **3.4.2 ELEMENTOS**

Los elementos de la aplicación se los separa por Servidor y Cliente, el detalle de cada uno es el siguiente:

#### **SERVIDOR**

El servidor es el que recibe y contesta las peticiones del cliente. La petición del cliente será la información de los POIs y las coordenadas de los mismos. La respuesta del servidor serán los datos consultados en formato JSON.

Los elementos del servidor son los siguientes:

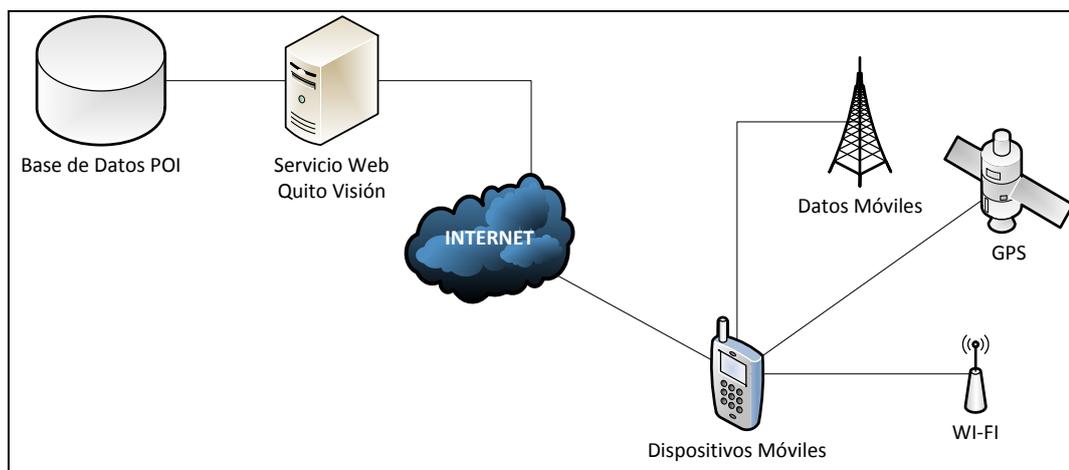
- **Servicio Web:** Elemento clave que sirve de puente entre el cliente y el servidor, se encuentra implementado en lenguaje PHP y se encarga de realizar las consultas a la base de datos y devolver la respuesta en formato JSON.
- **Base de Datos:** Con la ayuda de MySQL que permite el almacenamiento de la información (latitud, longitud, nombre, etc.) de cada POI.
- **Web Hosting:** Es en donde se encuentra publicado el servicio web.

#### **CLIENTE**

La aplicación es el cliente como tal y sus componentes son los siguientes:

- **Capa Negocio:** Consta de cada Activity que se presentará al usuario, la actividad principal es la que maneja y controla la vista de Realidad Aumentada.
- **Capa Presentación:** Consta de los layouts de la aplicación donde el usuario podrá interactuar, por ejemplo: panel de detalles o panel de rango de interés.

En la Figura 25 se puede observar los elementos como base de datos, servicio web y el cliente de la arquitectura física de la aplicación:



**Figura 25 Arquitectura Física**

### 3.4.3 DIAGRAMA DE CLASES

En la Figura 26 se presenta el diagrama de clases, interfaz, atributos y operaciones que se utilizaron en el desarrollo de la aplicación.

Dicha figura se encuentra en orientación horizontal para mejor visualización:

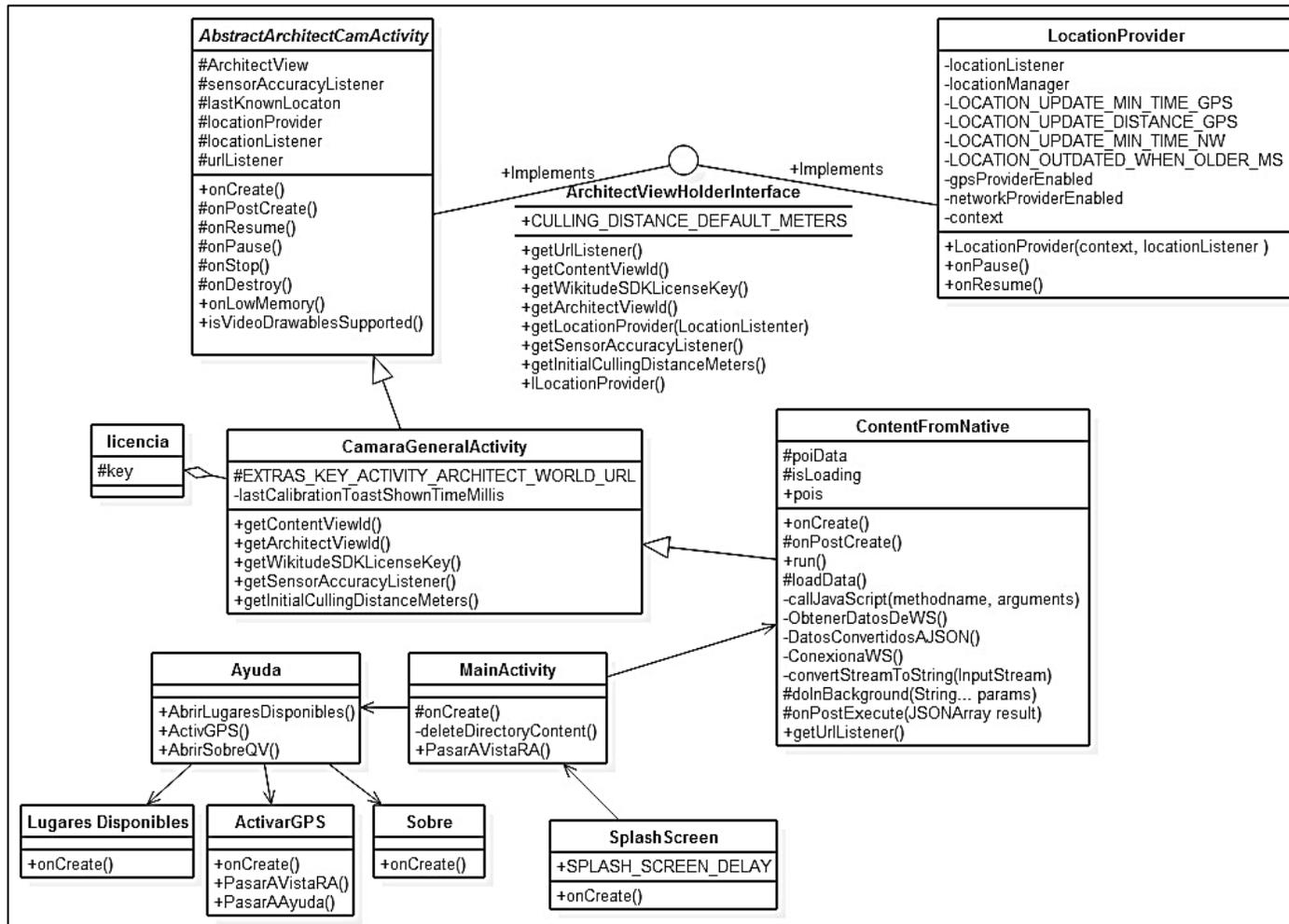
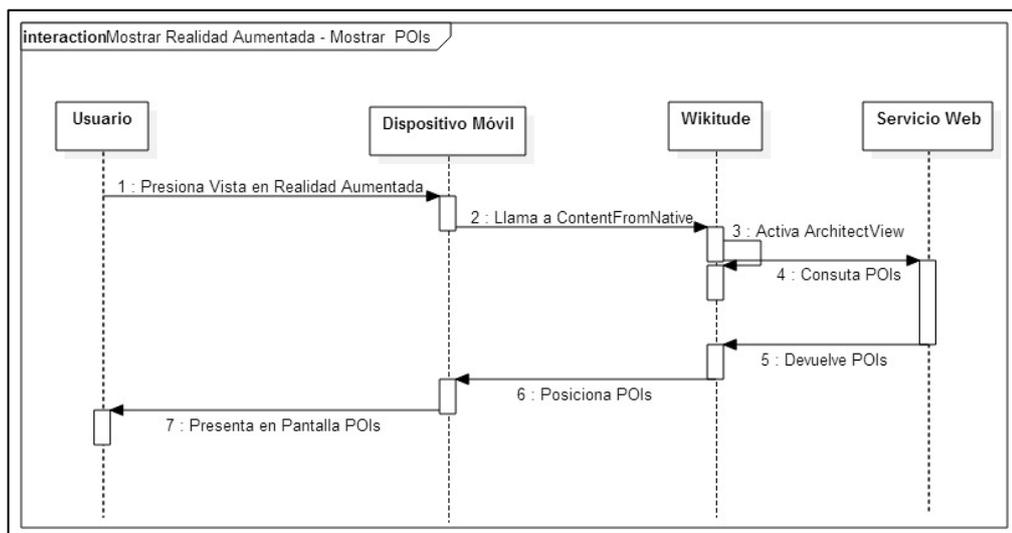


Figura 26 Diagrama de Clases de Quito Visión+

### 3.4.4 DIAGRAMA DE SECUENCIA

Con la ayuda de diagramas de secuencia se muestra las actividades de cada elemento y como es su funcionamiento en cada uso que tiene la aplicación.

La función más importante de la aplicación es presentar al usuario la vista en realidad aumentada con los POIs en las coordenadas correctas. Para ello intervienen en la acción actores como el usuario, dispositivo móvil, el SDK Wikitude y un servicio web, la Figura 27 presenta dicha secuencia:

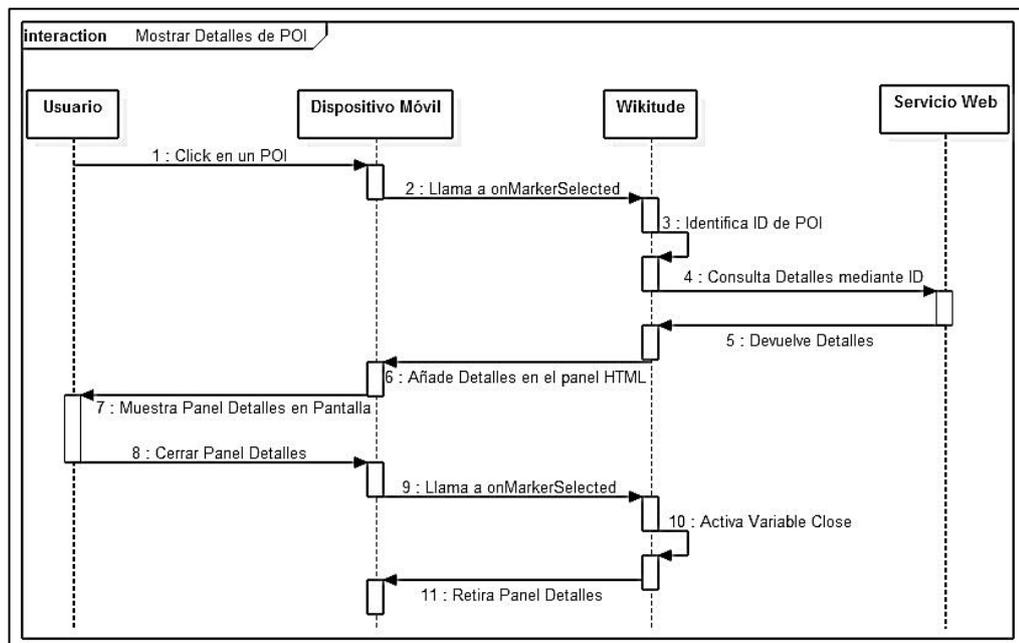


**Figura 27 Diagrama de Secuencia - Vista en Realidad Aumentada**

Una vez presentado al usuario los puntos de interés de acuerdo a su ubicación, éste tiene la opción de acceder a los detalles un POI seleccionado, escoger cuantos POIs desea observar de acuerdo a un rango, capturar una pantalla y si desea compartir la imagen.

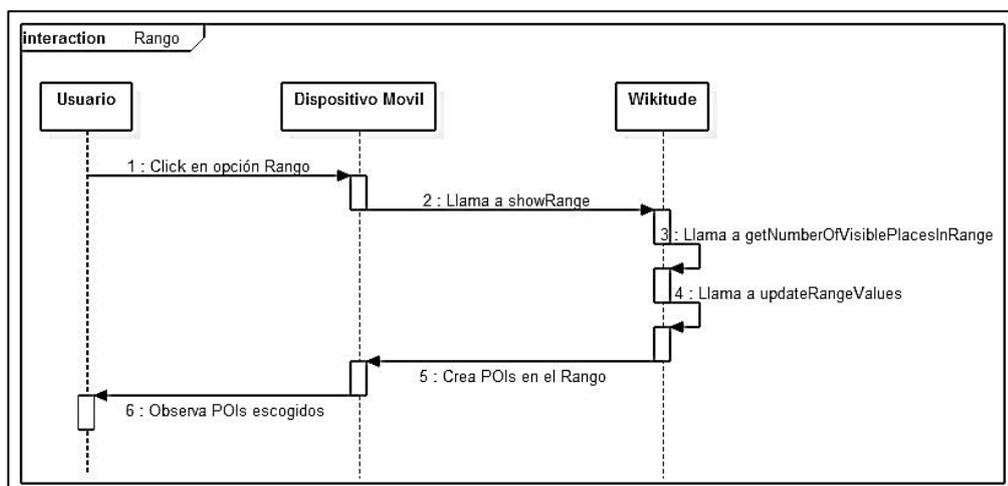
En la Figura 28 se puede observar el diagrama de secuencia cuando el usuario selecciona un POI para acceder a sus detalles, esta acción la realiza el usuario, el dispositivo móvil, el SDK Wikitude y el servicio web que es el

que por medio de una consulta a la base de datos obtiene la información de cada POI y se la presenta al usuario en un panel.



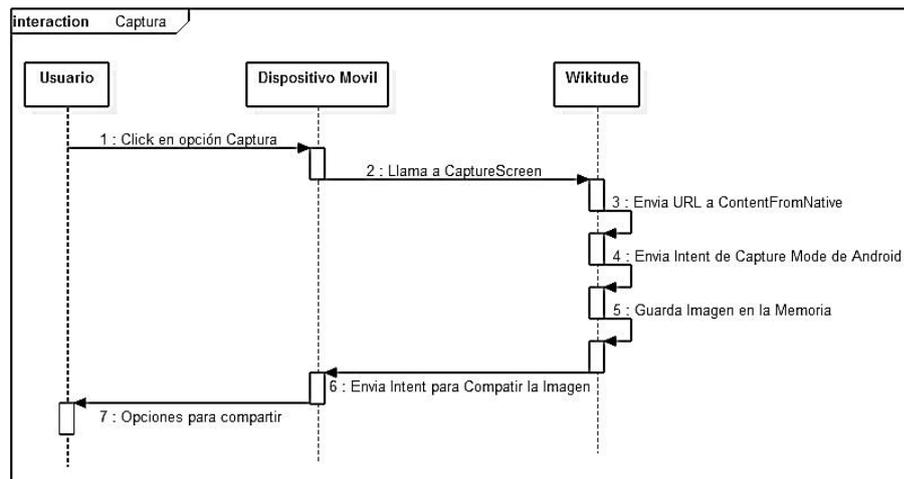
**Figura 28 Diagrama de Secuencia - Mostrar Detalles de POI**

Si el usuario desea puede acceder a un panel en donde se podrá escoger el número de POIs que desee observar de acuerdo a un rango, en la Figura 29 se observa que los actores que intervienen en esta función son usuario, dispositivo móvil y el SDK Wikitude:



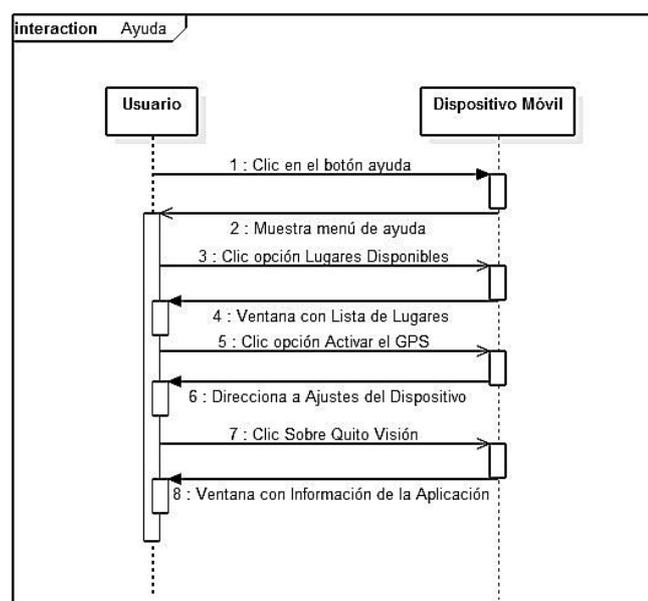
**Figura 29 Diagrama de Secuencia - Rango**

La opción de captura de pantalla es función en donde interviene el usuario, dispositivo móvil y el SDK Wikitude, su diagrama de secuencia se presenta en la Figura 30:



**Figura 30 Diagrama de Secuencia - Captura de Pantalla**

En la función ayuda se implementó netamente en lenguaje nativo Android, los actores son el usuario y dispositivo móvil, la siguiente figura muestra su diagrama de secuencia:



**Figura 31 Diagrama de Secuencia: Ayuda**

### 3.5 IMPLEMENTACIÓN

La cuarta etapa del plan de desarrollo muestra la implementación ya realizada de las funciones de la aplicación descritas en la sección 3.3 del presente capítulo.

#### 3.5.1 NOMBRE DE LA APLICACIÓN

El nombre de la aplicación es: “Quito Visión+”, se eligió el nombre ya que se quiere expresar y mostrar una visión aumentada del centro histórico de la ciudad de Quito.

Para agregarle vistosidad a la aplicación se creó una pantalla de inicio y un logotipo. La Figura 32 muestra el logotipo diseñado para “Quito Visión+”, este logotipo es el icono launcher que se encuentra en el menú de aplicaciones de los dispositivos móviles.



**Figura 32 Logotipo Quito Visión**

La Figura 33 muestra la pantalla de inicio que aparece cuando la aplicación ya fue activada, esta pantalla de bienvenida está disponible para el usuario por un lapso de 3 segundos.

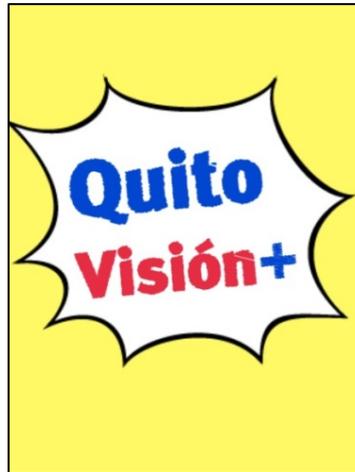


Figura 33 Pantalla de Inicio Quito Visión+

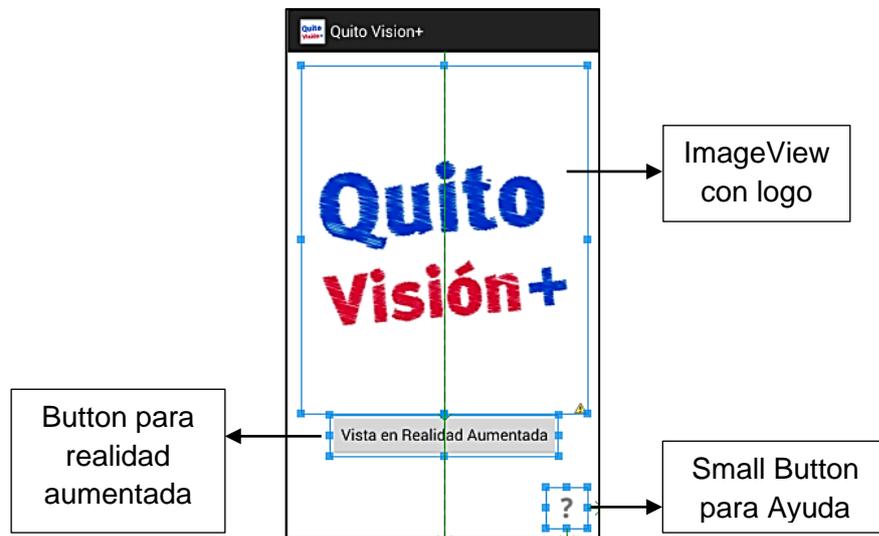
### 3.5.2 DISEÑO DE INTERFACES

Las tres interfaces de mayor importancia presentadas al usuario son las siguientes: Menú Inicio, Vista en Realidad Aumentada y Ayuda.

Cada interfaz se diseñó buscando la facilidad de su uso, agradable manejo e interacción, para su implementación se utilizó lenguaje nativo Android y elementos como: Botones, ImageView y TextView.

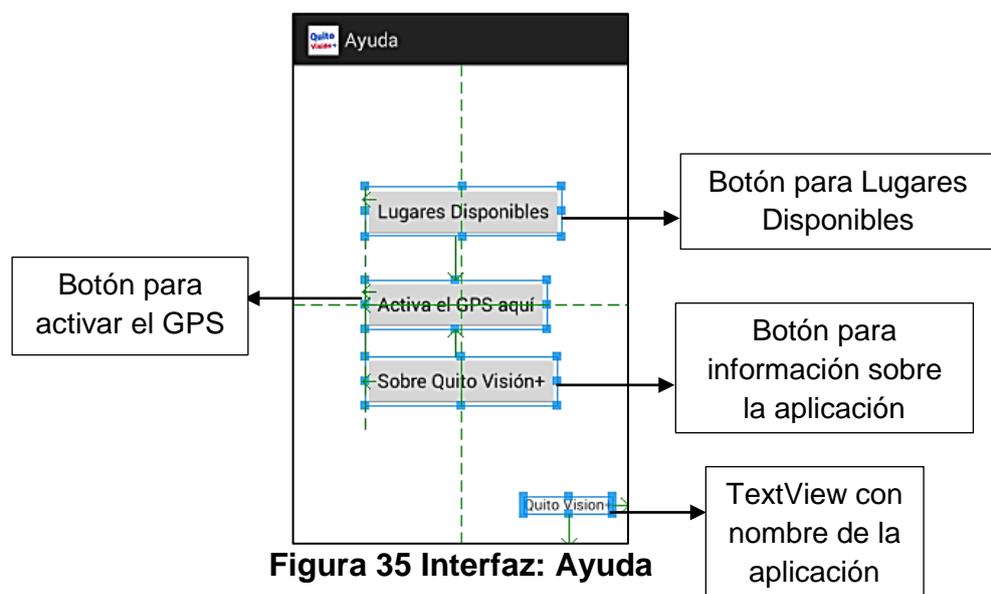
A continuación el diseño de cada una de las interfaces mencionadas:

- **Menú Inicio:** Esta interfaz dispone de un layout de color de fondo blanco, con dos botones: el primero para la opción de realidad aumentada, incluye texto y se encuentra centrado en la parte inferior de la pantalla y el segundo es un small button, para la opción de ayuda, incluye una imagen con el signo de interrogación, no incluye texto y es de fondo blanco, a todo esto se agrega a la interfaz el elemento ImageView, que se encuentra en la parte superior de la pantalla, en donde, se puede ver el logo de la aplicación.



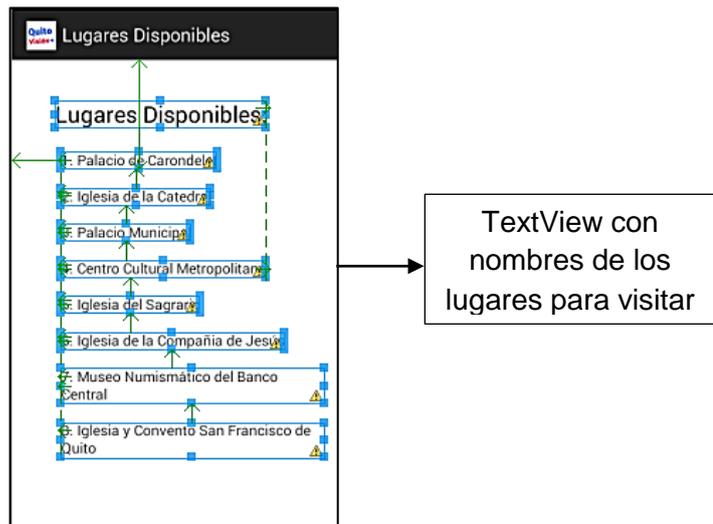
**Figura 34 Interfaz: Menú Inicio**

- **Ayuda:** Es una interfaz muy básica y fácil de manejar, tiene un layout de color blanco, un TextView en la esquina inferior derecha con el nombre de la aplicación y tres botones para dirigir a las opciones de ayuda, cada botón cuenta con texto y se encuentran en alineados en la mitad de la pantalla. A continuación, las siguientes figuras muestran el diseño de la interfaz principal de ayuda y de sus respectivas opciones:



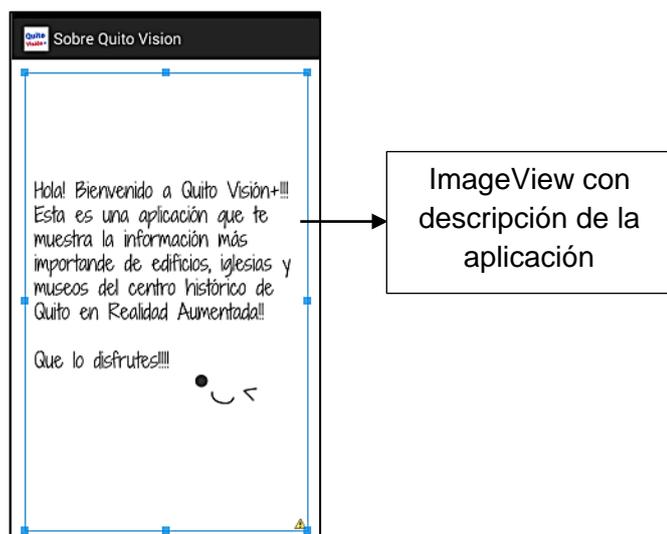
**Figura 35 Interfaz: Ayuda**

La interfaz de la opción “Lugares Disponibles” consta de TextView con los nombres de las edificaciones disponibles a visitar y un fondo de layout de color blanco:



**Figura 36 Interfaz: Lugares Disponibles**

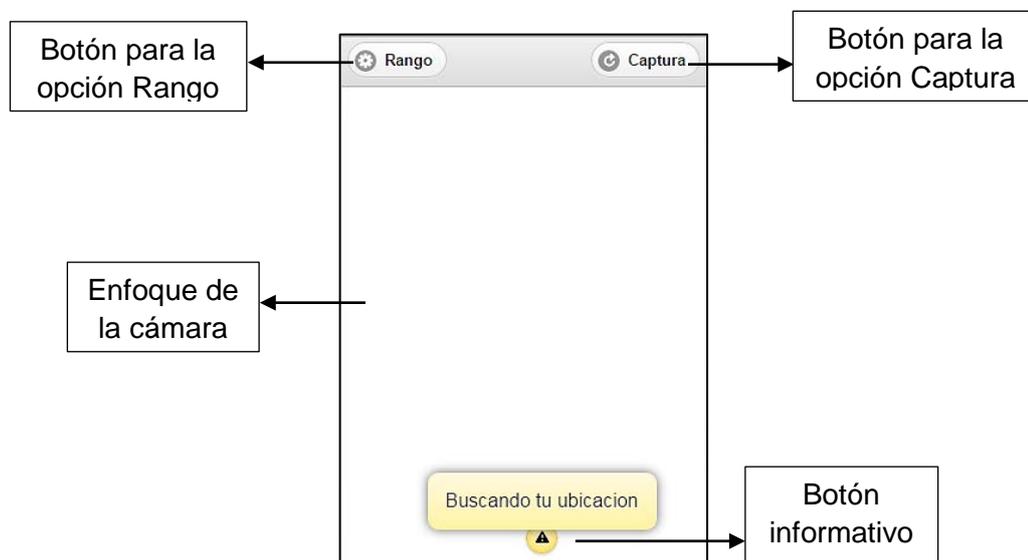
La interfaz de la opción Sobre “Quito Visión+” cuenta con un ImageView en donde se muestra las palabras de la creadora de la aplicación:



**Figura 37 Interfaz: Sobre Quito Visión**

La interfaz de la opción Activar GPS no fue diseñada, debido a que la aplicación redirige a los ajustes propios del teléfono.

- **Vista en Realidad Aumentada:** Esta interfaz es la que muestra la experiencia en realidad aumentada y es diferente a las anteriores mostradas, ya que fue diseñada en lenguaje HTML y no en lenguaje nativo Android. Sus elementos son los siguientes: un botón informativo que se encuentra centrado en la parte inferior de la pantalla y en la parte superior se encuentra una barra con las siguientes opciones: Captura y Rango que son representados con dos botones.



**Figura 38 Interfaz: Vista en Realidad Aumentada**

En la figura 38 se puede observar que el fondo de la interfaz es de color blanco, pero cuando la aplicación esté corriendo se mostrará el enfoque de la cámara.

### 3.5.3 MENÚ INICIO

Después de la pantalla de bienvenida, la aplicación dirige al usuario a un menú inicial en donde encuentra dos opciones, la primera que le permite acceder a la vista en realidad aumentada y la segunda que permite acceder a un menú de ayuda. La Figura 39 muestra la interfaz presentada al usuario:



**Figura 39 Pantalla de Menú Inicio**

- **Vista en Realidad Aumentada:** Es la opción más importante y fundamental de la aplicación, para acceder a la función se implementó un botón nativo de Android, que al momento de presionarlo abre el ArchitectView de Wikitude. Lo que el usuario puede ver es la imagen generada por la cámara y dependiendo de su ubicación y enfoque podrá observar los puntos de interés. Adicionalmente, el usuario tiene la posibilidad de ver los detalles de cada POI y escoger un rango de visibilidad.
- **Ayuda:** A esta opción se puede identificar en la pantalla con una imagen de un signo de interrogación, que permite el acceso a un submenú en donde el usuario puede activar el GPS, saber que lugares puede visitar y acceder a una breve descripción de la aplicación. Fue implementado con un botón de imagen.

### 3.5.4 VISTA EN REALIDAD AUMENTADA

La interfaz mostrada al usuario fue implementada con la ayuda del SDK Wikitude, además se agregó un panel en donde se podrá observar los

detalles de cada punto de interés y un panel en donde se puede escoger el rango de visibilidad.

El SDK Wikitude provee de licencias gratuitas para desarrolladores, es por eso que cuando se va a iniciar la vista en realidad aumentada se observa la pantalla del launcher de Wikitude, así como en la siguiente figura:



**Figura 40 Imagen Launcher de Wikitude**

La Figura 41 muestra como es la interfaz en realidad aumentada como tal y las opciones para interactuar con la aplicación:



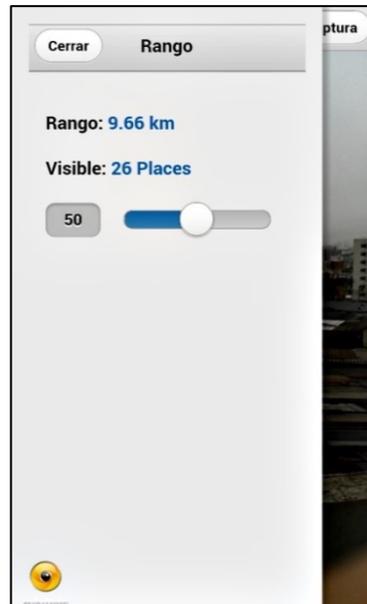
**Figura 41 Interfaz Realidad Aumentada**

- **Detalles:** La opción detalles muestra un panel desarrollado en lenguaje HTML y que llama a un método de una clase implementada en Java Script. Este panel muestra la información de POIs que se encuentra almacenada en un array de JSON luego de la consulta realizada al servicio web. Posee un scrollbar para cuando la información sea mayor al panel. La siguiente figura muestra el panel:



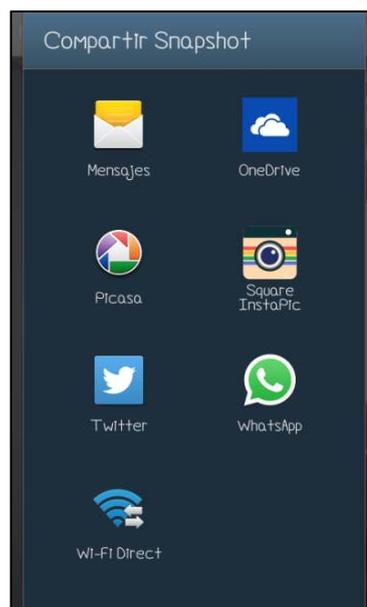
**Figura 42 Panel Detalles**

- **Rango:** Es una opción en donde el usuario puede escoger el número de POIs que desea ver en un rango, tras escoger la opción Rango en la pantalla se abre un panel implementado con lenguaje HTML y que llama a un método de una clase implementada en Java Script. El panel cuenta con un slider que permite al usuario escoger el rango deseado, dicho rango se encuentre entre 0Km a 50Km. A continuación, la figura 43 muestra el panel:



**Figura 43 Panel Rango**

- **Captura:** Es una opción que permite al usuario capturar la pantalla y compartirla en redes sociales, correo u otro dispositivo. La captura esta implementada en Java Script y en lenguaje nativo. El botón captura se encuentra implementado en HTML.



**Figura 44 Compartir Captura**

- **Status:** Es un pequeño icono centrado la parte inferior de la pantalla que notifica al usuario cuantos POIs se han cargado o que se encuentra buscando su ubicación. Se encuentra implementado en HTML. La figura 45 muestra cuando la aplicación muestra el número de lugares cargados:

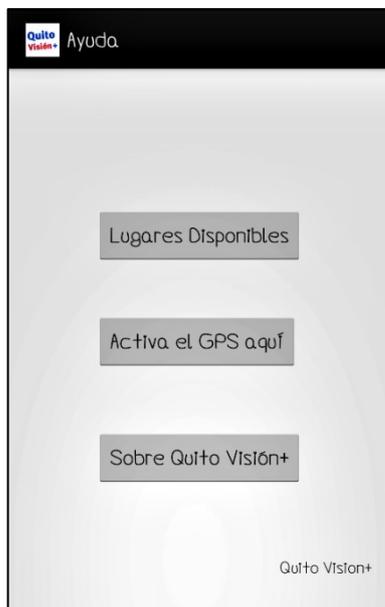


**Figura 45 Barra de Status**

### 3.5.5 MENÚ AYUDA

Función en donde el usuario dispone de tres opciones que son: lugares disponibles, activar GPS y una pequeña descripción de la aplicación. Implementado en lenguaje nativo Android, es decir, Java.

El submenú consta de tres botones para las tres opciones, tal como lo muestra la siguiente figura:



**Figura 46 Menú Ayuda**

- **Lugares Disponibles:** Le permite conocer al usuario los lugares que puede visitar con la aplicación. De acuerdo a las coordenadas de latitud y longitud los POIs de irán apareciendo.

A continuación, el siguiente cuadro muestra la lista de los sitios disponibles para visitar, junto con la información que se podrá observar al momento de seleccionar un punto de interés.

Información como: horarios de visita, condiciones de ingreso y contactos. El cuadro se encuentra en orientación horizontal para mejor apreciación de los datos.

**Cuadro 12**  
**Lugares Disponibles para el Usuario**

<b>INFORMACIÓN DEL SISTEMA</b>			
<b>LUGARES DISPONIBLES</b>	<b>HORARIOS DE VISITA</b>	<b>CONDICIONES DE INGRESO</b>	<b>CONTACTOS</b>
<b>Palacio de Carondelet</b>	Lunes de 15h00 a 19h00. Martes a Viernes de 9h00 a 19h00. Sábados de 9h00 a 22h00. Domingo de 09h00 a 16:00.	-Entrada gratuita. -Inscribirse con Cedula de Identidad o pasaporte. -No entrar con gorras, sombreros y/o gafas. -No tomar fotos con flash y filmar dentro del palacio. -Visita guiada.	Dirección: Gabriel García Moreno N10-43. Web oficial de la Presidencia: <a href="http://www.presidencia.gob.ec/museo/">http://www.presidencia.gob.ec/museo/</a>
<b>Iglesia de la Catedral</b>	Lunes a Sábado de 9h00 a 17h00. Abierto feriados.	-Entradas tiene un costo de: 2 dólares turistas nacionales y 4 dólares internacionales. -Entrada a las cúpulas con reserva previa y con un costo de 60 dólares por grupo de 10 personas.	Teléfonos: 2570-371 Fax: 2512-331 Direc: Venezuela N3-117 y Espejo.
<b>Palacio Municipal</b>	Horarios de Oficina	Ninguna condición	Web: <a href="http://www.quito.gob.ec">http://www.quito.gob.ec</a> Direc: Venezuela entre Chile y Espejo Telf.: 2395 23 00 / 1800 45 67 89
<b>Centro Cultural Metropolitano</b>	Martes a Sábado 09h00 a 17h30. Domingo 10h00 a 13h30.	Entrada libre.	Direc: García Moreno y Espejo esquina. Teléf.: (593 2) 395 2300. Reservaciones: 3952300 Ext. 15507

Continua →

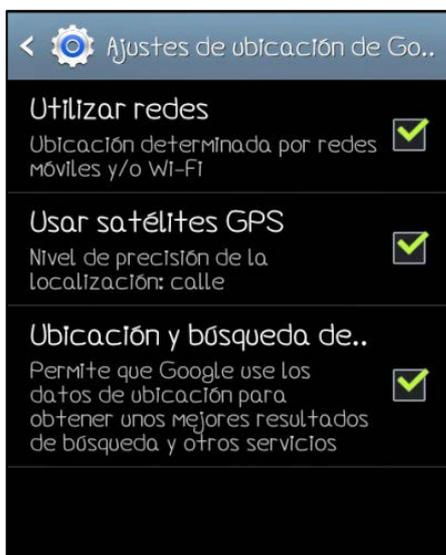
<b>Iglesia del Sagrario</b>	Misas Dominicales.	-No ingreso de turistas durante misa. -No tomar fotografías.	Direc: García Moreno y Espejo
<b>Iglesia de la Compañía de Jesús</b>	Lunes a Jueves de 9h30 a 18h30. Viernes de 9h30 a 17h30. Sábados 9h30 a 16h00. Feriados: 9h30 a 16h00.	No fotografías ni filmación.  Valor de la entrada para ecuatorianos: 2.00 dólares adultos, 0.50 centavos estudiantes primarios y 1.00 dólar estudiantes universitarios.	Direc: García Moreno y Sucre. Telf.22584175
<b>Museo Numismático del Banco Central</b>	Martes a viernes desde las 9:00 a 17:00. Sábado, domingo y feriados desde las 10:00 a las 16:00.	El ingreso para el público nacional es gratuito. Extranjeros adultos: 1.00 Dólar, estudiantes y Tercera edad: 0.50 ctvs. Entrada para turistas nacionales: 2.00 dólares adultos, 1.00 dólar estudiantes y 0.50 ctvs. Menores de edad.	Direc: García Moreno y Sucre. Tell: 22589-284 / 572784 /572776 Web: <a href="http://www.bce.fin.ec/">http://www.bce.fin.ec/</a>
<b>Iglesia y Convento San Francisco de Quito</b>	Lunes a Sábado de 9h00 a 17h00. Domingo 9h00 a 12h30.	Entrada para turistas extranjeros: 2.00 adultos y 1.00 dólar menores de edad.  Visitas guiadas.  Prohibido el ingreso de mascotas.	Direc: Cuenca 477 y Sucre. Telf.: 02 2952-911 / 0995621321. Web: <a href="http://www.museopedrogocial.org/">http://www.museopedrogocial.org/</a>

Además, la figura 47 muestra la pantalla que se presenta al usuario cuando escoge la opción 1:



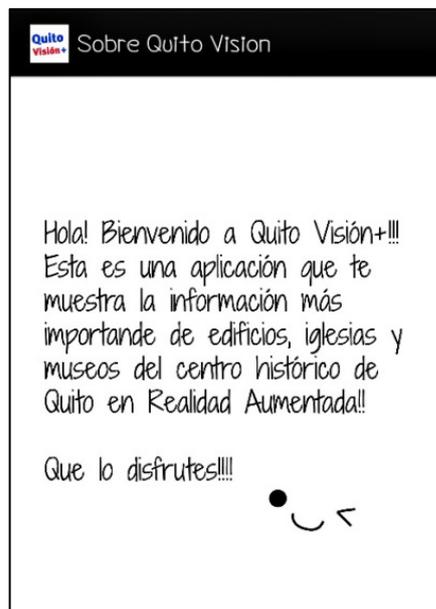
**Figura 47 Lugares Disponibles**

- **Activar GPS:** Dirige al usuario, mediante un intent, a los ajustes propios del dispositivo móvil para activar su GPS en caso de no tenerlo activado. A continuación, la figura de los ajustes:



**Figura 48 Pantalla de Ajustes GPS**

- **Sobre Quito Visión+:** Muestra una pequeña descripción de la aplicación escrita por la desarrolladora. Desarrollado en lenguaje nativo y consta de una imagen. El mensaje se muestra en la siguiente figura:



**Figura 49 Mensaje de la Desarrolladora**

### **3.5.6 SERVICIO WEB**

Para poder obtener la información de los puntos de interés que se encuentran almacenados en la base de datos, se realizó un servicio web que sirve de puente la aplicación y MySQL.

El servicio web fue implementado en lenguaje de programación PHP y realiza la conexión y consulta a la base de datos, con la respuesta obtenida transforma los datos a lenguaje de intercambio JSON y esto envía hacia la aplicación.

En la figura 50 se da un ejemplo de la respuesta por parte del servicio web, en donde se puede observar el JSONArray con la información:

```

{"id":"8","nombre":"Palacio de Carondelet","descripcion":"El palacio de Gobierno fue de propiedad del secretario de la Real Audiencia de Quito en 1611 y fue readecuado por el Presidente de la Audiencia, Hector de Carondelet, entre los siglos XVII y XIX. \r\n\r\nEste palacio es la residencia y sede oficial del presidente y vice-presidente del Ecuador. \r\n\r\nEs el presidente Rafael Correa quien permite la entrada a los ciudadanos ya que considera que el Palacio de Carondelet es un Patrimonio Cultural de los ecuatorianos.\r\n\r\nVisite la Web oficial de la Presidencia: http://www.presidencia.gob.ec/museo/", "longitud": "-78.512600", "latitud": "-0.220022", "altitud": "-32768"}, {"id":"9","nombre":"Horarios de Visita","descripcion":"Lunes de 15h00 a 19h00.\r\nMartes a Viernes de 9h00 a 19h00.\r\nSabados de 9h00 a 22h00.\r\nDomingo de 09h00 a 16:00.", "longitud": "-78.512545", "latitud": "-0.219942", "altitud": "-32768"}, {"id":"11","nombre":"Contactos","descripcion":"Direc: Gabriel Garcia Moreno N10-43.\r\nWeb oficial de la Presidencia: http://www.presidencia.gob.ec/museo/", "longitud": "-78.512294", "latitud": "-0.2197734", "altitud": "-32768"}, {"id":"10","nombre":"Condiciones de Ingreso","descripcion":"-Entrada gratuita.\r\n-Inscribirse con Cedula de Identidad o Pasaporte.\r\n-No entrar con gorras, sombreros y/o gafas.\r\n-No tomar fotos con flash y filmar dentro del palacio.\r\n-Visita

```

**Figura 50 Respuesta del Servicio Web**

Adicionalmente, se publicó el servicio web en un web hosting para alojarlo en Internet y que así sea accesible desde todo lugar. El web hosting es gratuito y la dirección en la que se publicó es la siguiente: <http://qvision.byethost11.com/QuitoVisionWS.php>.

Para la publicación, se necesitó la ayuda del web hosting: [www.bytehost.com](http://www.bytehost.com), que después de crear una cuenta entrega beneficios como base de datos, PHPMyAdmin y administración de documentos mediante FileZilla, alojamiento gratuito de hasta 5 bases de datos MySQL, ancho de banda 50000MB y espacio de disco 1000MB .

En FileZilla se cargan los archivos PHP y son transferidos hacia el web hosting mediante protocolo FTP, de esa manera y a través de la URL, previamente entregada, se puede acceder al servicio web en internet.

### 3.5.7 BASE DE DATOS

Se utilizó MySQL para la creación de la base de datos, en donde se encuentra almacenada información como nombre, descripción, latitud, longitud y altitud.

La siguiente figura muestra la base de datos con la información agregada:

id	nombre	descripcion	longitud	latitud	altitud
1	Calle 1	Calle 1	-78.474213	-0.160844	-32768
2	Calle 2	Calle 2	-78.474238	-0.160983	-32768
3	ESPE	ESPE	-78.443692	-0.314536	-32768
4	Labs Elec	Labs Elec	-78.4439495	-0.3141659	-32768
5	Biblioteca	Biblioteca	-78.4439407	-0.3140593	-32768
6	Garita	Garita	-78.4433184	-0.3159268	-32768
7	Edificio	Edificio	-78.4433184	-0.3159268	-32768
8	Palacio de Carondelet	El palacio de Gobierno fue de propiedad del secret...	-78.512600	-0.220022	-32768
9	Horarios de Visita	Lunes de 15h00 a 19h00. Martes a Viernes de 9h00 ...	-78.512545	-0.219942	-32768
11	Contactos	Direc: Gabriel Garcia Moreno N10-43. Web oficial ...	-78.512294	-0.2197734	-32768

**Figura 51 Información de Base de Datos en PHPMyAdmin**

### 3.6 DURACIÓN DEL PROCESO DE DISEÑO E IMPLEMENTACIÓN

La duración del diseño, implementación e investigación de los temas tuvo una duración de 4 meses, en el siguiente cuadro se describe cada etapa, aprendizaje y tiempo.

#### Cuadro 13

#### Duración del Proceso de Diseño e Implementación

DURACIÓN DEL PROCESO DE DISEÑO E IMPLEMENTACIÓN		
ETAPA	DESCRIPCIÓN	TIEMPO
<b>Investigación de Temas</b>	Conceptos de Android y Realidad Aumentada. Conceptos de Servicios Web y Base De Datos.	4 Semanas
<b>Familiarización con Android</b>	Utilización de Actividades, Android Manifest, Layouts, etc. Creación de una aplicación básica. Creación de una aplicación de apertura y	1 Semana
<b>Familiarización con Cámara y GPS</b>	permisos de la Cámara. Creación de una aplicación de manejo de GPS. Creación de una aplicación de manejo de sensores de orientación y acelerómetro.	1 Semana.
<b>Primeros Pasos con RA</b>	Creación de aplicaciones que muestren los valores de GPS, brújula y sensores en realidad aumentada.	1 Semana

Continúa →

<b>Diseño de Quito Visión+</b>	Especificación de requerimientos. Selección de SDK para realidad aumentada.	2 Días.
<b>Utilización básica de Wikitude</b>	Revisión de Documentación. Revisión de Ejemplos. Solicitud de licencia para estudiantes. Manejo de Layout. Manejo de ArchitectView. Creación Interfaz Inicial para el usuario. Manejo de Cámara.	2 Semanas
<b>Creación de Quito Visión+ (1)</b>	Manejo de LocationProvider. Validación de conexión a internet mediante Wi-Fi o Datos Móviles. Validación de requerimientos mínimos en versión Android y OPEN  GL.	2 Semanas
<b>Creación Servicio Web y Base de Datos</b>	Creación base de datos. Almacenamiento de Datos. Implementación de Servicio Web PHP. Respuesta en formato JSON. Publicación en internet del servicio web.	2 Semanas
<b>Creación de Quito Visión+ (2)</b>	Conexión con servicio web mediante AsyncTask. Creación de POIs. Creación de marcadores.	2 Semanas.
<b>Creación de Quito Visión+ (Final)</b>	Adición de detalles visuales a la vista de realidad aumentada. Adición panel de detalles de POI. Adición Opción de Captura. Adición panel de Rango. Adición de pantalla Splash, Menú Inicio y Menú Ayuda.	1 Semana.

## CAPÍTULO 4

### FUNCIONAMIENTO Y RESULTADOS DE LA APLICACIÓN

#### 4.1 FUNCIONAMIENTO DE LA APLICACIÓN

Las pruebas realizadas para comprobar el correcto funcionamiento de la aplicación se basaron en las 6 opciones de uso descritas en el Capítulo 3 sección 3.3.

Para cada función se realizó un cuadro con la descripción de la prueba, respuesta del sistema y estado de la respuesta. Las pruebas se realizaron en el centro histórico de Quito con diferentes dispositivos. A continuación, se presenta los cuadros de pruebas a las funciones de la aplicación:

- **Menú Inicio**

Esta función se probó mediante la realización de un clic en el icono launcher de la aplicación “Quito Visión+” por parte del usuario.

La aplicación por su parte responde a esta acción con una pantalla de bienvenida, conocida como Splash Screen, que dura 3 segundos seguida de la pantalla del Menú Inicio que muestra dos opciones: Vista en Realidad Aumentada y el signo de interrogación que simboliza opción de Ayuda.

Para dar como positivo y sin errores este caso de uso, la aplicación debe mostrar al usuario una pantalla que contenga: el logo “Quito Visión+” en dimensiones adecuadas para cada dispositivo, el botón de Vista en Realidad Aumentada debe encontrarse centrado en la parte inferior de la pantalla y con el tipo de letra del dispositivo y la opción ayuda debe mostrar su imagen con el signo “?” en la esquina inferior derecha.

## Cuadro 14

### Prueba Caso de Uso: Menú Inicio

CASO DE USO: MENÚ INICIO			
ORD.	CRONOLOGÍA	RESPUESTA DE LA APLICACIÓN	ESTADO
1	Clic en el Icono de Launcher.	Cargando aplicación Quito Visión+.	OK
2	Pantalla de Bienvenida	Muestra pantalla Splash Screen con Logo de la aplicación.	OK
3	Opciones de Menú	Muestra la opción de vista en realidad aumentada y la opción ayuda.	OK
4	Correctas dimensiones de elementos visuales	Logo “Quito Visión+” Botón Vista en Realidad Aumentada Opción Ayuda	OK

- **Vista en Realidad Aumentada**

Se ha mencionado con anterioridad que este caso de uso es la esencia de la aplicación por ello su correcto desarrollo es primordial.

La prueba para esta función, inicia con un requisito previo que es haber hecho clic en el botón “Vista en Realidad Aumentada” del menú inicial. Seguido de este paso, la aplicación muestra el logo de Wikitude, el cual aparece obligatoriamente por motivos de licencia, con un mensaje toast que informa al usuario que se encuentra buscando su ubicación.

Internamente la aplicación, valida el encendido del GPS y datos móviles o Wi-Fi, si uno de estos se encuentra inactivo el usuario recibirá una notificación por medio de un mensaje toast, que le solicita la activación de alguno de estos. Si la validación es correcta, es decir, GPS y datos móviles o Wi-Fi se encuentran activos, pasa a la siguiente pantalla que muestra el enfoque de la cámara con dos opciones: la primera, un icono informativo centrado en la parte inferior que al ser presionado indica cuantos lugares han sido cargados y la segunda, una barra en la parte superior, en donde se encuentran las opciones de Captura y Rango.

De manera importante, la aplicación debe mostrar los puntos de interés con sus marcadores, de color azul y letras blancas, en las coordenadas correctas.

Para que la prueba sea válida, la aplicación debe mostrar el enfoque de la cámara con la barra de opciones y el icono informativo, los puntos de interés deben estar en la posición correcta y las notificaciones deben aparecer cuando sean requeridas y con la opción de regreso al menú inicial. A todo esto cada elemento debe encontrarse en correctas dimensiones y acordes a la posición del teléfono, ya sea que éste se encuentre verticalmente u horizontalmente.

### **Cuadro 15**

#### **Prueba Caso de Uso: Vista en Realidad Aumentada**

<b>CASO DE USO: VISTA EN REALIDAD AUMENTADA</b>			
<b>ORD.</b>	<b>CRONOLOGÍA</b>	<b>RESPUESTA DE LA APLICACIÓN</b>	<b>ESTADO</b>
1	Clic en la opción Vista en Realidad Aumentada	Cargando vista en Realidad Aumentada y muestra del Logo de Wikitude.	OK
2	Mensaje de Ubicación	Mensaje tipo toast informando al usuario de que se encuentra en búsqueda de la ubicación	OK
3	Notificación de Activación de Datos Móviles, GPS y/o Wi-Fi	Presenta mensaje de notificación, en caso de no tener activos los datos móviles o Wi-Fi y GPS.	OK
4	Activación de Cámara	Muestra en pantalla el enfoque de la cámara principal del dispositivo.	OK
4	Muestra de POIs	Presenta en pantalla los puntos de interés posicionados de acuerdo a su longitud y latitud.	OK

- **Detalles de POIs**

Mostrar la información que contiene cada punto de interés también es importante en esta aplicación.

Para probar su correcto funcionamiento lo primero que debe realizar el usuario es hacer clic en uno de los POIs, al momento en el que se presiona el marcador, éste cambia de color, de azul a celeste y así se muestra que fue seleccionado. Cabe notificar que solo se puede presionar un marcador y no dos al mismo tiempo.

Seguido de esta acción la aplicación despliega un panel con la información en donde se encuentra: el título del POI, la información obtenida de la base de datos y la distancia a la que se encuentra dicho punto. Adicionalmente, el panel consta con un botón con la opción cerrar.

La prueba está aceptada cuando el panel se despliega sin retraso, de manera clara, modificable de acuerdo a la posición del teléfono y que se cierre cuando el usuario haya presionado el botón.

## Cuadro 16

### Prueba Caso de Uso: Detalles de POIs

CASO DE USO: DETALLES DE POIs			
CONDICIÓN PREVIA		<ul style="list-style-type: none"> <li>- Clic en la opción Vista en Realidad Aumentada.</li> <li>- Activado GPS, Datos móviles y/o Wi-Fi.</li> </ul>	
ORD.	CRONOLOGÍA	RESPUESTA DE LA APLICACIÓN	ESTADO
1	Clic en el marker de un POI	Color del marker cambia de color azul marino a celeste.	OK
2	Cambio de color de marker de POI	Muestra panel con la información de un solo POI específico.	OK
3	Clic en opción Cerrar	Cierra el panel de detalles.	OK

- **Captura de Pantalla**

La prueba realizada en esta opción inicia con el cumplimiento de un requisito previo que es haber activado la vista en realidad aumentada y activado el GPS y datos móviles o Wi-Fi. Ya que de lo contrario la opción no estará disponible.

Cuando el usuario hace clic en la opción “Captura”, que se encuentra en la barra superior de la pantalla, la aplicación congela por un mínimo de tiempo la imagen en demostración de que está realizando la captura de pantalla.

Esta imagen se guarda en la memoria del teléfono con el nombre:

*screenCapture\_(fecha y hora en milisegundos)*

De inmediato, la aplicación habilita la opción para compartir dicha captura en las redes sociales, correo electrónico o con otros dispositivos.

Se aprueba la función cuando se puede acceder a la imagen guardada en la memoria y compartirla luego de haber presionado el botón “Captura”.

### Cuadro 17

#### Prueba Caso de Uso: Captura de Pantalla.

CASO DE USO: CAPTURA DE PANTALLA			
CONDICIÓN PREVIA		- Clic en la opción Vista en Realidad Aumentada. - Activado GPS, Datos móviles y/o Wi-Fi.	
ORD.	CRONOLOGÍA	RESPUESTA DE LA APLICACIÓN	ESTADO
1	Clic en la opción CAPTURA	Realiza captura de pantalla de acuerdo al enfoque de la cámara.	OK
2	Guarda la imagen	La imagen es automáticamente guardada en la memoria del dispositivo.	OK
3	Compartir captura	Muestra opciones de aplicaciones para compartir la imagen. En caso de no querer compartir regresa a la vista de realidad aumentada.	OK

- **Rango**

Esta función de prioridad no alta, se probó haciendo clic en la opción “Rango” de la barra superior de la pantalla. Como condición previa se debe activar la vista en realidad aumentada.

“Quito Visión+” como respuesta despliega un panel que contiene un slider para poder escoger el rango de visibilidad, la distancia, cuantos lugares están disponibles y una opción para poder cerrar el panel.

Se da por aceptada la prueba cuando el usuario establece un rango moviendo el slider y la aplicación muestra en pantalla el número de POIs escogido.

### Cuadro 18

#### Prueba Caso de Uso: Rango

CASO DE USO: RANGO			
CONDICIÓN PREVIA		<ul style="list-style-type: none"> <li>- Clic en la opción Vista en Realidad Aumentada.</li> <li>- Activado GPS, Datos móviles y/o Wi-Fi.</li> </ul>	
ORD.	CRONOLOGÍA	RESPUESTA DE LA APLICACIÓN	ESTADO
1	Clic en la opción RANGO	Muestra panel con slider para escoger el rango de visibilidad.	OK
2	Escoge el Rango	Muestra número de POIs de acuerdo al rango escogido.	OK
3	Cerrar Panel	Cierra panel rango.	OK

- **Ayuda**

La prueba realizada a esta función inicia cuando se da clic en el signo de interrogación “?” que se encuentra en el menú inicial.

Seguido a este paso, la aplicación muestra una pantalla con tres botones que son las opciones de ayuda: Lugares Disponibles, Activar GPS y Sobre Quito Visión+.

Esta aceptada la evaluación cuando el usuario al dar clic en la primera opción es dirigido a pantalla con una lista de lugares disponibles, al dar clic en la segunda opción accede a los ajustes del teléfono y al dar clic en la tercera opción recibe una descripción corta de la aplicación. Adicionalmente,

la pantalla puede modificarse de acuerdo a la posición del teléfono y con opción a regresar al menú inicial.

### **Cuadro 19**

#### **Prueba Caso de Uso: Ayuda**

<b>CASO DE USO: AYUDA</b>			
<b>ORD.</b>	<b>CRONOLOGÍA</b>	<b>RESPUESTA DE LA APLICACIÓN</b>	<b>ESTADO</b>
1	Clic en la opción AYUDA	Muestra pantalla con tres opciones.	OK
2	Clic opción Lugares Disponibles	Muestra lista con los lugares para conocer.	OK
3	Clic opción Activar GPS	Redirige a los ajustes propios del dispositivo móvil.	OK
4	Clic opción Sobre Quito Visión+	Da una breve descripción del programa	OK

- **Servicio Web**

Como administradora de la aplicación también es importante validar el trabajo del servicio web que es el que realiza la consulta de información y responde en formato JSON.

Para la realización de la prueba ya debe estar la base de datos establecida y con información almacenada, el servicio web se conecta a la base de datos, escoge la tabla y consulta sus registros.

Para responder a la “Quito Visión+” el servicio web almacena los datos consultados en un JSON Array.

Finalmente, este servicio web se publica en un web hosting y puede ser accedido desde el dispositivo móvil con red Wi-Fi o datos móviles.

**Cuadro 20****Prueba a Servicio Web**

<b>SERVICIO WEB</b>			
<b>ORD.</b>	<b>CRONOLOGÍA</b>	<b>DESCRIPCIÓN</b>	<b>ESTADO</b>
1	Conexión a base de datos	Conecta con MySQL	OK
2	Consulta	Busca los datos en los registros de una tabla	OK
3	Cambio a formato JSON	Almacena información en un JSON Array.	OK
4	Publicación	En un web hosting gratuito con la transferencia FTP.	OK
5	Acceso	Acceso a los datos de consulta con Wi-Fi y datos móviles.	OK

**4.2 DISPOSITIVOS Y VERSIONES**

Las pruebas se realizaron con tres dispositivos que tienen como principales diferencias su versión Android, dimensiones de pantalla, resolución de cámara principal y procesador. Los dispositivos usados son los siguientes: Samsung Galaxy S3 Mini, Samsung Galaxy S5 y Sony Xperia Play. A continuación, se presenta un cuadro con las especificaciones de dichas características necesarias para las pruebas:

**Cuadro 21****Características de Dispositivos de Prueba**

<b>ESPECIFICACIÓN DE CARACTERÍSTICAS</b>			
<b>DISPOSITIVO</b>	<b>SAMSUNG GALAXY S3 MINI</b>	<b>SAMSUNG GALAXY S5</b>	<b>SONY XPERIA PLAY</b>
<b>VERSIÓN ANDROID</b>	Android V4.1.2 Jelly Bean	Android V4.4.2 KitKat	Android V2.3.3 Gingerbread
<b>RESOLUCIÓN CÁMARA PRINCIPAL</b>	5MP	16MP	5MP
<b>DIMENSIÓN DE PANTALLA</b>	4"	5"	4"
<b>PROCESADOR RAM</b>	1GHz	2.5GHz	1GHz

Las razones adicionales para trabajar con los teléfonos móviles descritos en el Cuadro 21 fueron las siguientes:

- Sony Xperia Play: tiene la versión mínima para utilizar el SDK Wiktitude es la 2.3.3 Gingerbread.
- Samsung Galaxy S3 Mini: es un dispositivo de mucha acogida en el mercado y se asemeja a las características del Sony Xperia Play.
- Samsung Galaxy S5: es uno de los teléfonos actualmente nuevos y su versión Android es 4.4.2 KitKat.

A continuación un cuadro con los resultados de cada función en los diferentes dispositivos móviles utilizados:

## Cuadro 22

### Resultados de las funciones en los Dispositivos

<b>RESULTADO LAS FUNCIONES DE ACUERDO A LOS DISPOSITIVOS</b>			
	<b>SAMSUNG GALAXY S3 MINI</b>	<b>SAMSUNG GALAXY S5</b>	<b>SONY XPERIA PLAY</b>
<b>Menú inicio</b>	✓	✓	✓
<b>Vista en realidad aumentada</b>	✓	✓	✓
<b>Detalles de POI</b>	✓	✓	✗
<b>Captura de pantalla</b>	✓	✓	✗
<b>Rango de interés</b>	✓	✓	✗
<b>3 opciones de ayuda</b>	✓	✓	✓

Como se observa en el Cuadro 22, el dispositivo Sony Xperia Play no permite mostrar los detalles de cada POI, esto se debe a que “Quito Visión+” realiza peticiones de información a un servicio web en internet con la ayuda de una clase llamada AsyncTask.

Android ofrece la clase AsyncTask para manejo de conexiones a internet mediante la creación de un hilo secundario y de esa manera evitar la

sobrecarga en el procesamiento del hilo principal (UI Thread) de la aplicación con el manejo de handlers y threads. Es obligatorio usar la clase AsyncTask a partir de la versión 3.0 de Android.

Por ese motivo, teléfonos con versiones inferiores a la 3.0 no podrán hacer uso de la aplicación “Quito Visión+”. Ya que los dispositivos internamente no sabrán cómo manejar las conexiones a internet con AsyncTask. La Figura 52 muestra el mensaje de error que el usuario recibe del teléfono al momento de abrir la vista en realidad aumentada:



**Figura 52 Mensaje de Error en Xperia Play**

En los dispositivos Samsung S3 mini y S5 no existió problemas y la aplicación corre sin inconvenientes, por ello en las siguientes figuras se muestran imágenes de los casos de uso en los teléfonos mencionados. En cada dispositivo se realizaron las pruebas como se encuentra descrito en la sección 4.1 del presente capítulo:



Figura 53 Menú Inicio en Samsung S3 Mini

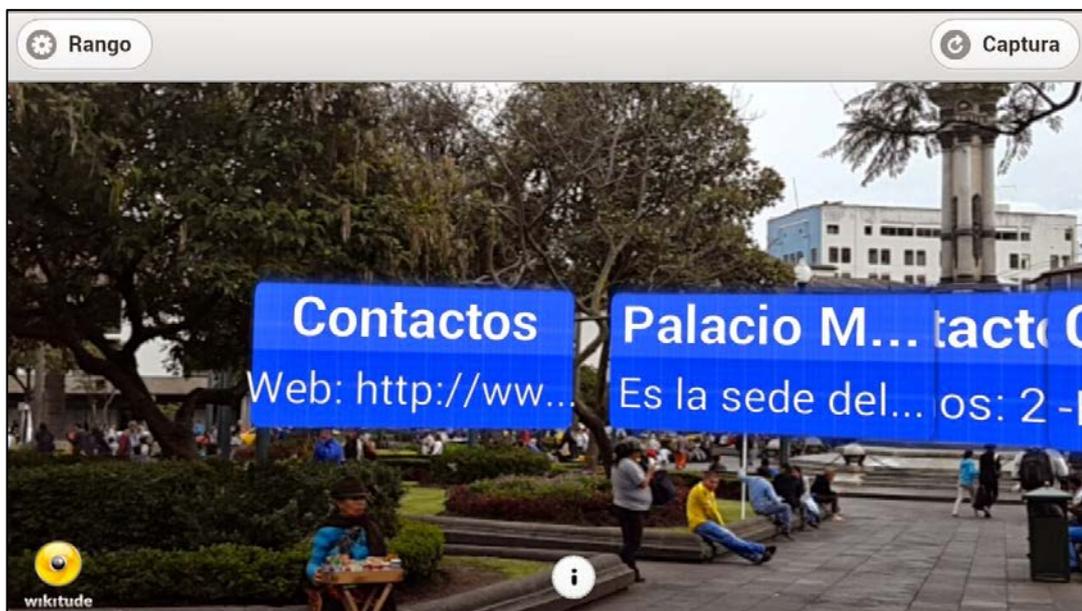


Figura 54 Vista en Realidad Aumentada en Samsung Galaxy S5



Figura 55 Detalles de POI en Samsung S3 Mini

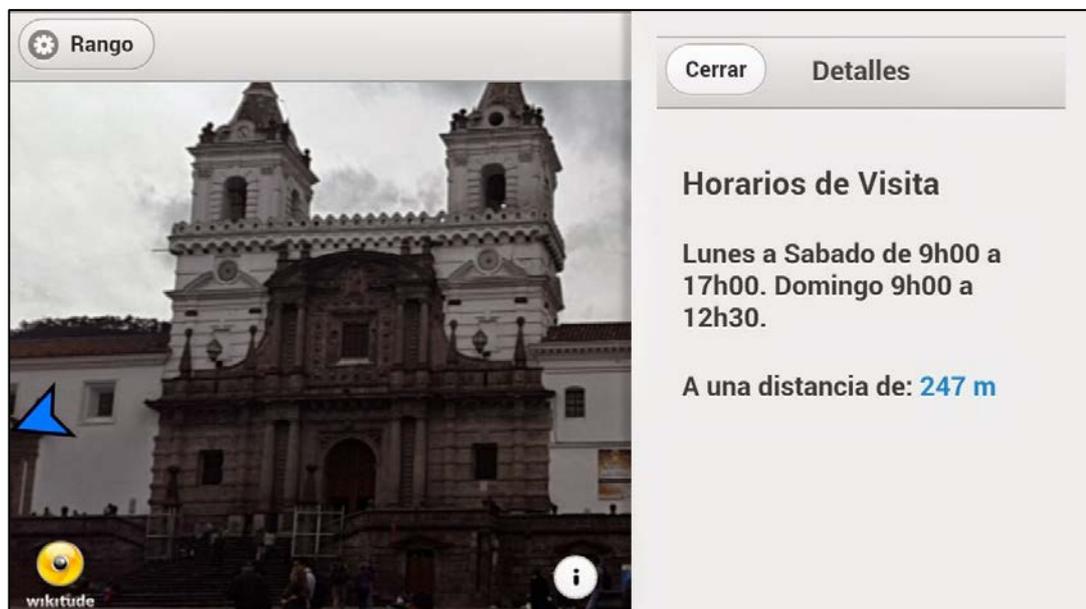


Figura 56 Detalles de POI Samsung Galaxy S5



Figura 57 Captura de Pantalla en Samsung S3 Mini

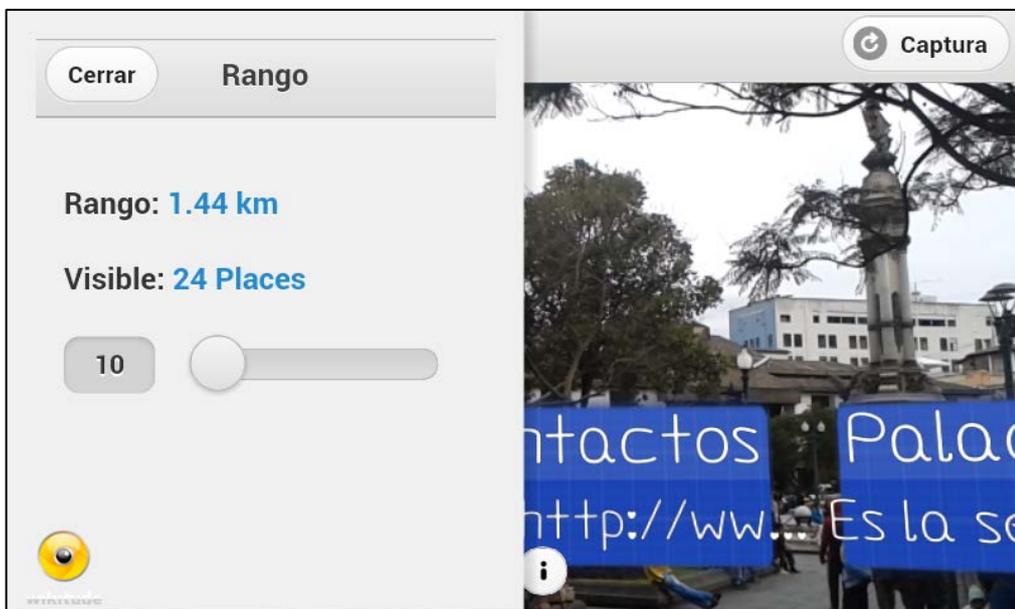
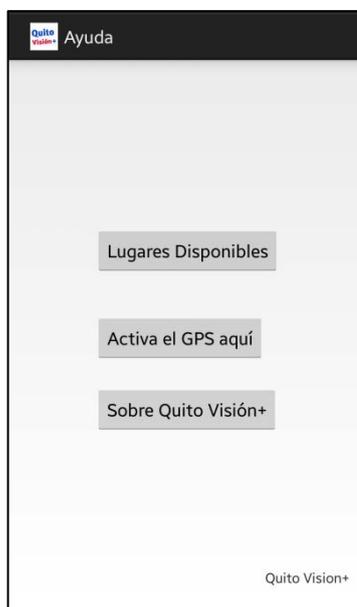


Figura 58 Rango en Samsung S3 Mini



**Figura 59 Menú Ayuda en Samsung Galaxy S5**

### 4.3 USABILIDAD Y PRECISIÓN

La precisión y usabilidad en esta aplicación es de vital importancia ya que cada punto de interés debe localizarse en una posición determinada y también las interfaces deben ser cómodas para el usuario. A continuación, se presenta un cuadro en donde se muestra los valores obtenidos después de realizar las pruebas:

#### Cuadro 23

#### Precisión y Usabilidad

PRECISIÓN Y USABILIDAD		
CASO DE USO	PRECISIÓN	USABILIDAD
Menú inicio	Correcto	Fácil
Vista en realidad aumentada	Desplazamiento y demora de detalles de POI-markers	Fácil
Detalles de POI	Correcto	Fácil
Captura de pantalla	Correcto	Medio
Rango de interés	Correcto	Medio
3 opciones de ayuda	Correcto	Fácil

El problema encontrado en la función, Vista en Realidad Aumentada, es que el posicionamiento de los markers, de acuerdo a su latitud y longitud, no son correctos, ya que en cada dispositivo en donde se realizaron las pruebas, Samsung Galaxy S3 Mini y S5, se encuentran levemente desplazados al momento de compararlos.

El cálculo del error se realizó tomando como valor teórico, las coordenadas oficiales del palacio de Carondelet y como valor experimental el obtenido de la aplicación, GPS Coordinates, que ayudó para obtener las coordenadas en formato decimal. Las coordenadas obtenidas con la aplicación ya mencionada, se almacenaron en la base de datos del servicio web.

A continuación, un ejemplo del cálculo del porcentaje de error en la latitud:

$$E\% = \frac{Valor_{teórico} - Valor_{experimental}}{Valor_{teórico}} * 100$$

$$E\%_{Latitud} = \frac{0,229722 - 0,219993}{0,219722} * 100$$

$$E\%_{Latitud} = 0,12\%$$

Diferentes publicaciones sobre realidad aumentada establecen que el margen de error de las aplicaciones, no solo basadas en geolocalización sino también en marcadores, tiene un error del  $\pm 5\%$  en la ubicación de la información virtual. Por ello es que, se dio como aceptado el porcentaje de error en cuanto al posicionamiento de POI-markers.

Otro problema que se encontró al momento de realizar las pruebas, fue la demora en la carga de los detalles de los POIs seleccionados, esto se debe a que cuando una aplicación utiliza elementos internos de los dispositivos, ya sean GPS, brújula o el tipo de conexión a internet, el

resultado es diferente de acuerdo a la ubicación del usuario y al modelo de teléfono que utilice.

“Quito Visión+” está diseñada para trabajar en lugares abiertos con conexión a internet vía datos móviles o Wi-Fi, pero como es conocido en el Ecuador, las empresas telefónicas no prestan un servicio 100% óptimo en los datos móviles, a esto adicionándole la condición climática y las características de cada dispositivo crea una demora al momento de cargar información.

#### 4.4 MEMORIA

El uso de la memoria RAM es importante, ya que muchos usuarios no desean descargar aplicaciones muy pesadas. Para conocer los valores se utilizó la aplicación 3C Toolbox.

Para conocer los valores del uso de la memoria RAM se dejó funcionando la aplicación por 3 minutos y con la aplicación 3C Toolbox se encontró el valor promedio del uso de la memoria.

A continuación, la figura 60 muestra el resultado de la prueba con la aplicación mencionada en el teléfono Samsung S5:

Uso de memoria			
Virtual	1015,00MB	RSS	160,38MB
Compartida	25,73MB	Data	90,93MB
Memoria usada realmente			124,96MB

**Figura 60 Resultado de 3CToolbox en Samsung S5**

La figura 61 muestra el resultado obtenido en dispositivo Samsung S3 Mini de la aplicación 3CToolbox:

Uso de memoria			
Virtual	534,93MB	RSS	98,67MB
Compartida	14,61MB	Data	73,57MB
Memoria usada realmente			68,75

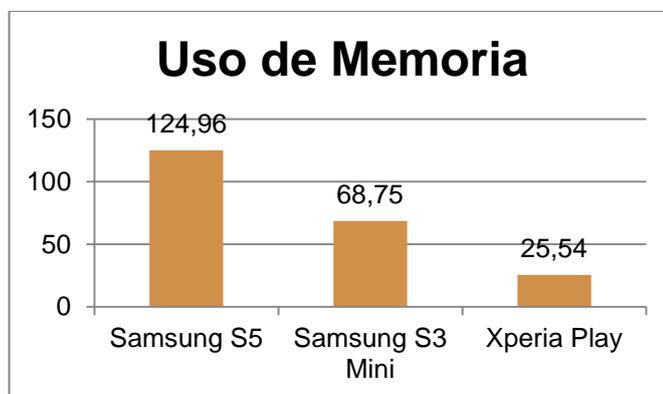
**Figura 61 Resultado de 3CToolbox en Samsung S3 Mini**

Finalmente en la figura 62, se puede observar el resultado de la prueba en el dispositivo Xperia Play:

Memory Usage			
Virtual	147.12MB	RSS	51.22MB
Shared	13.88MB	Data	48.53MB
Actual Memory Used			28.26MB

**Figura 62 Resultado de 3CToolbox en Xperia Play**

La figura 63 muestra las barras comparativas del uso de la memoria RAM en los dispositivos de prueba:



**Figura 63 Uso de la Memoria RAM**

El uso real de la aplicación, en el Samsung S5 es de 124,96, en el Samsung S3 mini el uso real es de 68,75 y en el Xperia Play su uso real es de 25,54 en un día.

El valor obtenido en el Xperia Play es solo del uso de la función Ayuda y no de la vista en realidad aumentada ya que el dispositivo no soporta AsyncTask. En los dispositivos restantes el valor es debido al uso de la

cámara y los diferentes tipos de conexiones a internet, además del uso de los sensores.

#### 4.5 CONSUMO DE DATOS MÓVILES Y WI-FI

Es necesario conocer el consumo de datos de la aplicación para evitar que usuario tenga que gastar más dinero de lo establecido en su plan de datos. Así, también es necesario conocer el uso de Wi-Fi por parte de la aplicación.

Para probar el uso de datos, se dejó corriendo la aplicación durante 10 minutos, en un lugar abierto y utilizando todas sus funciones, en especial la Vista en Realidad Aumentada.

La prueba se realizó con la ayuda de la opción Uso de Datos propia de Android y con la aplicación “My Data Manager”, que permiten conocer el consumo de datos y Wi-Fi por aplicación.

A continuación, las figuras muestran los resultados de los consumos de datos obtenidos de las dos aplicaciones en los dos dispositivos de prueba, Samsung S3 Mini y Samsung S5:

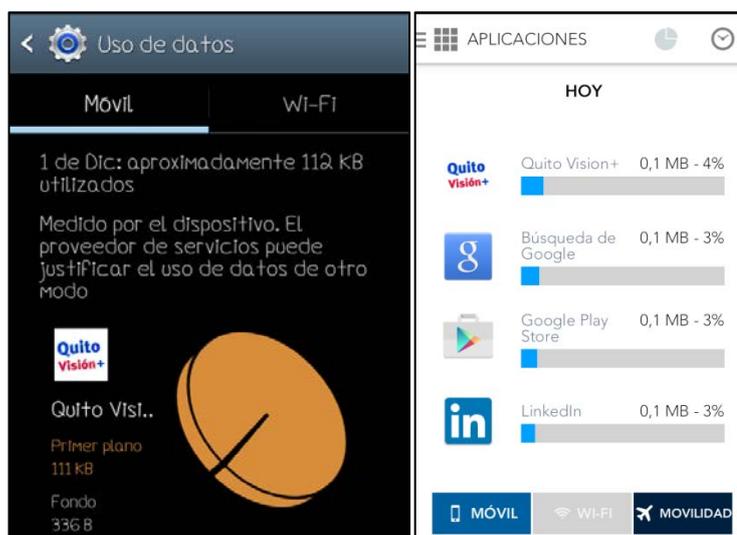
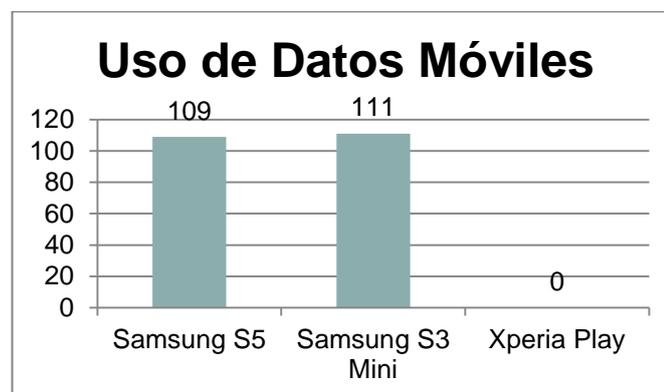


Figura 64 Uso de Datos obtenidos de Samsung S3 Mini



**Figura 65 Uso de Datos Obtenidos de Samsung S5**

Con los datos obtenidos se muestra que el consumo de datos móviles de la aplicación en primer plano, va de un valor de 100KB a 112KB en un día y con más aplicaciones corriendo en el teléfono y haciendo uso de datos en el S3 Mini. A diferencia del Samsung S5 que los datos consumidos en el mismo lapso de tiempo y bajo las mismas condiciones, va de entre 109KB a 112KB. En el dispositivo Xperia Play no hace uso de los datos móviles, debido a que no puede realizar una conexión a internet con la clase AsyncTask.

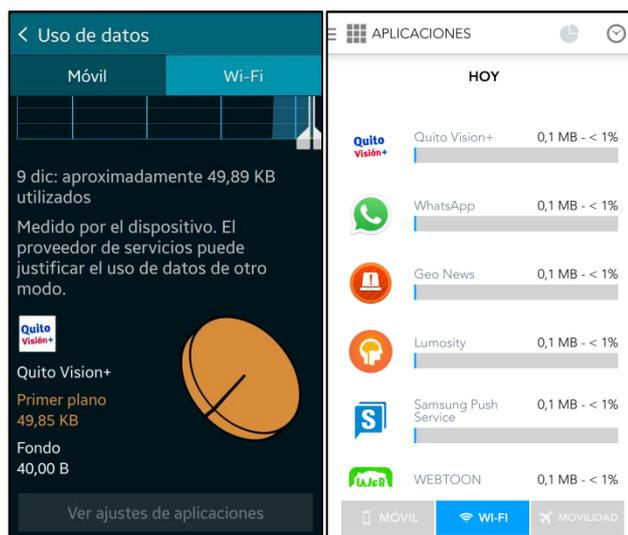


**Figura 66 Uso de Datos Móviles**

A continuación, de los resultados del consumo de red Wi-Fi obtenidos de las dos aplicaciones, en los dispositivos ya mencionados en la prueba:



**Figura 67** Uso de Wi-Fi obtenidos de Samsung S3 Mini



**Figura 68** Uso de Wi-Fi obtenidos de Samsung S5

Los resultados muestran que la aplicación utiliza Wi-Fi en 45,26 KB de un total de aplicaciones corriendo en un momento determinado en el Samsung S3 Mini y en el Samsung S5 consumen un aproximado de bajo las mismas condiciones de pruebas.

Con estos resultados se puede dejar claro que la aplicación no consume un valor alto de datos y no será tanto costo para el usuario.

## CAPÍTULO 5

### CONCLUSIONES Y RECOMENDACIONES

#### 5.1 CONCLUSIONES

- “Quito Visión+” es una aplicación que permite la vista en realidad aumentada de detalles de los edificios importantes del centro histórico de Quito para teléfonos Android.
- “Quito Visión+” es una aplicación de tamaño 12,72 MB, está disponible para dispositivos a partir de la versión 3.0 Honeycomb, que dispongan de conexión a internet, GPS y brújula en buen estado.
- La aplicación consta de funciones que permiten al usuario realizar captura de pantalla, conocer detalles de los sitios y escoger un rango de visibilidad, en metros, en donde se podrá observar los POIs de acuerdo a la elección.
- La función, Vista en Realidad Aumentada, no tiene una precisión del 100% debido a que existe un desplazamiento en la colocación de los POI-Markers, causado por motivos como: fabricación de los dispositivos, tipo de conexión a internet o condiciones climatológicas.
- En el mercado existen varios SDKs para realidad aumentada y tras un análisis, Wikitude fue la opción escogida para realizar “Quito Visión+”, no solo por su buen prestigio sino por sus principales características como ser multiplataforma, tracking por GPS y marcadores, lenguaje de programación Java, desarrollo de “mundos” en lenguaje HTML y JavaScript.
- La recopilación de los datos de importancia, que se muestran en los detalles de cada POI, fueron recolectados gracias a la campaña de “Datos Abiertos” del Municipio de Quito, que permite acceder a la

información tanto del mismo Municipio como de más centros de datos de la capital, para de esa manera ayudar al impulso del turismo.

- El servicio web fue desarrollado en lenguaje web PHP y permite la conexión con una base de datos. Su respuesta se encuentra en formato JSON, que consta de un JSONArray con los datos de la consulta hecha JSONObject, ya que hace que la transferencia entre servidor y cliente sea segura.
- Debido a que la respuesta del servicio web debe estar disponible a todo momento, se la publico en el web hosting gratuito, con la transferencia del archivo PHP mediante FileZilla que maneja el protocolo FTP.
- Se almacenó la información en una base de datos, ya que de esa manera si es necesario actualizar, borrar o agregar un punto de interés, se lo puede realizar de manera rápida y la actualización en la aplicación seria inmediata.
- “Quito Visión+” utiliza la clase AsyncTask para el manejo de conexiones a internet, por ello la versión mínima para utilizar la aplicación es la 3.0 Honeycomb, no tomando en cuenta que la versión mínima para desarrollar aplicaciones con Wikitude es la 2.3.3 Gingerbread porque dicha versión no soporta la clase AsyncTask.
- En los teléfonos inferiores a la versión 3.0 no se utilizó la clase AsyncTask debido a que no es soportada y tampoco se usó handlers y threads, ya que consumo de memoria sería mayor causando problemas a los dispositivos y saturando el UI Thread con las conexiones a internet.
- La licencia otorgada por Wikitude no permite la publicación de la aplicación en el Google Play debido a que es del tipo estudiantil.

## 5.2 RECOMENDACIONES

- Constatar que el dispositivo en donde se vayan a realizar las pruebas este en buen estado en cuanto a cámara, GPS, sensores de posición y brújula.
- Ingresar palabras en la base de datos sin la letra “ñ” y evitar colocar palabras con tilde ya que el SDK Wikitude es desarrollado en un país de lenguaje anglosajón y marca como null al texto.
- Establecer los permisos en el Android Manifest para poder tener acceso a la cámara, GPS y los diferentes tipos de conexiones a internet.
- Escribir el servicio web en lenguaje PHP ya que es un lenguaje para páginas de internet y así poder publicarlo en un web hosting.
- El mundo o experiencia en realidad aumentada debe tener como extensión mundo.html y no de esta manera mundo.html?myar=1.
- Para eliminar la marca de agua de la cámara, comunicarse personalmente con los directivos de Wikitude, para solicitar una licencia estudiantil dando el nombre del paquete de la aplicación y datos personales.
- Habilitar la instalación de aplicaciones de orígenes distintos a Google Play en los dispositivos para que la aplicación pueda correr sin problemas.
- Probar y utilizar “Quito Visión+” en lugares abiertos, ya que es una aplicación de realidad aumentada basada en localización.
- Para llamar a las funciones de los archivos JavaScript, colocar en la página HTML que será el “mundo”, la siguiente línea de código:  
<script src= “architect://architect.js”></script>.
- Como trabajo futuro se recomienda crear una aplicación que muestre los Tweets en realidad aumentada de personas cercanas a un centro turístico de Quito.

- Se podría mejorar la aplicación con la creación de un sistema de navegación en realidad aumentada que dirija a los turistas hacia más lugares turísticos del norte y sur de la ciudad. Además agregar los Tweets de personas cercanas a un centro turístico en realidad aumentada con la utilización de la API que ofrece Twitter.
- Crear aplicaciones de realidad aumentada que combinen tracking de imágenes y geolocalización para sectores como educación y turismo.
- Desarrollar aplicaciones de realidad aumentada para usuarios de Google Glass y Epson Moverio.

## BIBLIOGRAFÍA

- Abellán, B. (8 de Mayo de 2014). *WebService: Conexiones a base de datos Mysql desde Android. PARTE 1 (Insertar)*. Obtenido de <http://picarcodigo.blogspot.com/2014/05/webservice-conexiones-base-de-datos.html>
- Alejandro Alcalde. (5 de Marzo de 2013). *El Baul del Programador*. Recuperado el 7 de Julio de 2014, de Fundamentos Aplicaciones Android Parte II: [http://elbauldelprogramador.com/fundamentos-aplicaciones-android-parte\\_18/](http://elbauldelprogramador.com/fundamentos-aplicaciones-android-parte_18/)
- Android Developers. (2012). *Herramientas de Desarrollo*. Recuperado el 25 de Junio de 2014, de <http://developer.android.com/intl/es/sdk/index.html>
- Android Developers. (2012). *Using the Emulator*. Recuperado el 25 de Junio de 2014, de <http://developer.android.com/tools/devices/index.html>
- Android Developers. (25 de Junio de 2014). *Android Emulator*. Obtenido de <http://developer.android.com/tools/help/emulator.html>
- Android Developers. (25 de Junio de 2014). *Dashboards*. Obtenido de <http://developer.android.com/about/dashboards/index.html>
- Android Developers. (07 de Julio de 2014). *Desarrollo: Guías de la API*. Recuperado el 07 de Julio de 2014, de Fundamentos de las Aplicaciones: <http://developer.android.com/intl/es/guide/components/fundamentals.html>
- Android Developers. (7 de Julio de 2014). *Desarrollo: Guías de la API*. Recuperado el 7 de Julio de 2014, de Actividades: <http://developer.android.com/intl/es/guide/components/activities.html>
- Android Developers. (2014). *Guías de API*. Recuperado el 27 de Junio de 2014, de Sensores: [http://developer.android.com/intl/es/guide/topics/sensors/sensors\\_overview.html](http://developer.android.com/intl/es/guide/topics/sensors/sensors_overview.html)
- Android Developers. (25 de Junio de 2014). *SDK Manager*. Obtenido de <http://developer.android.com/tools/help/sdk-manager.html>
- AumentaME. (19 de Agosto de 2011). *AumentaME Blog*. Obtenido de <http://www.aumenta.me/?q=node/36>
- Business Wire. (18 de Septiembre de 2013). *Wikitude and Optinvent Show True Augmented Reality Glasses Experience*. Obtenido de

<http://www.businesswire.com/news/home/20130918005578/en/Wikitude-Optinvent-Show-True-Augmented-Reality-Glasses#.VFu4STSG9qU>

Climent, M. (30 de Junio de 2014). *El Mundo*. Recuperado el 03 de Julio de 2014, de Altamira, pionera del turismo 'aumentado' con Google Glass:  
<http://www.elmundo.es/economia/2014/06/30/53adae45ca47419a4d8b456b.html>

Del Alcazar Ponce, J. (21 de Abril de 2014). *Realidad Aumentada: Oportunidades y Usos*. Obtenido de <http://blog.formaciongerencial.com/2014/04/21/realidad-aumentada-oportunidades-y-usos/>

Di Nardo, E. (30 de Mayo de 2013). *OpenCV / Python – Primi passi*. Obtenido de <http://emanueldinardo.com/wp/opencv-python-primi-passi/>

Diario HOY, E. (13 de 05 de 2013). *Diario Hoy*. Obtenido de Diario Hoy:  
<http://www.hoy.com.ec/noticias-ecuador/hoy-entra-a-la-realidad-aumentada-581036.html>

Drgoldie. (23 de Julio de 2009). *Wikipedia*. Obtenido de [http://es.wikipedia.org/wiki/Realidad\\_aumentada](http://es.wikipedia.org/wiki/Realidad_aumentada)

Dubey, A., & Misra, A. (2013). *Android Security - Attacks and Defenses*. Boca Raton, FL: CRC Press.

Grubert, J., & Dr Grasset, R. (2013). *Augmented Reality for Android Application Development*. Birmingham, UK: Packt Publishing Ltd.

Jackson, W. (2011). *Android Apps for Absolute Beginners*. New York: Apress.

Kotzer, Z. (12 de Junio de 2014). *A New VR War Game Aims to Teach First Aid to Conflict Journalists*. Obtenido de [http://motherboard.vice.com/en\\_uk/read/a-new-vr-war-game-aims-to-teach-first-aid-to-conflict-journalists](http://motherboard.vice.com/en_uk/read/a-new-vr-war-game-aims-to-teach-first-aid-to-conflict-journalists)

LEGO. (21 de Juio de 2014). *LEGO Fusion*. Obtenido de <http://www.lego.com/en-us/fusion/apps>

Minetur. (17 de Noviembre de 2013). *Chaval.es*. (Secretaría de Estado de Telecomunicaciones y para la Sociedad de la Información) Recuperado el 24 de Abril de 2014, de <http://www.chaval.es/chavales/enterate/nuevos-usos/%C2%BFque-es-la-realidad-aumentada>

Momma, E., & Minagawa, T. (9 de Noviembre de 2011). *Introducing OpenCV for Developers*. Recuperado el 9 de Julio de 2014, de [http://www.jsk.t.u-tokyo.ac.jp/rsj2011/\\_downloads/2Q2-3.pdf](http://www.jsk.t.u-tokyo.ac.jp/rsj2011/_downloads/2Q2-3.pdf)

- Nieto, V. (24 de Junio de 2014). *McDonald's presentó una aplicación de Realidad Aumentada con motivo del Mundial*. Obtenido de <http://www.raparaguay.com/mcdonalds-presento-una-aplicacion-de-realidad-aumentada-con-motivo-del-mundial/>
- Ochoa, M. (30 de Enero de 2013). *Monografias.com*. Obtenido de <http://www.monografias.com/trabajos94/estado-del-arte-realidad-aumentada-educacion/estado-del-arte-realidad-aumentada-educacion.shtml>
- Planet Numbers. (30 de Enero de 2014). *Alice through the Google Glass*. Obtenido de <http://www.planet-numbers.co.uk/blog/alice-through-the-google-glass/>
- Qualcomm. (7 de Julio de 2014). *Develop*. Recuperado el 7 de Julio de 2014, de Augmented Reality (Vuforia): <https://developer.qualcomm.com/mobile-development/add-advanced-features/augmented-reality-vuforia>
- SOCIALetic. (27 de Junio de 2014). Realidad Aumentada y smartglasses Epson se dan cita en Droidcon Madrid. Barcelona, Barcelona, España.
- TechENT. (25 de Septiembre de 2014). *Qualcomm Introduce Vuforia, A virtual Reality App SDK*. Obtenido de <http://techent.tv/qualcomm-vuforia-virtual-reality-app-sdk/>
- The Geek Church. (27 de Enero de 2014). *Epson and the Moverio BT-200 smart glasses*. Obtenido de <http://thegeekchurch.com/?cat=1871>
- Universidad Carlos III De Madrid. (2012). *Software de Comunicaciones*. Recuperado el 19 de Junio de 2014, de Programación para Dispositivos Móviles: <https://sites.google.com/site/swcuc3m/home/android/generalidades/dalvikvm-1>
- Wikitude. (9 de Julio de 2014). *Wikitude*. Recuperado el 9 de Julio de 2014, de <http://www.wikitude.com/about/jobs-career/>
- WION. (4 de Febrero de 2013). *Guías turísticas con geolocalización, frictionless y realidad aumentada*. Obtenido de <http://www.realidad-aumentada.eu/guias-turisticas-con-geolocalizacion-frictionless-y-realidad-aumentada/>

# ANEXOS



## Tabla de contenido

<b>INTRODUCCIÓN.....</b>	<b>101</b>
<b>COMPONENTES.....</b>	<b>101</b>
<b>ICONOS .....</b>	<b>103</b>
<b>REQUISITOS DEL SISTEMA.....</b>	<b>103</b>
<b>INICIANDO LA APLICACIÓN .....</b>	<b>103</b>
<b>VISTA EN REALIDAD AUMENTADA .....</b>	<b>104</b>
<b>AYUDA.....</b>	<b>107</b>
<b>RECOMENDACIONES DE USO .....</b>	<b>108</b>

## 1. INTRODUCCIÓN

“Quito Visión+” es una aplicación para dispositivos móviles con sistema operativo Android a partir de la versión 3.0 Honeycomb, que muestra los datos de mayor importancia como: nombre, contactos, condiciones de ingreso, historia y horarios de entrada de los edificios patrimoniales del centro histórico de Quito en realidad aumentada.

La aplicación está dirigida principalmente para turistas extranjeros y para ciudadanos ecuatorianos en general, que desean conocer sobre los principales edificios de una manera dinámica y moderna.

Con “Quito Visión+” se impulsa y aporta en tratar de hacer de Quito una Smart City, cada vez más actualizada, inclusiva y tecnológica.

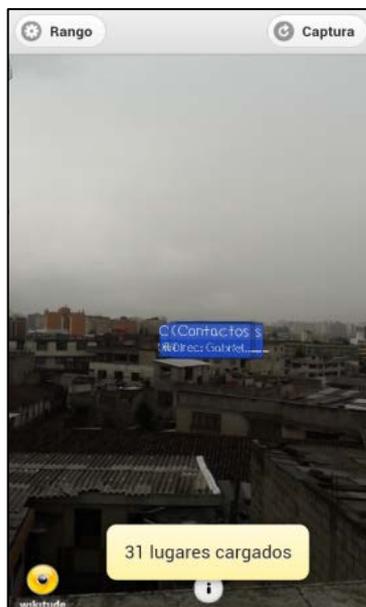
Para poder usar la aplicación, tan solo se necesita de un dispositivo Android con la cámara principal en buen estado, GPS activo y conexión a internet, ya sea vía Wi-Fi o datos móviles. Adicionalmente, el dispositivo móvil debe tener habilitado el permiso de instalación desde fuentes distintas a Google Play.

## 2. COMPONENTES

“Quito Visión+” se compone de dos interfaces principales, la primera es la Vista en Realidad Aumentada y la segunda un menú de Ayuda.

La vista en realidad aumentada muestra los puntos de interés mediante marcadores de color azul a través del enfoque de la cámara en la pantalla.

La siguiente figura muestra un ejemplo del enfoque de la cámara con información virtual agregada:



**Figura 1 Vista en Realidad Aumentada**

El menú ayuda le proporciona al usuario tres opciones que son: una lista de lugares disponibles, activar el GPS y una breve descripción de la aplicación.

La siguiente figura muestra las opciones de ayuda, representadas por un botón Android en un menú:



**Figura 2 Menú Ayuda**

### 3. ICONOS

**Cuadro 1**  
**Iconos del Sistema**

ICONO	SIGNIFICADO
	Launcher y Logo de Quito Visión+
	Ayuda
	Cargando Puntos de Interés
	Información de puntos de interés cargados
	Abre el Panel para Rango De Interés
	Realiza una captura de Pantalla

### 4. REQUISITOS DEL SISTEMA

Los requisitos mínimos para poder hacer uso de la aplicación “Quito Visión+” son los siguientes:

- Sistema Operativo Android versión 3.0 Honeycomb.
- OpenGL 2.0
- Cámara principal en buenas condiciones.
- Dispositivo con acelerómetro, brújula y GPS.
- Conexión a internet.

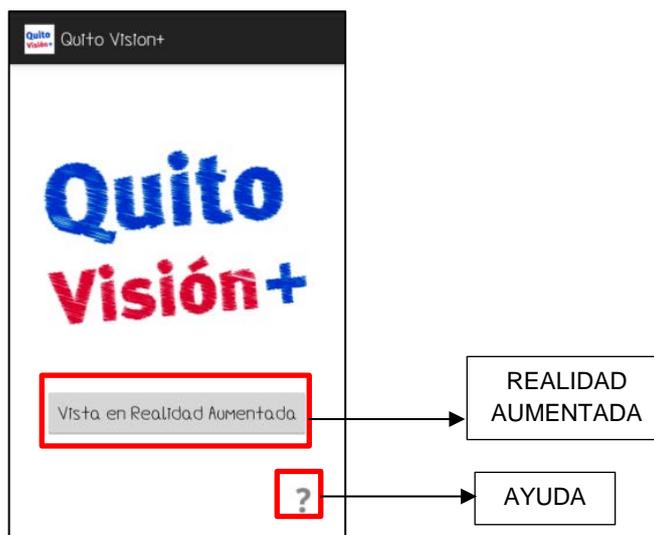
### 5. INICIANDO DE LA APLICACIÓN

Para abrir la aplicación se debe dar clic en el icono de “Quito Visión+” en la pantalla principal de las aplicaciones del dispositivo móvil.



**Figura 3 Menú de Aplicaciones Android**

Enseguida aparece una pantalla de bienvenida y se muestra el menú inicial de la aplicación con las opciones Vista en Realidad Aumentada y Ayuda.



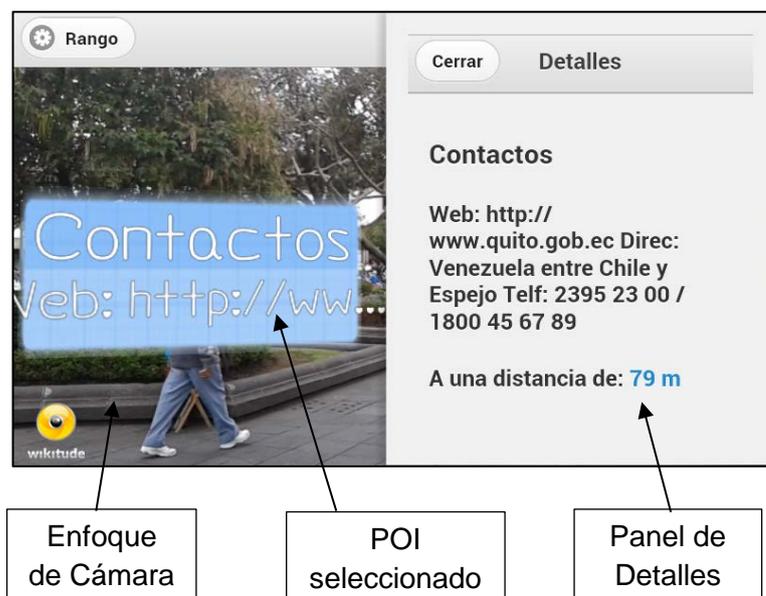
**Figura 4 Menú Inicial**

## 6. VISTA EN REALIDAD AUMENTADA

Para acceder debe realizar un clic en el botón: Vista en Realidad Aumentada, tras la aparición de una pantalla con el logo de Wikitude se puede visualizar el enfoque de la cámara mientras se cargan los puntos de interés.

Para asegurarse que los POIs han sido cargados correctamente y para saber el número de cuantos fueron cargados, dar clic en el botón que se encuentra centrado en la parte inferior de la pantalla.

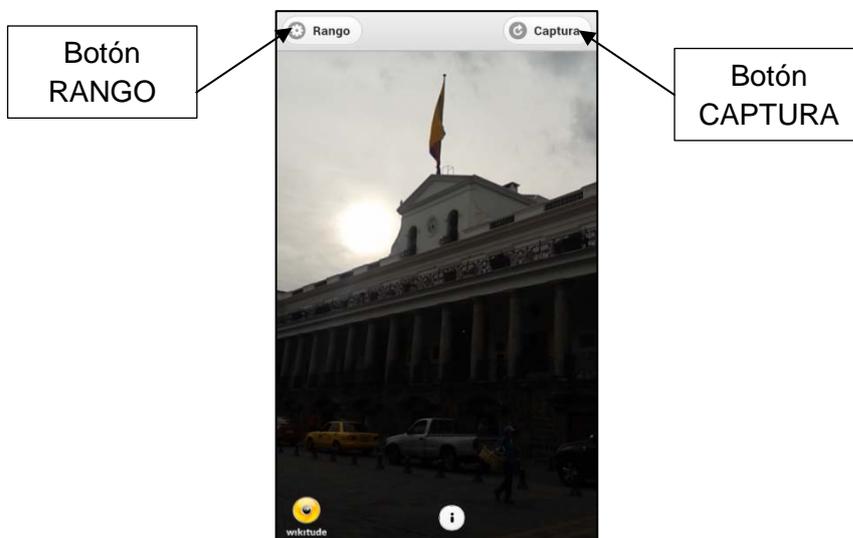
Los puntos de interés están representados por un marker de color azul, que contiene su respectivo nombre y una breve descripción. Para seleccionar un POI se debe dar clic sobre este en la pantalla, el marker cambia de color a celeste y en ese momento aparece un panel en donde se puede observar la información que contiene.



**Figura 5 Vista en Realidad Aumentada**

Esta función tiene la opción de captura de pantalla, a la cual se puede acceder mediante un clic en el botón CAPTURA, en la barra superior de la pantalla. Si desea puede compartir la imagen en las redes sociales o correo electrónico.

La imagen capturada es el resultado del enfoque de la cámara con los puntos de interés, sin la barra de opciones y botón informativo.



**Figura 6 Opción de Captura**



**Figura 7 Resultado de Captura de Pantalla**

También es una opción escoger un rango de interés, se puede acceder mediante un clic en el botón RANGO. Aparecerá un panel en donde el usuario escoge el valor deseado.

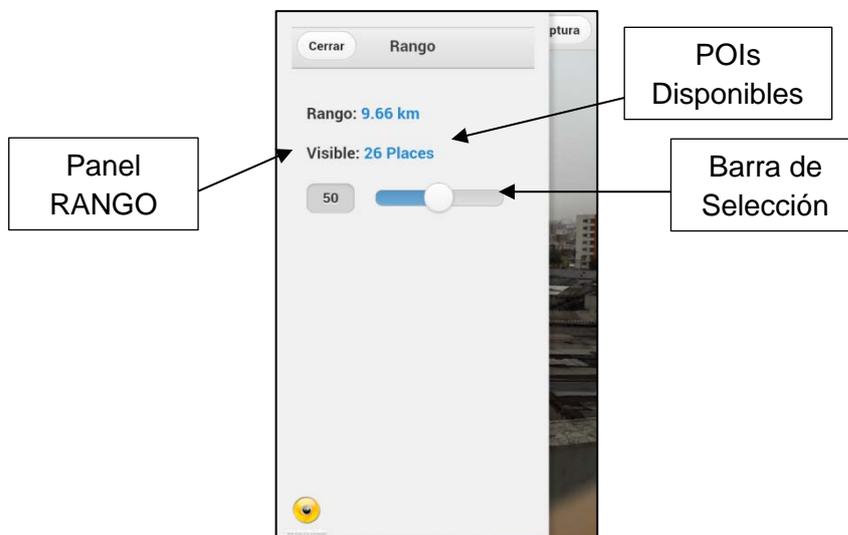


Figura 8 Opción Rango

## 7. AYUDA

Se puede acceder al menú ayuda mediante un clic en el signo de interrogación en la pantalla principal. Las opciones son: una lista de lugares disponibles para visitar, activar el GPS y una breve descripción de la aplicación.

La siguiente figura muestra las opciones de ayuda de la aplicación:

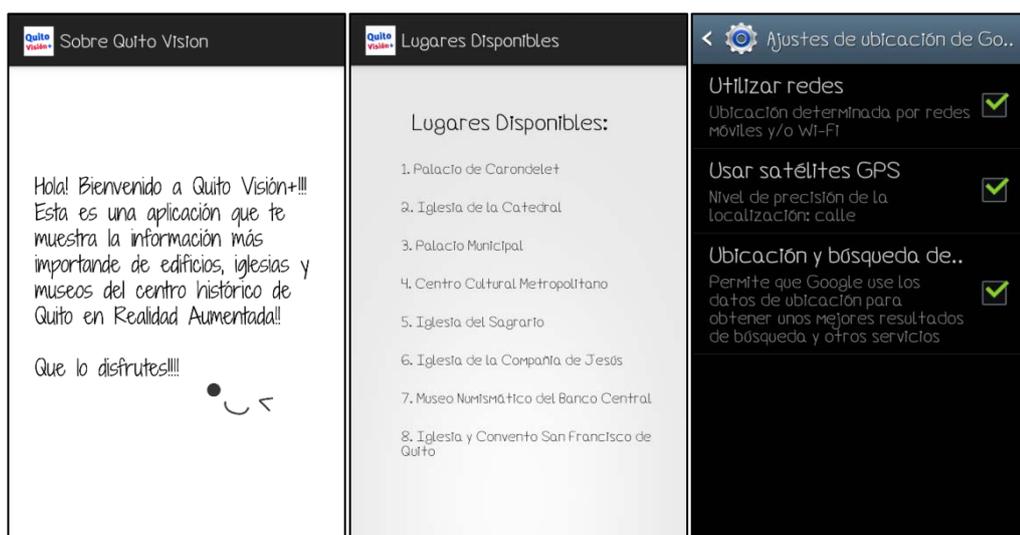


Figura 9 Opciones de Menú Ayuda

## 8. RECOMENDACIONES DE USO

- Verifique que el GPS este encendido, en caso de no estarlo dirigirse a ayuda y presionar el botón Activar GPS.
- No olvide que el dispositivo debe tener acceso a internet ya sea Datos Móviles o Wi-Fi.
- Recuerde que se puede seleccionar solo un punto de interés a la vez.
- Los puntos de interés aparecerán de acuerdo a la ubicación del dispositivo móvil.
- Al inicial la aplicación agite levemente el dispositivo para activar la brújula del mismo.
- Habilitar la instalación de aplicaciones desde fuentes desconocidas en los Ajustes del dispositivo móvil.
- Verificar cuantos puntos de interés han sido cargados, dando clic en el botón centrado en la parte inferior de la pantalla.
- Hacer uso de la aplicación en lugares abiertos ya que necesita encontrar la ubicación del usuario sin interferencias.

## ANEXO B

### INSTALACIÓN DEL ADT BUNDLE DE ANDROID

El sitio web oficial de Android permite la descarga de un conjunto de plugins necesarios para desarrollar aplicaciones que se llama ADT Bundle, en la sección 2.2.5 del Capítulo 2 se detalla cada uno de los componentes.

En el siguiente link se encuentra disponible la descarga para diferentes plataformas: <http://developer.android.com/sdk/index.html>.

#### 1. Requisitos del Sistema

##### 1.1. Sistemas Operativos

ADT plugin está disponible para Windows, Mac y Linux, a continuación las versiones con las cuales el ADT Bundle es compatible:

#### Cuadro 1

##### Requisitos para la instalación en Sistemas Operativos

Sistema operativo	Versiones
<b>Windows</b>	Windows XP 32-bit Windows Vista 32 y 64-bit Windows 7 32 y 64-bit
<b>Mac</b>	OSX 10.8.5 o superiores
<b>Linux</b>	Distribución 64-bit con la capacidad de correr aplicaciones de 32-bit. Librería GNU C (glibc) 2.11 o superiores.

##### 1.2. Herramientas de Desarrollo

En cuanto a las herramientas de desarrollo se necesita los siguientes componentes:

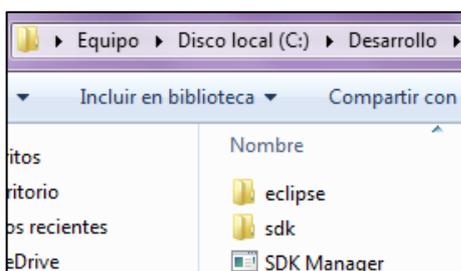
## Cuadro 2

### Requisitos De Herramientas de Desarrollo

Herramienta	Requisito
JDK	JDK 6
Apache	Apache Ant 1.8 o superiores

## 2. Instalación de ADT Bundle

Después de la descarga del ADT Bundle descomprimir el contenido y reubicarlos directamente en una carpeta en el disco C. Tal como se muestra en la siguiente figura:



**Figura 1 Ubicación de Archivos Descomprimidos**

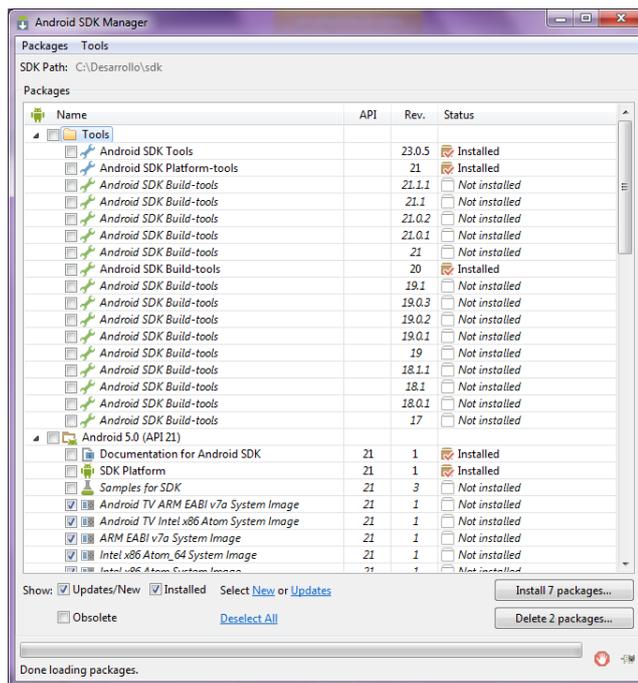
Dirigirse a la carpeta Eclipse y ejecutar el archivo eclipse.exe para agregar los plugins:

*C:\Desarrollo\eclipse*

Con la ayuda del Android SDK Manager se podrán agregar las herramientas, plataformas y más componentes necesarios. En la barra de herramientas Android dirigirse a este icono: 

Se abrirá una ventana en donde se debe escoger las últimas versiones de: Android SDK Tools, Android SDK Platform-tools y Android SDK Build-tools.

Adicionalmente, se debe actualizar el SDK Platform e instalar la imagen del emulador ARMEABI v7a System Image. La figura 2 muestra el Android SDK Manager:



**Figura 2 Android SDK Manager**

Una vez descargados todos los paquetes adicionales, eclipse se reiniciará y el IDE estará disponible para poder crear aplicaciones Android.

## ANEXO C

### CREACIÓN DE UNA APLICACIÓN BÁSICA ANDROID

Es necesaria la creación de una aplicación básica para poder familiarizarse con el desarrollo en Android. El lenguaje de desarrollo que utiliza es Java con el SDK de Android.

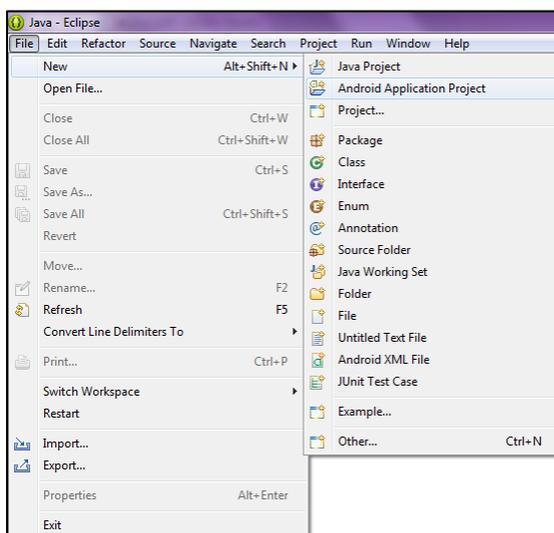
Previamente se debe realizar una correcta instalación de los paquetes adicionales del SDK de Android, para más información sobre la instalación ver el Anexo B o dirigirse a la página web oficial de desarrolladores Android:

<http://developer.android.com/sdk/index.html>

Los pasos para crear una aplicación son los siguientes:

#### 1. Creación del Proyecto

- 1.1. En eclipse dirigirse a Archivo → Nuevo → Android Application Project, como se muestra en la siguiente figura:

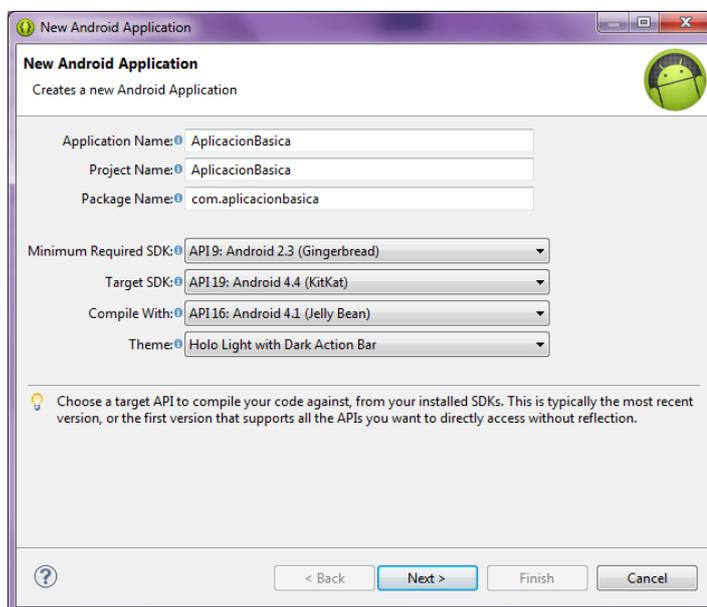


**Figura 1 Creación de un Nuevo Project**

- 1.2. En la siguiente ventana llenar los campos requeridos.

- Application Name: Será el nombre que con el que se identifica en el menú de aplicaciones del teléfono y Google Play.
- Project Name: Nombre que lo identifica en el IDE Eclipse.

- Package Name: identificador único de la aplicación, por lo general es el nombre de la compañía o dueño de la aplicación.
- Minimum Required SDK: la versión mínima que la aplicación soportará.
- Target SDK: la versión más alta con la que el teléfono trabajará.
- Compile With: la versión del dispositivo que ayudará a realizar la compilación.



**Figura 2 Detalles de Aplicación**

- 1.3. En la siguiente pantalla dejar las configuraciones por defecto.
- 1.4. Si desea crear un icono launcher, agregar su imagen o caso contrario dejar los valores por defecto.
- 1.5. Seleccionar BlankActivity en la siguiente pantalla.
- 1.6. Dar nombre a la actividad y layout principal o dejar los valores por defecto.
- 1.7. Dar clic en Finish.

## 2. Crear una Interfaz de usuario

La aplicación desde su creación cuenta un TextView en un layout tipo relative. A estos elementos se le agregará un botón para crear Intents y cambiar a una segunda interfaz de usuario.

Las siguientes figuras muestran la modificación del código y el layout con el botón agregado:

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.aplicacionbasica.MainActivity" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="98dp"
        android:text="Esta es una aplicación básica" />

    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_centerVertical="true"
        android:text="Siguiete Interfaz"
        android:onClick="SiguieteInterfaz" />

</RelativeLayout>
```

**Figura 3 Código de layout con botón**

Entre los atributos que tendrán estos elementos están los siguientes:

- Ancho y Altura: Se seleccionó "wrap\_content" para que el elemento se ajuste al texto o imagen que contiene.
- Texto: Entre comillas se coloca el texto deseado. Para mejor uso se recomienda la creación de strings.
- Id: El identificador del elemento.
- Alineado y centrado: existen varias formas de alinear y centrar elementos.
- onClick: Será el método que ejecutará al momento de ser presionado el botón.



**Figura 4 Primera Interfaz de Aplicación Básica**

### **3. Creación de Intents**

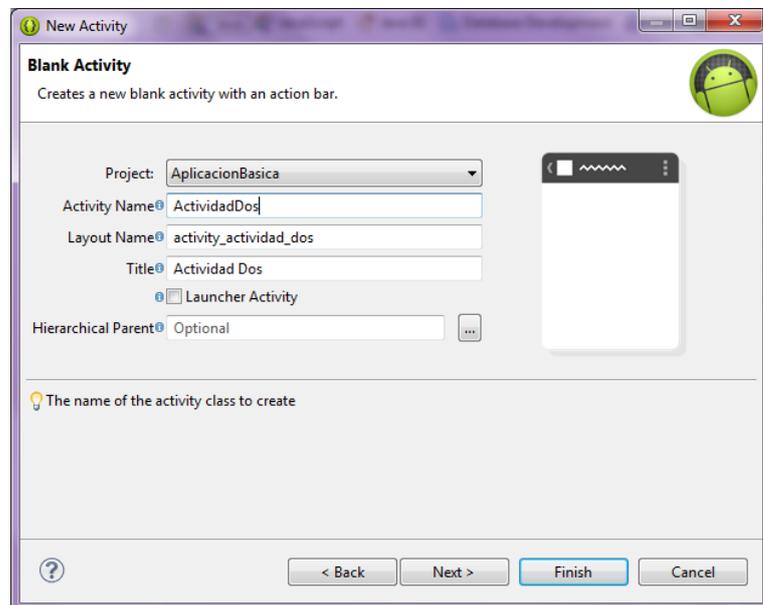
Los Intents son aquellos que pueden comunicar dos activities, la documentación de Android le define como “intento de hacer algo”. Se usan para varias tareas pero su principal función es para comenzar una nueva actividad.

#### **3.1. Crear una nueva actividad:**

Dirigirse a Nuevo → Otro → Carpeta Android → Android Activity.

#### **3.2. En la siguiente ventana escoger BlankActivity.**

#### **3.3. Dar nombre a la actividad, layout y título. Tal como se muestra en la siguiente figura:**



**Figura 5 Creación de Nueva Actividad**

- 3.4. Clic en Finish
- 3.5. Modificar el layout de la nueva actividad.
- 3.6. En el mainActivity.java escribir el siguiente código:

```
public void SiguienteInterfaz (View view){
    Intent intent = new Intent(MainActivity.this, ActividadDos.class);
    startActivity(intent);
}
```

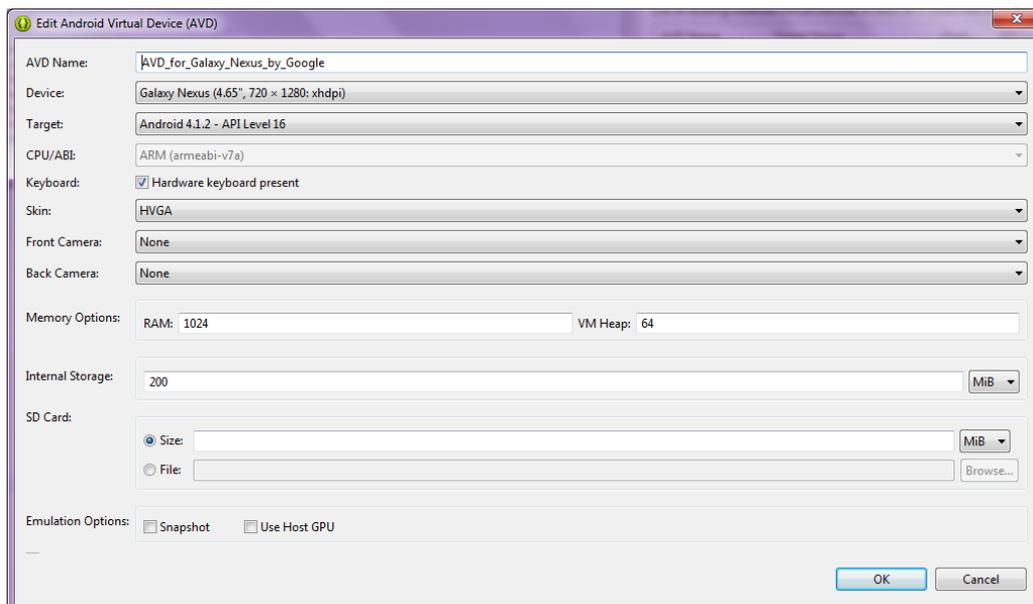
**Figura 6 Método para Dirigir a la siguiente Interfaz**

#### 4. Ejecución de la aplicación

La aplicación creada ya se puede ejecutar, para ello existen dos opciones: la primera con un dispositivo físico y la segunda con un emulador. Para probar la aplicación con un emulador se debe crear un dispositivo, los pasos a seguir son los siguientes:

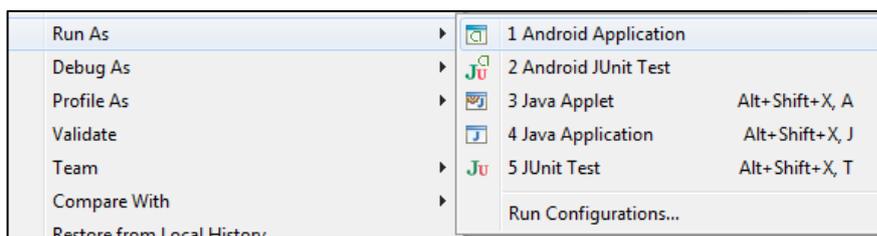
- 4.1. Abrir el AVD (Android Virtual Device) Manager.
- 4.2. Escoger la pestaña Device Definitions.

- 4.3. De los modelos presentados optar por que cumpla con las necesidades requeridas.
- 4.4. Clic en Create AVD y dar las características deseadas.



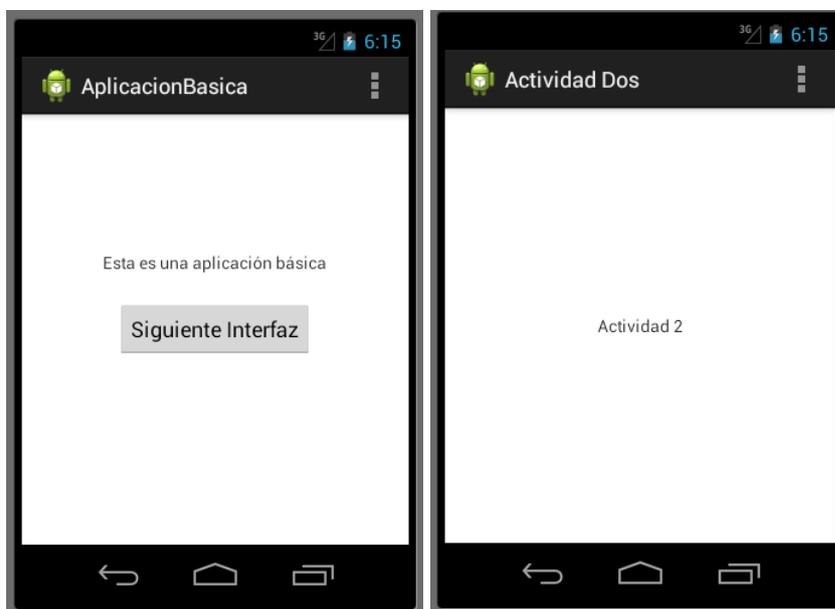
**Figura 7 Creación de Dispositivo Virtual**

- 4.5. Clic en OK y correr el programa dando clic derecho en la carpeta del proyecto → Correr Como → Android Application



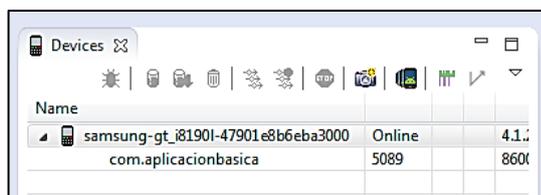
**Figura 8 Correr el Programa**

La siguiente figura muestra el resultado de la ejecución del programa en un dispositivo virtual:



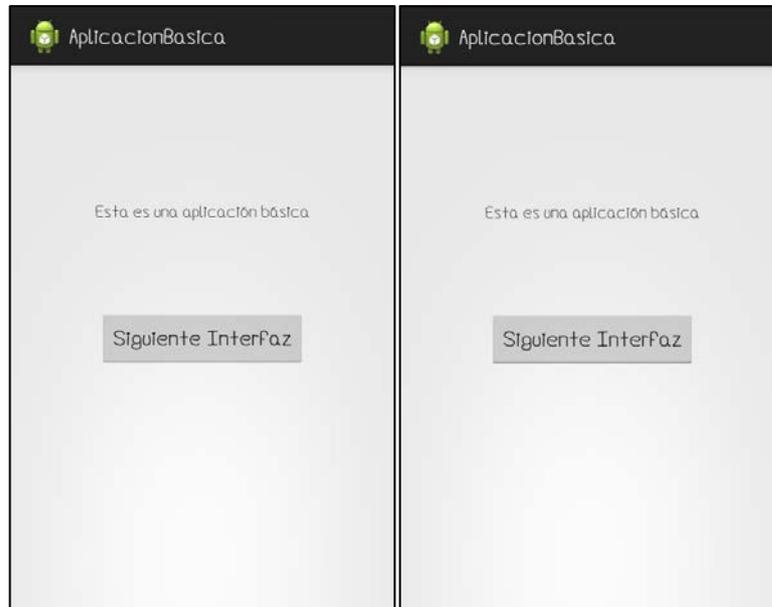
**Figura 9 Ejecución de Aplicación en Dispositivo Virtual**

Otra opción para ejecutar las aplicaciones es en un dispositivo físico, tan solo se debe conectar con cable USB a la computadora y el ADT lo reconocerá.



**Figura 10 Reconocimiento de Dispositivo Fisico**

La manera de correr el programa es de la forma que en el punto 4.5 del presente Anexo. La siguiente figura muestra los resultados de la ejecución del programa en un dispositivo físico:



**Figura 11 Ejecución de Aplicación en Dispositivo Físico**

## ANEXO D

### CREACIÓN DE APLICACIÓN BÁSICA DE REALIDAD AUMENTADA

Wikitude es uno de los SDK de renombre y mayor reconocimiento en el mercado de desarrollo, para poder tener una noción de cómo se trabaja la realidad aumentada se creó una aplicación, de acuerdo a la documentación otorgada por el fabricante.

Los pasos para la creación realidad aumentada con Wikitude son los siguientes:

#### 1. Crear Proyecto de Realidad Aumentada.

##### 1.1. Descarga de SDK Wikitude

La descarga del paquete de SDK que cuenta con documentación, ejemplos, herramientas, librería y referencias se lo puede realizar desde la página web oficial de Wikitude en el siguiente link: <http://www.wikitude.com/products/wikitude-augmented-reality-sdk-mobile/wikitude-sdk-android/>.

##### 1.2. Agregar librería

La manera más fácil de agregar una librería es copiarla en la carpeta libs del proyecto en eclipse.

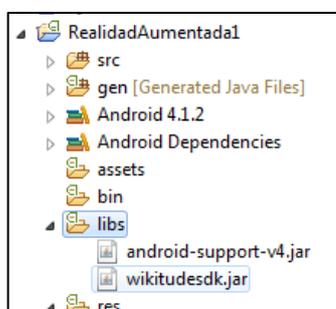
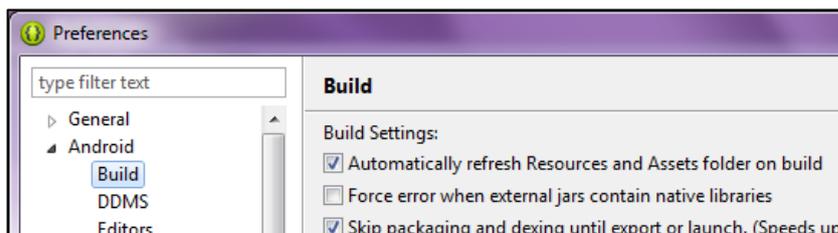


Figura 1 Carpeta libs

Se debe otorgar permisos para poder utilizar librerías externas, de la siguiente manera: Window→ Preferences→ Android →Build y desmarcar la opción “Force error when external jars contain native Libraries”.



**Figura 2 Agregar librería Wikitude**

## 2. Agregar permisos en el Manifest

En el Android Manifest deberán ir los permisos de uso de cámara, conexión a internet, requerimiento mínimo de OPEN GL, GPS y más. A continuación, del archivo XML:

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.ACCESS_GPS" />
<uses-permission android:name="android.permission.CAMERA" />
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />

<uses-feature
    android:name="android.hardware.camera"
    android:required="true" />
<uses-feature
    android:name="android.hardware.location"
    android:required="true" />
<uses-feature
    android:name="android.hardware.sensor.accelerometer"
    android:required="true" />
<uses-feature
    android:name="android.hardware.sensor.compass"
    android:required="true" />
<uses-feature
    android:glEsVersion="0x00020000"
    android:required="true" />
```

**Figura 3 Permisos en Android Manifest**

## 3. Creación de ArchitectView

El ArchitectView es la actividad que contiene la vista en realidad aumentada, para ello en el Android Manifest se debe otorgar características

como ocultar teclado, la orientación y el tamaño de la pantalla. Los pasos a seguir son los siguientes:

- 3.1. Agregar características a la actividad que contendrá el ArchitectView.

```
<activity
    android:name=".MainActivity"
    android:label="@string/app_name"
    android:configChanges="keyboardHidden|orientation|screenSize" >
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />

        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
```

**Figura 4 Características de ArchitectView**

- 3.2. Declarar la vista en realidad aumentada en el layout de la actividad:

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.realidadaumentada1.MainActivity" >

    <com.wikitudo.architect.ArchitectView
        android:id="@+id/architectView"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"/>

</RelativeLayout>
```

**Figura 5 Declaración de ArchitectView**

- 3.3. Definir el contentView con la licencia.

```
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

    //Hacer accesible solo para dispositivos con las características básicas.
    if(!ArchitectView.isDeviceSupported(this)){
        Toast.makeText(this, "Dispositivo No Cumple con Requisitos Mínimos", Toast.LENGTH_LONG).show();
        this.finish();
        return;
    }

    setContentView(R.layout.activity_main);
    this.architectView = (ArchitectView)this.findViewById( R.id.architectView );
    final ArchitectConfig config = new ArchitectConfig("VMxermoXpe2cniux9jD+gmQw0TGQ3+1drWChAkWswuL/asN
    this.architectView.onCreate( config );
}
```

**Figura 6 Definición Content View**

- 3.4. Cargar el mundo o experiencia de realidad aumentada en el `onPostCreate`. El mundo debe estar escrito en lenguaje HTML y guardado en la carpeta `assets` del proyecto.

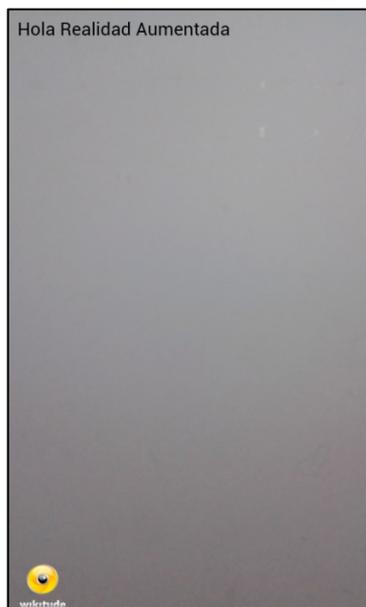
```
protected void onPostCreate(Bundle savedInstanceState) {
    super.onPostCreate(savedInstanceState);

    if (this.architectView != null)
        this.architectView.onPostCreate();

    if(!mundoCargado){
        try {
            architectView.load("index.html");
        } catch (IOException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
        mundoCargado=true;
    }
}
```

**Figura 7 Cargar Experiencia en Realidad Aumentada**

La figura 8 muestra el resultado de la ejecución de esta aplicación básica de realidad aumentada, se presenta el enfoque de la cámara con un mensaje en la parte superior:



**Figura 8 Resultado de Aplicación Básica de Realidad Aumentada**

## ANEXO E

### USO DE CLASE ASYNCTASK

La clase AsyncTask sirve para manejar las conexiones a internet desde una aplicación Android creando un segundo hilo y así evitar saturar de procesos al hilo principal de la aplicación.

La clase tiene tres tipos genéricos que son: Params, Progress y Result, además de cuatro pasos básicos que son: onPreExecute, doInBackground, onProgressUpdate y onPostExecute. Todos sus componentes y pasos ayudan a al desarrollador a no tener que lidiar con handlers y threads.

Se creó una aplicación que realiza una consulta a una base de datos y muestra el resultado en un TextView, todo esto con el uso de AsyncTask. A continuación, los pasos a seguir:

1. Crear un proyecto en Android.
2. Crear un Botón y 2 TextView en el layout. Véase la siguiente figura:



**Figura 1 Layout De Consulta**

3. Crear una clase llamada Personas.java con variables tipo String con sus respectivos getters y setters. Que servirá para colocar la información consultada y mostrarla en la pantalla.

```

public class Personas {
    private String id;
    private String nombre;

    public String getId() {
        return id;
    }
    public void setId(String id) {
        this.id = id;
    }
    public String getNombre() {
        return nombre;
    }
    public void setNombre(String nombre) {
        this.nombre = nombre;
    }
}

```

**Figura 2 Clase Personas**

4. En la actividad principal crear las siguientes variables, en el método onCreate declarar los TextView y darle al botón su acción al momento de ser presionado.

```

private TextView nombre;
private TextView id;

private Button mostrar;

private int posicion=1;
private List<Personas> listaPersonas;
private Personas personas;
@SuppressWarnings("InlinedApi")
@Override
protected void onCreate(Bundle savedInstanceState) {
    // TODO Auto-generated method stub
    super.onCreate(savedInstanceState);
    setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_SENSOR_PORTRAIT);
    setContentView(R.layout.activity_main);

    listaPersonas=new ArrayList<Personas>();
    nombre=(TextView)findViewById(R.id.nombre);
    id=(TextView)findViewById(R.id.id);

    //Mostramos los datos de la persona por pantalla.
    mostrar=(Button)findViewById(R.id.mostrar);
    mostrar.setOnClickListener(new OnClickListener(){

        @Override
        public void onClick(View v) {
            // TODO Auto-generated method stub
            new Mostrar().execute();
        }
    });
}

```

**Figura 3 Variables y método onCreate**

5. Crear un método con el nombre “mostrar” que será aquel que realice la conexión con el servicio web.

```

private String mostrar(){
    HttpClient httpclient = new DefaultHttpClient();
    HttpPost httppost = new HttpPost("http://192.168.0.106/QuitoVision/QuitoVisionWS.php");
    String resultado="";
    HttpResponse response;
    try {
        response = httpclient.execute(httppost);
        HttpEntity entity = response.getEntity();
        InputStream instream = entity.getContent();
        resultado= convertStreamToString(instream);
    } catch (ClientProtocolException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    } catch (IOException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    return resultado;
}

```

**Figura 4 Método Mostrar**

El método mostrar utiliza HttpClient y HttpPost para conectar con el servicio web que se encuentra ubicado en la dirección: <http://192.168.0.106/QuitoVision/QuitoVisionWS.php>

La variable resultado es la respuesta a la consulta realizada en el método convertStreamToString. Se ejecuta la consulta con la ayuda execute y se crea una entidad que contenga la conexión ejecutada.

6. A continuación, el código del método convertStreamToString que será el encargado de convertir datos del tipo InputStream a String. Es decir, cambia String la respuesta del servicio Web.

```

private String convertStreamToString(InputStream is) throws IOException {
    if (is != null) {
        StringBuilder sb = new StringBuilder();
        String line;
        try {
            BufferedReader reader = new BufferedReader(
                new InputStreamReader(is, "UTF-8"));
            while ((line = reader.readLine()) != null) {
                sb.append(line).append("\n");
            }
        }
        finally {
            is.close();
        }
        return sb.toString();
    } else {
        return "";
    }
}

```

**Figura 5 Método convertStreamToString**

7. El método “filtrar datos” será aquel que filtrará la información consultada en un ArrayList y convertirá la respuesta del servicio web de JSON a String.

```
private boolean filtrarDatos(){
    listaPersonas.clear();
    String data=mostrar();
    if(!data.equalsIgnoreCase("")){
        JSONObject json;
        try {
            json = new JSONObject(data);
            JSONArray jsonArray = json.optJSONArray("dataWS");
            for (int i = 0; i < jsonArray.length(); i++) {
                personas=new Personas();
                JSONObject jsonArrayChild = jsonArray.getJSONObject(i);
                personas.setId(jsonArrayChild.optString("id"));
                personas.setNombre(jsonArrayChild.optString("nombre"));
                personas.setLat(jsonArrayChild.optString("latitud"));
                personas.setLon(jsonArrayChild.optString("longitud"));

                listaPersonas.add(personas);
            }
        } catch (JSONException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
        return true;
    }
}
```

**Figura 6 Método Filtrar Datos**

8. Crear el método MostrarPersonas que es aquel que coloca la información obtenida en los TextView del layout.

```
private void mostrarPersona(final int posicion){
    runOnUiThread(new Runnable(){
        @Override
        public void run() {
            // TODO Auto-generated method stub
            Personas personas=listaPersonas.get(posicion);
            id.setText(personas.getId());
            nombre.setText(personas.getNombre());
        }
    });
}
```

**Figura 7 Método MostrarPersona**

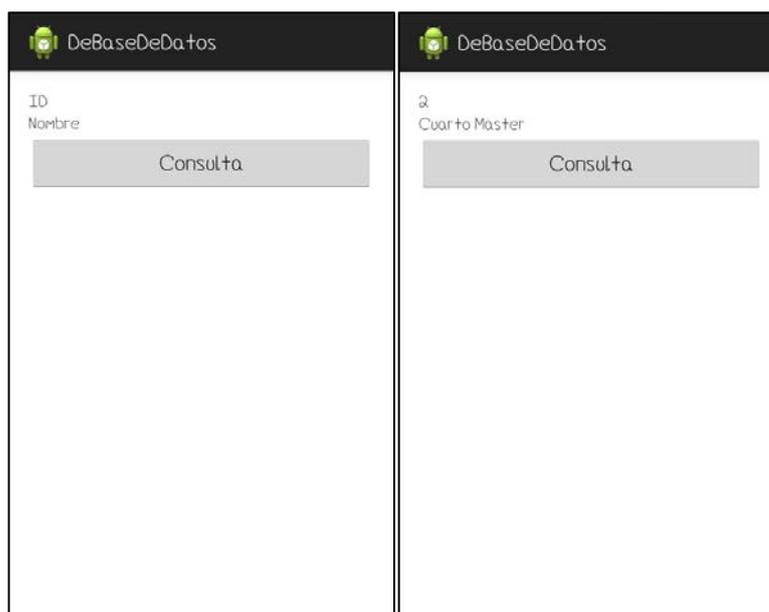
9. Finalmente el uso de la clase AsyncTask que obliga a trabajar a los métodos consulta de datos y filtro de información en uno de sus cuatro pasos, en este caso es en el doInBackground.

Como parámetro recibe una variable tipo String, en su proceso trabaja con datos tipo String y como resultado devuelve una variable String que en el método Mostrar se llama “resultado”.

```
class Mostrar extends AsyncTask<String,String,String>{
    @Override
    protected String doInBackground(String... params) {
        // TODO Auto-generated method stub
        if(filtrarDatos())mostrarPersona(posicion);
        return null;
    }
}
```

**Figura 8 Clase AsyncTask**

Los resultados de la aplicación creada son los siguientes:



**Figura 9 Resultados de Aplicación con AsyncTask**

## **ANEXO F**

### **CÓDIGO DE LA APLICACIÓN**

En el CD adjunto se puede encontrar la programación de la aplicación en la carpeta de nombre “Código de la Aplicación”, cada línea de código tiene su explicación.

La explicación de las líneas de código se encuentra en las clases tipo java y tipo java script, así también en los archivos XML pertenecientes a los layouts y al Android Manifest de la aplicación.

## **ANEXO G**

### **VIDEO TUTORIAL**

En el CD adjunto se puede encontrar en la carpeta de nombre “Video Tutorial”, un video en donde se muestra los pasos necesarios para crear una aplicación de realidad aumentada que realice consultas a una base de datos similar a “Quito Visión+”.