

**UNIVERSIDAD MAYOR DE SAN ANDRÉS**  
**FACULTAD DE CIENCIAS PURAS Y NATURALES**  
**CARRERA DE INFORMÁTICA**



**TESIS DE GRADO**

**“MODELO DE IDENTIFICACIÓN DEL PATRIMONIO  
CULTURAL Y TURÍSTICO  
MEDIANTE REALIDAD AUMENTADA”**

PARA OPTAR AL GRADO DE LICENCIATURA EN INFORMÁTICA  
MENCIÓN: INGENIERÍA DE SISTEMAS INFORMÁTICOS

**POSTULANTE:** WILSON RENE COSSIO MENDOZA  
**TUTORA METODOLÓGICA:** M. SC. FÁTIMA CONSUELO DOLZ DE MORENO  
**REVISOR:** LIC. JAVIER REYES PACHECO

**LA PAZ – BOLIVIA**

**2013**



**UNIVERSIDAD MAYOR DE SAN ANDRÉS  
FACULTAD DE CIENCIAS PURAS Y NATURALES  
CARRERA DE INFORMÁTICA**



**LA CARRERA DE INFORMÁTICA DE LA FACULTAD DE CIENCIAS PURAS Y NATURALES PERTENECIENTE A LA UNIVERSIDAD MAYOR DE SAN ANDRÉS AUTORIZA EL USO DE LA INFORMACIÓN CONTENIDA EN ESTE DOCUMENTO SI LOS PROPÓSITOS SON ESTRICTAMENTE ACADÉMICOS.**

**LICENCIA DE USO**

El usuario está autorizado a:

- a) visualizar el documento mediante el uso de un ordenador o dispositivo móvil.
- b) copiar, almacenar o imprimir si ha de ser de uso exclusivamente personal y privado.
- c) copiar textualmente parte(s) de su contenido mencionando la fuente y/o haciendo la referencia correspondiente respetando normas de redacción e investigación.

El usuario no puede publicar, distribuir o realizar emisión o exhibición alguna de este material, sin la autorización correspondiente.

**TODOS LOS DERECHOS RESERVADOS. EL USO NO AUTORIZADO DE LOS CONTENIDOS PUBLICADOS EN ESTE SITIO DERIVARA EN EL INICIO DE ACCIONES LEGALES CONTEMPLADOS EN LA LEY DE DERECHOS DE AUTOR.**

*DEDICATORIA*

*El presente trabajo va  
dedicado con mucho cariño  
a Martha mi madre que  
siempre estuvo a mi lado y  
me apoyó  
incondicionalmente.*

*Gracias por todo lo que me  
brindaste.*

## **AGRADECIMIENTO**

A mi madre, porque creyó en mí y porque me sacó adelante, dándome ejemplos dignos de superación y entrega, porque en gran parte gracias a ella, hoy puedo ver alcanzada mi meta, ya que siempre estuvo impulsándome en los momentos más difíciles de mi carrera, y porque el orgullo que siente por mí, fue lo que me hizo ir hasta el final. Para ti por que admiro tu fortaleza y por lo que has hecho de mí.

A mi hermano, que con sus consejos me ha ayudado a afrontar los retos que se me han presentado a lo largo de mi vida.

Quisiera hacer extensiva mi gratitud a M. Sc. Fátima Consuelo Dolz de Moreno mi tutora de tesis, por su valiosa guía y asesoramiento en la realización de la misma.

Al Lic. Javier Reyes Pacheco mi revisor, por toda la colaboración brindada, en la elaboración de este proyecto.

A Israel, por su apoyo incondicional en el transcurso de mi carrera universitaria, por compartir momentos de alegría, tristeza y demostrarme que siempre podré contar con él.

Gracias a todos por haber fomentado en mí el deseo de superación y el anhelo de triunfo en la vida.

Mil palabras no bastarían para agradecerles su apoyo, su comprensión y sus consejos en los momentos difíciles.

## **Resumen**

Este Proyecto consiste en el desarrollo de una aplicación destinada a los visitantes de nuestro país, para poder brindarles ayuda e información acerca de las riquezas históricas, la variedad cultural y los atractivos turísticos de Bolivia.

La aplicación se ejecuta en todos los dispositivos móviles que operan sobre la plataforma Android; se emplea tecnologías de Realidad Aumentada proporcionadas por el SDK de Vuforia de Qualcomm.

Esta aplicación servirá como un nexo de unión entre soportes, a priori incompatibles como ser una hoja de papel impresa, una foto o un objeto, unidos con el propio dispositivo electrónico haciendo uso de Realidad Aumentada; que proporcionará al medio abstracto de contenidos multimedia e interacción a través de la pantalla del dispositivo móvil.

Para llevar a cabo este proyecto se estudian las diferentes herramientas que se necesitan para poder desarrollar una aplicación para la plataforma Android, así como adquirir los conocimientos teóricos necesarios a tal efecto.

Además se hace un estudio sobre cuál es el estado actual de la Realidad Aumentada y de las herramientas capaces de proporcionarla. Se utilizará una de estas tecnologías, Vuforia, para dotar a la aplicación a desarrollar de capacidades de Realidad Aumentada y se explica con detalle cómo es el funcionamiento y arquitectura de esta tecnología.

Se estudiarán técnicas de modelado en tres dimensiones para incorporar Realidades Virtuales propias a la aplicación a desarrollar y se expone con detalle cómo ha sido el proceso de desarrollo de la aplicación, así como su funcionamiento.

## **Palabras clave**

Smartphone, Realidad Aumentada, Realidad Virtual, Android, Java, Vuforia, Rastreador, Marcador, Marcador de Imagen, detección, seguimiento, modelo 3D, OpenGL.

## **Abstract**

This project involves the development of an application for visitors to our country, to provide help and information about the historical richness, cultural variety and tourist attractions in Bolivia.

The application runs on all mobile devices that operate on the Android platform, and employs Augmented Reality technologies provided by Qualcomm Vuforia SDK.

This application or serve as a link between supports, a priori incompatible such as a printed sheet of paper, a photo or an object, together with the electronic device itself using Augmented Reality, which will provide the abstract medium and interaction of multimedia content through the mobile device screen.

To carry out this project the different tools needed to develop an application for the Android platform will be studied as well as acquire the knowledge necessary for this will be acquired.

In addition a study on what the current state of Augmented Reality and tools capable of providing it. One of these technologies will be used, Vuforia, to provide Augmented Reality capabilities to the application and explains in detail how is the operation and architecture of this technology.

Three dimensions of Modeling Techniques will be studied to incorporate own Virtual Realities develop the application to develop and described in detail the process of application development and operation.

## **Keywords**

Smartphone, Augmented Reality, Virtual Reality, Android, Java, Vuforia, Trackable, Target, Image Target, detection, tracking, 3D model, OpenGL

## INDICE

<b>CAPITULO I</b> .....	<b>1</b>
<b>MARCO REFERENCIAL</b> .....	<b>1</b>
<b>1.1 INTRODUCCIÓN</b> .....	<b>1</b>
<b>1.2 ANTECEDENTES</b> .....	<b>2</b>
<b>1.3 PROBLEMÁTICA</b> .....	<b>3</b>
<b>1.3.1 Problema Principal</b> .....	<b>3</b>
<b>1.3.2 Problemas Específicos</b> .....	<b>3</b>
<b>1.4 PLANTEAMIENTO DE LA HIPÓTESIS</b> .....	<b>4</b>
<b>1.4.1 Hipótesis</b> .....	<b>4</b>
<b>1.5 OBJETIVO</b> .....	<b>4</b>
<b>1.5.1 Objetivo General</b> .....	<b>4</b>
<b>1.5.2 Objetivos Específicos</b> .....	<b>5</b>
<b>1.6 JUSTIFICACIÓN</b> .....	<b>5</b>
<b>1.6.1 Económica.</b> .....	<b>5</b>
<b>1.6.2 Social.</b> .....	<b>6</b>
<b>1.6.3 Tecnológica.</b> .....	<b>6</b>
<b>1.7 ALCANCE</b> .....	<b>6</b>
<b>1.8 LIMITES</b> .....	<b>7</b>
<b>1.9 APORTE</b> .....	<b>7</b>
<b>1.10 METODOS Y MEDIOS DE INVESTIGACIÓN CIENTIFICA</b> .....	<b>8</b>
<b>1.11 METODOS Y MEDIOS DE INVESTIGACION INFORMATICA</b> .....	<b>8</b>
<b>CAPITULO II</b> .....	<b>9</b>
<b>MARCO TEORICO</b> .....	<b>9</b>
<b>2.1 PATRIMONIO TURISTICO</b> .....	<b>9</b>
<b>2.2 PATRIMONIO CULTURAL</b> .....	<b>9</b>
<b>2.2.1 Efectos Económicos y Sociales del Turismo Cultural</b> .....	<b>10</b>
<b>2.3 CARACTERÍSTICAS Y REQUERIMIENTOS DEL ENTORNO MOVIL</b> .....	<b>11</b>
<b>2.4 METODOLOGIA DE DESARROLLO DE SOFTWARE</b> .....	<b>13</b>
<b>2.5 METODOLOGÍAS AGILES</b> .....	<b>14</b>
<b>2.6.1 METODOLOGÍA XP</b> .....	<b>16</b>

<b>2.7 LENGUAJE UNIFICADO DE MODELADO (UML)</b>	<b>17</b>
<b>2.8 DISPOSITIVOS PORTATILES</b>	<b>18</b>
2.8.1. Smartphone	18
<b>2.9 SOFTWARE PARA DISPOSITIVOS MOVILES</b>	<b>19</b>
2.9.1 ANDROID	19
2.9.1.1 ARQUITECTURA	21
2.9.2 IOS APPLE	23
2.9.2.1 Programación en iOS	24
2.9.2.2 Arquitectura iOS	24
2.9.3 WINDOWS PHONE	25
2.9.3.1 ARQUITECTURA	26
2.9.3.2 HARDWARE	27
2.9.3.2.1 Modelo Windows Phone 7	27
<b>2.10 REALIDAD AUMENTADA</b>	<b>28</b>
2.10.1 Soluciones de Hardware	30
2.10.1.1 Display Montado Sobre la Cabeza del Usuario	30
2.10.1.2 Display de Mano	32
2.10.1.3 Display Espacial	34
2.10.2 Soluciones de Software	34
2.10.3 Aplicaciones	35
<b>2.11 PATRÓN DE ARQUITECTURA MODELO VISTA CONTROLADOR (MVC)</b>	<b>37</b>
2.11.1 Comunicación	39
2.11.2 Modelo pasivo	39
2.11.3 Unión del modelo con la vista y el controlador	40
<b>CAPITULO III</b>	<b>41</b>
<b>MARCO APLICATIVO</b>	<b>41</b>
<b>3.1 DISEÑO DE LA APLICACION</b>	<b>41</b>
3.1.1. Procedimiento	41
3.1.2 Datasets	46
<b>3.2 DIAGRAMA DE ARQUITECTURA</b>	<b>47</b>
<b>3.3 DIAGRAMA DE FLUJO DE DATOS</b>	<b>49</b>
<b>3.4 DIAGRAMA DE CASOS DE USO DEL SISTEMA</b>	<b>51</b>
3.4.1 CASO DE USO GENERAL	51
<b>3.5 DESCRIPCION DE LOS CASOS DE USO</b>	<b>52</b>
3.5.1 Caso de Uso General	52
3.5.2 Caso de Uso Reconocer Imagen	53
3.4.3 Caso de Uso Despliegue de Información	54
<b>3.6 DIAGRAMA DE SECUENCIA</b>	<b>57</b>
<b>3.7 DIAGRAMA DE COMPONENTES</b>	<b>58</b>

<b>3.6 PRUEBA DE HIPÓTESIS</b> -----	<b>59</b>
<b>CAPITULO IV</b> -----	<b>61</b>
<b>4.1 CONCLUSIONES</b> -----	<b>61</b>
<b>4.2 RECOMENDACIONES</b> -----	<b>61</b>
<b>BIBLIOGRAFIA</b> -----	<b>62</b>
<b>ANEXOS</b> -----	<b>65</b>
<b>A1 ARBOLDE PROBLEMAS</b> -----	<b>65</b>
<b>A2 ÁRBOL DE OBJETIVOS</b> -----	<b>66</b>
<b>A3 MARCO LÓGICO</b> -----	<b>67</b>

## ÍNDICE DE FIGURAS

<b>Figura 1.1:</b> Características del Desarrollo de Software Ágil.....	15
<b>Figura 1.2:</b> Características del SO Android .....	20
<b>Figura 1.3:</b> Características del SO Android continuación.....	21
<b>Figura 1.4:</b> Arquitectura en Capas del SO Android. ....	23
<b>Figura 1.5:</b> Capas de la arquitectura iOS.....	24
<b>Figura 1.6:</b> Modelos del software de Windows Phone.....	26
<b>Figura 1.7:</b> Características Windows Phone 7. ....	28
<b>Figura 1.8:</b> Realidad mixta .....	29
<b>Figura 1.9:</b> Ejemplo Display montado en la cabeza.....	31
<b>Figura 1.10:</b> Ejemplo display de mano .....	33
<b>Figura 1.11:</b> Distancia de reconocimiento de patrones de imágenes .....	37
<b>Figura 1.12:</b> Diagrama del Modelo Vista Controlador.....	38
<b>Figura 2.1:</b> Diagrama de Arquitectura.....	47
<b>Figura 2.3:</b> Diagrama de Casos de Uso general de la aplicación. ....	51
<b>Figura 2.4:</b> Flujo típico de acciones del usuario o respuesta del sistema.....	52
<b>Figura 2.5:</b> Flujo alternativo de acciones del usuario o respuesta del sistema.....	53
<b>Figura 2.6:</b> Flujo típico de acciones del usuario o respuesta del sistema.....	54
<b>Figura 2.8:</b> Diagrama de Secuencias del Sistema. ....	57
<b>Figura 2.9:</b> Diagrama de Componentes del Sistema. ....	58
<b>Figura 2.10:</b> Curva Cormal. ....	59
<b>Figura 2.11:</b> Distribución <i>t de student</i> . ....	60

## **CAPITULO I**

### **MARCO REFERENCIAL**

#### **1.1 INTRODUCCIÓN**

La informática constituye una de las más grandes revoluciones científicas y tecnológicas del mundo actual, la versatilidad del computador ha permitido su incursión en diferentes campos como la Medicina, Economía, Geografía, Turismo y otros; ahora ha irrumpido en los procesos tradicionales de exploración y difusión de la información creando expectativas y retos para su estudio en el aspecto turístico.

El incremento sin precedentes del número de turistas y la correspondiente tendencia hacia la progresiva masificación de este fenómeno, supone la urgente necesidad de abordar los problemas asociados a un turismo de calidad y a la protección del patrimonio cultural de un país, del uso excesivo o impropio. El gobierno nacional teniendo en cuenta este aspecto y reconociendo que es de interés colectivo para el país y la región ha elaborado una serie de afiches, folletos, pancartas, comerciales televisivos y portales web; sin embargo todo ello no es suficiente para promover y apropiar la riqueza cultural de la región, además que esta información no llega a ser objetiva, verás y atractiva para el visitante.

“En aras del interés general todas las diferentes formas de turismo usualmente deben adquirir, cada vez más, características de orden cultural, el pleno uso del patrimonio cultural, más que simple relación con una cosa o un evento, proporciona ocasión para adquirir nuevos conocimientos o para profundizar en los mismos, por esta razón, lejos de adquirir una mera fuente de ingresos, el turismo cultural es una ocasión de enriquecimiento espiritual.” [OMT, 2001]

Tomando en cuenta los factores del avance de la tecnología y su inserción en la vida cotidiana, es posible pensar en diseñar estrategias de difusión y entretenimiento que enriquezca la experiencia de los turistas que visitan la ciudad de

La Paz y sus alrededores; es así por lo que se plantea desarrollar un software para la identificación y reconocimiento del patrimonio cultural y turístico, mejorando la experiencia del visitante con contenidos multimedia, el cual pueda construir una herramienta tanto para los turistas nacionales y extranjeros, incluso para un guía turístico convencional. De esta manera se espera que el turista asuma un rol más activo en su visita a la ciudad a partir de una exploración interactiva de los sitios culturales y turísticos, mediada por el software el cual proporcionara información, fotos, videos, narraciones y toda la interactividad que la realidad aumentada puede brindar.

Las tecnologías de Realidad Aumentada permite una interacción transparente entre el mundo real y virtual, de modo que el participante puede ver el mundo real mientras que se superponen objetos 3D virtuales e información similar que aumenta la visión de la realidad.

## **1.2 ANTECEDENTES**

La tecnología de Realidad Aumentada actualmente es usada en un sin número de campos, como medicina, educación, y educación entre otros. También es usada en el sector de turismo, apuntando a mejorar la experiencia de los turistas. Por otro lado, muchos ejemplos han demostrado que la Realidad Aumentada puede ayudar a las empresas turísticas y a profesionales en turismo ya que pueden alcanzar una mayor audiencia ofreciendo la tecnología con contenido multimedia y aplicaciones móviles finamente diseñadas. Por otro lado los sistemas de información de Realidad Aumentada pueden ayudar a los turistas accedendo a información importante y mejorando su conocimiento con respecto a una atracción turística o cultural, mientras mejora la experiencia del visitante y proponiendo incrementar los niveles de entretenimiento a lo largo del proceso.

Otros tipos de proyectos y trabajos desarrollados en el área son:

- Tesis de Grado propuesta por Monje Zalles, José M. titulada “Red de internet como estrategia alternativa de Marketing promocional para el desarrollo del Turismo receptivo en La Paz”, propone utilizar la red para incrementar el turismo utilizando el marketing electrónico.
- Tesis de Grado realizada por Farfán Mariaca, Erika titulada “Formulación de una estrategia para el desarrollo del turismo” la tesista propone aplicar nuevas herramientas principalmente tecnológicas para fomentar el turismo nacional e internacional en la ciudad.
- Tesis de Grado propuesta por Mamani Castro, Andrés titulada “Inyección de Código en Aplicaciones Móviles”, propone las técnicas básicas para realizar el desarrollo de software en dispositivos móviles, utilizando metodologías ágiles.

## **1.3 PROBLEMÁTICA**

### **1.3.1 Problema Principal**

De qué manera se puede promover el patrimonio cultural y turístico brindando al visitante una experiencia dinámica, entretenida y enriquecedora.

### **1.3.2 Problemas Específicos**

- No existe un medio adecuado y fácil de usar con el que un visitante pueda obtener información acerca de los sitios turísticos de la ciudad.
- Los recorridos turísticos no son flexibles de acuerdo al criterio del turista.

- ✦ El costo de los servicios de un guía turístico muchas veces es elevado.
- ✦ La información ofrecida por el guía turístico a veces es inconsistente y tiene un alto grado de variabilidad de acuerdo a la persona que guía.
- ✦ No existe una promoción adecuada de los sitios turísticos de la ciudad.
- ✦ Se pierde la identidad de la cultura y el patrimonio en la ciudad.

## **1.4 PLANTEAMIENTO DE LA HIPÓTESIS**

### **1.4.1 Hipótesis**

El uso de la Realidad Aumentada basada en reconocimiento de imágenes como una herramienta de identificación de patrimonio cultural y turístico brindando información sostenible, consistente y oportuna permite promover la cultura y turismo.

## **1.5 OBJETIVO**

### **1.5.1 Objetivo General**

Desarrollar una aplicación informática para la identificación del patrimonio cultural y turístico (estructuras, ruinas, monumentos, pinturas) por medio de la realidad aumentada basada en reconocimiento de imágenes para que brinde información sostenible, consistente y oportuna.

### 1.5.2 Objetivos Específicos

- Desarrollar software turístico con el uso de herramientas multimedia, principalmente la aplicación de Realidad Aumentada.
- Desarrollar un modulo de reconocimiento de imágenes.
- Desarrollar un BD con una cantidad mínima de imágenes a ser identificadas (trajes típicos, monumentos, ruinas representativas).
- Desarrollar un modulo de acceso a base de datos.
- Implementar una base de datos con la información de los sitios turísticos y culturales de la ciudad.
- Desarrollar una interfaz sencilla y amigable.
- Validar el prototipo y proponerlo como medio de difusión y apropiación del patrimonio cultural y turístico.

## 1.6 JUSTIFICACIÓN

### 1.6.1 Económica.

En nuestro país no se invierte debida y oportunamente en el área de turismo, si bien existe un presupuesto que el estado predispone para la elaboración de panfletos, pancartas y spots publicitarios estos no llegan hasta el turista de forma oportuna y concisa, además de que la contratación de los servicios de un guía profesional tienen un costo elevado, en base a ello el sistema mitigará estas deficiencias brindando una herramienta amigable y dinámica con un costo reducido.

### **1.6.2 Social.**

El presente proyecto se justifica socialmente debido a que ayudara a todos los turistas tanto extranjeros como nacionales a conocer e informarse acerca de la historia, costumbres y cultura, de la región, para poder formar un sentido de conciencia y apropiación del patrimonio turístico y cultural.

### **1.6.3 Tecnológica.**

Debido a los grandes avances tecnológicos y el desarrollo de nuevos medios para la difusión y desarrollo del turismo el prototipo de localización de sitios turísticos mediante realidad aumentada, permitirá al visitante el uso de una herramienta dinámica y amigable que le ayude en su recorrido por los diferentes sitios turísticos.

Mediante la Realidad Aumentada se permitirá una mejor comprensión de la cultura de la región, permitiendo una interacción dinámica entre el entorno y el turista, presentando gran contenido multimedia, como ser imagen narraciones y videos, por este hecho la justificación tecnológica del presente trabajo se basa en el uso de un medio informático (hardware y software) que interactúa con el usuario por medio de un dispositivo móvil.

## **1.7 ALCANCE**

Al tratarse de un prototipo el presente prototipo tendrá como objeto de estudio y referencia los museos de la ciudad de La Paz, debido a su gran riqueza turística, arqueológica y cultural. El prototipo logrará reconocer las pinturas, objetos

y fotografías de uno de los museos ubicados en la ciudad, para poder brindar al visitante una experiencia enriquecedora y dinámica brindando información multimedia.

## **1.8 LIMITES**

El sistema de información será realizado para el reconocimiento de los objetos situados en un museo de la ciudad de La Paz, la aplicación en una primera instancia reconocerá las obras de arte visuales como ser cuadros, pinturas, fotografías, panfletos y banners; todos estos deberán tener una adecuada iluminación para que el dispositivo pueda identificarlo.

## **1.9 APORTE**

El presente proyecto de tesis está orientado a la promoción y apropiación del patrimonio turístico y cultural del departamento de La Paz, mediante una herramienta entretenida e innovadora como es la Realidad Aumentada, emparejándola con el reconocimiento de imágenes y la tecnología multimedia.

La aplicación inicialmente permitirá reforzar el conocimiento del turista acerca de un sitio turístico en particular, en este caso las pinturas y fotografías de un museo de la ciudad de La Paz, posteriormente se podrán agregar más ubicaciones y contenido, acerca de sitios turísticos y culturales, asimismo se tiene previsto que el sistema ayude a rescatar el patrimonio cultural de la ciudad pudiendo reconocer la vestimenta típica de las danzas de la región, de esta manera fortalecer la cultura de nuestro país y difundir el patrimonio turístico y cultural.

## 1.10 METODOS Y MEDIOS DE INVESTIGACIÓN CIENTIFICA

En cuanto a los métodos involucrados para el desarrollo, mencionaremos:

**Método Analógico-Sintáctico:** Se contempla la necesidad de realizar un análisis y síntesis en varias etapas del proyecto.

**Método Inductivo-Deductivo:** La inducción viene siendo la única forma de conseguir la deducción o deducciones propuestas en los objetivos.

**Método Abstracción-Concreción:** Necesitamos tener una visión exacta del trabajo de investigación.

## 1.11 METODOS Y MEDIOS DE INVESTIGACION INFORMATICA

Las herramientas involucradas en el trabajo son las siguientes:

- ✓ Lenguaje de programación Java Jdk1.6.0.
- ✓ IDE Eclipse.
- ✓ Kit de Desarrollo de Software de Android (Android SDK).
- ✓ Kit de Desarrollo Nativo (Native Development kit NDK).
- ✓ Librería de código abierto Open CV.
- ✓ Motor de base de datos de código abierto SQL Lite.

## **CAPITULO II**

### **MARCO TEORICO**

#### **2.1 PATRIMONIO TURISTICO**

El patrimonio puede significar un recurso estratégico para el desarrollo socioeconómico de las poblaciones, dado que en muchos casos representa el componente básico y fundamental del turismo, que dinamiza y sostiene medianos y pequeños emprendimientos.

El término “patrimonio” contiene elementos tanto naturales como culturales que podrían resumirse de la siguiente manera: por un lado, la características del ambiente natural, los ecosistemas, la biodiversidad existente y el entorno paisajístico; por el otro, la historia, las tradiciones, las festividades, la gastronomía, las costumbres y creencias populares, arquitectura, monumentos y yacimientos arqueológicos, así como cualquier otra manifestación de la cultura propia del lugar, ya sea tanto en aspectos tangibles como intangibles. [Borrega, 2009]

#### **2.2 PATRIMONIO CULTURAL**

El concepto de patrimonio cultural es subjetivo y dinámico, no depende de los objetos o bienes sino de los valores que la sociedad en general les atribuyen en cada momento de la historia y que determinan qué bienes son los que hay que proteger y conservar para la posteridad.

La visión restringida, singular, antigua, monumental y artística del patrimonio del siglo XIX será superada durante el siglo XX con la incorporación del concepto de valor cultural.

Hoy son varios los documentos internacionales que consolidan una visión amplia y plural del patrimonio cultural, que valoran todas aquellas entidades materiales e inmateriales significativas y testimoniales de las distintas culturas, sin establecer límites temporales ni artísticos, considerando así las entidades de carácter tradicional, industrial, inmaterial, contemporáneo, subacuático o los paisajes culturales como garantes de un importante valor patrimonial. [OMT, 2009]

### **2.2.1 Efectos Económicos y Sociales del Turismo Cultural**

Una contribución manifiesta se da cuando el desarrollo del turismo está encaminado a establecer mecanismos que estén enfocados al afianzamiento de un mercado receptivo con una serie de medidas encaminadas a cubrir y desarrollar adecuadamente la puesta en valor del patrimonio cultural, evitando en lo posible los impactos negativos en la comunidad.

En este sentido, se logra la máxima satisfacción del visitante pero también un aceptable nivel de beneficio para el destino. Un beneficio, que no sólo es económico, sino también es social y cultural, que debe orientarse a partir de políticas específicas a la conservación de los bienes culturales, controlando el uso de esos bienes como atractivos turísticos.

El turismo como hecho social, económico y cultural, genera fenómenos socioeconómicos, culturales y jurídicos, reconocidos por organismos y congresos tanto nacionales como internacionales, tanto oficiales como privados, que son aspectos sobresalientes del desarrollo y de los mercados turísticos, con los objetivos particulares que son la preservación y puesta en valor de los bienes histórico-culturales como parte esencial del patrimonio turístico, con la puesta en valor del ser humano y su

obra, por lo que la participación de la sociedad como comunidad receptora debe ser su sustento básico. [Borrega, 2009]

No hay que olvidar que el conocer otras culturas y formas de vida no son sólo motivaciones del turista extranjero, sino que hay un turismo nacional o doméstico que también debe ser considerado y desarrollado adecuadamente. La corriente de visitantes tanto nacionales como extranjeros motivada adecuadamente, va a contribuir a valorizar los bienes culturales insertos en un circuito económico nacional. Así, el patrimonio cultural debe ser entendido en un doble sentido cultural y económico, al estar conformado por el conjunto de bienes materiales, muebles e inmuebles, e inmateriales, donde estos bienes deben ser objeto de un adecuado aprovechamiento económico, social y cultural, sin desvirtuar su esencia y autenticidad. Es decir, no deben ser objeto de mistificaciones y/o adulteraciones que atenten contra su valor bajo el pretexto del turismo, que el turismo esté al servicio de la cultura y no a la inversa. Es lo que se ha denominado “La Dysneificación de la Realidad”. Una característica del turismo cultural es que es de muy baja permanencia, pero ofrece una gran elasticidad al permitir combinarlo con otros tipos de turismo, permitiendo que las actividades turísticas enfocadas al patrimonio sean muy rentables. [Borrega, 2009]

### **2.3 CARACTERÍSTICAS Y REQUERIMIENTOS DEL ENTORNO MOVIL**

El desarrollo de aplicaciones móviles difiere del desarrollo de software tradicional en muchos aspectos, lo que provoca que las metodologías usadas para estos entornos también difieran de las del software clásico.

El desarrollo de software móvil tienen que satisfacer una serie de requerimientos y condicionantes especiales como las siguientes:

**Canal de radio:** consideraciones tales como la disponibilidad, las desconexiones, la variabilidad del ancho de banda, la heterogeneidad de redes o los riesgos de seguridad han de tenerse especialmente en cuenta en este entorno de comunicaciones móviles.

**Movilidad:** aquí influyen consideraciones como la migración de direcciones, alta latencia debido a cambio de estación base o la gestión de la información dependiente de la localización. Sobre esta última, de hecho, se un sinnúmero de aplicaciones, pero la información de contexto asociada resulta muchas veces incompleta y varía frecuentemente.

**Portabilidad:** la característica de portabilidad de los dispositivos terminales implica una serie de limitaciones físicas directamente relacionadas con el factor de forma de los mismo, como el tamaño de las pantallas (algo que ha variado sustancialmente con la popularización de las pantallas táctiles) o del teclado, limitando también el número de teclas y su disposición.

**Fragmentación de la industria:** la existencia de una considerable variedad de protocolos de red diferentes añaden complejidad al escenario de desarrollo móvil.

**Capacidades limitadas de terminales:** aquí debemos incluir factores como la baja potencia de cálculo o gráfica, los riesgos en la integridad de datos, las interfaces de usuario poco funcionales en muchos aspectos, la baja capacidad de almacenamiento, la duración de las baterías o la dificultad para el uso de periféricos en movilidad. Factores todos que, por otro lado, están evolucionando en la dirección de la convergencia de los ultra-portátiles (netbooks o tablets) con los dispositivos inteligentes constituyendo cada vez menos un elemento diferencial.

**Diseño:** desde el punto de vista del desarrollo, el diseño multitarea y la interrupción de tareas es la clave para el éxito de las aplicaciones de escritorio; pero la oportunidad y frecuencia de estas es mucho mayor que el software tradicional, debido al entorno móvil que manejan, complicándose todavía más debido a la limitación de estos dispositivos.

**Usabilidad:** las necesidades específicas de amplios y variados grupos de usuarios, combinados con la diversidad de plataformas tecnológicas y dispositivos, hacen que el diseño se convierta para todos en un requisito que genera una complejidad creciente difícil de acotar.

**Time-to-market:** en un sector con un dinamismo propio, dentro de una industria en pleno cambio, los requisitos que se imponen en términos de tiempo de lanzamiento son muy estrictos y añaden dificultad en la gestión de los procesos de desarrollo.

El diseño de software para dispositivos móviles es más complejo que el tradicional para ordenadores de escritorio, es por eso que los desarrolladores deben reconsiderar el uso de las metodologías actuales de desarrollo de software. El uso de metodologías ágiles de desarrollo, es el medio más apropiado para el desarrollo de tecnología en móviles, aunque las características especiales de los terminales y de las redes de telefonía móviles demandan algunos ajustes sobre las actuales metodologías ágiles. [Campero, 2010]

## **2.4 METODOLOGIA DE DESARROLLO DE SOFTWARE**

Al llevar un proyecto como la realización de software, es necesario adoptar una metodología; muchas veces el éxito de una aplicación depende del orden en el que se realizan las cosas.

La mayoría de los líderes de proyecto realiza un plan de trabajo el cual consiste en dividir el sistema en procesos y éstos asignarlos a los desarrolladores, la implementación de una metodología que se adecue a las necesidades del equipo de trabajo disminuye mucho los problemas y evita confusión el lo que el cliente quiere y lo que los desarrolladores realizan. [Campero, 2010]

## 2.5 METODOLOGÍAS AGILES

Tras una reunión celebrada en Utah en Febrero de 2001, nace el término “ágil” aplicado al desarrollo de software. El objetivo fue esbozar los valores y principios que debería permitir a los equipos desarrollar software rápidamente y responder a los cambios que pueden surgir a lo largo del proyecto. Se creó The Agile Alliance dedicada a promover el desarrollo ágil de software y ayudar a las empresas que lo adoptaran. El punto de partida fue el Manifiesto Ágil, documento que resume esa filosofía y que expone cuatro valores a tener en cuenta.

El individuo y las interacciones del equipo de desarrollo están por encima del proceso y las herramientas. Construir un buen equipo y que esta configure su propio entorno de desarrollo en base a sus necesidades.

Desarrollar software que funciona, más que conseguir buena documentación. No producir documentos a menos que sean necesarios de una forma inmediata. Si el software no funciona, los documentos no valen nada.

La colaboración con el cliente es más importante que la negociación de contratos. Tiene que haber una interacción constante entre el cliente y el equipo de desarrollo.

La respuesta ante el cambio es más importante que el seguimiento de un plan. La planificación no deben ser estricta en cambio debe ser flexible y abierta, la fiabilidad de responder a los cambios que surjan determina el éxito o fracaso del proyecto.

En definitiva el desarrollo ágil de software intenta evitar los tortuosos y burocráticos caminos de las metodologías tradicionales, enfocándose en las personas y los resultados. [Campero, 2010]

Aunque existe una variedad de metodologías ágiles, casi ninguna se ha centrado en los requerimientos tan específicos que el desarrollo móvil necesita. Como se puede observar en la tabla 2.1, las metodologías ágiles poseen ciertas propiedades que las hacen totalmente aplicables al dominio del software en

dispositivos móviles. En la reunión celebrada en Utah, se identificaron los métodos ágiles con la solución potencial para el desarrollo de software en móviles.

Características ágiles	Motivación lógica	En el caso de desarrollo para plataformas móviles
Alta volatilidad del entorno	Debido a la alta frecuencia en el cambio que sufren los requerimientos, tendremos menos necesidad de diseño y planificación inicial, en cambio mayor necesidad de desarrollos incrementales e iterativos.	Alta incertidumbre, entornos dinámicos, cientos de nuevos terminales cada año.
Equipos de desarrollo pequeños	Capacidad de reacción más rápida trabajo basado en la compartición de la información, menos documentación.	La mayor parte de los proyectos de desarrollo de software para plataformas móviles se lleva a cabo en microempresas y PyME.
Ciente Identificable	Desaparecen los malentendidos	Potencialmente, hay un número ilimitado de usuarios finales, pero los clientes son fáciles de identificar.
Entornos de desarrollo orientados a objetos	La mayoría de las herramientas de desarrollo ágil existen bajo plataformas orientadas a objetos.	Por ejemplo Java y C++, se usan algunos problemas en herramientas como refactorización o primeros test.
Software crítico no asegurado	Los fallos no causan gran impacto, como la pérdida de vidas. Se puede buscar mayor agilidad en el desarrollo.	La mayoría del software es para entretenimiento. Los terminales no son fiables del todo.
Software a nivel de aplicación	Sistemas embebidos grandes requieren comunicación exhaustiva y mecanismos de verificación.	Mientras que los sistemas móviles son complejos y altamente dependientes, las aplicaciones son muy autónomas.
Sistemas pequeños	Menos necesidad de diseño inicial	Las aplicaciones, aunque variables en tamaño no suelen superar las 10.000 líneas de código.
Ciclos de desarrollo cortos	Propósito de realimentación rápida.	Periodos de desarrollo de 1 a 6 meses.

**Figura 1.1:** Características del Desarrollo de Software Ágil.

**Fuente:** [Campero, 2010]

### 2.6.1 METODOLOGÍA XP

Es una metodología ágil de desarrollo de software, esta metodología trata de dar al cliente el software que el necesita, por tanto se debe responder muy rápido a las necesidades del cliente.

Básicamente las cuatro actividades por las que está compuesta la metodología son las siguientes:

**Codificar:** Es imprescindible plasmar las ideas a través del código, se debe codificar constantemente.

**Realizar pruebas:** Las pruebas dan la oportunidad de saber si lo implementado es lo que realmente se buscaba, las pruebas son referentes para saber si el sistema tiene la funcionalidad correcta. También es eficiente programar y realizar pruebas ya que se tendrá un panorama más completo de cómo va el proceso de desarrollo. Las pruebas deben contemplar la mayor cantidad de casos posibles para que no existan problemas sin resolver.

**Escuchar:** Se debe escuchar las necesidades del cliente o de alguna fuente que exprese que problemas debería resolver el sistema o en qué puntos debería reforzarse la solución.

**Diseñar:** El diseño crea una estructura que organiza la lógica del sistema, un buen diseño permite que el sistema sea adaptable a cambios y actualizaciones. [Campero, 2010]

## 2.7 LENGUAJE UNIFICADO DE MODELADO (UML)

Es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el OMG (Object Management Group). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un “plano” del sistema, incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y compuestos reciclados.

UML se puede usar para modelar distintos tipos de sistemas: sistemas de software, sistemas de hardware y organizaciones del mundo real. UML Ofrece nueve diagramas para modelar los sistemas.

**Diagramas de casos de uso:** para modelar los procesos del negocio.

**Diagramas de Secuencia:** para modelar el paso de mensajes entre objetos

**Diagramas de Colaboración:** Para modelar interacciones entre objetos.

**Diagramas de Estado:** para modelar el comportamiento de los objetos en el sistema.

**Diagramas de Actividad:** para modelar el comportamiento de los casos de uso, objetos u operaciones.

**Diagramas de clases:** para modelar la estructura estática de las clases del sistema.

**Diagramas de objetos:** para modelar la estructura estática de los objetos en el sistema.

**Diagramas de Componentes:** para modelar componentes.

**Diagramas de Implementación:** para modelar la distribución del sistema.

## 2.8 DISPOSITIVOS PORTATILES

### 2.8.1. Smartphone

El término Smartphone hace referencia a “teléfono móvil inteligente”; el hecho de llamarlo “inteligente” es porque esta clase de dispositivos ofrecen más funciones que los móviles tradicionales. En la actualidad, para que un teléfono móvil se considere “inteligente” ha de disponer de una serie de características entre las que destacan:

-  Permiten la instalación de aplicaciones.
-  Deben contar con algún sistema operativo.
-  Cuentan con GPS.
-  Permiten el acceso a internet.
-  Soporte a sistemas de correo electrónico.
-  Deben disponer de cámara digital.
-  Funciones de agenda digital, gestión de contactos, calendarios, etc.
-  Lectura de documentos de distinto formato.
-  Reproducción de audio y vídeo.

Todas estas características implican que este tipo de dispositivos tengan una gran capacidad de cómputo, en forma de potentes procesadores, memoria RAM, almacenamiento, etc.; que además permitan el uso simultáneo de varias de estas funciones.

Sobre su diseño, es habitual que los Smartphone posean un tamaño significativamente mayor al de un móvil tradicional, esto se debe a la necesidad de incorporar ciertas características especiales como pantallas

táctiles de grandes dimensiones. De todo lo anterior se concluye que, las características de los teléfonos inteligentes los convierten en dispositivos potenciales para proveer funciones de Realidad Aumentada; en concreto son “Displays de mano” sin el software necesario para proporcionar Realidad Aumentada. En la actualidad, estos dispositivos se han convertido en los dispositivos de Realidad Aumentada por antonomasia, gracias a las aplicaciones que proporcionan funciones de Realidad Aumentada, que además han sido las causantes de acercar este mundo a una gran cantidad de público. [Villarán, 2012]

## **2.9 SOFTWARE PARA DISPOSITIVOS MOVILES**

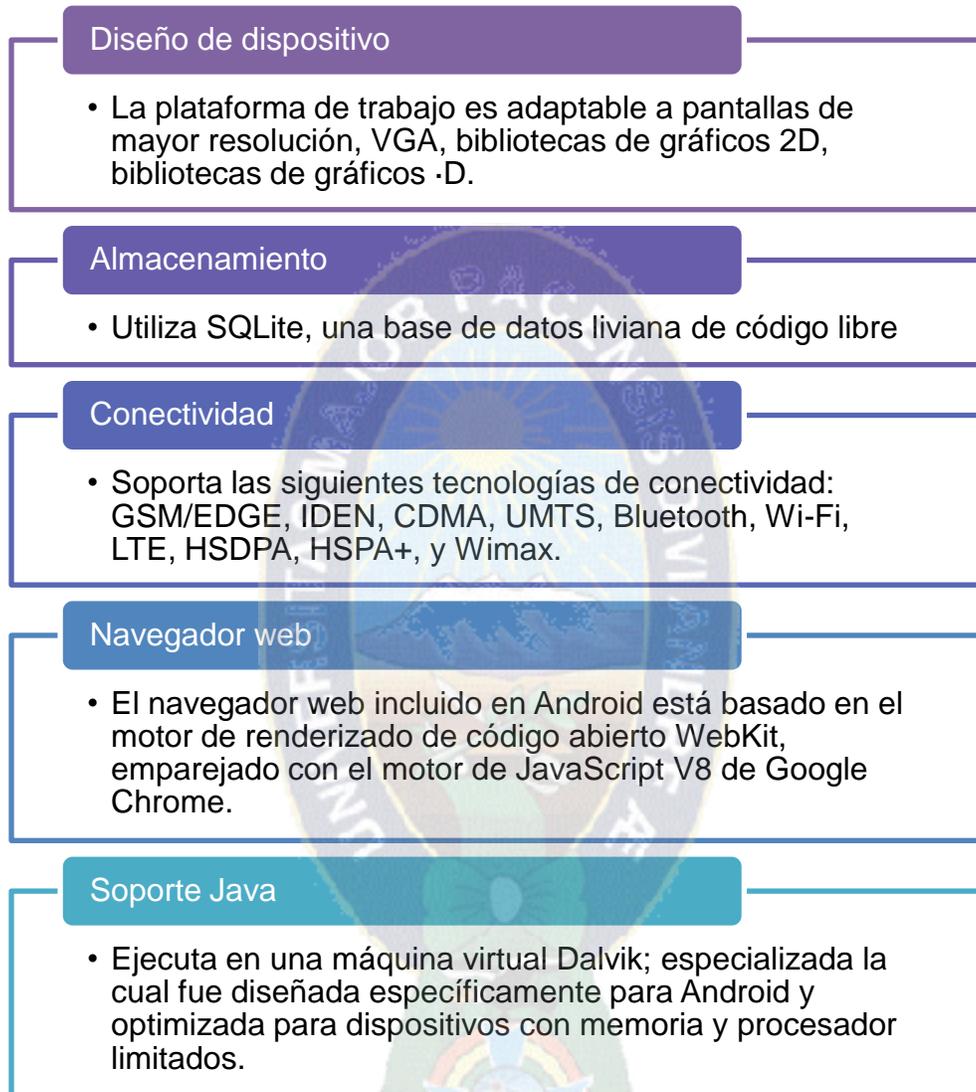
### **2.9.1 ANDROID**

Es un sistema operativo basado en Linux, principalmente para dispositivos portátiles con pantalla táctil como ser teléfonos inteligentes, tabletas y otros dispositivos.

Android es código abierto y Google libera el código bajo la Licencia de Apache. Esta característica de ser código abierto y una licencia permisiva permite al software ser libremente modificado y distribuido por los fabricantes de dispositivos y desarrolladores de software.

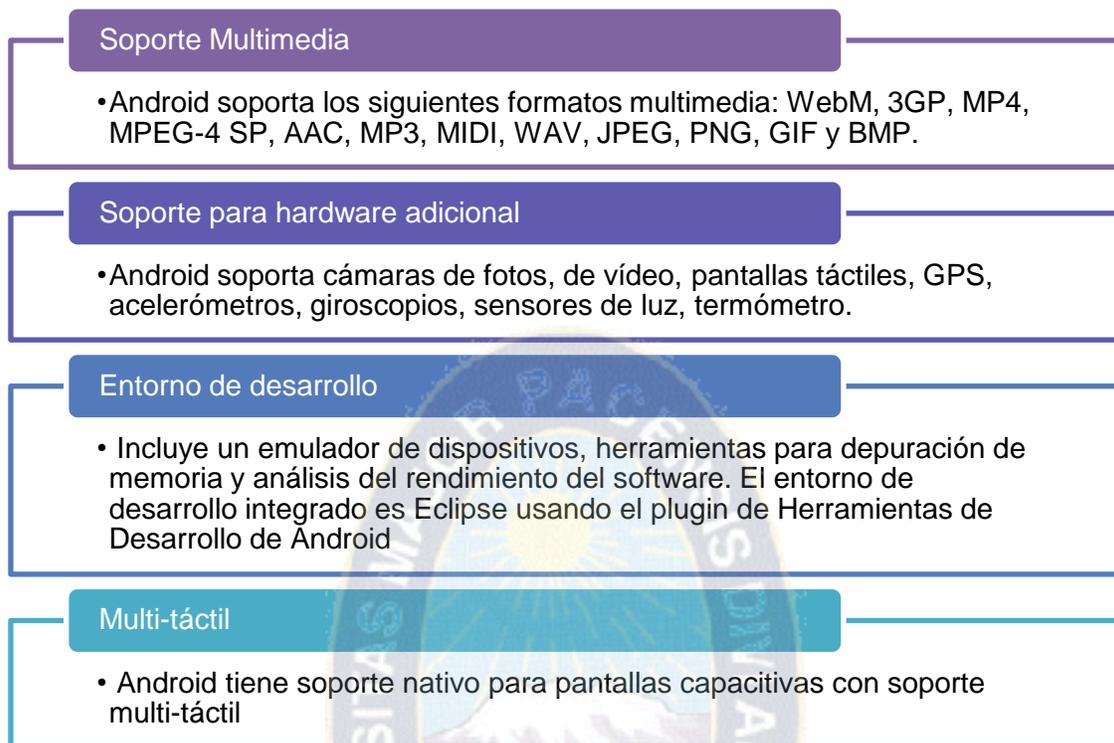
Estos factores han permitido a Android convertirse en la plataforma para dispositivos móviles más usada mundialmente, sobreponiéndose a Symbian en el último trimestre de 2010. En un principio fue diseñado principalmente para teléfonos inteligentes, pero tiene aplicaciones adicionales en televisores, consolas de juego, cámaras digitales, tabletas, relojes y otros dispositivos electrónicos. [Maier, 2012]

Las principales características del SO son:



**Figura 1.2:** Características del SO Android

**Fuente:** [Maier, 2012]



**Figura 1.3:** Características del SO Android continuación.

**Fuente:** [Maier, 2012]

### 2.9.1.1 ARQUITECTURA

Los componentes principales del sistema operativo de Android son:

**Aplicaciones:** las aplicaciones base incluyen un cliente de correo electrónico, programa de SMS, calendario, mapas, navegador, contactos y otros. Todas las aplicaciones están escritas en lenguaje de programación Java.

**Marco de trabajo de aplicaciones:** los desarrolladores tienen acceso completo a los mismos API's del framework usados por las aplicaciones base. La arquitectura está diseñada para simplificar la reutilización de

componentes; cualquier aplicación puede publicar sus capacidades y cualquier otra aplicación puede luego hacer uso de esas capacidades (sujeto a reglas de seguridad del framework). Este mismo mecanismo permite que los componentes sean reemplazados por el usuario.

**Bibliotecas:** Android incluye un conjunto de bibliotecas de C/C++ usadas por varios componentes del sistema. Estas características se exponen a los desarrolladores a través del marco de trabajo de aplicaciones de Android; algunas son: System C library (implementación biblioteca C estándar), bibliotecas de medios, bibliotecas de gráficos, 3D y SQLite, entre otras.

**Runtime de Android:** Android incluye un set de bibliotecas base que proporcionan la mayor parte de las funciones disponibles en las bibliotecas base del lenguaje Java. Cada aplicación Android corre su propio proceso, con su propia instancia de la máquina virtual Dalvik. Dalvik ha sido escrito de forma que un dispositivo puede correr múltiples máquinas virtuales de forma eficiente. Dalvik ejecuta archivos en el formato Dalvik Executable (.dex), el cual está optimizado para memoria mínima. La Máquina Virtual está basada en registros y corre clases compiladas por el compilador de Java que han sido transformadas al formato.dex por la herramienta incluida "DX".

**Núcleo Linux:** Android depende de Linux para los servicios base del sistema como seguridad, gestión de memoria, gestión de procesos, pila de red y modelo de controladores. El núcleo también actúa como una capa de abstracción entre el hardware y el resto de la pila de software. [Maier, 2012]



**Figura 1.4:** Arquitectura en Capas del SO Android.

**Fuente:** [Maier, 2012]

## 2.9.2 IOS APPLE

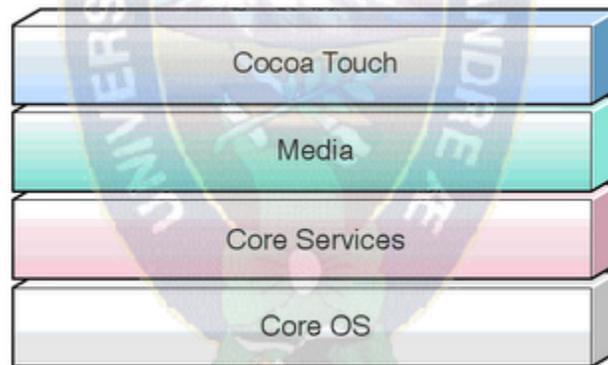
IOS es un sistema operativo desarrollado por Apple Inc. para los dispositivos móviles iPod touch, iPhone e iPad. Está basado en una variante del Mach kernel de Mac OS X. Inicialmente, Apple desarrolló este software para instalarlo en su dispositivo móvil Iphone. El sistema operativo de Iphone, desde su nacimiento en 2007, ha destacado por su uso multitáctil de la pantalla (deslizando los dedos o pellizcando la pantalla), por su diseño funcional e intuitivo del menú de tareas, así como por la gran variedad de aplicaciones que ofrece como el correo electrónico, el navegador Web Safari, el reproductor multimedia, un gestor de contactos, etc. [Apple, 2012]

### 2.9.2.1 Programación en iOS

iPhone comparte base con Mac OS X, la cadena de instrumentos para desarrollar aplicaciones para iPhone está también basada en Xcode e incluye compiladores cruzados para el procesador ARM y un emulador de iPhone llamado Aspen. El lenguaje de programación principal para iPhone OS, al igual que en Mac OS, es Objective-C. [Pastor, 2012]

### 2.9.2.2 Arquitectura iOS

La arquitectura iOS está basada en capas, donde las capas más altas contienen los servicios y tecnologías más importantes para el desarrollo de aplicaciones, y las capas más bajas controlan los servicios básicos.



**Figura 1.5:** Capas de la arquitectura iOS.

**Fuente:** [Pastor, 2012]

**Cocoa Touch:** Cocoa Touch es la capa más importante para el desarrollo de aplicaciones iOS. Posee un conjunto de Framework que proporciona el API de Cocoa para desarrollar aplicaciones. Se podría decir que Cocoa Touch proviene de Cocoa, la API ya existente en la plataforma MAC.

Esta capa está formada por dos Framework fundamentales:

**UIKit:** contiene todas las clases que se necesitan para el desarrollo de una interfaz de usuario

**Foundation Framework:** define las clases básicas, acceso y manejo de objetos, servicios del sistema operativo

**Media:** Provee los servicios de gráficos y multimedia a la capa superior.

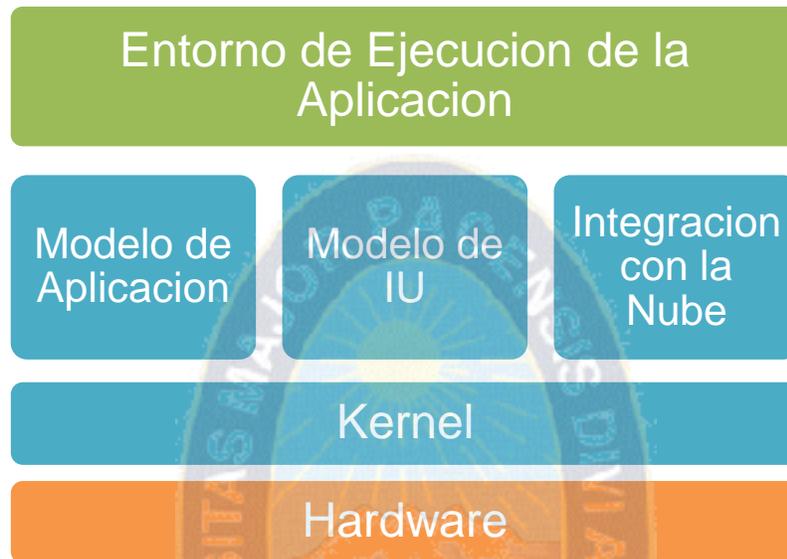
**Core Services:** Contiene los servicios fundamentales del sistema que usan todas las aplicaciones

**Core OS:** Contiene las características de bajo nivel: ficheros del sistema, manejo de memoria, seguridad, drivers del dispositivo. [Pastor, 2012]

### 2.9.3 WINDOWS PHONE

Debemos saber que Microsoft ya se había enfocado en llevar su propio sistema operativo a los teléfonos, entonces es así como surge Windows Mobile, pero uno de los grandes problemas a los que Microsoft debió hacer frente fue la fragmentación de Hardware y Software existente en el ecosistema, es decir a la diversidad de modelos que había en el mercado, aprendiendo de sus errores, en Windows Phone 7 nos encontramos un modelo de Hardware y Software claro y conciso que ayudara a evitar la repetición de esta situación.

### 2.9.3.1 ARQUITECTURA



**Figura 1.6:** Modelos del software de Windows Phone.

**Fuente:** [Gómez & Ortega, 2012]

Vamos a clasificar el software que funciona en Windows Phone 7 en 3 modelos:

**Modelo de Aplicación.** En Windows Phone 7 las aplicaciones se despliegan en forma de paquete XAP, básicamente es un archivo dentro del cual podemos encontrar los ensamblados y recursos originales de nuestra aplicación.

**Modelo de UI.** El modelo de interface de usuario de Windows Phone 7 se compone de elementos y una sesión es el conjunto de interacciones que

realiza un usuario sobre nuestra aplicación e incluso puede involucrar a otras aplicaciones.

**Integración con la Nube.** Windows Phone 7 nace con una clara integración con la nube. Por defecto tenemos integración con servicios como Exchange, Google Mail, Hotmail, Xbox Live, Skydrive, Facebook o Bing. En la versión actual del kit de desarrollo no existen API's que permitan a nuestras aplicaciones acceder a estos servicios directamente, pero se espera que aparezcan en próximas actualizaciones. [Gómez & Ortega, 2012]

### 2.9.3.2 HARDWARE

Hasta la llegada de Windows Phone 7 al mercado podíamos encontrar 2 modelos, o planteamientos distintos a nivel de hardware en los principales competidores (Modelo Iphone y Modelo Windows Mobile/Android).

#### 2.9.3.2.1 Modelo Windows Phone 7

Microsoft como fabricante del sistema, requiere que todo teléfono que desee ejecutar Windows Phone 7 disponga de unas características mínimas, para asegurar la consistencia de todos los usuarios del sistema, con este modelo todos los usuarios obtienen la misma experiencia de uso y los desarrolladores sabemos que nuestra aplicación funcionara de forma idéntica en todos los dispositivos que tengan Windows Phone 7, y no estamos encerrados en un solo hardware, existen distintos dispositivos de varios fabricantes, todos con unas características mínimas comunes.

Con este modelo en mente, Microsoft creo unas especificaciones mínimas iniciales, llamadas **Chassis 1**, por las que todo fabricante que desee crear terminales para el nuevo sistema debe guiarse.



**Procesador:** ARMv7 Cortex/Scorpion a 1Ghz.  
**Procesador gráfico:** Soporte hardware completo de DirectX9  
**Memoria:** 256Mb RAM / 8Gb ROM.  
**Sensores:** A-GPS, Acelerómetro, brújula, iluminación, proximidad.  
**Cámara:** 5mpx con flash y botón físico de disparo.  
**Multimedia:** Aceleración de audio y video por hardware.  
**Pantalla:** Capacitiva, resolución: 800x480.  
**Botones Físicos:** Inicio, Buscar, Atrás

**Figura 1.7:** Características Windows Phone 7.

**Fuente:** [Gómez & Ortega, 2012]

## 2.10 REALIDAD AUMENTADA

En la actualidad se aceptan dos definiciones para explicar qué es la Realidad Aumentada. La primera, (Milgram & Kishino, 1994) fue realizada por Paul Milgram y Fumio Kishino en 1994. Definieron un concepto que llamaron Continuo-Virtualidad (Virtuality-Continuum) que consiste en una escala continua entre lo completamente virtual (Entorno Virtual), y lo completamente real (Entorno Real). El Continuo Realidad-Virtualidad por tanto, abarca todas las posibles variaciones y composiciones de objetos reales y virtuales. El área que está entre los extremos, donde tanto lo real como lo virtual se mezcla, recibe el nombre de Realidad Mixta (Mixed Reality); ésta a su vez se dice que consta de Realidad Aumentada

(Augmented Reality), donde lo virtual aumenta lo real; y la Virtualidad Aumentada (Augmented Virtuality), donde lo real aumenta lo virtual. Mientras que el término Virtualidad Aumentada raramente es utilizado, Realidad Aumentada y Mixta se suelen usar como sinónimos. [Andrade, 2010]



Figura 1.8: Realidad mixta

[Andrade, 2010]

La segunda de las definiciones la hizo el Doctor Ronald T. Azuma en 1997. Según él (Azuma, 1997), para poderse considerar Realidad Aumentada, ésta ha de mezclar elementos de la realidad con elementos inexistentes en la misma, es decir Realidades Virtuales; el individuo que la experimente ha de poder interactuar con ella en tiempo real y ha de estar en tres dimensiones. Las tecnologías existentes en la actualidad para producir Realidad Aumentada, se pueden englobar en dos categorías: Tecnología hardware dedicada que proporcione capacidades de Realidad Aumentada y software aplicable a soluciones hardware ya existentes y cuyo propósito general no es el de proporcionar Realidad Aumentada, por ejemplo Smartphone, tabletas, etc. Por tanto, para simplificar se dirá que existen soluciones de hardware y soluciones de software. [Azuma, 1997]

### **2.10.1 Soluciones de Hardware**

En rasgos generales, los dispositivos dedicados de Realidad Aumentada están formados por unos cascos o auriculares que se acoplan a la cabeza del usuario (u otro tipo de soluciones que se expondrán a continuación) y un sistema de visualización para mostrarle la Realidad Virtual que se incluye al mundo real. Los cascos o auriculares pueden ser susceptibles de llevar incorporado sistemas de posicionamiento GPS para permitir determinar con exactitud la localización del usuario. Los principales sistemas de visualización que se vienen empleando son pantallas de óptica transparente (Optical See\_through Display) y pantallas de mezcla de imágenes (Video\_mixed Display).

A continuación se explicarán las tres técnicas más destacadas para la representación de la Realidad Aumentada.

#### **2.10.1.1 Display Montado Sobre la Cabeza del Usuario**

Este tipo de pantallas se conoce por el acrónimo HMD (head\_mounted display). Estos displays representan tanto imágenes del mundo real que está percibiendo el usuario como imágenes de Realidades Virtuales en tres dimensiones sobre la vista del mundo real. Se trata de un dispositivo de visualización parecido a un casco en el que sobre una lente muy próxima al ojo del usuario, que le permite ver el mundo real, se proyectan los elementos virtuales que, con la combinación de la visión del mundo real, genera la Realidad Aumentada. La proyección de la Realidad Virtual, en lugar de proyectarse sobre una lente, puede ser proyectada directamente sobre la retina de los ojos del usuario; en este caso el HMD recibe el nombre de monitor virtual de retina. Los HMD se pueden clasificar

atendiendo a dos parámetros; según su configuración física sobre la cabeza del usuario y según el campo de visión del mundo real que se le permita ver. Según el primer parámetro, los HMD pueden ser:

**Monocular:** Las imágenes virtuales sólo pueden ser percibidas por un ojo.

**Binocular:** Las imágenes virtuales pueden ser percibidas por ambos ojos, obteniendo de este modo una imagen estereoscópica.

Según el segundo parámetro, existen HMD destinados a Realidad Virtual y destinados a la representación de Realidad Aumentada. Cuando el dispositivo HMD reduce tanto el campo de visión del usuario, éste se ve totalmente inmerso en una Realidad Virtual, ya que solamente será capaz de ver las imágenes creadas artificialmente por ordenador sobre el display. Mientras, los Head\_Mounted Displays que permiten ver al usuario el entorno en el que está inmerso y añadir a esa visión objetos virtuales son los destinados a la representación de Realidad Aumentada.



**Figura 1.9:** Ejemplo Display montado en la cabeza

[Villarán, 2012]

Un ejemplo de actualidad de dispositivos dedicados de Realidad Aumentada es el PGLass que está realizando Google (Villarán, 2012). Las gafas prototipo fueron presentadas en Abril de 2012 en Google+, si bien la idea no es nueva, lo que ha sorprendido de este Proyecto es que las gafas son mucho más pequeñas y delgadas que todas las soluciones existentes hasta el momento; además Google ha indicado que el siguiente paso será llevar el diseño del dispositivo a uno nuevo que permita la integración con cualquier gafa convencional existente. Por todo esto, el Proyecto Glass de Google ha tenido tanta repercusión, debido a las posibilidades reales de implantación en el mercado de las que dispone, así como de su potencial.

#### **2.10.1.2 Display de Mano**

Constan de pantallas portátiles cuyo tamaño se ajusta a la mano del usuario. Todas las soluciones disponibles de este tipo hasta la fecha consisten en la visualización de la realidad a través de éstas (a través de la cámara digital de la que dispone el dispositivo) y sobre ellas se montan las diferentes Realidades Virtuales para dotar al usuario de la percepción de Realidad Aumentada.

Al principio, los dispositivos de Realidad Aumentada con display de mano empleaban para realizar el seguimiento del dispositivo, y por tanto, del usuario; marcadores con códigos QR, para más adelante usar unidades de geoposicionamiento GPS y sensores MEMS como brújulas digitales y acelerómetros/giroscopios de seis grados de libertad. Posteriormente, sistemas como ARToolKit, permitieron añadir información digital a las secuencias de vídeo en tiempo real. Actualmente, sistemas de visión como

SLAM3 o PTAM4 son los que se están empezando a emplear para realizar el seguimiento.

Este tipo de dispositivo parece que va ser el que mayor éxito comercial tenga de entre los diferentes tipos de dispositivos que proporcionan Realidad Aumentada; puesto que las ventajas más significativas que ofrece son el carácter portátil de los dispositivos de mano y la posibilidad de ser aplicada en los teléfonos con cámara, tan habituales en la sociedad de hoy. Sin embargo, tienen como desventaja las limitaciones físicas, por el hecho de que los usuarios han de sostener el dispositivo por delante de ellos en todo momento, así como el efecto distorsionador de ángulo típico de las cámaras de los teléfonos móviles en comparación con el mundo real visto a través de los ojos humanos. [Feiner, 2011]



**Figura 1.10:** Ejemplo Display de Mano

[Feiner, 2011]

### **2.10.1.3 Display Espacial**

En lugar de que el usuario tenga que llevar puesto o tenga que sostener tanto un dispositivo Head\_Mounted Display o Display de mano, los dispositivos de Realidad Aumentada Espacial (SAR, Spatial Augmented Reality) hacen uso de proyectores digitales para mostrar información gráfica sobre los objetos físicos. [Raskar & Henry, 1998]

### **2.10.2 Soluciones de Software**

Se trata de proporcionar software a dispositivos electrónicos tales como Smartphone o tabletas, los cuales disponen de las tecnologías hardware necesarias para ofrecer funcionalidades de Realidad Aumentada. El software ha de ser capaz de realizar una fusión coherente entre las imágenes obtenidas con la cámara del dispositivo del mundo real y las imágenes virtuales en 3D; estas imágenes en 3D han de posicionarse en lugares del mundo real. Con las imágenes capturadas por la cámara, el mundo real ha de ser situado en un sistema de coordenadas; esto recibe el nombre de registro de imágenes. Este proceso involucra diversos métodos de visión computacional, gran parte de ellos relacionados con el seguimiento de secuencias de vídeo. Gran número de métodos de visión por ordenador de Realidad Aumentada se heredan de modo similar a los métodos de odometría visual.

En el mercado se encuentran disponibles multitud de soluciones software de Realidad Aumentada que se pueden englobar en software de código abierto (open source) y software de código cerrado gratuito (free closed source). [Villarán, 2012]

### 2.10.3 Aplicaciones

Como se ha visto hasta ahora, la Realidad Aumentada proporciona una nueva forma de interacción que hace que esta tecnología se aplique en diferentes campos:

**Educación:** Puesto que el coste que supone realizar soluciones de Realidad Aumentada todavía no es lo suficientemente bajo; su uso está básicamente restringido a museos, parques temáticos, etc. En estos lugares es habitual la presencia de redes wi-fi, que junto con las aplicaciones de Realidad Aumentada, permiten que los usuarios obtengan información sobre lugares, obras o puedan visualizar recreaciones en 3D de obras arquitectónicas, artefactos científicos, etc. [Educause, 2005]

**Entretenimiento:** El campo de aplicación por excelencia de la Realidad Aumentada en el mundo del entretenimiento son los videojuegos, ya que proporciona una forma muy novedosa a la hora de jugar; sobre todo teniendo en cuenta que en los últimos años han ido ganando popularidad plataformas de juegos con controles no tradicionales, que se escapan del mando tradicional, como es el caso de la Wii de Nintendo y los juegos táctiles de las plataformas móviles. Por ello, esta nueva forma de jugar puede gozar de gran éxito en los próximos años. Un proyecto que goza de bastante éxito es ARQuake Project, que permite jugar al conocido Quake pero en exteriores disparando a enemigos virtuales. [INTEC, 2010]

**Arquitectura:** Para la recreación de edificios deteriorados o derruidos, así como para la representación de edificios de nueva construcción y la generación de planos de planta en 3D que proporcionan una visión más realista de los mismos.

**Dispositivos de navegación:** Facilitan el uso de navegación dentro de un edificio mostrando dónde se ubican cada una de las estancias del mismo; además de, en automóviles emplear la luna para superponer información de tráfico, destino, etc.

**Publicidad:** En la actualidad se emplea la Realidad Aumentada en este campo, sobre todo, para llamar la atención del usuario mediante anuncios interactivos, susceptibles de proporcionar interactividad.

**Turismo:** Existen plataformas que no requieren prácticamente conocimientos técnicos para desarrollar aplicaciones a terceros; un par de ejemplos son Junaio o Layar; que han fomentado el lanzamiento de gran número de aplicaciones sobre turismo, etc.

Se debe tomar en cuenta que el reconocimiento de patrones de imágenes en los dispositivos móviles está sujeto a diversas consideraciones, como ser que la cámara del dispositivo debe encontrarse por lo menos a una distancia de 25 cm a 30 cm del objeto o patrón a identificar, además q la iluminación debe ser la adecuada sin obstruir la percepción del objeto.

Se puede observar una tabla con las distancias máximas a las que la cámara puede reconocer el patrón u objeto. [Educause, 2005]

TAMAÑO	DISTANCIA
16 x 16 cm	1,90 m
11 x 11 cm	1,30 m
7,5 x 7,5 cm	1,00 m
3,5 x 3,5 cm	0,40 m
1,3 x 1,3 cm	0,16 m

**Figura 1.11:** Distancia de reconocimiento de patrones de imágenes

Fuente: [Educause, 2005]

El dispositivo detectará el patrón de acuerdo a su tamaño y a la distancia que tenga con la cámara. Mientras más grande sea el patrón, más lejos de la cámara puede ubicarse.

## 2.11 PATRÓN DE ARQUITECTURA MODELO VISTA CONTROLADOR (MVC)

El patrón de arquitectura MVC (Modelo Vista Controlador) es un patrón que define la organización independiente del **Modelo** (Objetos de Negocio), la **Vista** (interfaz con el usuario u otro sistema) y el **Controlador** (controlador del workflow de la aplicación).

De esta forma, dividimos el sistema en tres capas donde, como explicaremos más adelante, tenemos la encapsulación de los datos, la interfaz o vista por otro y por último la lógica interna o controlador. [Soto, 2010]

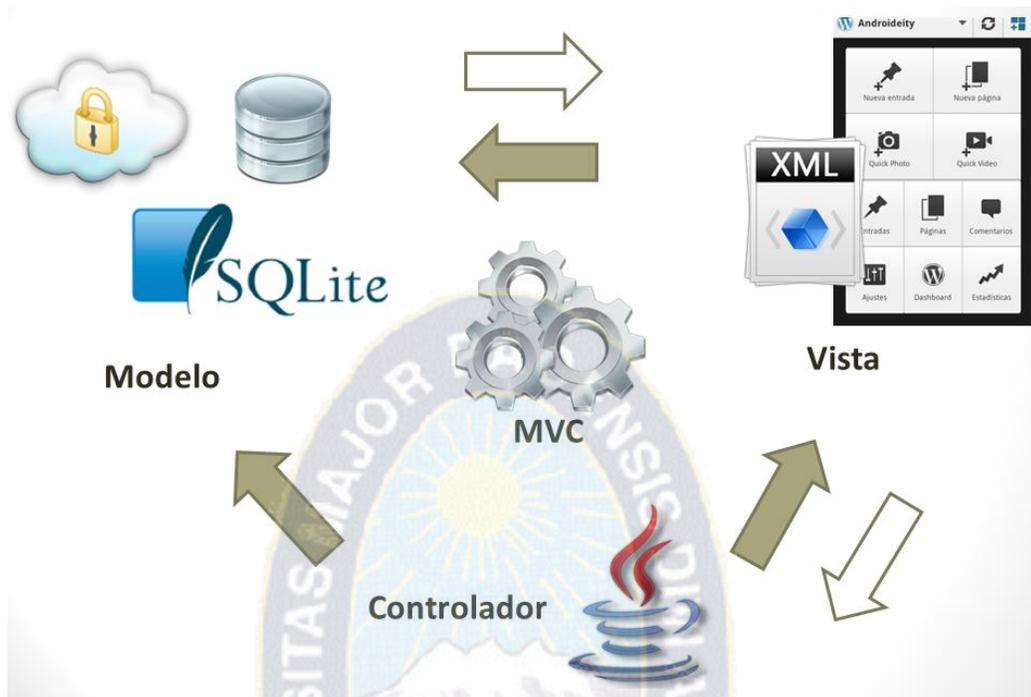


Figura 1.12: Diagrama del Modelo Vista Controlador

Fuente: [Soto, 2010]

El patrón de arquitectura "modelo vista controlador", es una filosofía de diseño de aplicaciones, compuesta por:

- Modelo:** Contiene el núcleo de la funcionalidad (dominio) de la aplicación.  
 Encapsula el estado de la aplicación.  
 No sabe nada / independiente del Controlador y la Vista.
- Vista:** Es la presentación del Modelo.  
 Puede acceder al Modelo pero nunca cambiar su estado.  
 Puede ser notificada cuando hay un cambio de estado en el Modelo.

**Controlador:** Reacciona a la petición del Cliente, ejecutando la acción adecuada y creando el modelo pertinente.

Para entender cómo funciona nuestro patrón Modelo vista controlador, se debe entender la división a través del conjunto de estos tres elementos y como estos componentes se comunican unos con los otros y con otras vistas y controladores externos al modelo principal. Para ello, es importante saber que el controlador interpreta las entradas del usuario (tanto teclado como el ratón), enviado el mensaje de acción al modelo y a la vista para que se proceda con los cambios que se consideren adecuados. [Soto, 2010]

### **2.11.1 Comunicación**

El modelo, la vista y el controlador deben comunicarse de una manera estable los unos con los otros, de manera que sea coherente con las iteraciones que el usuario realizara. Como es lógico la comunicación entre la vista y el controlador es bastante básica pues están diseñados para operar juntos, pero los modelos se comunican de una manera diferente, un poco más sutil. [Soto, 2010]

### **2.11.2 Modelo pasivo**

No es necesario para el modelo hacer ninguna tener alguna disposición a él, simplemente basta con tener en cuenta su existencia. El modelo no tiene ninguna responsabilidad para comunicar los cambios a la vista porque ocurren solo por orden del usuario, por lo que esta función la llevara a cabo el controlador porque será el que interprete las ordenes de este usuario debido a que solo debe comunicar que algo ha cambiado. Por esto, el

modelo es se encuentra en modo inconsciente y su participación en este caso es irrisoria. [Soto, 2010]

### 2.11.3 Unión del modelo con la vista y el controlador

Como no todos los modelos pueden ser pasivos, necesitamos algo que comunique al controlador y a la vista, por lo que en este caso, si que necesitamos el modelo, ya que solo este puede llevar a cabo los cambios necesarios al estado actual en el que estos se encuentran.

Al contrario que el modelo, que puede ser asociado a múltiples asociaciones con otras vistas y controladores, cada vista solo puede ser asociada a un único controlador, por lo que han de tener una variable de tipo controlador que notificara a la vista cual es su controlador o modelo asignado. De igual manera, el controlador tiene una variable llamada *View* que apunta a la vista. De esta manera, pueden enviarse mensajes directos el uno al otro y al mismo tiempo, a su modelo.

Al final, la vista es quien lleva la responsabilidad de establecer la comunicación entre los elementos de nuestro patrón MVC. Cuando la vista recibe un mensaje que concierne al modelo o al controlador, lo deja registrado como el modelo con el cual se comunicara y apunta con la variable controlador al controlador asignado, enviándole al mismo su identificación para que el controlador establezca en su variable *view* el identificador de la vista y así puedan operar conjuntamente. El responsable de deshacer estas conexiones, seguirá siendo la vista, quitándose a sí misma como dependiente del modelo y liberando al controlador. [Soto, 2010]

## **CAPITULO III**

### **MARCO APLICATIVO**

#### **3.1 DISEÑO DE LA APLICACION**

##### **3.1.1. Procedimiento**

Para la creación de una aplicación basada en Realidad Aumentada haciendo uso del SDK Vuforia de Qualcomm, su desarrollo se puede realizar siguiendo dos procedimientos bien diferenciados, a través de Eclipse con el plugin de Android o mediante una extensión de Unity. A continuación se describirán estas dos formas de proceder y se argumentará el por qué de la elección de una de ellas.

La primera forma de proceder, consiste en emplear el entorno de desarrollo Eclipse para programar la aplicación del modo tradicional, igual que se hace habitualmente para la creación de aplicaciones para escritorio; teniendo en cuenta su arquitectura, su lenguaje de programación principal java , su API y framework, etc. Junto con ello, el empleo del SDK Vuforia, que establece su propio API en C++, arquitectura, etc.

La segunda de ellas es a través de Unity. Unity es un Motor 3D, propiedad de Unity Technologies, que se emplea para el desarrollo de videojuegos para Windows, OS X, iOS, Android, Xbox 360, PlayStation 3, y Wii. Qualcomm ofrece una extensión para Unity que permite crear aplicaciones basadas en Aumentada con Vuforia para dispositivos Android.

Si bien, como se ha mencionado en apartados anteriores el SDK Vuforia es gratuito, al igual que la extensión para el Motor 3D; para poder cargar aplicaciones en Android, Unity requiere de una licencia de pago. Además, la programación de la aplicación se realiza a través de Unity, que

actúa como una capa de abstracción por encima de java, de la arquitectura de Android y del API en C++ de Vuforia. La programación en Unity se realiza de un modo más visual, situando posiciones de cámara, planos (Targets), Realidades Virtuales sobre los Targets, etc. a modo de un editor 3D convencional, combinada con lenguajes de programación Java Script, C# o un dialecto de Python llamado Boo.

De estas dos maneras de proceder, Qualcomm recomienda la programación a través de la extensión de Unity debido a que, como se dijo antes, se evita la programación directa contra Java, conocer la arquitectura de Android, etc.

Sin embargo, para la realización de este Proyecto, se ha optado por la programación directamente contra código en Java, haciendo uso del API y framework de Eclipse para Android, así como con el SDK Vuforia. El fundamento de esta elección radica en uno de los objetivos que se quieren alcanzar con este Proyecto y que consiste en aprender cómo es la arquitectura y funcionamiento de Android, así como aprender a programar en su lenguaje de programación Java.

El hándicap que presenta esta forma de proceder, a parte de la dificultad extra que supone, se produce en la representación de las Realidades Virtuales; y es que si se sigue este procedimiento; las animaciones de los objetos 3D sólo pueden ser animaciones simples. A pesar de ello, se ha considerado más provechosa esta forma de proceder, por lo expuesto anteriormente.

Puesto que comenzar a programar una aplicación basada en Vuforia desde cero es muy complicado y no hay suficiente documentación para realizarlo; Qualcomm pone a disposición de los desarrolladores varias

aplicaciones simples de ejemplo con su código fuente disponible para que, a partir de ellas y a modo de plantilla, comenzar a programar aplicaciones propias sea más sencillo. Gracias a estas aplicaciones de ejemplo se puede comprobar cuáles son las posibilidades básicas y características que proporciona este SDK.

Aunque para desarrollar la aplicación final se haya tomado la aplicación ImageTargets como plantilla; tanto ésta como el resto de aplicaciones de ejemplo, comparten una serie de ficheros comunes, clases, que se alojan en el directorio RCommon, y de los que heredarán muchas de las clases de la. Alguno de estos componentes ha tenido que ser reescritos para satisfacer características de la aplicación a desarrollar. Éstos son:

**ARParentViewController:** Definición e implementación del ViewController principal de la aplicación. Se encarga de añadir la EAGLView y la Overlay view a la Ventana Principal, manejadores de eventos táctiles, etc. No se ha modificado ningún aspecto de las mismas para la realización del Proyecto.

**ARViewController:** Gestiona el ciclo de vida de la cámara y de las Realidades Aumentadas, llamando a QCAR:createAR, QCAR:destroyAR, QCAR:pauseAR y QCAR:resumeAR cuando sea necesario. Asimismo, gestiona los datos necesarios para la Vista, como la carga de texturas que se requieren para pintar las Realidades Aumentadas. Además se encarga de tratar los cambios de orientación del dispositivo rotando la Vista como proceda.

**AR\_EAGLView:** Superclase que contiene la configuración de OpenGL para su sub\_clase, EAGLView. La clase AR\_EAGL envuelve el CAEAGLLayer de CoreAnimation en una subclase de UIView. El contenido de la Vista es básicamente una superficie

EAGL donde se reproduce la escena OpenGL. Este aspecto no se ha modificado. Aquí se define la clase Object3D, que se trata de la estructura que apuntará a un objeto para que se renderice; es decir, un objeto de esta clase se corresponderá con un objeto 3D que se presentará al usuario como Realidad Aumentada. Esta clase hereda de NSObject (clase base de un gran número de librerías que se utilizan tanto en OS X como en iOS) y ha sido modificada para satisfacer las características de los ficheros de cabecera de los diferentes Modelos 3D y videos.

**OverlayViewController:** Se encarga de la gestión del ActionSheet de configuración de la cámara así como de la disposición de los elementos en pantalla cuando se gira el dispositivo. No se ha modificado ningún aspecto de las mismas para la realización del Proyecto.

**OverlayView:** Se encarga de mostrar los elementos de la interfaz de usuario sobre otra Vista. No se ha modificado ningún aspecto de las mismas para la realización del Proyecto.

**QCARutils:** Contiene todo el código que inicializa y maneja el ciclo de vida de QCAR (Vuforia); además de algunas funciones útiles para acceder a los Targets, etc. Se trata de una clase que hace que QCAR sea accesible desde cualquier lugar dentro de la aplicación. No se ha modificado ningún aspecto de las mismas para la realización del Proyecto.

**ShaderUtils:** Aquí se definen una serie de métodos que servirán de ayuda para la manipulación de las matrices que representan la posición de los objetos 3D sobre la escena: Translaciones, rotaciones, escalados, etc. No se ha modificado ningún aspecto de las mismas para la realización del Proyecto.

**Texture:** En ella se definen una serie de atributos y métodos necesarios para que, partiendo de un fichero de imagen (JPG, PNG), crear una textura y podérsela aplicar de forma conveniente al Objeto 3D. No se ha modificado ningún aspecto de las mismas para la realización del Proyecto.

Una vez vistos los componentes comunes a todas las aplicaciones Vuforia, se citarán los componentes propios de la aplicación final:

**ImageTargetsAppDelegate:** Se encarga de gestionar el ciclo de vida de la aplicación de acuerdo al ciclo de vida de las aplicaciones bajo la arquitectura Android. Se controla si la resolución de la pantalla del dispositivo es retina, mantiene la pantalla de carga de la aplicación hasta que la cámara esté inicializada, etc.

Las modificaciones necesarias en estos ficheros tienen que ver con especificar a la aplicación qué datasets han de cargarse para poder detectar y realizar el seguimiento de los diferentes marcadores de imagen.

**EAGLView:** Es la clase encargada de la importación de los diferentes Modelos 3D y su renderización, tanto estática como animada; en función de los Targets que se estén visualizando en el momento.

**DomParentViewController:** Se trata de una sub\_clase de ARParentViewController que se encarga de pasar los eventos táctiles que se producen sobre la Ventana a la clase de la que hereda.

**ButtonOverlay:** Definición, implementación y fichero de Interface Builder de la clase ButtonOverlay. Aquí se configuran todos los elementos de la interfaz gráfica: botones, etiquetas, imágenes, WebViews, ActionSheets, cuadros de diálogo, así como los eventos

multitáctiles a los que responderán y a que métodos han de llamar cuando éstos se produzcan. Además se definen una serie de métodos que afectan a la interfaz de usuario, mostrando u ocultando botones, etiquetas, etc. que sirven para adaptar la interfaz gráfica al contexto de utilización de la aplicación.

**VideoPlaybackController:** gestiona el control del objeto tipo video de la aplicación, verifica el estado del video (en reproducción, detenido, pausa).

### 3.1.2 Datasets

Es conveniente hacer referencia a los datasets que contienen la información necesaria para que la aplicación pueda identificar y realizar el seguimiento de las diferentes imágenes. Este dataset se crea on\_line mediante la herramienta proporcionada por Qualcomm, Target Management System; para ello se han de subir a las diferentes imágenes correspondientes a cada una de las pinturas, objetos, lugares a dicha herramienta on\_line. Una vez son procesadas las imágenes, se procede a la descarga del dataset. En el caso de este Proyecto, los ficheros correspondientes al dataset son: Turismo.dat y Turismo.xml; es en este último fichero donde se encuentra la información relativa a los diferentes trackables que se emplean en la aplicación, a saber, nombre de la variable que servirá para identificar a un Target dentro del código y su tamaño.

### 3.2 DIAGRAMA DE ARQUITECTURA

La arquitectura del Kit de desarrollo de software ofrecido por Vuforia, A continuación se exponen con más detalle las características del SDK Vuforia de Qualcomm, puesto que se trata de la herramienta de desarrollo que dotará a la aplicación a implementar en este Proyecto de las funcionalidades de Realidad Aumentada.

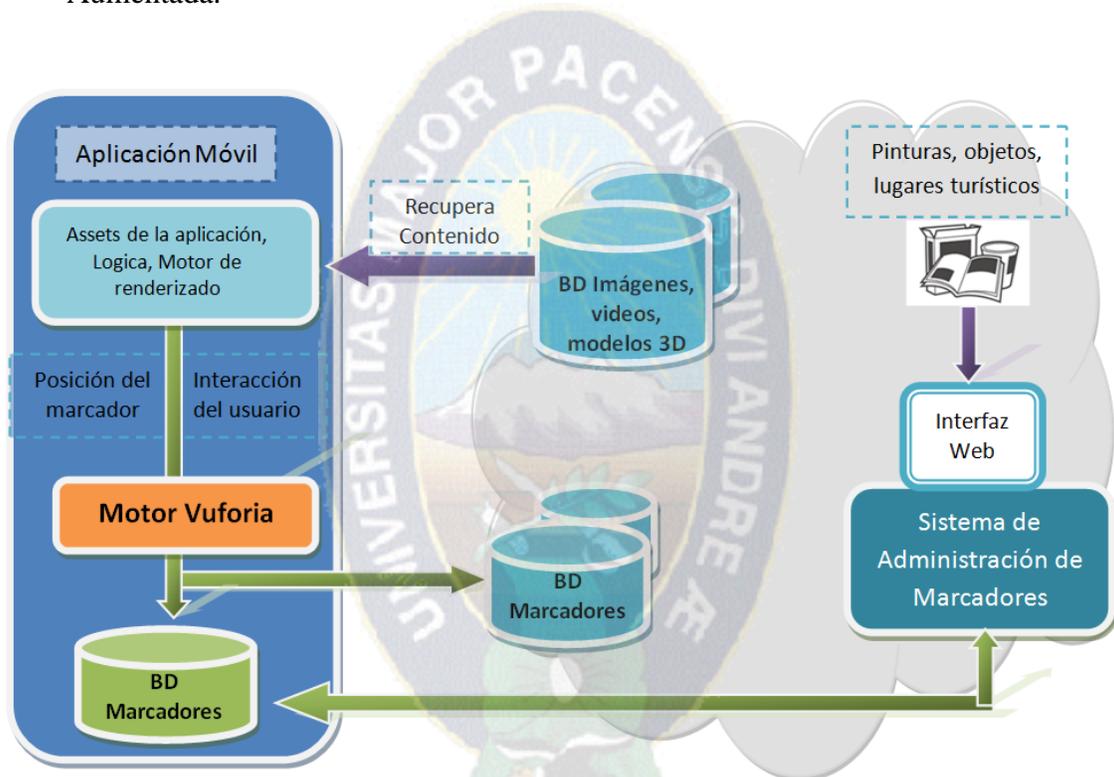


Figura 2.1: Diagrama de Arquitectura

Fuente: Elaboración Propia

Una aplicación basada en el SDK Vuforia de Realidad Aumentada hace que la pantalla del dispositivo móvil funcione como unas “lentes mágicas” que al mirar a través de ellas, el usuario percibe un mundo aumentado en el que el mundo real y

virtual coexisten. La aplicación superpone objetos virtuales 3D sobre la visión en directo capturada por la cámara del dispositivo, de modo que estos objetos virtuales parecen que realmente se encuentran en la escena real. La figura que se muestra a continuación muestra una vista general de cómo se realiza el desarrollo de una aplicación con la plataforma Vuforia. Esta plataforma consta del SDK Vuforia y un Target Management System (Sistema de gestión de Imágenes de Referencia) alojado en el portal de desarrolladores de Qualcomm, QDevNet5El desarrollador puede subir imágenes para generar la imagen de referencia (trackable) para realizar el seguimiento y una vez generada por el sistema, descargarse los recursos necesarios que serán incluidos en la aplicación a desarrollar. El SDK Vuforia proporciona una biblioteca (objeto compartido “libQCAR.so” en Android, y biblioteca estática “libQCAR.a” en iOS) que debe ser vinculada a la aplicación.



### 3.3 DIAGRAMA DE FLUJO DE DATOS

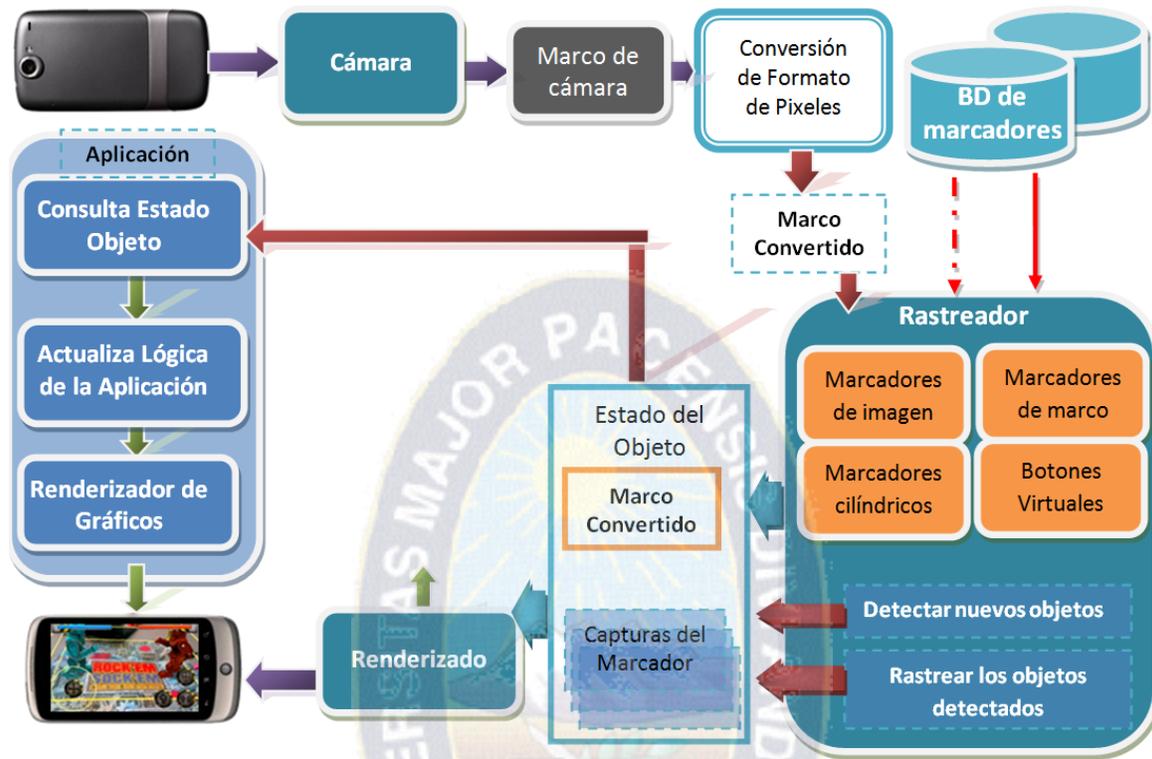


Figura 2.2: Diagrama de Flujo de Datos

Fuente: Elaboración Propia

Una aplicación de Realidad Aumentada basada en Vuforia, consta de una serie de componentes esenciales. A continuación se describirán términos referentes a la arquitectura de clases utilizadas en el desarrollo muchos de estos componentes se tratan de singletons; cuya traducción literal sería “hijo único”, y que consiste en una clase diseñada para tener sólo una instancia (o un número limitado de ellas).

- **Cámara:** La instancia de la cámara se encarga de que cada fotograma capturado por la cámara digital se pase de forma eficiente al rastreador.

- **Convertidor de imágenes:** La instancia del conversor de formato de pixel realiza la conversión entre el formato con el que trabaja la cámara (por ejemplo, YUV12) a un formato adecuado para el renderizado en OpenGL ES (por ejemplo,

RGB565) y para el seguimiento (por ejemplo, luminancia). Esta conversión implica un submuestreo para tener la imagen capturada por la cámara en diferentes resoluciones disponible en la pila de fotogramas convertidos.

- **Rastreador:** El rastreador (tracker) contiene los algoritmos de visión computacional para detectar y seguir (detect & track) los objetos en los fotogramas capturados por la cámara. Basado en la imagen tomada por cámara, diferentes algoritmos se ocupan de detectar nuevas imágenes de referencia (Targets) o marcadores (Markers), y evaluar los botones virtuales (Virtual Buttons). Los resultados se almacenan en un objeto de estado que es utilizado por el procesador de vídeo de fondo y al que puede accederse desde el código de la aplicación. El tracker puede cargar múltiples conjuntos de datos (datasets), pero sólo uno puede estar activo a la vez.

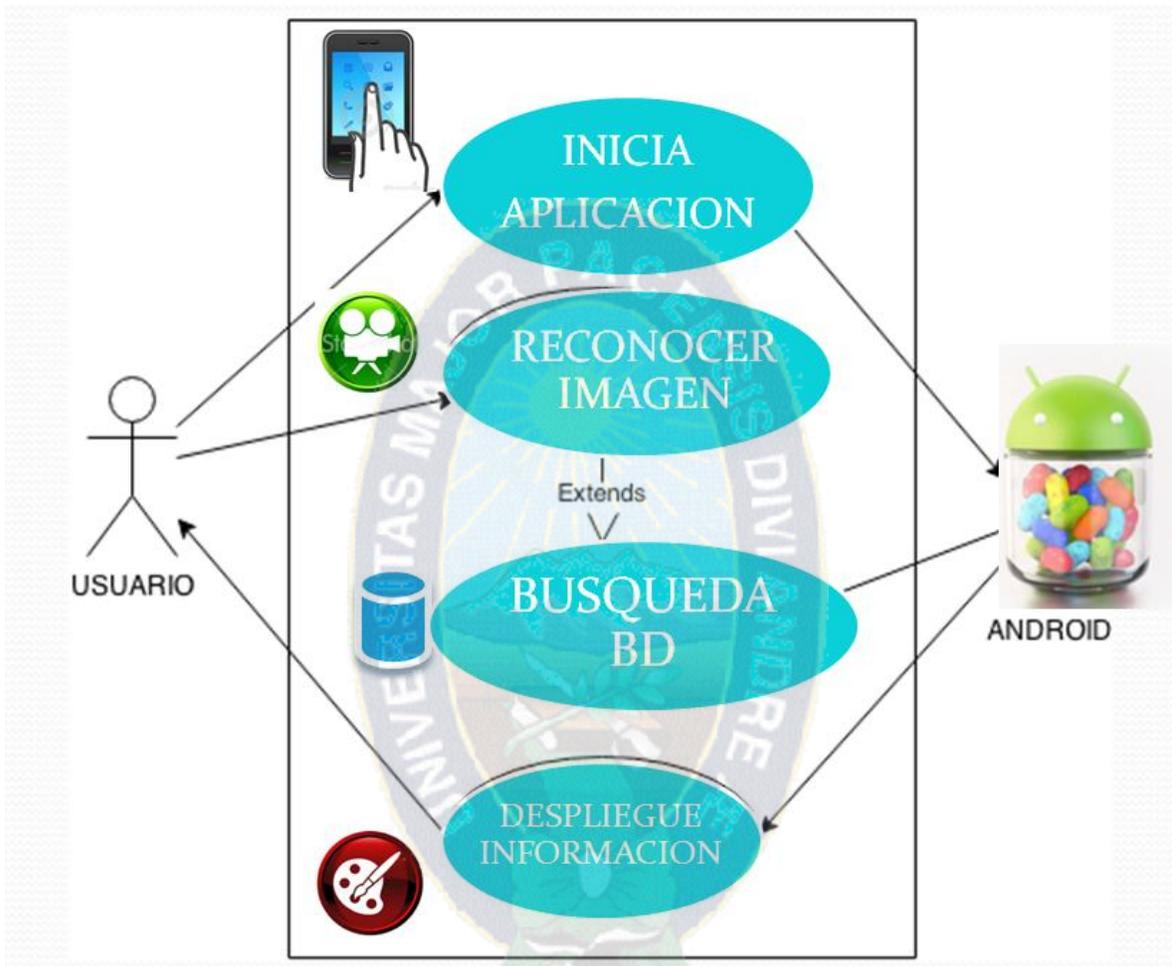
- **Procesador de vídeo de fondo:** La instancia del procesador de vídeo de fondo procesa la imagen capturada por la cámara que se encuentra almacenada en el objeto de estado. El rendimiento del renderizado del vídeo de fondo está optimizado para dispositivos específicos.

- **Código de la aplicación:** El desarrollador ha de inicializar todos los componentes anteriores y llevar a cabo tres pasos fundamentales en el código de la aplicación. Por cada fotograma procesado, el objeto de estado se actualiza y se llama al método de procesamiento de la aplicación. Se consulta el objeto de estado para los Targets y/o Markers nuevos que puedan aparecer en escena, se actualiza su estado, actualizar la lógica de la aplicación con nuevos datos de entrada y renderizar la capa de Realidad Aumentada.

- **Recursos de imágenes de referencia:** Los Target Resources se crean mediante el Target Management System, que se encuentra disponible on\_line. El dataset descargado contiene un fichero de configuración XML que permite configurar ciertas características de los trackables y un fichero binario que contiene la base de datos de los trackables. Estos elementos (XML y binario) son compilados por la aplicación

### 3.4 DIAGRAMA DE CASOS DE USO DEL SISTEMA

#### 3.4.1 CASO DE USO GENERAL



**Figura 2.3:** Diagrama de Casos de Uso general de la aplicación.

**Fuente:** Elaboración propia.

### 3.5 DESCRIPCIÓN DE LOS CASOS DE USO

#### 3.5.1 Caso de Uso General

**Nombre:** Caso de uso General.

**Actores:** Usuario; Android.

**Descripción:** Identifica el proceso de uso del sistema de reconocimiento de imágenes para dispositivos Android, para la identificación y despliegue de información.

**Precondiciones:** Tener dispositivo móvil con sistema operativo Android, con cualquier versión instalada del mismo, con conexión a internet y una cámara de video disponible.

**Postcondiciones:** el sistema mantiene el acceso a internet.

**Flujo típico de interacciones:**

Paso		ACCIONES DEL USUARIO O RESPUESTA DEL SISTEMA
1	Usuario	Inicia la aplicación desde el menú del dispositivo.
2	Android	Ingresa a la opción de identificación de imágenes.
3	Android	Inicia el servicio de intercambio de datos.
4	Android	Limpia la pantalla.
5	Android	Activa la cámara del dispositivo.
6	Usuario	Enfoca la cámara hacia el objeto a reconocer.
7	Android	Identifica la imagen.
8	Android	Muestra una interfaz con información del objeto.
9	Usuario	Reproduce video.
10	Usuario	Puede continuar con la aplicación o terminarla.

**Figura 2.4:** Flujo típico de acciones del usuario o respuesta del sistema.

**Fuente:** Elaboración propia.

### Flujo excepcional de Interacciones:

En caso de que falle la conexión a la Base de Datos en el paso 7.

<i>Pasos</i>	<i>ACCIONES DEL USUARIO O RESPUESTA DEL SISTEMA</i>	
<i>1</i>	<i>Android</i>	<i>Se muestra un error en conexión a BD</i>
<i>2</i>	<i>Usuario</i>	<i>Elige continuar intentando conexión o salir de la aplicación.</i>

**Figura 2.5:** Flujo alternativo de acciones del usuario o respuesta del sistema.

**Fuente:** Elaboración propia.

**Prioridad:** Alta.

**Frecuencia de uso:** Diario.

**Requerimientos especiales:** El sistema debe ser capaz de establecer conexión al servicio de transferencia de datos vía web, usando un dispositivo móvil con sistema operativo Android.

**Suposiciones:** El usuario cuenta con un dispositivo móvil con Android, conexión a internet vía Wi-Fi o plan de datos.

#### 3.5.2 Caso de Uso Reconocer Imagen

**Nombre:** Reconocer Imagen.

**Actores:** Usuario, Android.

**Descripción:** Identifica el proceso para reconocer una imagen.

**Precondiciones:** Existe una cámara disponible y activada en el dispositivo.

**Postcondiciones:** el sistema queda en espera de una acción por parte del usuario.

### Flujo típico de Interacciones.

	<i>Acciones del usuario o respuesta del sistema</i>	
1	<i>Usuario</i>	<i>Enfoca la cámara hacia el objeto o imagen a reconocer</i>
2	<i>Android</i>	<i>Activa un rastreador que busca la imagen para reconocer</i>
3	<i>Android</i>	<i>Una vez enfocada la imagen el rastreador activa la librería de reconocimiento de imágenes</i>
4	<i>Android</i>	<i>Genera un identificador de la imagen y la envía vía web.</i>
5	<i>Android</i>	<i>La BD retorna los datos codificados</i>
6	<i>Android</i>	<i>Decodifica los datos</i>
7	<i>Android</i>	<i>Despliega una interfaz con la información deseada.</i>
8	<i>Usuario</i>	<i>Puede continuar con la aplicación o terminarla.</i>

**Figura 2.6:** Flujo típico de acciones del usuario o respuesta del sistema.

**Fuente:** Elaboración propia.

**Prioridad:** Alta.

**Frecuencia de uso:** Diario.

**Suposiciones:** La base de datos contiene información acerca de la imagen.

#### 3.4.3 Caso de Uso Despliegue de Información

**Nombre:** Despliegue de información.

**Actores:** Usuario, Android.

**Descripción:** Identifica el proceso para el despliegue de la información para el usuario.

**Precondiciones:** Existe una respuesta afirmativa de la Base de datos.

**Postcondiciones:** El sistema mantiene encendido el tráfico de datos.



## Flujo típico de interacciones

<i>Acciones del usuario o respuesta del sistema</i>		
1	<i>Android</i>	<i>El sistema recibe la respuesta de la Base de Datos en forma de un objeto codificado en Json.</i>
2	<i>Android</i>	<i>Decodifica el objeto y lo vuelca en un contenedor de tipo Array.</i>
3	<i>Android</i>	<i>Un adaptador lo reagrupa el objeto y lo despliega en un list view de tres campos para el usuario.</i>
4	<i>Usuario</i>	<i>Puede reproducir un video informativo o deslizar la pantalla para poder observar imágenes y texto informativo, o deslizar la pantalla y observar su posición en un mapa.</i>
5	<i>Usuario</i>	<i>Puede ir atrás para identificar otra imagen o salir de la aplicación</i>

**Figura 2.7:** Flujo típico de acciones del usuario o respuesta del sistema.

**Fuente:** Elaboración propia.

### 3.6 DIAGRAMA DE SECUENCIA

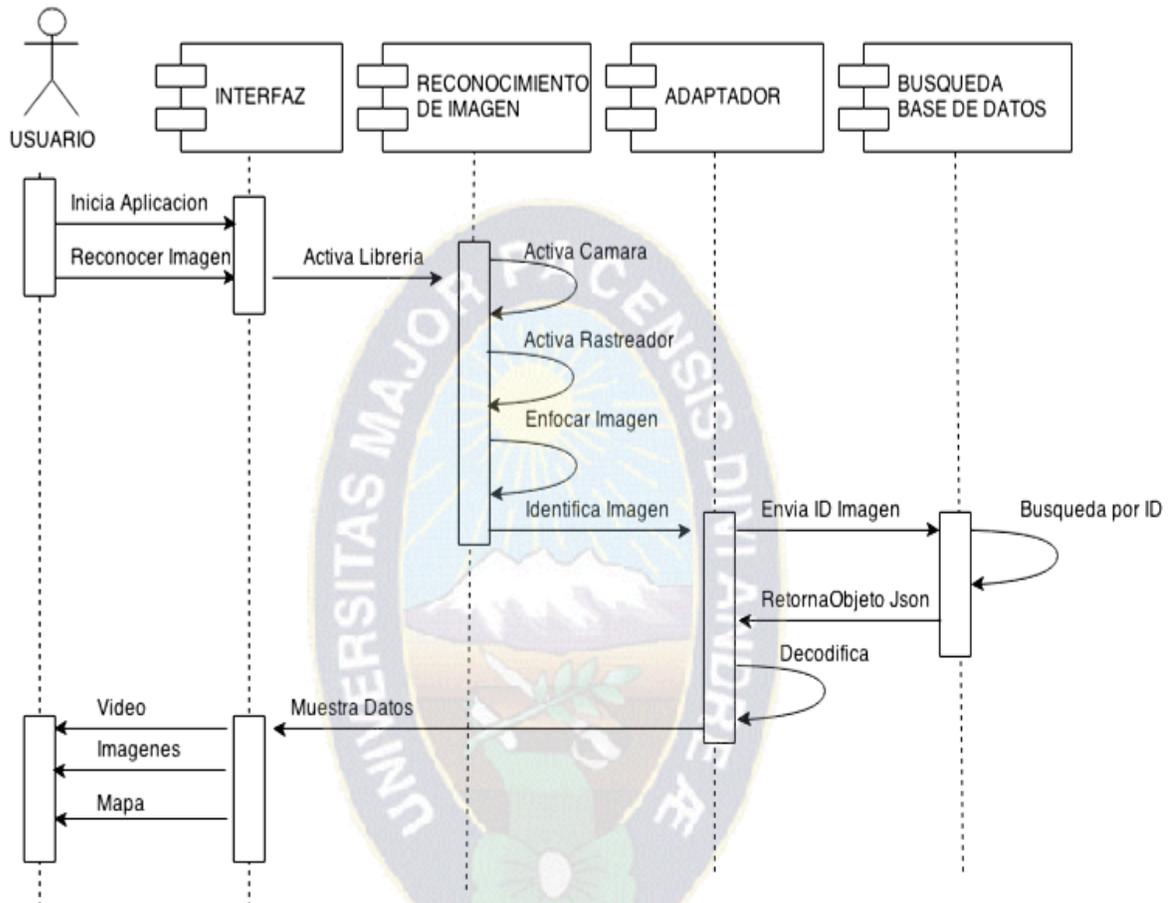
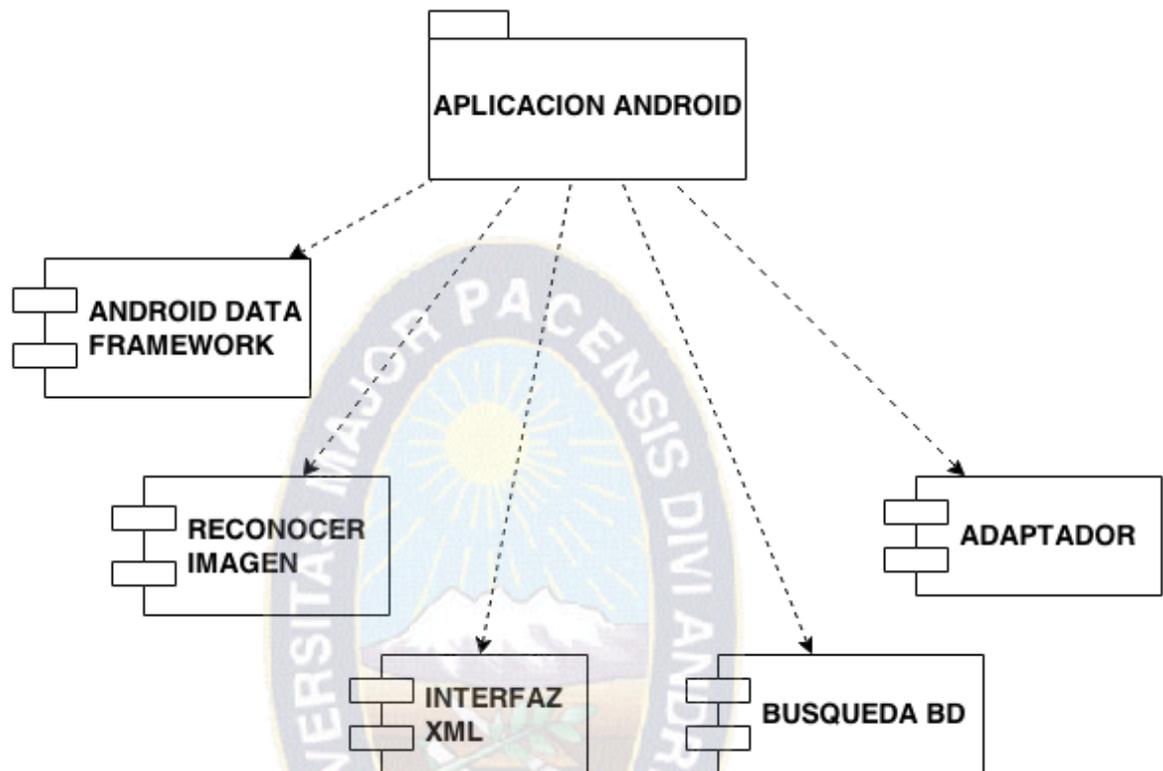


Figura 2.8: Diagrama de Secuencias del Sistema.

Fuente: Elaboración propia.

### 3.7 DIAGRAMA DE COMPONENTES



**Figura 2.9:** Diagrama de Componentes del Sistema.

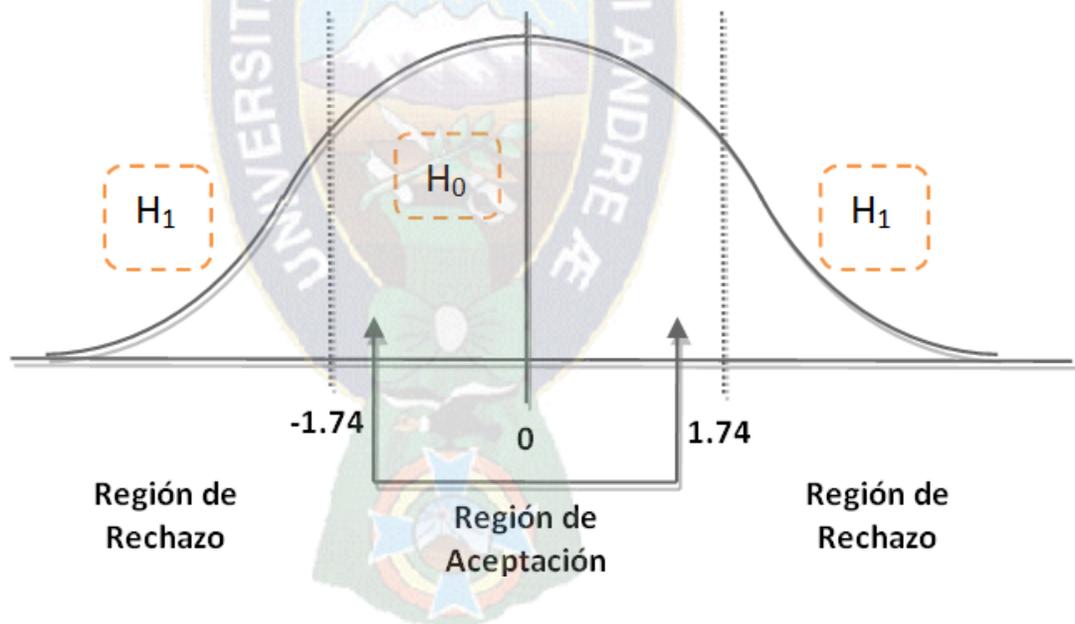
**Fuente:** Elaboración propia.

### 3.6 PRUEBA DE HIPÓTESIS

Es una afirmación sobre un parámetro poblacional, la cual será aceptada o rechazada en base a evidencia estadística.

**H<sub>0</sub>:** El uso de la Realidad Aumentada basada en reconocimiento de imágenes como una herramienta de identificación de patrimonio cultural y turístico logrará mejorar la experiencia del usuario brindando información consistente y oportuna.

**H<sub>a</sub>:** El uso de la Realidad Aumentada basada en reconocimiento de imágenes como una herramienta de identificación de patrimonio cultural y turístico no logrará mejorar la experiencia del usuario brindando información consistente y oportuna.



**Figura 2.10:** Curva Normal.

**Fuente:** Elaboración Propia.

Se vio por conveniente utilizar la distribución de *t de student* por tener una muestra pequeña. Menor a treinta.

Para el cálculo del parámetro utilizamos la formula de t de student:

$$t = \frac{(x - \mu)\sqrt{n}}{S}$$

**Figura 2.11:** Distribución *t de student*.

**Fuente:** Elaboración Propia.

Donde:

$x$	es la media	= 3
$\mu$	es el valor de la hipótesis	= 1
$S$	es la desviación típica	= 5.62
$n$	es el tamaño de la muestra	= 18

Se obtuvo un valor de 1.509 lo cual se encuentra en la región de aceptación ( $\pm 1.74$ ). Consecuentemente se acepta la hipótesis  $H_0$ .

**H<sub>0</sub>:** El uso de la Realidad Aumentada basada en reconocimiento de imágenes como una herramienta de identificación de patrimonio cultural y turístico logrará mejorar la experiencia del usuario brindando información consistente y oportuna.

## **CAPITULO IV**

### **4.1 CONCLUSIONES**

Se implementó una base de datos con la información de los marcadores, imágenes a reconocer.

Se implementó una base de datos con la información Multimedia para los visitantes.

Se desarrolló un módulo de reconocimiento de imágenes que gestiona los marcadores de la aplicación.

Para una mayor fluidez en la transmisión de datos se implementó la Base de datos de manera local con el fin de reducir los tiempos de descarga de contenidos.

Todo ello se implemento mediante una interfaz sencilla y amigable, implementando el modelo Vista Controlador para tener un diseño limpio y ordenado.

Se logró diseñar una aplicación de software para dispositivos móviles que permite identificar el patrimonio turístico y cultural de la ciudad de la paz, enriqueciendo la experiencia del turista, brindando información útil y oportuna de forma interactiva.

### **4.2 RECOMENDACIONES**

Se recomienda utilizar una combinación de reconocimiento de imágenes y geolocalización par que el usuario tenga una experiencia completa.

Se recomienda utilizar animaciones en 3D además de los videos para que dé la sensación de contar con un guía convencional y no con uno virtual.

Se recomienda mejorar la transferencia de información, por que esta se ve limitada por el acceso a internet del usuario en el momento del uso de la aplicación.

## BIBLIOGRAFIA

- [Andrade, 2010] ANDRADE M., 2010. "Que es la realidad Aumentada?". [En línea]. Disponible en: <<http://www.slideshare.net/tecnotic/realidad-aumentada-12579895>>. [Fecha de acceso: mayo 2012].
- [Azuma, 1997] AZUMA RONALD T, 1997. "A Survey of Augmented Reality". [En línea]. Disponible en: <<http://www.cs.unc.edu/azuma>>. [Fecha de acceso: junio 2012].
- [Conder & Darcey, 2010] CONDER SHANE y LAUREN DARCEY, 2010. "Android Wireles Application Development". Boston. USA. Addison-Wesley
- [Borrega, 2009] BORREGA REYES YOLANDA, 2013. "El patrimonio y su relación con el turismo". Licenciatura en Turismo. La Paz .UMSA. [En línea]. Disponible en: <<http://www.TurismoRuralbolivia.com>>. [Fecha de acceso: agosto 2013].
- [Educause, 2005] EDUCAUSE, 2012. "7 Things You Should Know About Augmented Reality". [En línea]. Disponible en: <<http://www.educase.edu/eli>>. [Fecha de acceso mayo 2012].
- [Gómez & Ortega, 2012] GOMEZ ISABEL y EDUARDO ORTEGA, 2012. "Arquitectura de la plataforma de desarrollo Windows phone 7". [En línea]. Disponible en: <<http://www.slideshare.net/movilforum/arquitectura-de-la-plataforma-de-desarrollo-de-windows-phone-7-5518091>>. [Fecha de acceso: agosto 2013].
- [Hochkofler, 2000] HOCHKOFER PATTY FRANKS OSWALDO, 2000. "Sistema de Información Geográfica Para el Registro Documental de Atractivos Turísticos (SIGRDAT)". Licenciatura en Informática. Biblioteca de informática. UMSA.
- [Intec ,2010] INTEC-DIRECCIÓN OPERATIVA DE INCORPORACIÓN DE TECNOLOGÍAS , 2011. "Plan Integral de Educación Digital". Buenos Aires, Argentina.

- [IAPH, 2010] INSTITUTO ANDALUZ DEL PATRIMONIO HISTÓRICO, 2013. "Que es Patrimonio Cultural". Consejería de Cultura. España. En línea. Disponible en: <<http://www.iaph.es/web/canales/patrimonio-cultural/>>. [Fecha de acceso: agosto 2013].
- [Jordan & Gryling, 2010] JORDAN LUCAS y GREYLING PIETER, 2011. "Practical Android Projects". Indianapolis. USA. APRESS.
- [Lee, 2012] LEE WEI-MENG, 2012. "Android 4 Application Development". Indianapolis. USA. JhonWiley & Sons Inc.
- [Maier, 2012] MAIER ROLANDO, 2012. "Todo lo que Deseas Saber de Android". [En línea]. Disponible en: <<http://www.elandroidelibre.com/2011/03/todo-lo-que-quieres-saber-de-android.html>>. [Fecha de acceso: junio 2013].
- [Martin, 2012] MARTIN EVA, 2012. "Cuanto Espacio de Almacenamiento Necesitas En tu Smartphone". [En línea]. Disponible en: <<http://www.tuexperto.com/2012/12/10/%C2%BFcuanto-espacio-de-almacenamiento-necesitas-en-tu-smartphone/>> [Fecha de acceso: abril 2013].
- [Murphy, 2009] MARK L. MURPHY, 2009. "Beginning Android". New York. USA. APRESS.
- [OMT, 2009] ORGANIZACIÓN MUNDIAL DE TURISMO, 2013. "Patrimonio Turístico" En línea. Disponible en: <<http://www.omt.com/resources/>>. [Fecha de acceso: agosto 2013].
- [Raskar, 1998] RASKAR RAMESH, GREG WELCH, HENRY FUCHS 1998. "Spatially Augmented Reality". Department of Computer Science. University of North Carolina at Chapel Hill. USA. [En línea]. Disponible en: <[http://web.media.mit.edu/~raskar/UNC/Office/0~IWAR\\_SAR.pdf](http://web.media.mit.edu/~raskar/UNC/Office/0~IWAR_SAR.pdf)> [Fecha de acceso: octubre 2013.]
- [Soto, 2010] SOTO V. ANDREA, 2013. "Modelo Vista Controlador: que es?", [En línea]. Disponible en: <<http://www.lab.inf.uc3m.es/~a0080802/RAI/mvc.html>>. [Fecha de acceso: noviembre 2013].
- [Villarán, 2012] VILLARÁN MOLINA DIEZO, 2012. "Realidad Aumentada en Dispositivos IOS Como sistema de ayuda a los alumnos". Trabajo de Fin de Grado. España. Universidad Carlos II de Madrid. [En línea]. Disponible en:

<[http://e.archivo.uc3m.es/bitstream/handle/10016/16921/TFG\\_Diego\\_Villaran\\_Molina.pdf;jsessionid=4B0BC088D0C685D847F73FCE3ADD9AE7?sequence=2](http://e.archivo.uc3m.es/bitstream/handle/10016/16921/TFG_Diego_Villaran_Molina.pdf;jsessionid=4B0BC088D0C685D847F73FCE3ADD9AE7?sequence=2)>. [Fecha de acceso: agosto 2013].

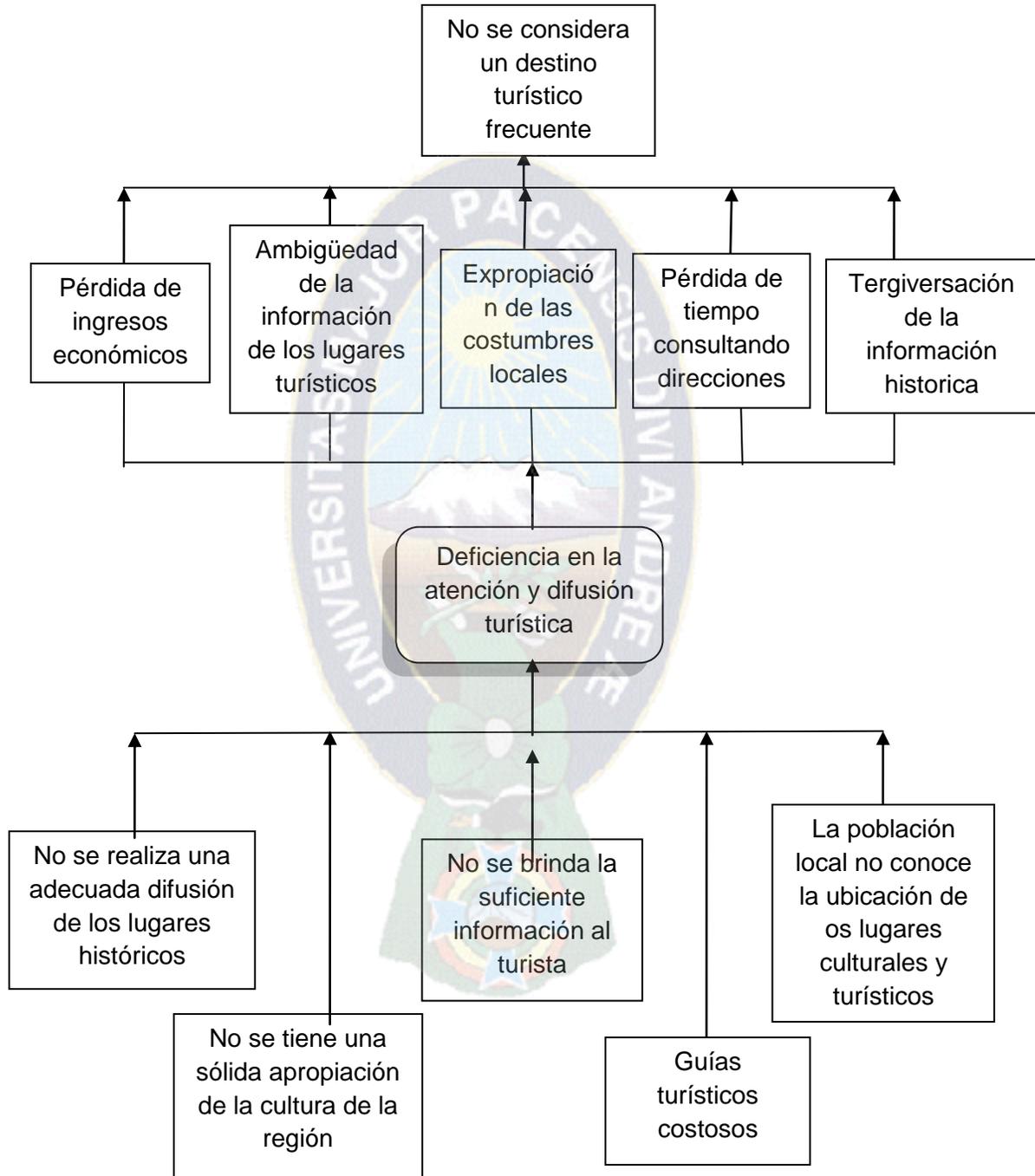
[Pastor, 2012]

PASTOR JAVIER E., 2012. "Apple 7 y su Arquitectura Plantean un Futuro Diferente" [En línea]. Disponible en:<<http://www.xataka.com/componentes-de-pc/apple-a7-y-su-arquitectura-plantean-un-futuro-diferente>>. [Fecha de acceso: septiembre 2013].

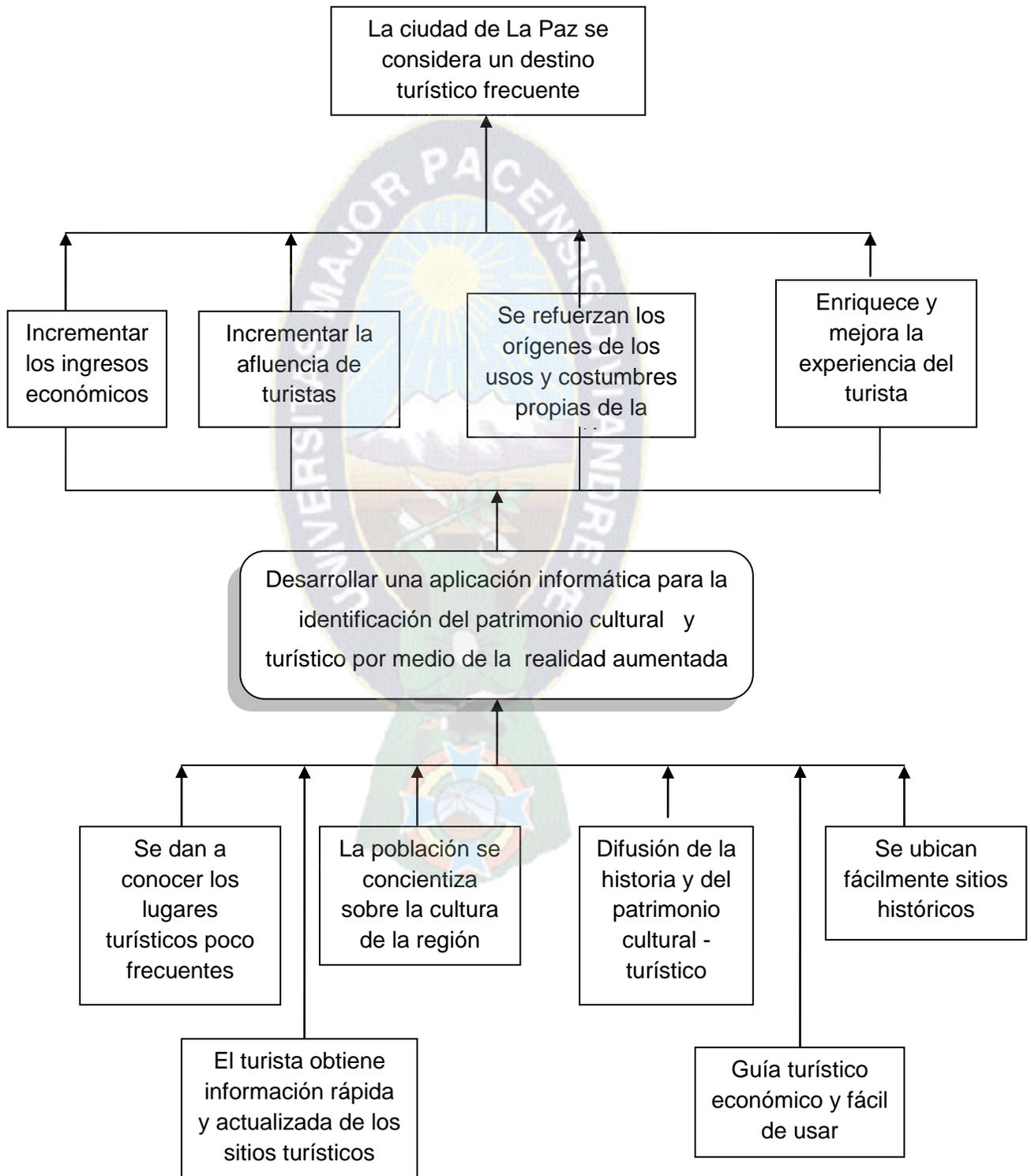


## ANEXOS

### A1 ARBOLDE PROBLEMAS



## A2 ÁRBOL DE OBJETIVOS



**A3 MARCO LÓGICO**

