

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO  
FACULTAD DE INGENIERÍA  
ESCUELA DE INGENIERÍA INFORMÁTICA

# **GUÍA DE TURISMO UTILIZANDO GOOGLE PLACES Y REALIDAD AUMENTADA**

**ESTEBAN ALBERTO PAVEZ AGUIRRE**

**CAROLINA ELIZABETH VILLANUEVA LASCEVENA**

INFORME FINAL DEL PROYECTO  
PARA OPTAR AL TÍTULO PROFESIONAL DE  
INGENIERO DE EJECUCIÓN EN INFORMÁTICA

Enero de 2014

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO  
FACULTAD DE INGENIERÍA  
ESCUELA DE INGENIERÍA INFORMÁTICA

# **GUÍA DE TURISMO UTILIZANDO GOOGLE PLACES Y REALIDAD AUMENTADA**

**ESTEBAN ALBERTO PAVEZ AGUIRRE**

**CAROLINA ELIZABETH VILLANUEVA LASCEVENA**

Profesor Guía: **José Miguel Rubio León**

Profesor Co-referente: **Claudio Cubillos Figueroa**

Carrera: **Ingeniería de Ejecución en Informática**

Enero de 2014

***Esteban Pavez:*** *Dedico este proyecto a mis padres, quienes me dieron apoyo incondicional en todo momento en el transcurso de esta carrera.*

***Carolina Villanueva:*** *Este proyecto va dedicado a mi familia, que gracias a su esfuerzo y apoyo logré terminar con éxito mis estudios.*

# Índice

<b>Resumen .....</b>	<b>iii</b>
<b>Abstract .....</b>	<b>iii</b>
<b>Glosario .....</b>	<b>iv</b>
<b>Índice de Figuras .....</b>	<b>vi</b>
<b>Índice de Tablas.....</b>	<b>vii</b>
<b>1    Introducción.....</b>	<b>1</b>
<b>2    Análisis del Problema .....</b>	<b>2</b>
2.1   Descripción del Problema.....	2
2.2   Objetivo General.....	2
2.3   Objetivos Específicos .....	2
<b>3    Descripción de la Solución.....</b>	<b>3</b>
3.1   Metodología.....	4
3.2   Plan de Trabajo .....	5
3.2.1   Carta Gantt.....	7
3.3   Herramientas.....	8
<b>4    Análisis de Tecnologías .....</b>	<b>9</b>
4.1   Realidad Aumentada.....	9
4.2   Sistema Operativo Android .....	10
4.3   Layar .....	10
4.4   Bot.....	10
4.5   API Google Places .....	12
4.6   Lenguajes de Programación.....	12
4.6.1   PHP.....	12
4.6.2   SQL .....	12
<b>5    Estudio de Factibilidad .....</b>	<b>13</b>
5.1   Técnica.....	13
5.2   Económica .....	13
5.3   Operativa.....	14
5.4   Legal .....	14
<b>6    Análisis y Diseño del Sistema .....</b>	<b>15</b>
6.1   Requerimientos Funcionales.....	15
6.2   Requerimientos no Funcionales.....	15
6.3   Arquitectura .....	15
6.4   Diagramas del Sistema .....	17
6.4.1   Caso de uso Seleccionar POI.....	17

6.4.2	Diagrama de Actividad seleccionar POI .....	18
6.4.3	Plan de Pruebas.....	19
6.4.4	Especificación de Casos de Prueba .....	21
<b>7</b>	<b>Desarrollo del Sistema .....</b>	<b>25</b>
7.1	Primer Prototipo.....	25
7.2	Segundo Prototipo .....	27
7.2.1	Simbología de Puntos de Interés .....	27
7.2.2	Desarrollo del Acceso directo.....	29
7.3	Prototipo Final .....	30
7.3.1	Creación del Servidor .....	30
7.3.2	Extracción de POIs de la API.....	30
7.3.3	Creación de Base de datos .....	31
7.3.4	Creación de Web Service .....	31
7.3.5	Desarrollo del Bot.....	32
7.3.6	Creación de Cron Job .....	33
7.3.7	Interfaces de la Aplicación .....	34
<b>8</b>	<b>Conclusiones y Trabajo Futuro .....</b>	<b>35</b>
<b>9</b>	<b>Referencias.....</b>	<b>36</b>

## **Resumen**

El presente informe describe las etapas de desarrollo del proyecto: “Guía de turismo utilizando Google Places y Realidad Aumentada”, el cual consiste en la implementación de una aplicación móvil que permita buscar lugares o puntos turísticos de la ciudad en la que se encuentre el usuario, y entregar información relevante sobre ellos, tales como número telefónico, dirección, sitio web, ruta a seguir para llegar al punto de interés, entre otros. Este proyecto se desarrollará con la intención de fomentar el turismo en las ciudades por medio de la prometedora e innovadora tecnología de realidad aumentada.

## **Abstract**

This document describes the development stages of the project: "Tourist Guide using Google Places and Augmented Reality", which consists in the implementation of a mobile application that allows finding places or tourist spots in the city where the user is, and deliver relevant information about them, such as phone number, address, website, route to follow to arrive to the point of interest, among others. This project will be developed with the intention of promoting tourism in the city, by one of the most promising and innovative technologies: augmented reality.

# Glosario

**API (Application Programming Interface):** Es el conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción. Son usadas generalmente en las bibliotecas.

**Base de datos (BD):** Es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.

**Bot:** Es un programa informático, imitando el comportamiento de un humano. Un *bot* puede estar diseñado en cualquier lenguaje de programación, funcionar en un servidor o en un cliente, o ser un agente móvil, etc.

**Capa (de aplicación):** Capa de aplicación de Laya, la cual es la que se ve en el Smartphone del usuario, donde se ven los puntos de interés ingresados en el gestor de POIs.

**Cliente:** Aplicación informática o un computador que consume un servicio remoto en otro computador, conocido como servidor, normalmente a través de una red de telecomunicaciones

**Cron:** Es un administrador de servicios que ejecuta un servicio o varios, programados para ejecutarse entre intervalos regulares.

**Cron Job:** Es una tarea guardada en los crontabs (archivos donde se guardan las listas de tareas) que el Cron verifica cada minuto para ver si existe alguna tarea para realizarse, todo esto se realiza en segundo plano.

**Framework (marco de trabajo):** Define, en términos generales, un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.

**Geolocalización:** Posicionamiento con el que se define la localización de un objeto espacial (representado mediante punto, vector, área, volumen) en un sistema de coordenadas y datum determinado.

**GPS (Global Positioning System):** Sistema de posicionamiento global que permite determinar en todo el mundo la posición de un objeto, una persona o un vehículo con una precisión hasta de centímetros (si se utiliza GPS diferencial), aunque lo habitual son unos pocos metros de precisión.

**Hardware:** Se refiere a todas las partes tangibles de un sistema informático; sus componentes son eléctricos, electrónicos, electromecánicos y mecánicos.

**Hosting:** Es el servicio que provee a los usuarios de Internet un sistema para poder almacenar información, imágenes, vídeo, o cualquier contenido accesible vía web.

**Kernel:** Parte del S.O. residente permanentemente en memoria. Dirige los recursos del sistema, memoria, E/S y procesos. Podemos distinguir dos partes: sección de manejo de procesos y sección de manejo de dispositivos.

**Link:** Elemento de un documento electrónico que hace referencia a otro recurso, por ejemplo, otro documento o un punto específico del mismo o de otro documento.

**Punto de Interés (POI):** Punto de ubicación específica que alguien puede encontrar útil o interesante. Como por ejemplo un restaurant, hospital, playas, etc.

**Software:** Se conoce como software al equipamiento lógico o soporte lógico de un sistema informático, que comprende el conjunto de los componentes lógicos necesarios que hacen posible la realización de tareas específicas.

**Smartphone:** Es un teléfono móvil construido sobre una plataforma informática móvil, con una mayor capacidad de almacenar datos y realizar actividades semejantes a una minicomputadora y conectividad que un teléfono móvil convencional.

**String:** Es una sucesión de caracteres (letras, números u otros signos o símbolos).

**Tablet:** Es una computadora portátil de mayor tamaño que un teléfono inteligente, integrada en una pantalla táctil con la que se interactúa primariamente con los dedos, sin necesidad de teclado físico ni ratón.

**URL:** Localizador de recursos Uniforme es una secuencia de caracteres, de acuerdo a un formato modélico y estándar, que se usa para nombrar recursos en Internet para su localización o identificación.

**Web Service:** Servicio web en donde se encuentran métodos que posibilitan el traspaso de información entre el Smartphone a la base de datos, y desde esta al gestor de puntos de interés.



# Índice de Figuras

Figura 1: Metodología del proyecto. ....	4
Figura 2: Estructura de Desglose de Trabajo .....	6
Figura 3: Carta Gantt – Proyecto I .....	7
Figura 4: Carta Gantt – Proyecto II .....	7
Figura 5: Esquema del concepto RA .....	9
Figura 6: Diagrama funcionamiento Bot.....	11
Figura 7: Arquitectura .....	16
Figura 8: Caso de uso seleccionar POI.....	17
Figura 9: Diagrama de Actividad Seleccionar POI.....	18
Figura 10: Creación de capa en Hoppala .....	25
Figura 11: Creación de capa en Layar.....	26
Figura 12: Administración de POIs en Hoppala .....	26
Figura 13: Modo lista .....	27
Figura 14: Modo mapa .....	27
Figura 15: Simbología de puntos de interés .....	28
Figura 16: Inicio de aplicación.....	28
Figura 17: Acceso directo de aplicación .....	29
Figura 18: Aplicación de prueba: POIs de Google Places .....	30
Figura 19: Base de Datos de la aplicación. ....	31
Figura 20: Código fuente de <i>bot</i> que agrega POI.....	32
Figura 21: Formularios de ingreso de datos POI.....	33
Figura 22: Creación de Cron Job.....	33
Figura 23: Interfaz inicial. ....	31
Figura 24: Opciones Aplicación.....	32
Figura 25: Ver como mapa.....	33
Figura 26: Mostrar ruta .....	33

# Índice de Tablas

Tabla 1: Costos del Proyecto.....	13
Tabla 2: Plan de Prueba Ubicación de POI's.....	19
Tabla 3: Plan de Prueba Mostrar ruta de POI.....	20
Tabla 4: Plan de Prueba Funcionalidades de POI's .....	21
Tabla 5: Caso de Prueba: Ingreso de POIs a la Base de Datos .....	22
Tabla 6: Caso de Prueba: Mostrar ruta.....	23
Tabla 7: Caso de Prueba: Funcionalidades de POI's .....	24

# 1 Introducción

No cabe duda que la tecnología avanza a pasos agigantados, en consecuencia, el manejo de información por parte de la sociedad ha cambiado radicalmente. En la actualidad una persona puede estar conectada a internet las 24 horas del día, en cualquier lugar a través de los dispositivos móviles que lo permiten, tales como *smartphones* o *tablets*, que son cada vez más necesarios, asequibles e indispensables para la mayoría de la sociedad.

Debido a esto, hoy en día existe un sin número de aplicaciones para dispositivos móviles que cumplen diversas funciones y satisfacen las necesidades de los distintos tipos de usuarios. El presente proyecto se enfoca en los usuarios turistas y se centra en el desarrollo de una aplicación para dispositivos móviles que se utilice como guía turística para la ciudad, conteniendo los puntos de interés más importantes y su información respectiva. Para facilitar aún más la utilización de la aplicación y la ubicación de los usuarios, se pretende incorporar la tecnología Realidad Aumentada.

La realidad aumentada es una tecnología que ayuda a enriquecer nuestra percepción de la realidad a través de un “lente” donde la información del mundo real se complementa con la del digital. Esta tecnología permite la superposición en tiempo real, de imágenes marcadores o información generados virtualmente, sobre imágenes del mundo real. De esta forma, se fusiona la información de objetos virtuales con la realidad, dando la impresión de que ambas pertenecen al mismo entorno. En el caso de este proyecto se utilizará la realidad aumentada en una aplicación para *smartphones*, ya que estos dispositivos cuentan con todos los componentes necesarios para su funcionamiento, estos son la cámara, pantalla, hardware, GPS, brújula y acelerómetro del *Smartphone*.

Este informe tiene el objetivo de describir el desarrollo del proyecto propuesto, detallando todas sus etapas, entre ellas los objetivos, las herramientas que se utilizarán para implementar la aplicación, el plan de trabajo, el estudio de factibilidad del proyecto, los requerimientos funcionales y no funcionales, el plan de pruebas y finalmente, diseño y desarrollo de los prototipos.

## **2      Análisis del Problema**

### **2.1    Descripción del Problema**

El turismo, tanto nacional como internacional, es actualmente una de las actividades económicas y culturales más importantes para un país. La relevancia del turismo residirá en el hecho de que es a través de éste, que el ser humano puede conocer y enriquecerse de otras culturas, sociedades, formas de vivir y ambientes geográficos.

En Chile, el turismo es una de las actividades que más influye en la economía del país, aportando con el 3,2% del PIB, recibiendo durante el 2013, 3.569.744 turistas internacionales, según datos que entregó el director nacional de turismo, Javier Vergara, quien agrega que al realizar un balance, se puede decir que entre los años 2010-2013, se produjo un aumento acumulado en la llegada de turistas extranjeros al país cercano al 30%. En cuanto al turismo nacional el 2103 fueron cerca de 8,5 millones de chilenos los que recorrieron el país, cifra que se pretende incrementar año tras año.

En consecuencia, es fundamental contar con nuevas e innovadoras acciones de promoción para fomentar el turismo en Chile. Sin embargo, en la actualidad no existen aplicaciones para dispositivos móviles con información turística dinámica y con fuentes confiables. Es por esto que se optó por implementar una aplicación que se utilice como guía turística, basándose en realidad aumentada, que permite visualizar la información de manera más simple, facilitando el uso de la aplicación, y que utilice la API de Google Places la cual proporciona información fidedigna de gran cantidad de puntos de interés.

### **2.2    Objetivo General**

El objetivo general del proyecto es crear una aplicación para *smartphones*, que incorpore los puntos de interés de cualquier ciudad donde se encuentre el usuario (turista), utilizando la Realidad Aumentada.

### **2.3    Objetivos Específicos**

- Comprender las tecnologías y herramientas para desarrollo de aplicaciones basadas en realidad aumentada.
- Desarrollar una aplicación móvil para teléfonos inteligentes orientada a turistas.
- Probar y validar la aplicación desarrollada.

### 3 Descripción de la Solución

El proyecto consiste en desarrollar una aplicación para dispositivos móviles que permita ser una guía turística para cualquier ciudad, utilizando Realidad Aumentada (RA). La aplicación será desarrollada para *Smartphones* con sistema operativo Android debido a que es uno de los sistemas más populares en la actualidad. En la aplicación se utilizará realidad aumentada móvil, que se basa en la geolocalización, así el usuario, a través de un *Smartphone*, puede observar información del entorno en tiempo real, como por ejemplo, donde se encuentra la estación de metro más cercana, la ubicación de un restaurante, tener información sobre algún monumento histórico, etc.

Para desarrollar este proyecto se utilizará un gestor de puntos de interés y la plataforma Layar, los cuales permiten a los desarrolladores trabajar con realidad aumentada. El primero posibilita el ingreso, almacenamiento y administración de los puntos de interés (POIs) que incorporará la aplicación, añadiendo toda la información requerida del lugar, además de permitir agregar opciones adicionales, tales como realizar llamadas telefónicas, enviar correos electrónicos, entre otras. La plataforma Layar, muestra en tiempo real los datos de los POIs captados por la cámara del *Smartphone*. Para ello, Layar utiliza una combinación entre la cámara, la brújula, la conexión a internet, el GPS y el acelerómetro del teléfono. Con ello se identifica la posición del usuario y su orientación, después se reciben los datos pertenecientes a la posición y se muestra en la pantalla sobre la imagen capturada por la cámara.

Para desarrollar esta aplicación se realizarán primeramente dos prototipos, los cuales incorporarán puntos de interés que serán ingresados manualmente a través de la plataforma Hoppala. En estos prototipos la aplicación contendrá aproximadamente 30 puntos de interés, junto con sus opciones. Se desea desarrollar de esta manera para poder comprender el funcionamiento de la realidad aumentada utilizando las plataformas nombradas anteriormente, y así continuar con el desarrollo del último prototipo.

Luego se desarrollará el prototipo final, en el cual se extraerán los puntos de interés automáticamente, utilizando la API que nos proporciona Google llamada Google Places, para luego ser insertados en un gestor de POIs mediante un *bot*. Para realizar esto se utilizará el proyecto de Android que provee Layar, el cual exporta un acceso directo a la capa de nuestra aplicación, para extraer los POIs usando la API, la cual retornará los datos de los puntos más cercanos al usuario. Luego esta información se enviará a un *web service* que la insertará en una base de datos. Finalmente, para ingresar los puntos al gestor de POIs, se utilizará un *bot*, que será desarrollado en lenguaje PHP y será ejecutado cada cierto tiempo a través de un *Cron Job*, e irá extrayendo la información de la base de datos e insertándola en el gestor de POIs. Así el cliente de Layar tomará esos puntos y los mostrará en la pantalla del celular del usuario utilizando la realidad aumentada.

### 3.1 Metodología

La metodología que se utilizará para desarrollar el proyecto será la de prototipado evolutivo. Se aprovechará esta para ir formulando los requerimientos, que si bien se especificaron los básicos para este proyecto, se prevé que no serán los definitivos y que según el avance en el desarrollo de la aplicación se irán modificando. Esta metodología nos da la posibilidad de ir observando, en cortos periodos de tiempo, el desarrollo de la aplicación. De esta manera, se pueden realizar cambios en los requerimientos para no tener inconsistencias en las futuras etapas del desarrollo del proyecto. En la figura adjunta se muestra gráficamente la forma en que se utilizó esta metodología [1].

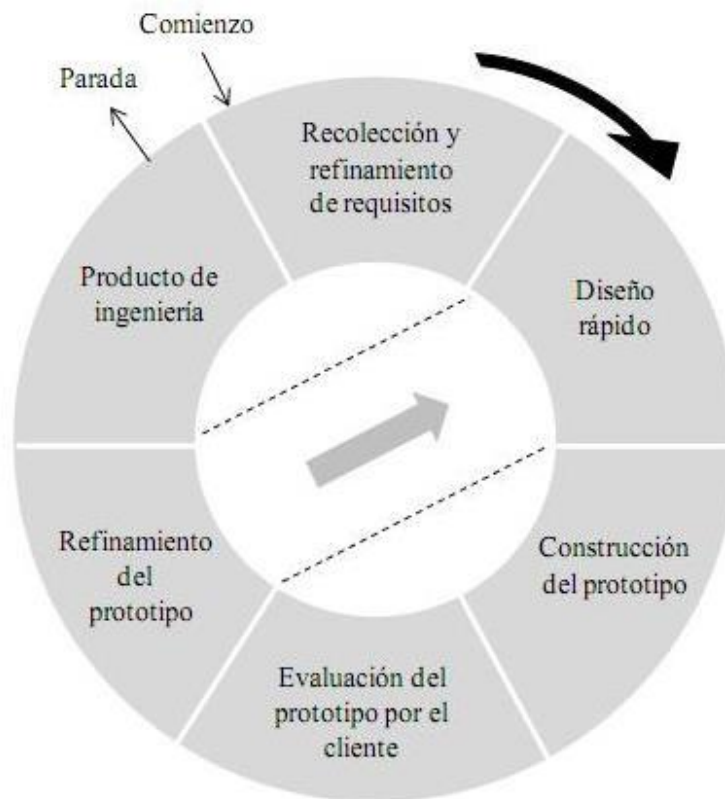


Figura 1: Metodología del proyecto. Fuente [1]

Para el primer prototipo se tendrá la arquitectura completa de Layar y Hoppala implementada, teniendo la capa de Layar hecha y funcionando en el *Smartphone*. Se espera que los puntos de interés oficiales estén ingresados junto a sus opciones correspondientes, y con la mayoría de la información adicional que estos deben tener. Para el segundo prototipo se tendrá la aplicación que funcionará como acceso directo a la capa en Layar, lo cual facilitará al usuario el ingreso a la aplicación de Turismo para Viña del Mar. Para el prototipo final se utilizará el proyecto Android de esta aplicación para realizar la extracción automática de puntos de interés usando la API de Google Places, para luego ser ingresados al gestor de POIs con el *bot*.

## 3.2 Plan de Trabajo

Para comenzar a realizar la planificación del proyecto, se analizaron las actividades que se desarrollarán durante el periodo de realización del sistema. Para esto se realizó una estructura de desglose de trabajo (*Work Breakdown Structure* o *WBS*), la cual nos ayuda a visualizar de una manera más ordenada las distintas actividades que son necesarias para desarrollar el proyecto. El diagrama WBS se dividió en 8 etapas, las cuales son: especificación de necesidades, análisis del problema, diseño del prototipo 1, construcción del prototipo 1, diseño del prototipo 2, construcción del prototipo 2, desarrollo del sistema automático y pruebas del sistema automático. En la figura 4 se puede visualizar el diagrama anteriormente descrito, diferenciando por color las actividades. Este proyecto se dividió en dos fases, Proyecto 1 y 2, las tareas de Proyecto 1 son las de color rojo, mientras que las verdes pertenecen a las actividades de Proyecto 2.

Con el diagrama de WBS finalizado se puede realizar una matriz de responsabilidades, donde se les asigna a los alumnos las actividades de las que estarán encargados. Además se procede a la elaboración de la carta Gantt del proyecto (basada en las actividades establecidas en el diagrama WBS) donde se pueden identificar las fases y actividades con su respectiva duración, de esta forma las actividades quedan ordenadas de forma cronológica y se puede apreciar el avance de forma gráfica. Asimismo, se incluyen los hitos estipulados para el proyecto, los cuales son las entregas de informes para la universidad, y las fechas en que los prototipos deben estar finalizados. La carta Gantt de proyecto 1 se puede visualizar en la figura número 5, mientras que la planificación del proyecto 2, en la figura 6 del informe.

Con estas herramientas se puede planificar de manera eficaz el trabajo a realizar, permitiendo analizar los tiempos necesarios para realizar las tareas que se estipularon en un comienzo, y de esta forma, guiar a los desarrolladores en la realización del proyecto.

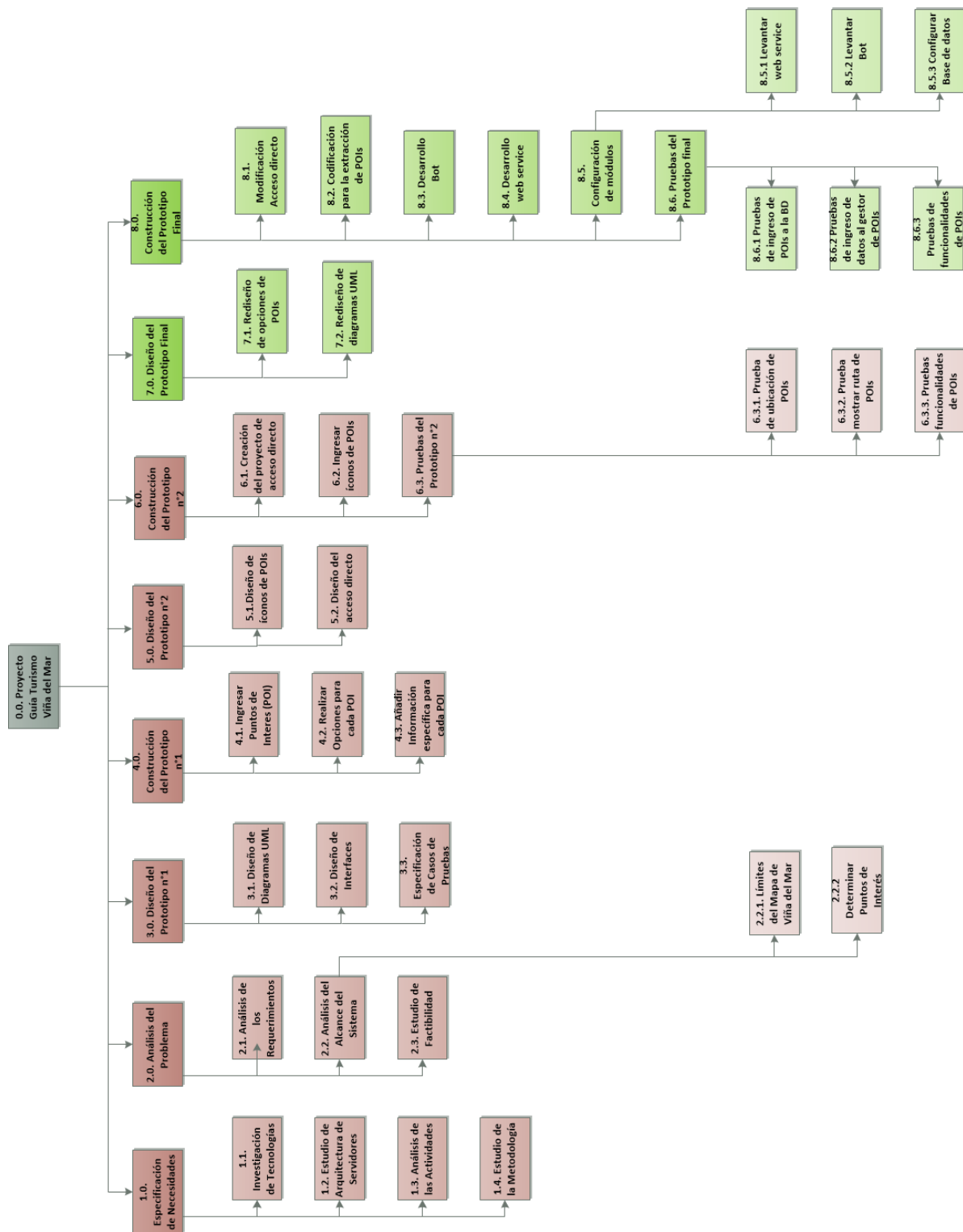


Figura 2: Estructura de Desglose de Trabajo



## 3.2.1 Carta Gantt

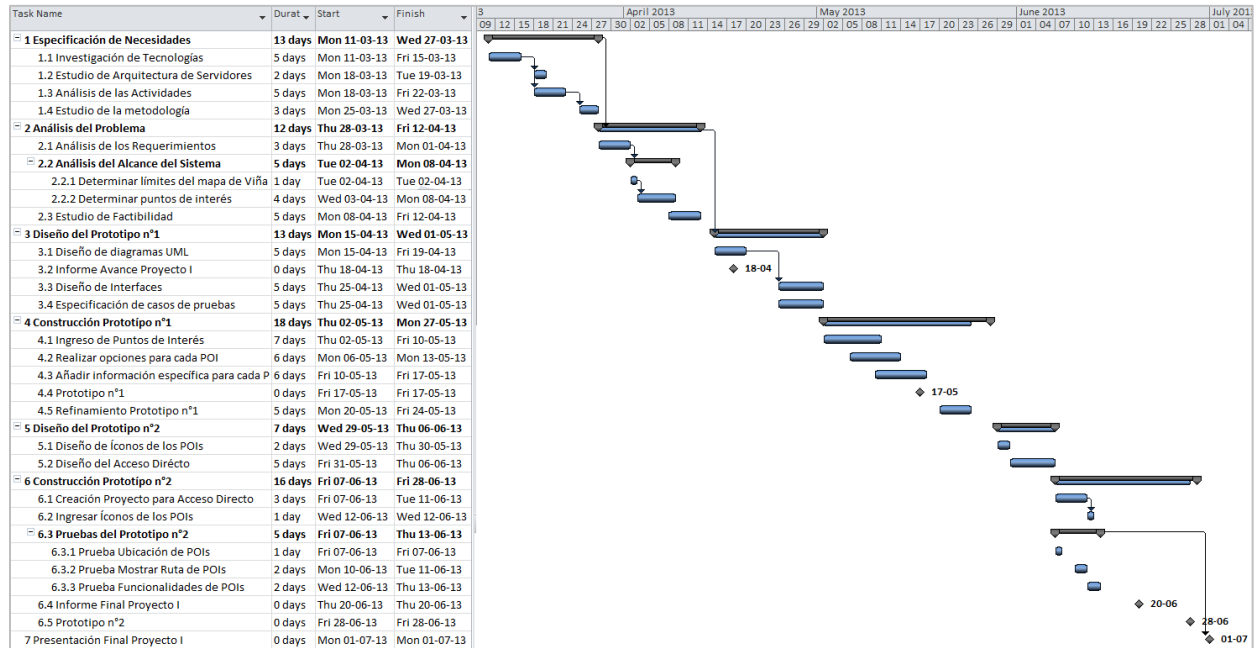


Figura 3: Carta Gantt – Proyecto I

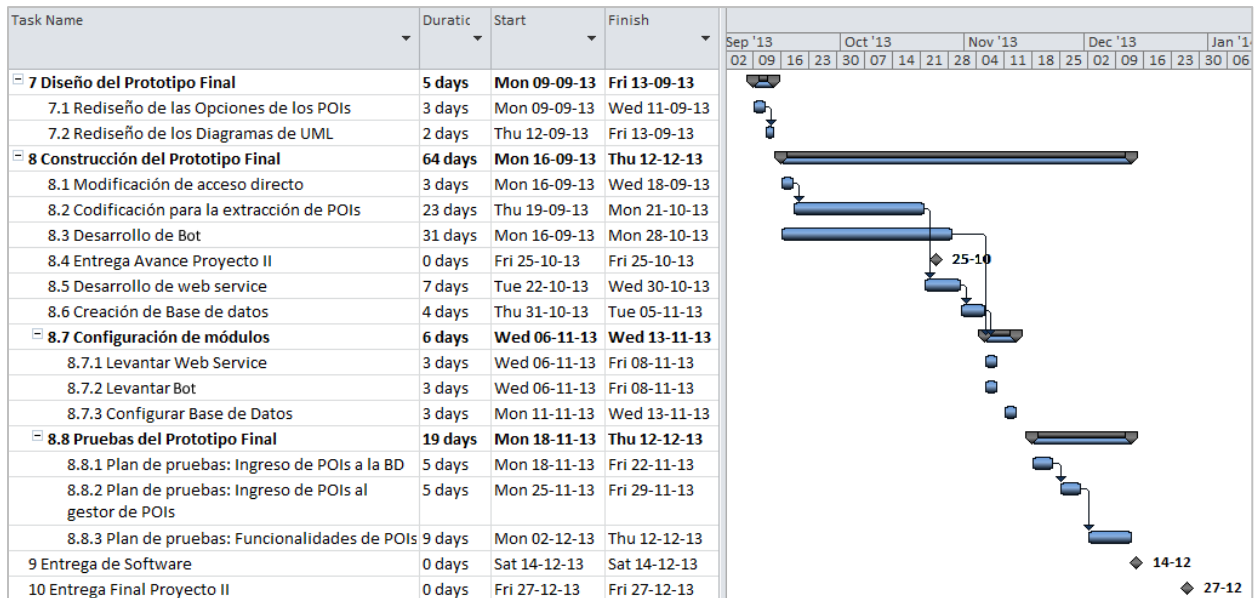


Figura 4: Carta Gantt – Proyecto II

### 3.3 Herramientas

A continuación se definen las herramientas que se utilizarán en el desarrollo del proyecto, tanto en la etapa de implementación, como en la de pruebas.

1. **Eclipse IDE:** Plataforma utilizada para desarrollar típicamente entornos de desarrollo integrados, como por ejemplo el de Java llamado Java Development Toolkit (JDT) y el compilador (ECJ). Además es un sistema extensible gracias a la integración de diferentes *plugins*, como por ejemplo algunos utilizados para complementar el desarrollo en aplicaciones en Android.
2. **Android SDK:** Es un kit de desarrollo de *software* (*Software Development Kit* en inglés), en donde están las herramientas necesarias para la creación de aplicaciones basadas en el sistema operativo Android. Este kit provee las librerías de la API y herramientas como tutoriales, depuradores, entre otras.
3. **Layar Platform:** Es una herramienta online la cual facilita a los desarrolladores crear diversos tipos de experiencias en realidad aumentada, como por ejemplo usar la geolocalización para mostrar los distintos puntos de interés seleccionados.
4. **Hoppala:** Es una plataforma online que permite el almacenamiento y administración de la información de los puntos de interés que incorpora la aplicación. Será utilizado en los prototipos 1 y 2 del proyecto.
5. **Gestor de POIs:** Es un sistema web desarrollado por ex alumnos de la Universidad de Playa Ancha [2], que permite ingresar automáticamente los puntos de interés mediante un *bot*. Será utilizado en el prototipo final del proyecto.
6. **API Google Places:** Es un servicio gratuito que permite extraer información de los puntos de interés incorporados en los mapas de Google, tales como coordenadas geográficas, nombre de POI, dirección, número telefónico, imagen referencial, etc.
7. **NetBeans:** Entorno de desarrollo integrado (IDE) libre, creado principalmente para el lenguaje de programación Java. Será utilizado para desarrollar, en lenguaje PHP, el *web service* y el *bot* del prototipo final.
8. **cURL:** Es una herramienta para usar en un intérprete de comandos para transferir archivos con sintaxis URL, soporta distintos protocolos, subidas FTP, Kerberos, subidas mediante formulario HTTP, proxies, cookies, autenticación mediante usuario+contraseña y muchas otras prestaciones [3]. cURL es utilizado para desarrollar el bot.

## 4 Análisis de Tecnologías

### 4.1 Realidad Aumentada

La realidad aumentada (en inglés *Augmented Reality*) es una tecnología que funciona como un “lente”, el cual permite la superposición en tiempo real, de imágenes marcadores o información generados virtualmente, sobre imágenes del mundo real. De esta forma, se fusiona la información de objetos virtuales con la realidad, dando la impresión de que ambas pertenecen al mismo entorno. Básicamente, la información del mundo real se complementa con la del digital, a través del lente de un dispositivo. En la figura adjunta se aprecia el esquema del concepto de la RA.



Figura 5: Esquema del concepto RA Fuente: [4]

Para componer un sistema de realidad aumentada son necesarios cuatro requisitos básicos:

- Contar con un dispositivo que capture las imágenes del entorno, por ejemplo, una cámara incorporada en un teléfono móvil.
- Contar con una pantalla que proyecte las imágenes capturadas por una cámara y las imágenes virtuales.
- Es preciso tener un elemento de procesamiento, que interpreten del mundo real que capta el usuario, generar la información virtual y mezclarla de forma adecuada.
- Se necesita un activador de realidad aumentada, que podría ser, la imagen que están visualizando los usuarios, ya que a partir de ella el sistema debería reaccionar. Sin embargo, este proceso tiene una complejidad alta y se utilizan otros elementos que lo sustituyen, tales como el GPS de los *Smartphone*, brújulas y acelerómetros, los cuales permiten posicionar y orientar dichos dispositivos. [4]

## 4.2 Sistema Operativo Android

Android es un sistema operativo diseñado en primera instancia para dispositivos móviles con pantalla táctil, tales como smartphones y tablets. Fue desarrollado por Android, Inc., una empresa que más tarde compró Google. Es un sistema operativo que está basado en Linux, disponiendo de un Kernel y utilizando una máquina virtual sobre este, que es la responsable de convertir el código en Java de las aplicaciones, a código capaz de comprender el Kernel.

Android posee una tienda oficial llamada Google Play donde se encuentran todas las aplicaciones disponibles para este sistema, en la actualidad existen más de 600.000 aplicaciones, de las cuales dos tercios son gratuitas, superando la cifra de 25.000 millones de descargas, convirtiéndose así en el sistema operativo más utilizado en *smartphones*. [5]

La arquitectura de Android está formada por varias capas que facilitan al desarrollador la creación de aplicaciones, cada una de estas utiliza elementos de la capa inferior para realizar sus funciones. La capa inferior es el kernel de Linux, luego las librerías, la capa de Android *runtime*, el *Framework* de aplicaciones, y en la capa superior se encuentran las aplicaciones. [6]

## 4.3 Layar

Layar es un navegador de realidad aumentada para Android, su funcionamiento se basa en usar la información que proporciona el GPS y la brújula que posee el terminal, mientras la pantalla muestra lo que la cámara capta y sobre ella información relativa en tiempo real de lo que se tiene en frente.

Resulta realmente potente lo que se puede llegar a hacer con este sistema, al apuntar con el terminal Android en una dirección se puede recibir detalles como puntos de interés, casas en venta, restaurantes, cajeros automáticos, etc. Además la aplicación muestra con una interfaz bastante intuitiva la distancia hasta los puntos en cuestión y la ruta a seguir para llegar al elemento mediante la comunicación que se realiza con Google Maps.

La idea de Layar no termina en este funcionamiento, los desarrolladores creen que un punto importante estará en la posibilidad de compartir información entre usuarios, ya que la aplicación permite a los usuarios marcar un punto en el mapa e introducir datos sobre la ubicación. [7]

## 4.4 Bot

Un bot de internet, también conocido como *web robot* (robot de la web en inglés), WWW robot o simplemente bot, es un software que corre tareas programas en internet. Por lo general los bots realizan tareas simples y estructuralmente repetitivas, a una velocidad mucho más alta que como las haría un humano. Los bots más utilizados son los llamados robot araña o *web spider*, los cuales utilizan scripts automatizados para analizar y extraer información de servicios webs.

En sitios wiki, como Wikipedia, un bot puede realizar funciones rutinarias de edición. En otros sitios, como Youtube, el bot puede responder a cuestiones sobre el propio contenido del sitio. En sitios de chat en línea, algunos bots fueron utilizados para simular una persona, intentando hacer creer al "ciber inter locutor" que chateaban con una persona real. Es por ello

que, aquellos sitios que hacen un uso positivo de los bots, como Wikipedia o Encarta, tienen reglas muy estrictas para su uso.

Asimismo, otros bots pueden abrir cuentas en sitios de correo electrónico gratuito, por lo que ahora la mayoría de esos sitios cuenta con un paso adicional en el que el usuario debe escribir una palabra presentada en formato gráfico algo muy difícil de interpretar para un bot. [8]

En el caso de este proyecto se utilizó un bot de inserción de datos en formularios HTML de un sitio de internet. Este fue desarrollado en lenguaje PHP y se utilizó la herramienta cURL. A continuación se muestra un diagrama de actividad que explica gráficamente la forma en que funciona el bot mediante cURL.

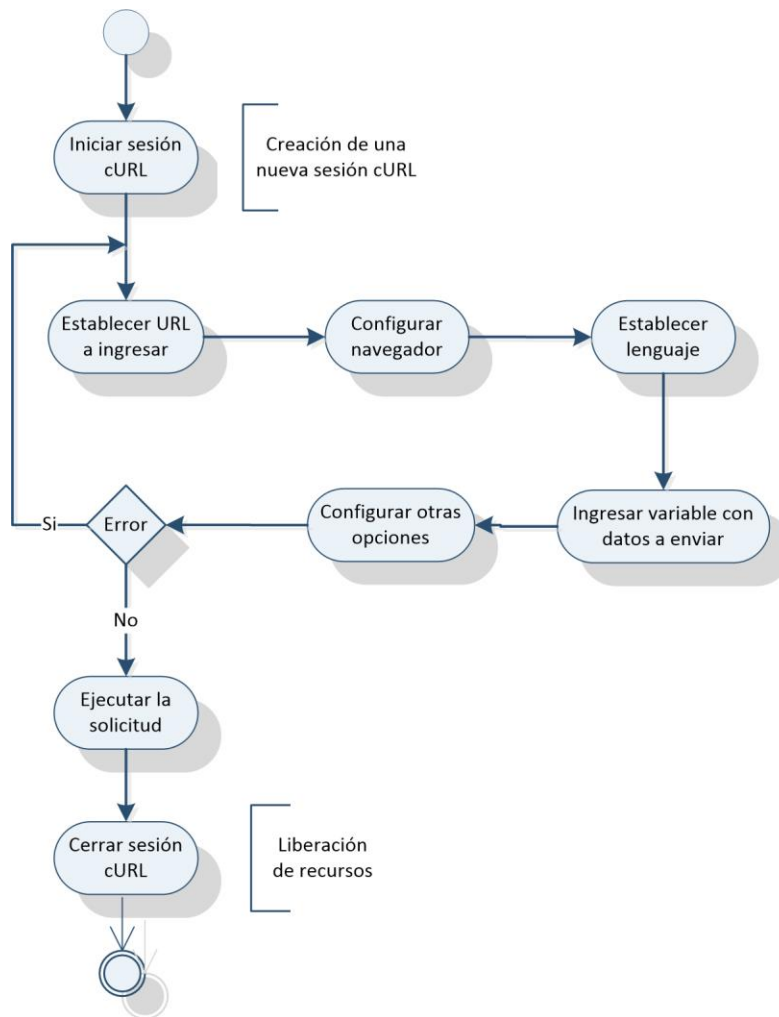


Figura 6: Diagrama funcionamiento Bot

## 4.5 API Google Places

El API de Google Places es un servicio gratuito que retorna información sobre lugares definidos dentro de esta API tales como establecimientos o importantes puntos de interés, mediante peticiones HTTP. La solicitud de un lugar especifica ubicaciones como coordenadas geográficas (latitud/longitud).

El servicio dispone de distintas solicitudes, tales como búsqueda, detalles, acciones y fotos de un lugar específico. Cada uno de los servicios es accedido como una petición HTTP y devuelve una respuesta JSON o XML. Todas las solicitudes para un servicio deben utilizar el protocolo https:// e incluir una clave y un parámetro sensor.

El API de Google Places utiliza una clave de API para identificar la aplicación. Las claves API se gestionan a través de la consola de API Google. Algunas características de la API, como resúmenes de revisión y ciertos filtros de peticiones, requieren licencias adicionales, disponibles como parte de la licencia de la empresa API Places. [9]

## 4.6 Lenguajes de Programación

### 4.6.1 PHP

PHP *Hypertext Pre-processor* es un lenguaje de programación de uso general de código del lado del servidor, originalmente diseñado para el desarrollo web de contenido dinámico. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera la página Web resultante.

Una de las ventajas de este lenguaje es que permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, PostgreSQL, Oracle, ODBC, DB2, Microsoft SQL server Server, Firebird y SQLite. Además tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos, tales como Unix (y de ese tipo, como Linux o Mac OS X) y Microsoft Windows, y puede interactuar con los servidores de web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI. [10]

### 4.6.2 SQL

El lenguaje de consulta estructurado o SQL (*structured query language*) es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones en ellas. Una de sus características es el manejo del álgebra y el cálculo relacional que permiten efectuar consultas con el fin de recuperar de forma sencilla información de interés de bases de datos, así como hacer cambios en ella.

Es un lenguaje declarativo de "alto nivel" o "de no procedimiento" que, gracias a su fuerte base teórica y su orientación al manejo de conjuntos de registros (y no a registros individuales) permite una alta productividad en codificación y la orientación a objetos. [11]

## 5 Estudio de Factibilidad

Con la finalidad de conocer si el desarrollo del proyecto es viable, se han analizado comprensivamente los aspectos económicos, técnicos, operacionales, y legales. Estos resultados se detallan a continuación.

### 5.1 Técnica

El equipo dispone de las herramientas necesarias para llevar a cabo el proyecto, tanto en las etapas de diseño, implementación y pruebas del *software*. Con respecto a los materiales requeridos, se cuenta con dos computadores portátiles para el desarrollo de la aplicación, además de dos dispositivos móviles para la etapa de pruebas.

En cuanto a las habilidades y conocimientos requeridos para el desarrollo del proyecto, los programadores no poseen experiencia en la tecnología a utilizar, sin embargo, cuentan con grandes capacidades de aprendizaje por medio de investigación, asimismo de las competencias y habilidades que han adquirido a lo largo de la carrera.

### 5.2 Económica

El proyecto será desarrollado por dos alumnos de la escuela de informática de la Pontificia Universidad Católica de Valparaíso, netamente con fines académicos. El sistema no posee fines de lucro, por lo tanto no habrá ganancias a futuro. A continuación se detallan los costos de *software*, hardware y mano de obra asociados al proyecto.

<i>Aspectos</i>	<i>Costos</i>
<i>Costo de Software</i>	Las herramientas que se utilizarán corresponden a <i>software</i> gratuitos, por lo tanto no habrá costos en las licencias de los programas.
<i>Costo de Hardware</i>	Los alumnos cuentan con dos computadores portátiles avaluados en \$800.000. Además de dos dispositivos móviles con sistema operativo Android, estos están avaluados en \$200.000 aproximadamente. Debido a que los materiales descritos pertenecen a los desarrolladores, no habrá costos asociados al hardware.
<i>Costo de Mano de obra</i>	Debido a que la implementación de la aplicación será desarrollada por los alumnos, no se considerará el costo de personal.

Tabla 1: Costos del Proyecto

## 5.3 Operativa

Este análisis se realiza para determinar si existe una demanda por la aplicación que justifique el desarrollo de esta. En este caso analizamos tres aspectos de factibilidad operacional, los cuales son los siguientes:

1. El uso del sistema puede resultar demasiado complejo: Debido a que la aplicación está destinada a usuarios que poseen *Smartphones*, se supone que ellos están familiarizados con otras aplicaciones disponibles para estos dispositivos móviles, por consiguiente, no sería difícil para ellos acostumbrarse al uso del sistema. Además, la aplicación no posee una complejidad muy alta respecto a su utilización, lo que se traduce en un *software* de fácil manejo y comprensión.
2. El sistema puede hacer que los usuarios se resistan a él como consecuencia del miedo a ser desplazados, intereses en el sistema antiguo u otras razones: La finalidad del uso de la aplicación es fomentar el turismo, por lo tanto, este existirá para ayudar a los turistas, de acuerdo a esto, se supone que las personas no se sentirán desplazadas por el sistema. En cuanto a los intereses en el sistema antiguo, la aplicación no sustituirá una existente, en consecuencia, se aprueba el aspecto estudiado.
3. La probabilidad de la obsolescencia subsecuente en el sistema: Debido a que la tecnología que se utilizará en la aplicación es interesante para muchos de los usuarios de *Smartphones*, además de ser una tecnología innovadora y relativamente nueva, se supone que no quedará obsoleta en un corto plazo, al contrario, se prevé que seguirá evolucionando de acuerdo madura la tecnología, sobretodo en términos de usabilidad y funcionalidad.

## 5.4 Legal

El equipo cuenta con todas las licencias de las herramientas para desarrollar el *software*, ya que el servicio que entrega Layar y Hoppala es gratuito, y las plataformas Eclipse y NetBeans, son *software* libres. Con respecto al gestor de POIs, contamos con los permisos de los propietarios del sistema, para hacer uso del mismo. Y en cuanto a los servicios proporcionados por Google Places, se pueden obtener a través de un registro gratuito de Google. Por lo tanto, no se infringe ninguna norma o ley vigente establecida, lo que permite realizar legalmente el proyecto.

Respecto a lo planteado en los cuatro puntos anteriores, se observa que el desarrollo de la aplicación es factible en todos los aspectos analizados, por consiguiente, se confirma la viabilidad del proyecto.



## 6 Análisis y Diseño del Sistema

A continuación se presentan los requerimientos funcionales y no funcionales del *software* que guiarán el desarrollo de éste.

### 6.1 Requerimientos Funcionales

- Mostrar los puntos de interés más cercanos al usuario, de acuerdo al rango de búsqueda de la aplicación. (ID1)
- Poder seleccionar uno de los puntos de interés que aparezcan en la pantalla y ver sus respectivas opciones. (ID2)
- El sistema debe permitir al usuario cambiar el modo de vista de la aplicación a modo mapa. (ID3)
- Acceder a la página web del punto seleccionado. (ID4)
- Realizar una llamada telefónica al lugar seleccionado. (ID5)
- Conocer la ruta a seguir para llegar a un punto de interés. (ID6)
- Permitir compartir la aplicación, a través de Facebook y Twitter. (ID7)

### 6.2 Requerimientos no Funcionales

- El sistema deberá estar disponible a toda hora.
- La información de los puntos de interés debe ser fidedigna.
- La aplicación será desarrollada para implementarse en *smartphones* con sistema operativo Android 2.2 o superior.
- El teléfono del usuario debe tener acceso a internet a través de Wi-Fi, 3G o 4G.

### 6.3 Arquitectura

A continuación se presenta la arquitectura diseñada para realizar el prototipo final, la cual se puede visualizar gráficamente en la figura N°7. Esta arquitectura está formada por los siguientes componentes:

- Aplicación Android: Encargada de extraer la información de los puntos de interés usando la API Google Places, y luego proceder con los métodos correspondientes para abrir la aplicación de Layar y la capa de nuestra aplicación.
- Web Services: Intermediario entre la aplicación Android, la base de datos y el robot. Programado en PHP.
- Base de Datos: Lugar donde se recopilarán los datos de los puntos de interés proporcionados por la API de Google Places. Estará alojada en un servidor gratuito.
- Bot: Será el encargado de extraer los datos de los POIs de la base de datos cada cierto tiempo, y agregarlos al gestor de POIs. Será desarrollado en PHP.

- Gestor de POIs: Son los servicios que entregan terceras personas no vinculadas directamente con el sitio Layar. En nuestro caso se utilizará el gestor de POIs creado por estudiantes de la UPLA.
- Proxy Layar: Intermediario entre el gestor de POIs y el cliente de layar, enviará los datos al servidor de Layar donde se verán en el cliente ubicado en el celular.
- Cliente Layar: Es la aplicación que los usuarios utilizan en sus *smartphones*.

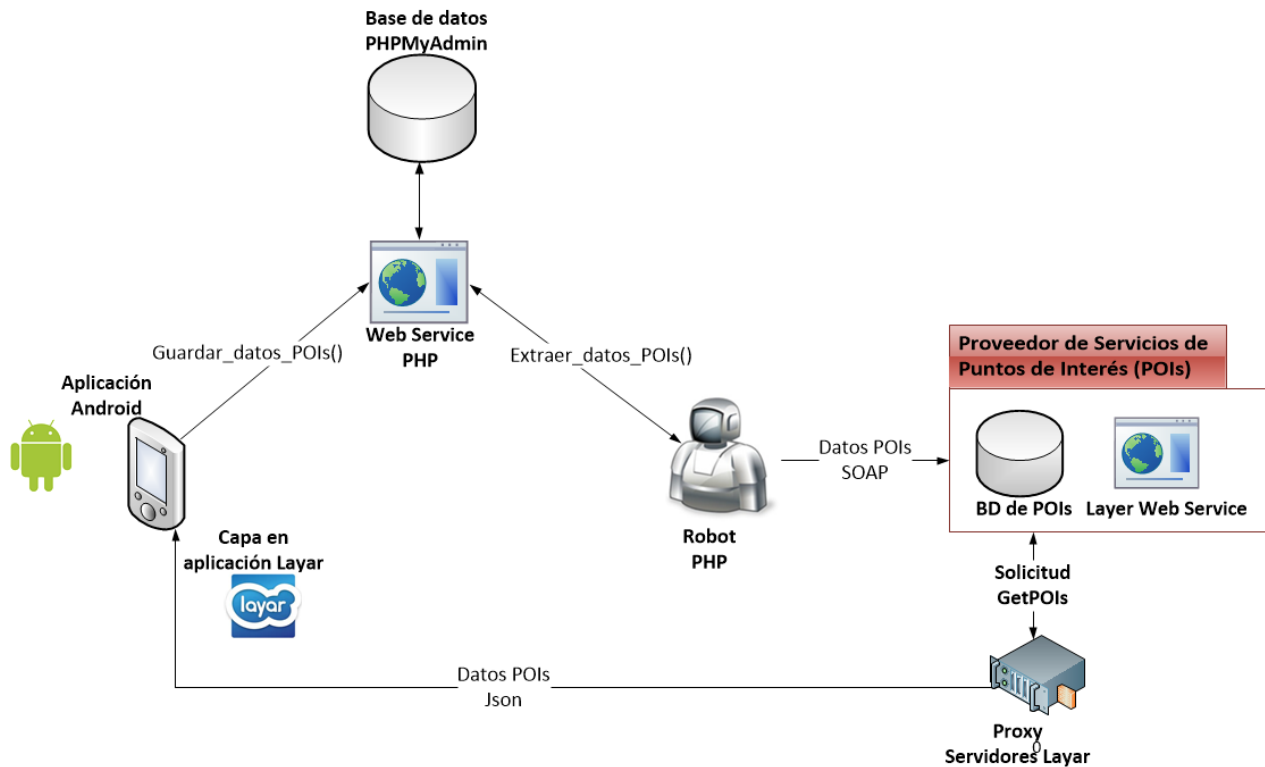


Figura 7: Arquitectura

## 6.4 Diagramas del Sistema

A continuación se presentan algunos diagramas del proyecto para comprender de forma gráfica las funcionalidades del sistema.

### 6.4.1 Caso de uso Seleccionar POI

Con la finalidad de modelar la interacción del usuario con el sistema y visualizar las funcionalidades de este, se presenta el diagrama de caso de uso seleccionar POI, que muestra las opciones que se pueden realizar tras seleccionar un punto de interés en la aplicación.

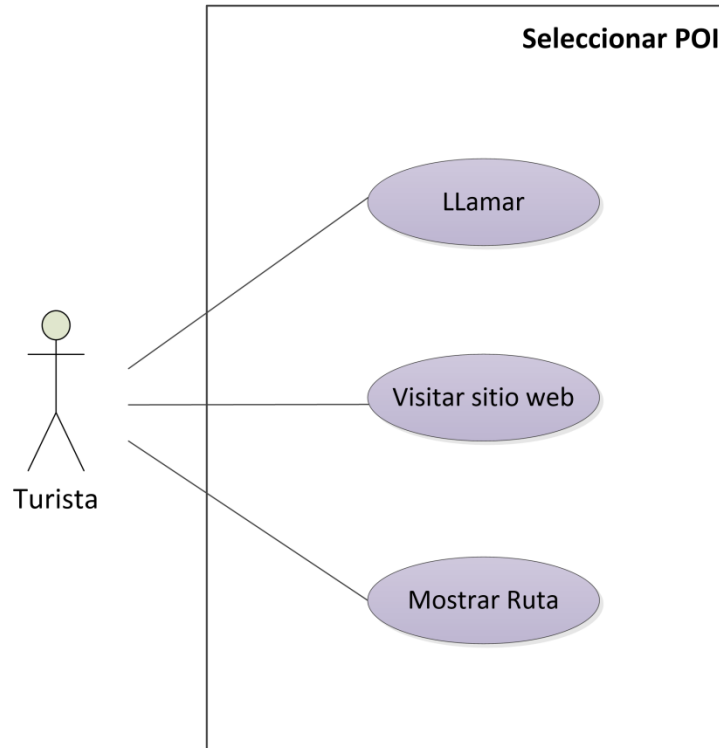


Figura 8: Caso de uso seleccionar POI

### 6.4.2 Diagrama de Actividad seleccionar POI

Un diagrama de actividades muestra una visión simplificada de lo que ocurre durante una operación o proceso. La figura adjunta corresponde al diagrama de actividad tras seleccionar un POI.

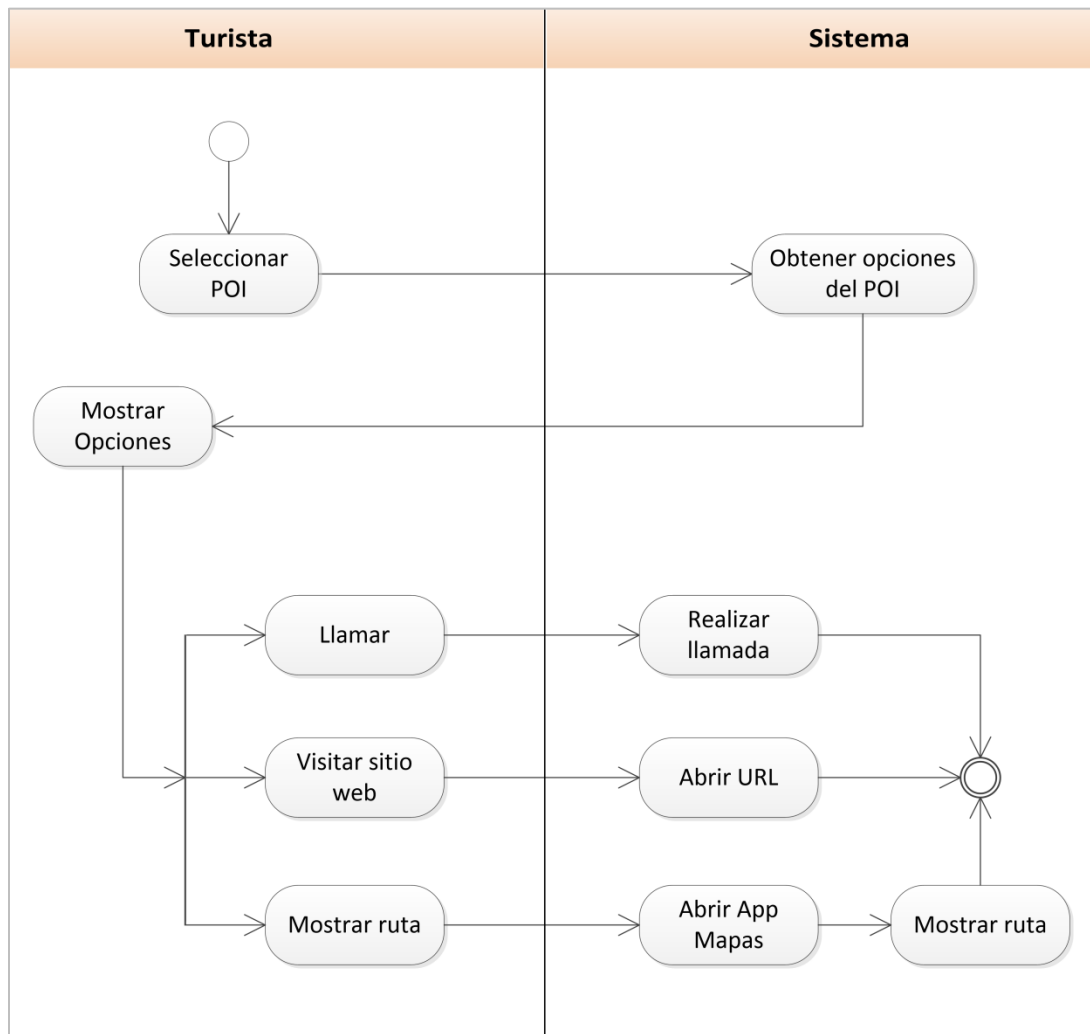


Figura 9: Diagrama de Actividad Seleccionar POI

### 6.4.3 Plan de Pruebas

El plan de pruebas sirve para planificar los diferentes escenarios de pruebas con mayor importancia. Estos escenarios de pruebas corresponden al de los prototipos y se van refinando hasta llegar al prototipo final. Su objetivo es asegurar que el *software* cumpla con las especificaciones requeridas y eliminar los posibles defectos que este pudiera tener, y están orientadas al quebramiento del sistema, por lo tanto las realizan habitualmente desarrolladores que no estuvieron involucrados en la construcción del sistema. Para llevar a cabo el plan de pruebas se a continuación se explicitan el alcance, enfoque, recursos requeridos, calendario, responsables y manejo de riesgos de un proceso de pruebas. Además se realizarán pruebas funcionales, las cuales tienen por objetivo probar que los sistemas desarrollados, cumplan con las funciones específicas para los cuales han sido creados. A continuación se detalla el plan de prueba para cada uno de los escenarios estipulados.

#### 6.4.3.1 Plan de Prueba Ingreso de POI's a la Base de Datos

<b>Identificador del plan</b>	PP-Ingreso-POIs-BD
<b>Alcance</b>	Se probará el ingreso de la información de los puntos de interés, desde la aplicación Android, a la base de datos.
<b>Items a probar</b>	Se realizarán pruebas de sistema utilizando el proyecto Android, el <i>Web Service</i> y la base de datos. Se debió realizar la pertinente conexión entre estos 3 ítems.
<b>Estrategia</b>	Se realizarán pruebas de caja negra para la verificación del ingreso de los datos a la BD.
<b>Categorización de la configuración</b>	El plan se debe suspender en el caso de que uno o más de los datos de los puntos de interés no fueron ingresados con éxito.
<b>Tangibles</b>	Especificaciones de casos de pruebas.
<b>Procedimientos especiales</b>	<ul style="list-style-type: none"><li>• Se debe tener la cuenta de Layar donde se encuentra la capa de la aplicación.</li></ul>
<b>Recursos</b>	<ul style="list-style-type: none"><li>• Es necesario para utilizar la plataforma Layar un celular <i>smartphone</i> con sistema operativo Android 2.2 o superior.</li><li>• Se requiere de un integrante para poder realizar este plan.</li></ul>
<b>Calendario</b>	<ul style="list-style-type: none"><li>• Comienzo: 18 de noviembre del 2013</li><li>• Duración: 5 días.</li></ul> Hito: Ingreso de todos los datos de los POIs fue con éxito.
<b>Manejo de riesgos</b>	No existen riesgos en cuanto a la manipulación de datos, ya que solo se ingresarán puntos de interés.
<b>Responsables</b>	Esteban Pavez.

Tabla 2: Plan de Prueba Ubicación de POI's

### 6.4.3.2 Plan de Prueba Ingreso de POIs al Gestor de POIs

<b>Identificador del plan</b>	PP-Ingreso-POIs-Gestor
<b>Alcance</b>	Se probará la extracción de la información de los POIs desde la base de datos al gestor de puntos de interés.
<b>Items a probar</b>	Se realizarán pruebas de sistema utilizando la base de datos, el <i>Bot</i> y el gestor de POIs.
<b>Estrategia</b>	Se realizarán pruebas de caja negra para la verificación de los puntos de interés al gestor de POIs.
<b>Categorización de la configuración</b>	El plan se debe suspender en el caso de que uno o más de los datos de los puntos de interés no fueron ingresados con éxito.
<b>Tangibles</b>	Especificaciones de casos de pruebas.
<b>Procedimientos especiales</b>	<ul style="list-style-type: none"><li>• Se debe tener una cuenta registrada en el gestor de POIs.</li><li>• Se debe haber ejecutado el <i>Bot</i> para que extraiga los datos de la BD.</li></ul>
<b>Recursos</b>	<ul style="list-style-type: none"><li>• Se requiere de un integrante para poder realizar este plan.</li></ul>
<b>Calendario</b>	<ul style="list-style-type: none"><li>• Comienzo: 25 de noviembre del 2013</li><li>• Duración: 5 días.</li></ul> Hito: Ingreso de todos los datos de los POIs fue con éxito.
<b>Manejo de riesgos</b>	No existen riesgos en cuanto a la manipulación de datos, ya que solo se ingresarán puntos de interés.
<b>Responsables</b>	Carolina Villanueva.

Tabla 3: Plan de Prueba Mostrar ruta de POI

### 6.4.3.3 Plan de Prueba Funcionalidades de POIs

<b>Identificador del plan</b>	PP-Funcionalidades-POI
<b>Alcance</b>	Se probarán las funcionalidades de cada punto de interés exceptuando la de mostrar ruta.
<b>Items a probar</b>	Se probarán las opciones del punto de interés. Los puntos de interés deben haber sido ingresados en el mapa junto con sus opciones.
<b>Estrategia</b>	Se aplicará la estrategia de caja-negra.
<b>Categorización de la configuración</b>	Si se encuentra alguna imperfección en un punto de interés específico se puede continuar el plan a menos que sea un fallo general.
<b>Tangibles</b>	Especificaciones de casos de pruebas.
<b>Procedimientos especiales</b>	<ul style="list-style-type: none"><li>• Se debe tener la cuenta de Layar donde se encuentra la capa</li></ul>

	de la aplicación. <ul style="list-style-type: none"> <li>• Ubicarse a una distancia de 5 km como máximo del área estipulada para los POI's de la aplicación.</li> </ul>
<b>Recursos</b>	<ul style="list-style-type: none"> <li>• Es necesario para utilizar la plataforma Laya un celular <i>smartphone</i> con sistema operativo Android 2.2 o superior.</li> <li>• Se requiere de dos integrantes para poder realizar este plan.</li> </ul>
<b>Calendario</b>	<ul style="list-style-type: none"> <li>• Comienzo: 2 de diciembre del 2013.</li> <li>• Duración: 9 días.</li> </ul> Hito: Revisión total de los puntos de interés
<b>Manejo de riesgos</b>	No existen riesgos en cuanto a la manipulación de datos ya que solo se manejarán los puntos de interés para verificar las funcionalidades de cada punto de interés.
<b>Responsables</b>	Esteban Pavez.

Tabla 4: Plan de Prueba Funcionalidades de POI's

## 6.4.4 Especificación de Casos de Prueba

Los casos de pruebas son un conjunto de condiciones o variables bajo las cuáles el analista determinará si el requisito de una aplicación es parcial o completamente satisfactorio. Se pueden realizar muchos casos de prueba para determinar que un requisito es completamente satisfactorio. Con el propósito de comprobar que todos los requisitos de una aplicación son revisados, debe haber al menos un caso de prueba para cada requisito. A continuación se presenta la especificación de los casos de prueba, en los que se explica cómo realizar los diferentes escenarios identificados en el plan de pruebas. Además, se muestran los resultados obtenidos para cada uno de los escenarios estipulados anteriormente.

### 6.4.4.1 Caso de Prueba de Ingreso de POIs a la Base de Datos

Visión general	
<b>Identificador</b>	P1.0
<b>Creador de caso de prueba</b>	Esteban Pavez
<b>Versión</b>	1.0
<b>Nombre</b>	Caso de Prueba de Ingreso de POIs a la Base de Datos
<b>Id requerimientos</b>	ID1
<b>Propósito</b>	Poder verificar el correcto ingreso de la información de los puntos de interés a la base de datos.
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• <i>web service</i> levantado en un servidor en línea.</li> <li>• Base de datos configurada para recibir datos de forma remota.</li> </ul>
Actividades	
<b>Ambiente de prueba/</b>	Free Hosting Nixiweb

<b>Configuración</b>	
<b>Inicialización</b>	<ul style="list-style-type: none"> <li>• Abrir a la aplicación Guía de Turismo.</li> </ul>
<b>Finalización</b>	<ul style="list-style-type: none"> <li>• Si algún dato de uno de los puntos de interés es errónea o no se ingresó se debe revisar el procedimiento almacenado y el <i>web service</i>.</li> </ul>
<b>Acciones</b>	<ul style="list-style-type: none"> <li>• Revisar la tabla Datos_POIs en la base de datos BDTURISMO.</li> </ul>
<b>Datos de entrada</b>	<ul style="list-style-type: none"> <li>• Nombre de (los) POI(s).</li> <li>• Dirección del (de los) POI(s).</li> <li>• Teléfono del (de los) POI(s).</li> <li>• Latitud y Longitud del POI(s).</li> </ul>
<b>Resultados</b>	
<b>Salida esperada</b>	Visualizar la información de los puntos de interés, en la base de datos, que se ubican a un radio de 1km del usuario.
<b>Salida obtenida</b>	Efectivamente se ingresan los puntos de interés en la base de datos,
<b>Resultado</b>	Correcto.
<b>Severidad</b>	Grave
<b>Seguimiento</b>	No se encontraron defectos en este caso de pruebas.
<b>Estado</b>	Finalizado.

Tabla 5: Caso de Prueba: Ingreso de POIs a la Base de Datos

#### 6.4.4.2 Caso de Prueba Ingreso de Puntos de Interés al Gestor de POIs

<b>Visión general</b>	
<b>Identificador</b>	P2.0
<b>Creador de caso de prueba</b>	Carolina Villanueva
<b>Versión</b>	1.0
<b>Nombre</b>	Caso de Prueba de Ingreso de Puntos de Interés al Gestor de POIs.
<b>Id requerimientos</b>	ID1
<b>Propósito</b>	Poder verificar el correcto ingreso de la información de los puntos de interés al gestor de POIs.
<b>Dependencias</b>	<ul style="list-style-type: none"> <li>• <i>web service</i> levantado en un servidor en línea.</li> <li>• Base de datos con información verídica de los puntos de interés entregada por la API de Google Places.</li> <li>• <i>Bot</i> ejecutándose y programado para extraer información de la base de datos.</li> </ul>
<b>Actividades</b>	



<b>Ambiente de prueba/ configuración</b>	Gestor de POIs.
<b>Inicialización</b>	<ul style="list-style-type: none"> <li>• Ejecutar el <i>Bot</i>.</li> </ul>
<b>Finalización</b>	<ul style="list-style-type: none"> <li>• Si algún dato de uno de los puntos de interés es errónea o no se ingresó se debe revisar el procedimiento almacenado, el <i>web service</i> y el bot.</li> </ul>
<b>Acciones</b>	<ul style="list-style-type: none"> <li>• Revisar el mapa con los puntos de interés ingresados.</li> <li>• Revisar la información de los POIs.</li> </ul>
<b>Datos de entrada</b>	<ul style="list-style-type: none"> <li>• Nombre de (los) POI(s).</li> <li>• Dirección del (de los) POI(s).</li> <li>• Teléfono del (de los) POI(s).</li> <li>• Latitud y Longitud del POI(s).</li> </ul>
<b>Resultados</b>	
<b>Salida esperada</b>	Visualizar los puntos de interés en el gestor de POIs, con sus correspondientes acciones, íconos y tipos.
<b>Salida obtenida</b>	Efectivamente se ingresan los puntos de interés en el gestor con su información correspondiente.
<b>Resultado</b>	Correcto.
<b>Severidad</b>	Grave
<b>Seguimiento</b>	No se encontraron defectos en este caso de pruebas.
<b>Estado</b>	Finalizado

Tabla 6: Caso de Prueba: Mostrar ruta

#### 6.4.4.3 Caso de Prueba de funcionalidades de POIs

<b>Visión general</b>	
<b>Identificador</b>	P3.0
<b>Creador de caso de prueba</b>	Esteban Pavez.
<b>Versión</b>	1.0
<b>Nombre</b>	Caso de Prueba de funcionalidades de POI.
<b>Id requerimientos</b>	ID2, ID3, ID4, ID5, ID6, ID7.
<b>Propósito</b>	Poder comprobar que las opciones de los puntos de interés funcionen correctamente.
<b>Dependencias</b>	La base de datos del gestor de POIs tenga puntos ingresados con las acciones que le corresponda.
<b>Actividades</b>	
<b>Ambiente de prueba/ configuración</b>	Ambiente de desarrollo de layar.

<b>Inicialización</b>	<ul style="list-style-type: none"> <li>• Ingresar a la plataforma layar.</li> <li>• Ingresar la cuenta creada donde se ubica la capa de la aplicación.</li> <li>• Seleccionar algún Punto de interés.</li> </ul>
<b>Finalización</b>	<ul style="list-style-type: none"> <li>• Si la aplicación no muestra el número de teléfono correcto se debe modificar en la base de datos.</li> <li>• Si la aplicación no muestra la página web del punto de interés se debe modificar en la base de datos.</li> <li>• Si la aplicación no muestra la ruta de cómo llegar al lugar indicado.</li> </ul>
<b>Acciones</b>	<ul style="list-style-type: none"> <li>• Seleccionar la funcionalidad de llamar al lugar y verificar si corresponde o no.</li> <li>• Seleccionar la funcionalidad de ingresar a la página web del lugar y verificar si corresponde o no.</li> <li>• Seleccionar la funcionalidad de “cómo llegar” y verificar si corresponde o no.</li> </ul>
<b>Datos de entrada</b>	<ul style="list-style-type: none"> <li>• Número telefónico del lugar.</li> <li>• Página web del lugar.</li> </ul>
<b>Resultados</b>	
<b>Salida esperada</b>	Visualizar las opciones correspondientes al punto de interés seleccionado, y revisar la correcta funcionalidad de cada una de ellas.
<b>Salida obtenida</b>	Se visualizan las opciones correspondientes al punto de interés seleccionado, y se realiza cada opción de forma exitosa.
<b>Resultado</b>	Correcto.
<b>Severidad</b>	Mayor.
<b>Seguimiento</b>	No se encontraron defectos en este caso de prueba.
<b>Estado</b>	Finalizado.

Tabla 7: Caso de Prueba: Funcionalidades de POI's

## 7 Desarrollo del Sistema

### 7.1 Primer Prototipo

El objetivo de la realización del primer prototipo es el de poder armar la arquitectura que es necesaria para trabajar con aplicaciones utilizando la realidad aumentada. Para esto en primer lugar se debió crear una capa en el servidor de Hoppala, en donde está nuestra base de datos de puntos de interés. Esta capa posee una dirección u “overlay URL”, la cual se utiliza para realizar la conexión al servidor que provee la realidad aumentada, que en nuestro caso es la plataforma Layar. En la figura a continuación se puede ver que la capa está creada con el nombre de “Guía Turismo Vina” y junto a ella, enmarcada en rojo, su dirección URL.

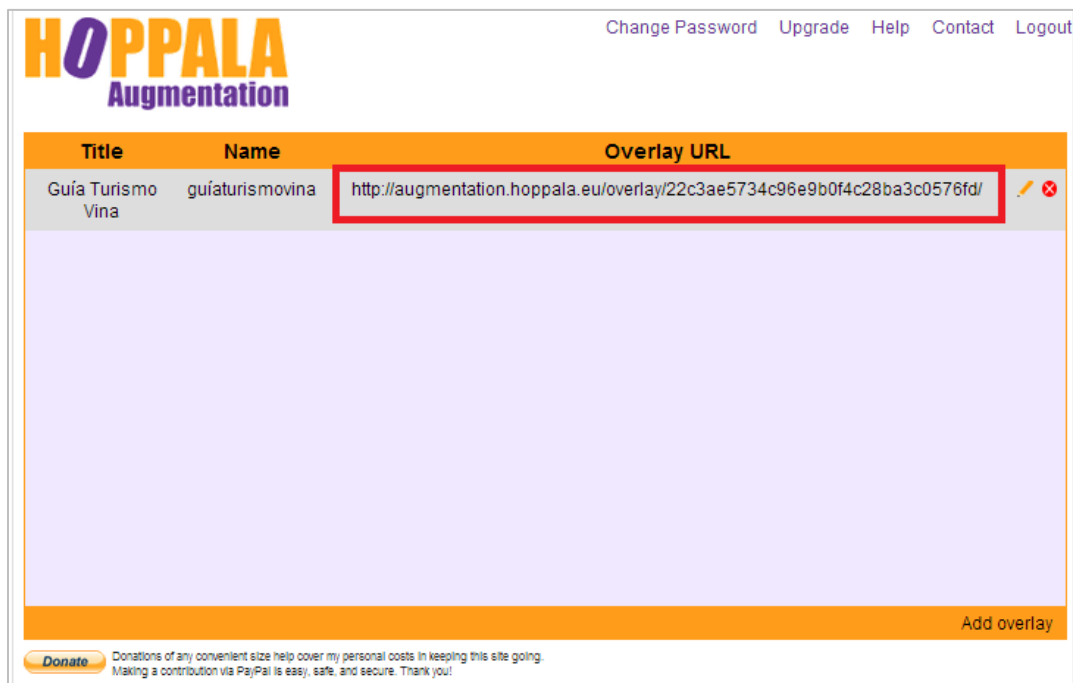


Figura 10: Creación de capa en Hoppala

Para utilizar esta dirección es necesario tener creada la capa de la aplicación en el servidor de Layar. Luego de haberla creado se debió ingresar al panel de edición de la capa e ingresar la dirección indicada en Hoppala a la casilla llamada “API endpoint”, de esta manera la base de datos se conectará a la capa de la aplicación de Layar. En la figura a continuación se puede ver el panel de edición de nuestra capa llamada “Guía Turismo Vina”, donde además se destaca la casilla en que se debe ingresar la dirección de la capa de Hoppala.

Layers
My information
Stats & Errors
Authorized partners
Support

« Back to all layers

## Edit Layer: Guia Turismo Vina (guiaturismovina)

General
API endpoint
Listing & indexing
Look & feel
Coverage
Filters
Permissions
Additional settings

API endpoint URL
http://augmentation.hoppala.eu/overlay/22c3a

### Hosted Screenshots

URL to post screenshots to:

Button text:

### Layer signing

OAuth signing required

OAuth consumer key

OAuth consumer secret

Save
Cancel

Figura 11: Creación de capa en Layar

Al realizar estos pasos realiza la conexión entre las capas de Hoppala y de Layar, y es posible visualizar los puntos de interés en la aplicación del celular. Luego se ingresaron manualmente los puntos de interés con sus respectivas funcionalidades en la capa de datos de Hoppala, mediante las herramientas que provee la plataforma. A continuación se puede ver la base de datos de puntos de interés del prototipo 1.

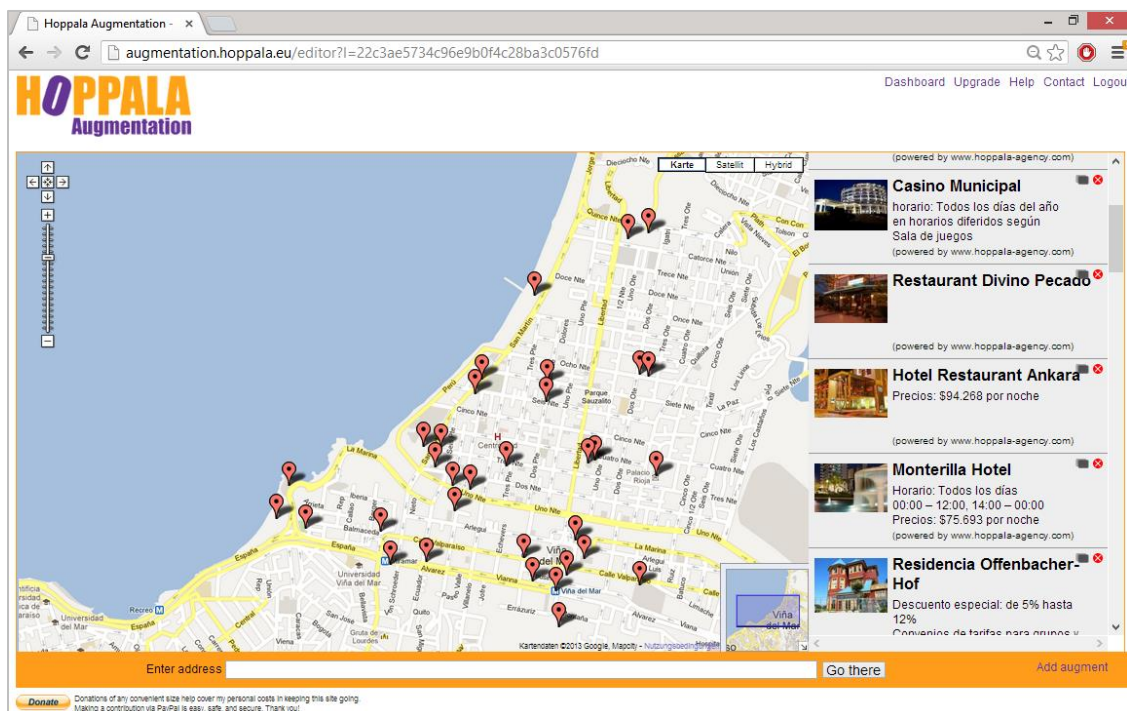


Figura 12: Administración de POIs en Hoppala

En la figura 11 se visualizan los puntos de interés en modo de lista, donde se muestra una foto referencial a lugar y la distancia en metros entre este y el usuario. En la figura 12 se presentan el modo de vista como mapa, en el cual se observa un plano que señala el lugar de ubicación del usuario y los puntos de interés.



Figura 13: Modo lista



Figura 14: Modo mapa

## 7.2 Segundo Prototipo

El prototipo 2 tiene como finalidad realizar un refinamiento del primer prototipo donde se agregaron más puntos de interés y se creó una simbología para diferenciar los distintos tipos de POIs. Además se realizó el proyecto de Android que exporte el instalador de la aplicación, con esto se logra tener un acceso directo que abrirá la aplicación sin la necesidad de pasar por la plataforma Layar.

### 7.2.1 Simbología de Puntos de Interés

En la figura 15 se presenta la simbología de la aplicación donde se pueden observar los íconos que representan los distintos tipos de puntos de interés que posee la aplicación. Esta simbología se verá al inicio de la aplicación utilizando la realidad aumentada y el tamaño de los íconos variará según la ubicación del usuario, es decir, se visualizarán más grandes o más pequeños dependiendo de la distancia entre el usuario y el punto del interés. En la figura 16 se presenta la pantalla de inicio de la aplicación donde se observan POIs con sus respectivos íconos.

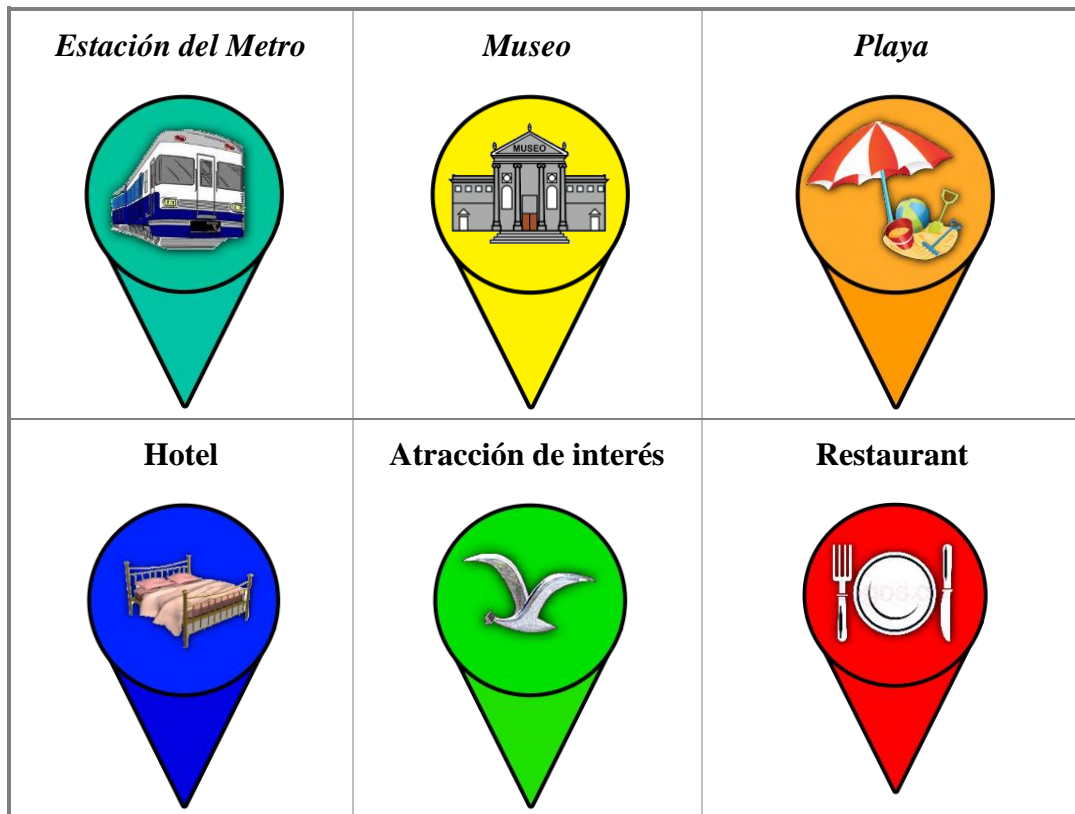


Figura 15: Simbología de puntos de interés



Figura 16: Inicio de aplicación

## 7.2.2 Desarrollo del Acceso directo

Para efectuar el ingreso a la aplicación de forma más expedita, se desarrolló una aplicación que funciona como acceso directo y abre automáticamente Layar y la capa de la aplicación desarrollada. A continuación se presenta una captura de pantalla del celular donde se ve el ícono del acceso directo creado.



Figura 17: Acceso directo de aplicación

El acceso directo favorece a los desarrolladores la etapa de pruebas, ya que permite un ingreso rápido a la aplicación. Además resulta más fácil publicitar la aplicación, ya que sin esto, la única forma de que otras personas la encuentren es vía Layar, pero no todos la conocen. Entonces al tener el APK o instalador de la aplicación se puede publicar en distintos sitios o guardarlo como código QR.



## 7.3 Prototipo Final

El prototipo final tiene como objetivo integrar a la aplicación, el sistema automático de ingreso de puntos de interés al gestor de POIs. A continuación se describen las actividades que se realizaron para llevar a cabo lo descrito.

### 7.3.1 Creación del Servidor

Para el prototipo final es fundamental poseer un servidor, el cual contenga todos los archivos necesarios para el funcionamiento de la aplicación, estos son el web service, la base de datos y el bot. Para esto, se optó por crear una cuenta en la página web <http://www.nixiweb.com/> que entrega alojamiento web gratuito, con un potente panel de control que nos permite crear y administrar la base de datos MySQL con la herramienta PhpMyAdmin, además posibilita la creación de tareas programadas mediante Cron Jobs.

### 7.3.2 Extracción de POIs de la API

Se debió utilizar la API que proporciona Google para trabajar con los puntos de interés que están integrados en Google Maps, llamada Google Places. Esta API envía información de los puntos de interés cercanos al usuario (rango modificable en el código) la cual se extrajo desde nuestro proyecto Android con Eclipse. Para poder probar la extracción exitosa de puntos de interés desde la API, se realizó una aplicación de prueba que enlista los POIs y muestra información adicional al momento de seleccionar alguno en particular. En la figura a continuación se puede ver una captura de pantalla del celular con la lista descrita.

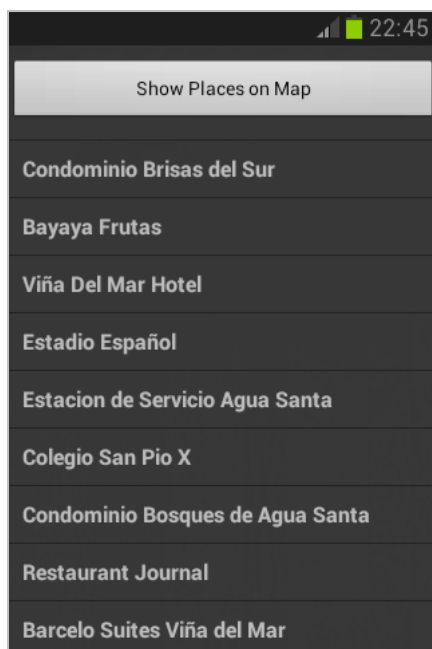
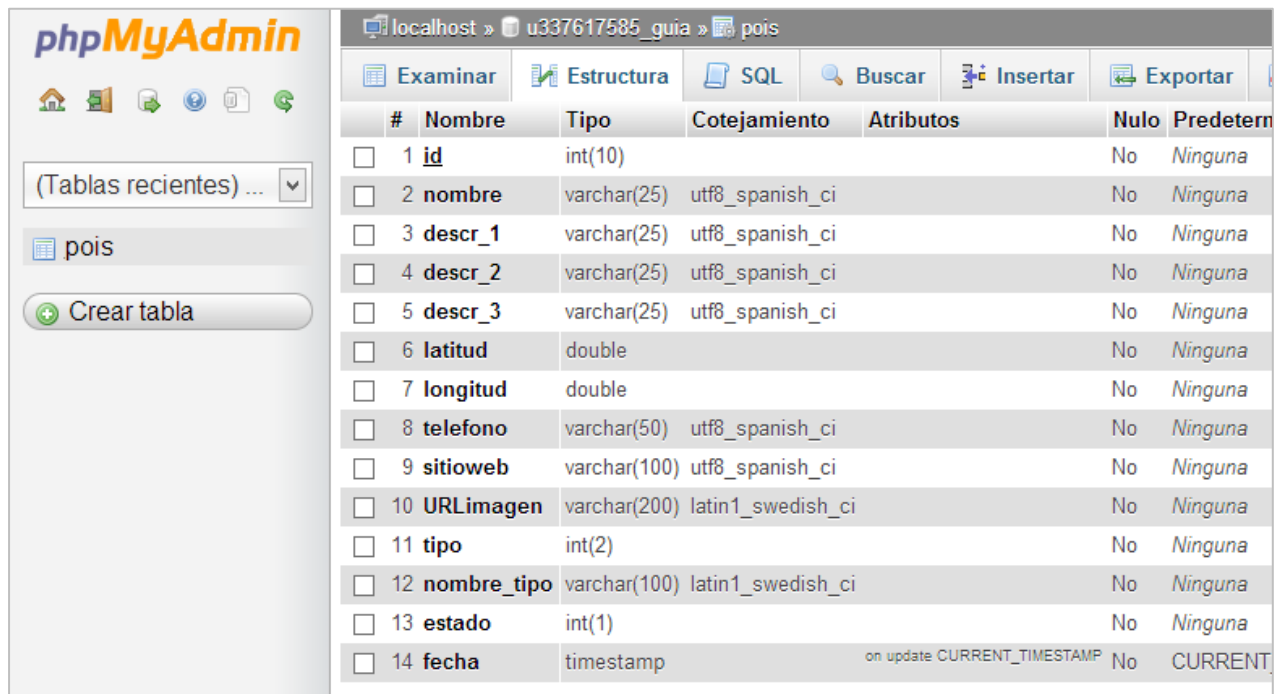


Figura 18: Aplicación de prueba: POIs de Google Places



### 7.3.3 Creación de Base de datos

En el servidor de Nixiweb se creó una base de datos denominada “guía”, que almacena la información de los puntos de interés en una tabla llamada “pois”, la cual posee todos los atributos de los puntos, estos son las coordenadas geográficas, junto con sus datos respectivos y el estado, que hace referencia a su existencia en el gestor de POIs (estado 1: existente en el gestor de POIs. Estado 0: No existente en el gestor de POIs.). En la figura a continuación se visualiza la base de datos y la tabla pois con sus atributos.



The screenshot shows the phpMyAdmin interface for a database named 'u337617585\_guia'. The 'pois' table is selected, and its structure is displayed. The table has 14 columns: id, nombre, descr\_1, descr\_2, descr\_3, latitud, longitud, telefono, sitioweb, URLimagen, tipo, nombre\_tipo, estado, and fecha. The 'id' column is the primary key. The 'estado' column is a tinyint(1) with a default value of 0. The 'fecha' column is a timestamp that updates to the current time on update.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predetermin
1	id	int(10)			No	Ninguna
2	nombre	varchar(25)	utf8_spanish_ci		No	Ninguna
3	descr_1	varchar(25)	utf8_spanish_ci		No	Ninguna
4	descr_2	varchar(25)	utf8_spanish_ci		No	Ninguna
5	descr_3	varchar(25)	utf8_spanish_ci		No	Ninguna
6	latitud	double			No	Ninguna
7	longitud	double			No	Ninguna
8	telefono	varchar(50)	utf8_spanish_ci		No	Ninguna
9	sitioweb	varchar(100)	utf8_spanish_ci		No	Ninguna
10	URLimagen	varchar(200)	latin1_swedish_ci		No	Ninguna
11	tipo	int(2)			No	Ninguna
12	nombre_tipo	varchar(100)	latin1_swedish_ci		No	Ninguna
13	estado	int(1)			No	Ninguna
14	fecha	timestamp		on update CURRENT_TIMESTAMP	No	CURRENT

Figura 19: Base de Datos de la aplicación.

### 7.3.4 Creación de Web Service

El *web service* fue desarrollado en lenguaje PHP y se encuentra alojado en el servidor de Nixiweb. Es consumido por la aplicación Android con el objeto de almacenar los puntos de interés captados de la API de Google places, mediante el método Guardar\_Datos\_POI. Asimismo es consumido por el *bot*, con el fin de obtener los POIs que se agregarán al gestor, mediante el método Extraer\_Datos\_POI.

El primer método inserta en la base de datos la información de los puntos de interés. Primeramente, se le asigna al POI el tipo de punto que corresponda, de acuerdo a las categorías existentes en el gestor. Luego se verifica que las coordenadas geográficas del POI no se encuentren en la base de datos. Finalmente se inserta la información del POI.

El método Extraer\_Datos\_POI obtiene de la base de datos, los puntos de interés cuyos estados sean 0. Luego modifica el estado de los POIs a 1, para que estos no se vuelvan a agregar al gestor de POIs.

### 7.3.5 Desarrollo del Bot

Paralelamente se trabajó en el desarrollo del *Bot*, que es el encargado de ingresar los datos de los puntos de interés al gestor de POIs, rellenando un formulario en HTML de manera automática. Para esto se programaron 2 archivos en lenguaje PHP. Uno que consume el web service llamado cliente, y otro que contiene los *bots* que insertan, tanto las propiedades, como las acciones de un POI, denominado funcionesBot.

El archivo cliente se conecta al *web service* y obtiene los puntos de interés que se deben agregar al gestor de POIs. Luego, se crea una variable que contiene un *string* con los datos concatenados requeridos por el formulario que agrega un POI, y se envía por parámetro a la función *bot* que se encarga de insertarlo en el gestor. Finalmente, se crea la variable con los datos de la acción del POI, ya sea llamar o visitar sitio web, y se envía por parámetro a la función del *bot* que realiza la inserción de acciones.

Para desarrollar los *bots* fue necesario utilizar la herramienta cURL (explicada en páginas anteriores). En primer lugar, la función recibe la variable con los datos del POI a ingresar. Luego se inicia una sesión en cURL y se establece la URL a la cual se desea acceder para ingresar los datos. Después se establecen las configuraciones del navegador y lenguaje utilizado. Luego, se fija la variable que contiene los datos a agregar. Finalmente, se cierra la sesión de cURL.

En las siguientes imágenes se visualiza el código de la función del *bot* que ingresa un punto al gestor y los formularios que insertan propiedades y acciones de un POI.

```
3 function bot_POI($parametro_post) {
4
5     $ch = curl_init();
6     curl_setopt($ch, CURLOPT_URL, 'http://artour.elementfx.com/insertar_poi.php');
7     curl_setopt($ch, CURLOPT_USERAGENT, 'Chrome/30.0.1599.66');
8     curl_setopt($ch, CURLOPT_HTTPHEADER, array("Accept-Language: es-es,en"));
9
10    curl_setopt($ch, CURLOPT_POST, true);
11    curl_setopt($ch, CURLOPT_POSTFIELDS, $parametro_post);
12
13    curl_setopt($ch, CURLOPT_HEADER, false);
14    curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
15    curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
16
17    $result = curl_exec($ch);
18    $error = curl_error($ch);
19    curl_close($ch);
```

Figura 20: Código fuente de *bot* que agrega POI

Figura 21: Formularios de ingreso de datos POI

### 7.3.6 Creación de Cron Job

Para automatizar el funcionamiento de la aplicación se debió crear una tarea programada que ejecute el *bot* periódicamente. Para esto, se hizo uso de una de las características del free hosting de Nixiweb, la cual permite crear Cron Jobs, y se programó la ejecución del *bot* cada cinco minutos. De esta forma, se verificará periódicamente la existencia de nuevos puntos de interés en la base de datos para ser agregados al gestor POIs. En la figura adjunta se visualiza lo anteriormente descrito.

Figura 22: Creación de Cron Job

### 7.3.7 Interfaces de la Aplicación

A continuación se presentan imágenes de las principales interfaces de la aplicación. En la figura 23 se presenta la pantalla de inicio, donde se observan los POIs con sus respectivos íconos. La imagen 24 muestra lo que se despliega al seleccionar el ícono de la aplicación en la esquina superior derecha, donde se visualizan sus opciones. En la figura 25 se muestra el modo de vista como mapa, en el cual se observa un plano que señala el lugar de ubicación del usuario y los puntos de interés. Finalmente, en la imagen 26 se visualiza la ruta a seguir para llegar al lugar seleccionado, además muestra la distancia en tiempo y kilómetros entre el punto y el usuario.



Figura 23: Interfaz Inicial

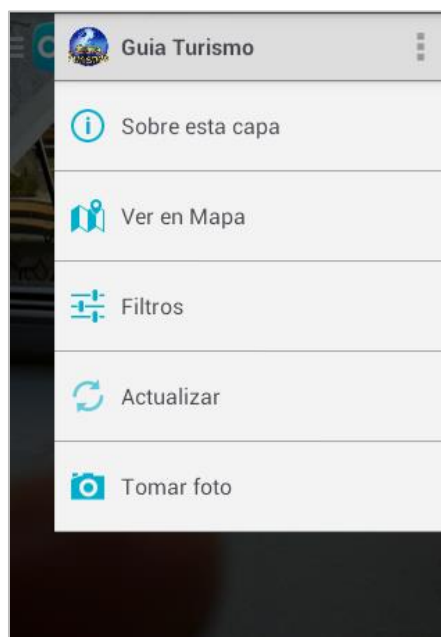


Figura 24: Opciones Aplicación



Figura 25: Ver como mapa

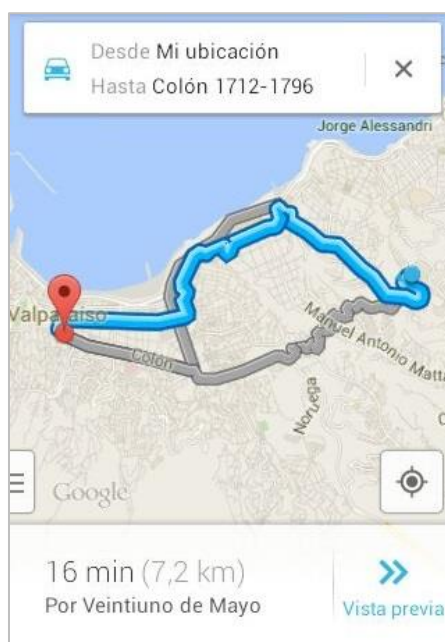


Figura 26: Mostrar ruta

## 8 Conclusiones y Trabajo Futuro

Respecto a la tecnología utilizada, la realidad aumentada es una ciencia relativamente joven que cada vez toma más protagonismo debido al desarrollo de los *smartphones*, por lo que es previsible que siga evolucionando acorde a la maduración y progreso de las tecnologías de la información. En consecuencia, es de esperar que el campo de aplicación de la realidad aumentada se amplifique mucho más, y será la imaginación la que ponga límites en su desarrollo.

En relación al proyecto propuesto, se desarrolló el primer prototipo el cual cumplió exitosamente el objetivo de conectar los servidores de Hoppala y Layar, en el que se logra visualizar los primeros puntos de interés incorporados para esa etapa, además de gran parte de sus funcionalidades. Asimismo, se agregó una simbología en los marcadores que permite distinguir el tipo de punto de interés que esta frente al usuario, ya sea alojamiento, restaurant, etc. Respecto al segundo prototipo, se incorporó en la base de datos de Hoppala la totalidad de los puntos de interés con todas sus opciones respectivas, además se instauró el lanzador que permite al usuario acceder directamente a la capa desarrollada en Layar, mediante su *smartphone*.

Con respecto al prototipo final, el cual consiste en un sistema que añade los puntos de interés automáticamente al gestor de POIs, se utilizó la API de Google Places para obtener la información de los puntos de interés, y se desarrolló un *web service*, el cual recibe los datos y los inserta en la base de datos alojada en un *free hosting*. Para ingresar los puntos de interés al gestor automáticamente, fue necesario utilizar una plataforma que permitiera rellenar formularios HTML mediante un *bot*. Debido a que Hoppala no permitía ser accedida por un *bot* (por motivos de seguridad del sitio), se tuvo que hacer uso de otro servidor denominado “Insert POI”, el cual fue desarrollado por ex alumnos de la Universidad de Playa Ancha.

Gracias a la API de Google Places, que permite obtener información de los puntos de interés, y al *bot* desarrollado para ingresar automáticamente la información al gestor de POIs, la aplicación alcanzará un nivel de masificación mucho mayor, ya que no solo se podría utilizar en el área establecida para Viña del Mar, sino que en cualquier ciudad que cuente con puntos de interés en la API de Google Places.

Se espera desarrollar una mejora para la aplicación realizando un algoritmo de corrección de altitud que pueda capturar y mostrar, de una forma más exacta, los lugares que se obtienen a través del GPS. Ya que en ciertos lugares, existen puntos de interés que se ubican en el mismo sitio que el usuario pero a diferente altura, por ejemplo en lugares con cerros, tales como Valparaíso, Río de Janeiro, etc. En este caso la aplicación no debiera mostrar el ícono del POI ya que el usuario no puede visualizar el lugar, sin embargo, actualmente lo mostraría. Por este motivo, se espera desarrollar la solución para el problema descrito, y así dar a los usuarios una correcta experiencia de la realidad aumentada.

## 9 Referencias

- [1] Metodología de prototipado evolutivo.  
<http://sisteminformacii.wikispaces.com/METODOLOG%C3%8DA+DE+ROGER+PRESSMAN> Fecha último acceso: 10-05-13
- [2] Gestor de POIs: Proyecto de Título “Guía Turística Móvil con Tecnología de Realidad Aumentada para la Ciudad de Valparaíso”. Dagoberto Pavez G., Marco Riquelme L.  
<http://artour.elementfx.com> Fecha último acceso: 27-12-13
- [3] cURL  
<http://curl.haxx.se> Fecha último acceso: 20-10-13
- [4] Realidad Aumentada: Una nueva lente para ver el mundo.  
[http://www.fundacion.telefonica.com/es/que\\_hacemos/media/publicaciones/Realidad\\_Aumentada\\_Completo.pdf](http://www.fundacion.telefonica.com/es/que_hacemos/media/publicaciones/Realidad_Aumentada_Completo.pdf) Fecha último acceso: 10-04-13
- [5] Sistema operativo Android.  
<http://actualidadg.com/2012/10/que-es-android-historia-conceptos-y-evolucion-del-sistema-operativo-para-dispositivos-moviles/> Fecha último acceso: 22-10-13
- [6] Arquitectura de S.O Android.  
<http://androideity.com/2011/07/04/arquitectura-de-android/> Último acceso: 13-04-13
- [7] Layar.  
<http://www.xatakamovil.com/aplicaciones/layar-primer-navegador-android-con-realidad-aumentada> Fecha último acceso: 16-04-13
- [8] Bot.  
<http://www.dailycommonsense.com/web-bot-what-is-it-can-it-predict-stuff/> Fecha último acceso: 15-12-13
- [9] API de Google Places.  
<https://developers.google.com/places/documentation/> Fecha último acceso: 04-10-13
- [10] Lenguaje PHP.  
<http://www.php.net> Fecha último acceso: 04-10-13
- [11] Lenguaje SQL.  
<http://databases.about.com/od/sql/a/sqlfundamentals.htm> Fecha último acceso: 04-10-13