

ESCUELA POLITÉCNICA DE EJÉRCITO

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CARRERA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

**“DESARROLLO DE UN SISTEMA DE INFORMACION PARA
LA ESCUELA POLITÉCNICA DEL EJÉRCITO BASADO EN
VISITAS VIRTUALES Y REALIDAD AUMENTADA”**

Previa a la obtención de Título de:

INGENIERO EN SISTEMAS E INFORMÁTICA

ELABORADO POR:

PABLO IVAN OCHOA VÁSQUEZ

Sangolquí, Octubre del 2012

REGISTRO DE PUBLICACIÓN

Yo, Pablo Iván Ochoa Vásquez con cédula de identidad No. 1104627169, autorizo a la biblioteca Alejandro Segovia de la Escuela Politécnica del Ejército se publique la Tesis “Desarrollo de un sistema de información para la Escuela Politécnica del Ejército basado en visitas virtuales y realidad aumentada” realizada por mi persona.

Sangolquí, Octubre del 2012

Pablo Ochoa
1104627169

CERTIFICACIÓN

Certifico que el presente trabajo fue realizado en su totalidad por el Sr. OCHOA VASQUEZ PABLO IVAN como requerimiento parcial a la obtención del título de INGENIERO EN SISTEMAS E INFORMÁTICA

Sangolquí, Octubre del 2012

ING. DIEGO MARCILLO.

DEDICATORIA

Quiero dedicar este proyecto a mis padres, ya que gracias a su apoyo incondicional durante todos estos años, han hecho que me sea posible estar en el lugar que me encuentro hoy en día.

Pablo Iván Ochoa Vásquez

AGRADECIMIENTOS

A todas las personas que directa o indirectamente han colaborado con este trabajo, a toda mi familia especialmente a mis padres por estar siempre a mi lado y brindarme su sabiduría, comprensión y apoyo durante toda la vida. Gracias.

Pablo Iván Ochoa Vásquez

TABLA DE CONTENIDOS

CAPÍTULO 1: INTRODUCCIÓN.....	12
1.1 Antecedentes.....	12
1.2 Planteamiento del Problema.....	12
1.3 Delimitación Espacial.....	15
1.4 Delimitación Temporal	15
1.5 Formulación del problema.....	15
1.6 Sistematización del problema	16
1.7 Objetivos.....	16
1.7.1 Objetivo general	16
1.7.2 Objetivos específicos.....	17
1.8 Justificación	17
1.9 Alcance	19
1.10 Factibilidad	24
1.10.1 Factibilidad Técnica	24
1.10.2 Factibilidad Económica	25
1.10.3 Factibilidad Operativa	27
1.11 Metodología	27
CAPÍTULO 2: MARCO TEÓRICO	28
2.1 Metodologías de Desarrollo.....	28
2.1.1 Introducción.....	28
2.1.2 Extreme Programming.....	29
2.1.3 Software Prototyping.....	33
2.2 Content Management Systems.....	38
2.2.1 Introducción.....	38
2.2.2 Ventajas de un CMS	39
2.2.3 Drupal	41
2.2.3.3 Arquitectura	46
2.2.3.4 Módulos.....	48
2.3 Imágenes Panorámicas.....	51
2.3.1 Introducción.....	51
2.3.2 Usos	52
2.3.3 Tipos	53
2.3.4 Equipo.....	56
2.3.5 Stitching.....	59

2.3.6	Krpano	61
2.3.6.1	Introducción	61
2.3.6.2	Licencia	64
2.4	Realidad Aumentada	66
2.4.1	Definición	66
2.4.2	Hardware Requerido.....	67
2.4.3	Aplicaciones	72
2.4.4	Servicios Realidad Aumentada.....	74
2.4.5	Teoría de AR basada en geo localización.....	77
2.4.5.1	Operaciones matemáticas con coordenadas	80
2.5	Desarrollo Móvil	82
2.5.1	Introducción.....	82
2.5.2	iOS.....	83
2.5.2.1	Requerimientos	83
2.5.2.2	Herramientas.....	83
2.5.3	Android.....	87
2.5.3.1	Requerimientos	87
2.5.3.2	Herramientas.....	88
CAPÍTULO 3: DISEÑO		90
3.1	Historias de Usuario.....	90
3.1.1	Introducción.....	90
3.1.2	Definición Prioridades	105
3.1.3	Definición de Iteraciones.....	106
3.2	Casos de Uso.....	108
3.2.1	Diagrama de casos de uso.....	108
3.2.2	Descripción de casos de uso	109
3.3	Diagrama de Entidad Relación	126
3.4	Diagrama de Clases del Aplicativo Móvil	127
3.5	Diagrama de Componentes del Módulo Web	128
3.6	Diagrama de Despliegue	130
CAPÍTULO 4: DESARROLLO		132
4.1	Desarrollo de Módulos del Sistema	132
4.2	Descripción del API del Módulo Web.....	132
4.2.1	Estándares de Codificación	132
4.2.2	Descripción de Módulos.....	133

4.2.3	Descripción de Módulo de Visitas Virtuales	135
4.2.3.1	Listado de Archivos	135
4.2.3.2	Descripción de Funciones.....	137
4.3	Descripción del API del Aplicativo Móvil.....	151
4.3.1	Estándares de Codificación	151
4.3.2	Listado de Clases	152
4.4	Pruebas de Aceptación.....	161
CAPÍTULO 5	173
5.1	CONCLUSIONES	173
5.2	RECOMENDACIONES	176
BIBLIOGRAFÍA	178

ÍNDICE DE GRÁFICAS

Gráfica 2.1: Planificación y Retroalimentación en XP	30
Gráfica 2.2: Prácticas XP	32
Gráfica 2.3: Modelo de Desarrollo por Prototipos.....	35
Gráfica 2.4: Arquitectura Física Drupal	47
Gráfica 2.5: Invocación Página Web en Drupal	47
Gráfica 2.6: Arquitectura Lógica Drupal	48
Gráfica 2.7: Estructura básica de un módulo en Drupal	49
Gráfica 2.8: Implementación de una función hook.....	50
Gráfica 2.9: Tipos de Imágenes Panorámicas.....	55
Gráfica 2.10: Tamaño imágenes dependiendo de su formato.....	56
Gráfica 2.11: Pérdida de campo de visión en la foto	57
Gráfica 2.12: Cabezal Panorámico	58
Gráfica 2.13: Lente gran angular	59
Gráfica 2.14: Stitching de imágenes.....	60
Gráfica 2.15: Hardware Realidad Aumentada.....	72
Gráfica 2.16: Navegadores de AR basados en geo localización	76
Gráfica 2.17: Principio Matemático de la AR geo localizada.....	78

ÍNDICE DE TABLAS

Tabla 1.1: Software	25
Tabla 1.2: Recursos Materiales	25
Tabla 1.3: Licencias de Desarrollo	26
Tabla 1.4: Recursos Humanos	26
Tabla 1.5: Valor Total	26
Tabla 2.1: Comparativa WordPress, Joomla, Drupal.....	45
Tabla 2.2: Licencias Krpano	65
Tabla 2.3: Licencias Desarrollo iOS.....	86

RESÚMEN

El presente trabajo muestra el desarrollo de un sistema de información basado en fotografía inmersiva y realidad aumentada; la combinación de estas tecnologías ha permitido demostrar que se mejora la experiencia de usuario en los actuales sistemas de información ya que muestra datos de un sitio en particular antes de que el usuario se encuentre físicamente en el mismo y posteriormente cuando se encuentra en el lugar, mediante el uso de fotografías panorámicas de 360 grados que son visualizadas en un navegador de internet y la inclusión de información georeferenciada en una aplicación de realidad aumentada utilizando un dispositivo móvil.

Se realizaron pruebas de usuario a quienes se les solicitó ubicar un lugar de interés que no conocían de antemano contando con la ayuda del aplicativo desarrollado. De los resultados se desprende que la utilización de estas tecnologías permite que el usuario se ubique con facilidad en un sitio desconocido, sin necesidad de buscar otro tipo de ayudas, dándole además información relevante y geolocalizada de acuerdo al punto donde se encuentre.

CAPÍTULO 1: INTRODUCCIÓN

1.1 Antecedentes

La Escuela Politécnica del Ejército sin duda es una de las universidades más prestigiosas del Ecuador, contando con uno de los mejores y más amplios campus existentes en el país, es por esto que constantemente recibe visitas de instituciones secundarias, nuevos estudiantes, otras universidades, profesionales de postgrados, entre otros.

Es importante que las personas puedan ubicarse con facilidad dentro de los límites universitarios, y si bien existe información de la universidad en el portal web, esta no cubre las necesidades reales de brindar información mucho más específica del campus universitario.

1.2 Planteamiento del Problema

Los sistemas de información actuales ayudan a automatizar procesos en prácticamente todas las áreas profesionales. El auge del internet ha posibilitado que el acceso a la información sea mucho más fácil, rápido y eficiente.

Gracias a su desarrollo actualmente existen muchas facilidades para que actividades muy comunes se puedan llevar a cabo desde la comodidad de nuestro hogar, así se puede realizar desde un simple registro para recibir

boletines de noticias hasta inscripciones o procesos complejos dentro de una institución.

Sin embargo, a pesar del amplio abanico de posibilidades que estos sistemas y herramientas ofrecen, no llegan a cubrir todos los posibles escenarios y casos en los cuales es necesario obtener información de manera localizada y en tiempo real. Esto sumado a un campus e infraestructura muy amplia, obliga a las personas que visitan estas instalaciones a perder mucho tiempo y constantemente preguntar por información en cualquier punto o persona que vean a su alrededor.

Una manera de obtener información muy clara de un sitio desconocido, es utilizando tecnologías como son las visitas virtuales, que consisten en tomar fotos panorámicas del lugar que se pretende mostrar a través de la web, y luego con la ayuda de un software especial, estas son generadas y mostradas en el computador. El nivel de realismo e inmersión es adecuado, pudiendo además agregar información interactiva a puntos estratégicos que sean de importancia para el visitante.

Por otro lado la evolución del hardware para dispositivos móviles abre un nuevo campo y nuevas soluciones para estos aspectos que por su naturaleza, los sistemas tradicionales de escritorio o web no pueden solucionar de manera adecuada. Es así que con el continuo incremento en la producción de Smartphone y tablets se vuelve un punto estratégico primordial para llegar a la comunidad de una manera más efectiva.

Una tecnología que en estos últimos años ha ganado mucha fuerza es la realidad aumentada (AR)¹, que básicamente es la visualización directa o indirecta de un ambiente del mundo real, cuyos elementos son aumentados por imágenes, videos y sonidos generados con la ayuda de un computador.

De esta manera el entorno del usuario se vuelve interactivo y manipulable digitalmente, ofreciendo nuevas herramientas para mejorar la inmersión del usuario final con nuestras aplicaciones.

También resulta muy interesante la posibilidad de combinar la tecnología de realidad aumentada con la geo localización, lo cual permitiría que puntos de interés debidamente establecidos dentro del campus, sean fácilmente encontrados y descritos por la aplicación, mostrando imágenes, videos e información relacionada con estos puntos.

Es por estas razones que surge la necesidad de contar con un sistema que cubra en plenitud con la información mostrada en la universidad y se convierta en un mecanismo realmente efectivo para que cualquier persona pueda ubicarse fácilmente dentro de ella.

Además el sistema tiene que ser accesible y sobre todo permitir que la movilidad no se vea afectada, es decir que en cualquier punto del campus se pueda obtener información, acceso a servicios y direcciones a cada uno

¹ http://en.wikipedia.org/wiki/Augmented_reality

de los puntos significativos de la universidad, debidamente geo referenciados mediante aplicativos como Google Maps².

1.3 Delimitación Espacial

La investigación y el desarrollo del sistema de información basado en visitas virtuales y realidad aumentada se lo llevará a cabo en mi domicilio y en las oficinas del director del Centro de Innovación y Desarrollo Empresarial, auspiciante del presente trabajo de tesis.

1.4 Delimitación Temporal

La investigación y el desarrollo de un sistema de información basado en visitas virtuales y realidad aumentada se realiza en base a las necesidades tecnológicas que requiere un sitio web a partir del lanzamiento del manejador de contenidos Drupal 6 en el año 2008 y del kit de desarrollo de software para la plataforma iOS, SDK 5 disponible a partir del año 2011.

1.5 Formulación del problema

¿De qué manera el implementar un sistema de información basado en visitas virtuales y realidad aumentada mejorará la comunicación de información relevante de la institución?

² <http://maps.google.com/>

1.6 Sistematización del problema

Con el desarrollo del sistema se podrá almacenar la información susceptible de ser mostrada en un recorrido virtual, garantizando se conserven las características de la información.

Será posible presentar la información de forma que se garantice un incremento en la efectividad de los procesos llevados a cabo en la universidad.

El sistema logrará, hacer más ágil e intuitiva la gestión de recorridos virtuales y el procesamiento de imágenes panorámicas de tal forma que muestren lo más importante del campus universitario.

Se puede tener disponibilidad de la información almacenada en cualquier punto dentro del campus universitario con la ayuda de dispositivos móviles.

1.7 Objetivos

1.7.1 Objetivo general

Desarrollar un sistema de información para la Escuela Politécnica del Ejército, basado en visitas virtuales utilizando imágenes panorámicas de 360 grados y realidad aumentada.

1.7.2 Objetivos específicos

- Establecer la información institucional que puede ser susceptible de mostrarse dentro de un recorrido virtual.
- Diseñar con la ayuda de una herramienta CASE los diagramas útiles para el sistema y determinar estándares para un sistema testeable, entendible, explorable y explicable.
- Diseñar una base de datos que permita almacenar la información a ser mostrada por las visitas virtuales juntamente con datos de geo localización.
- Desarrollar un módulo web que permita crear, configurar y visualizar visitas virtuales de los lugares más importantes de la universidad.
- Integrar el módulo web con herramientas de procesamiento de imágenes panorámicas, para transformarlas en formatos amigables para los motores de visualización.
- Desarrollar un aplicativo móvil, que permita mostrar información interactiva utilizando imágenes panorámicas en 360 grados, realidad aumentada y geo localización.

1.8 Justificación

El desarrollo de un sistema de información para la Escuela Politécnica del Ejército, permitirá exponer datos institucionales de manera más eficiente

para las personas que se acerquen a sus instalaciones. Esto, brindará muchos beneficios para la universidad, y muy especialmente en temporadas especiales como son el tiempo de admisiones, campeonatos, torneos, congresos científicos, tecnológicos entre otros.

La información será manejada por un sistema informático, y almacenada de manera correcta en una base de datos, minimizando de esta forma errores que puedan surgir al momento de trabajar con dicha información, los cuales además de originar ineficiencia al transmitir los datos institucionales pueden llegar a causar pérdidas de dinero y tiempo a la universidad.

Integrará la información estática, es decir datos que tienden a permanecer mucho tiempo sin modificaciones, como lo es la infraestructura, departamentos, biblioteca, edificios, áreas de recreación; entre otros. Con información dinámica, es decir datos que son actualizados con mucha frecuencia, tales como proyectos estudiantiles, congresos de ciencia y tecnología, conferencias. Todo esto gracias a la utilización de puntos de interés o hotspots dentro de las visitas virtuales, que serán administrables por el módulo web del sistema.

Para que toda la información mencionada pueda ser visualizada de igual forma por los usuarios, tanto en el portal web como en el aplicativo móvil, esta debe tener un estándar de presentación, es por esta razón que los datos deberán ser procesados y puestos en un formato común, con lo cual se garantizará que no se pierda ningún detalle al momento de mostrarlos al usuario.

Para el desarrollo del sistema se usará la metodología XP³ combinada con un desarrollo por prototipos debido a que estas permiten tener entregables funcionales en tiempos cortos, con los cual los interesados en el sistema pueden estar al tanto de su evolución. Además al fundamentarse en pruebas y en la optimización de código, el sistema, desarrolla una base sólida, de alto rendimiento y con un nivel muy bueno de respuesta ante el cambio.

Por otro lado el uso de dichas metodologías permite tener un panorama mucho más claro de los requerimientos reales que el cliente desea sean implementados, y gracias a la interacción continua entre el desarrollador con el cliente, se puede cumplir con los plazos de entrega de acuerdo a lo planeado con anticipación.

1.9 Alcance

El sistema tendrá dos partes, la primera será el módulo web que permitirá visualizar fotos panorámicas en 360 grados junto con información institucional, conformando las visitas virtuales a la universidad, en las cuales se incluirán panoramas de los sitios más significativos.

La segunda parte la representa un aplicativo móvil que utilizará tecnologías de realidad aumentada para mostrar la información creada por el módulo

³ http://en.wikipedia.org/wiki/Extreme_programming

web en tiempo real. El aplicativo móvil será desarrollado para la plataforma iOS⁴ específicamente para el dispositivo iPhone.

Para el desarrollo de ambas partes se establecen los siguientes puntos que marcarán el alcance del presente proyecto.

- Módulo de creación de visitas virtuales

Este módulo permite la creación y administración de las visitas virtuales, el usuario que tenga acceso a este modulo podrá crear o editar una visita y agregar los diferentes panoramas que conformaran a la misma, los cuales quedarán geo localizados en los puntos establecidos por el usuario, utilizando aplicativos como Google Maps.

- Módulo de Configuración de Visitas Virtuales

En este módulo permite la inclusión de puntos de interés o hotspots dentro de las visitas virtuales, los cuales contendrán información e imágenes que serán ingresadas por el usuario administrador. Este módulo se encarga de geo localizar los puntos de interés procesando datos de los panoramas creados con anterioridad.

- Módulo de extensión e integración de componentes con herramientas externas

⁴ <http://en.wikipedia.org/wiki/IOS>

Este módulo brindará la posibilidad de subir imágenes panorámicas para ser usadas en la administración de visitas virtuales, las mismas que se procesarán y se transformarán a un formato que pueda ser interpretado por motores de visualización de fotos panorámicas en 360 grados.

- Módulo de visualización de visitas virtuales

Mediante el uso de este módulo, los visitantes del sitio web podrán observar las visitas virtuales que han sido creadas y configuradas en el sistema, también, podrán navegar a través de las fotos panorámicas que conforman la visita y ver los datos descriptivos que representan a cada una de ellas.

- Módulo de servicios web

Este módulo permite la integración de servicios web con el manejador de contenidos, además expone los datos que fueron ingresados en el sistema a clientes externos utilizando una arquitectura REST⁵ y en formatos de transporte soportados por el CMS⁶.

- Módulo de integración con sistema de seguridad incorporado de CMS

Con la utilización de este módulo se integrarán roles personalizados con el sistema de seguridad de Drupal CMS, lo cual permite dar acceso a la funcionalidad del sistema únicamente a los usuarios que sean debidamente autorizados por el administrador del portal web.

⁵ http://en.wikipedia.org/wiki/Representational_state_transfer

⁶ http://en.wikipedia.org/wiki/Content_management_system

- Módulo de acceso a sensores del dispositivo móvil

Con la implementación de este módulo se podrá acceder a los sensores que tenga disponible el dispositivo móvil y utilizar la información que puedan brindar al usuario del aplicativo.

Entre los sensores de los cuales se requiere constante actualización se tiene:

- Cámara
- GPS
- Acelerómetro
- Compás Digital

Es importante señalar que la información recibida en conjunto de todos estos sensores es fundamental para poder mostrar información de manera eficaz al utilizar realidad aumentada, por lo cual es muy importante se obtengan estos datos con la mayor precisión posible.

- Módulo de presentación de sitios geo localizados para el dispositivo móvil

Este módulo hace uso de los servicios de mapas y localización dentro del dispositivo móvil para mostrar la ubicación actual del usuario y de cada uno de los puntos de interés creados en el sistema web, de una manera fácil y conveniente que ayude al usuario a orientarse de mejor forma dentro del campus universitario.

- Módulo de integración del dispositivo móvil con servicios web

Con la implementación de este módulo se puede acceder a las capacidades de conectividad y networking del dispositivo móvil, utilizando tanto librerías nativas del sistema operativo como librerías externas proporcionadas por usuarios en internet.

Para el desarrollo del proyecto se tomarán en cuenta las siguientes especificaciones técnicas:

- **Drupal CMS:** Drupal es un sistema de gestión de contenidos que permite publicar, artículos, blogs, foros, encuestas y administrar usuarios con sus respectivos roles. Es un sistema de gestión dinámico, en el sentido que el contenido de la página es almacenado en una base de datos, para que luego pueda ser editada usando un entorno web. Es un sistema libre con licencia GNU/GPL⁷.
- **KRPano Viewer:** Krpano es un motor de visualización de imágenes panorámicas y tours virtuales interactivos, es pequeño y de alto rendimiento. El visor está disponible como una aplicación de Adobe Flash Player⁸ para su uso en computadores de escritorio y dispositivos Android que soporten y tengan instalado Flash Player. También, Krpano ofrece un

⁷ <http://www.gnu.org/copyleft/gpl.html>

⁸ <http://www.adobe.com/products/flashplayer.html>

visor puramente implementado en HTML5⁹ y JavaScript que puede ser utilizado por dispositivos iOS (iPhone y iPad).

- **Google Maps:** Es un servicio proporcionado por Google, ofrece imágenes de mapas desplazables así como fotos satelitales, también ofrece datos de rutas entre diferentes ubicaciones ya sea a pie, automóvil o transporte público. También existe la posibilidad de integrar muchas características de Google Maps con cualquier sitio Web que incorpore e implemente los archivos y API ¹⁰proporcionadas por Google.
- **iOS SDK:** Es un kit de desarrollo de software proporcionado por Apple Inc. con el cual se pueden implementar aplicaciones nativas para la plataforma iOS y probarlas dentro del simulador de iPhone o iPad. Cabe señalar que la instalación de aplicaciones dentro de un dispositivo físico es únicamente posible una vez haya sido pagado un “iOS Developer Program”, cuyo costo es \$99.00 por año.

1.10 Factibilidad

1.10.1 Factibilidad Técnica

El desarrollador del presente sistema cuenta con los recursos necesarios tanto económicos, de hardware y de software para llevar a cabo y culminar

⁹ <http://en.wikipedia.org/wiki/HTML5>

¹⁰ http://es.wikipedia.org/wiki/Interfaz_de_programaci%C3%B3n_de_aplicaciones

el presente proyecto, además el conocimiento que posee en el área de desarrollo le es suficiente para llevar a cabo los requerimientos del proyecto.

1.10.2 Factibilidad Económica

Software

Material	Valor en dólares
Drupal 6 CMS	0
Mac OSX Lion	29,00
Xcode 4	4,99
XAMPP	0
Krpano Viewer	111,4
Krpano HTML 5 Add-On License	48,28
Subtotal	\$ 193,67

Tabla 1.1: Software

Recursos Materiales

Material	Valor en dólares
Internet	240
Impresiones	210
Computador Macbook	999,99
Apple iPhone 4S	649,99
Subtotal	\$ 2099,98

Tabla 1.2: Recursos Materiales

Licencias de Desarrollo

Material	Valor en dólares
Apple Developer Program	99
Subtotal	\$ 99,00

Tabla 1.3: Licencias de Desarrollo

Recursos Humanos

Desarrollador por 6 meses	Valor
Pablo Ochoa	\$4000 por 6 mes
Subtotal	\$4000,00

Tabla 1.4: Recursos Humanos

Valor Total

Recurso	Valor
Software	\$ 193,67
Recursos Materiales	\$ 2099,98
Licencias de Desarrollo	\$ 99,00
Recursos Humanos	\$4000,00
TOTAL	\$6392,65

Tabla 1.5: Valor Total

1.10.3 Factibilidad Operativa

El tema es totalmente factible ya que cuenta con el auspicio del “Centro de Innovación y Desarrollo Empresarial CIDE” de la Escuela Politécnica del Ejército.

Departamentos que manipulan con mucha frecuencia datos informativos susceptibles de ser mostrados en las visitas, pueden estar muy interesados en el desarrollo del proyecto, como serian el Departamento de Admisión y Registro, el Departamento de Marketing y el mismo Departamento de Innovación y Desarrollo Empresarial, CIDE.

1.11 Metodología

Para el desarrollo de este proyecto se analizó la adopción de una metodología ágil, esto por la velocidad con la cual se puede avanzar el proyecto, dando entregables rápidos y que pueden reaccionar ante el cambio de manera muy fácil. Sin embargo, la mayoría de metodologías ágiles están enfocados en un trabajo de equipo, por lo cual muchos de sus principios y prácticas no se pueden aplicar a desarrolladores independientes.

La metodología que se piensa adoptar es la de programación extrema XP, conjuntamente con un desarrollo por prototipos. De las cuales se utilizarán sus principales prácticas.

CAPÍTULO 2: MARCO TEÓRICO

2.1 Metodologías de Desarrollo

2.1.1 Introducción

Las metodologías de desarrollo de software hicieron su aparición alrededor de los años 60, con lo cual buscaban la implementación de sistemas de información de una forma muy deliberada, estructurada y siguiendo una metodología, la cual requería que cada fase del ciclo de vida del sistema sea llevado a cabo de manera rígida y secuencial.

En la actualidad las metodologías de desarrollo de software proveen un marco de trabajo el cual es utilizado para estructurar, planear y controlar el proceso de desarrollo de un sistema informático, lo que incluye la predefinición de los entregables y artefactos que serán creados y completados por el equipo de desarrollo y mantenimiento de la aplicación.

Con el paso de los años una gran variedad de metodologías han evolucionado con sus fortalezas y debilidades, las mismas que no necesariamente son efectivas y aplicables para todo tipo de proyecto, la elección de una metodología está basada en muchas consideraciones tanto técnicas, como organizacionales.

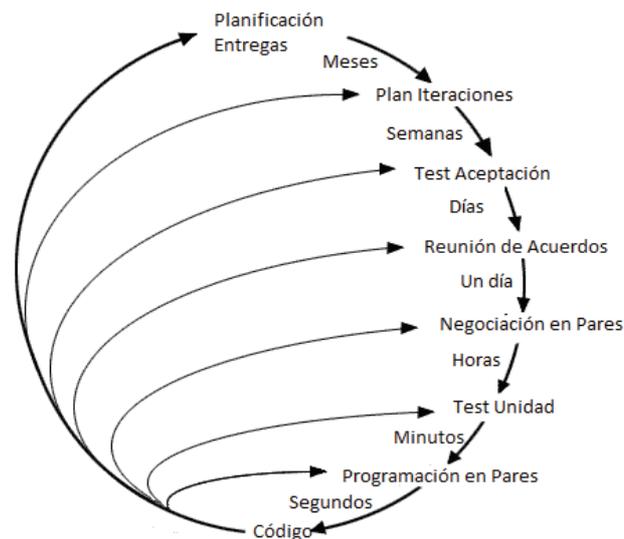
Muchas metodologías están ligadas a organizaciones, las cuales se encargan de mejorar, dar soporte y promover el uso de la misma. Generalmente la metodología es definida en un documento oficial que incluye todos los detalles de sus principios, prácticas y aplicación.

2.1.2 Extreme Programming

Extreme Programming es una metodología de desarrollo que tiene como objetivo el mejorar la calidad del software y la respuesta ante cambios en los requerimientos establecidos. Se basa en valores que a lo largo de los años han probado mejorar la eficiencia y productividad en el desarrollo de sistemas informáticos, de esta forma Extreme Programming se basa en los siguientes valores:

- **Simplicidad:** Es la base de la metodología, siempre se intenta simplificar el diseño con lo cual se agiliza el diseño y facilita el mantenimiento. Para mantener la simplicidad es necesario refactorar el código a medida que este va creciendo, también se aplica el valor de simplicidad al momento de documentar el código, haciéndolo en su justa medida e intentando que el mismo este autodocumentado.
- **Comunicación:** La comunicación con el cliente es muy buena gracias a que forma parte del equipo de desarrollo, por otro lado con la ayuda de programación en parejas se logra una mayor interacción entre los programadores del proyecto.

- Retroalimentación: Al realizarse ciclos cortos en los cuales ya se pueden mostrar resultados, se conoce la opinión del cliente, con lo que se minimiza el impacto de tener que rehacer parte del código que no cumpla con los requerimientos.
- Coraje o valentía: Significa que los desarrolladores tienen la libertad de reconstruir su código, modificarlo o rechazar código que resulte obsoleto; si con ello aseguran que futuros cambios se implementaran más fácilmente.
- Respeto: Los programadores no pueden realizar cambios que afecten a funcionalidad existente o implique demora en el trabajo de sus compañeros. Se respeta el trabajo de los demás sin minimizarlo, elevando el ritmo de producción del equipo.



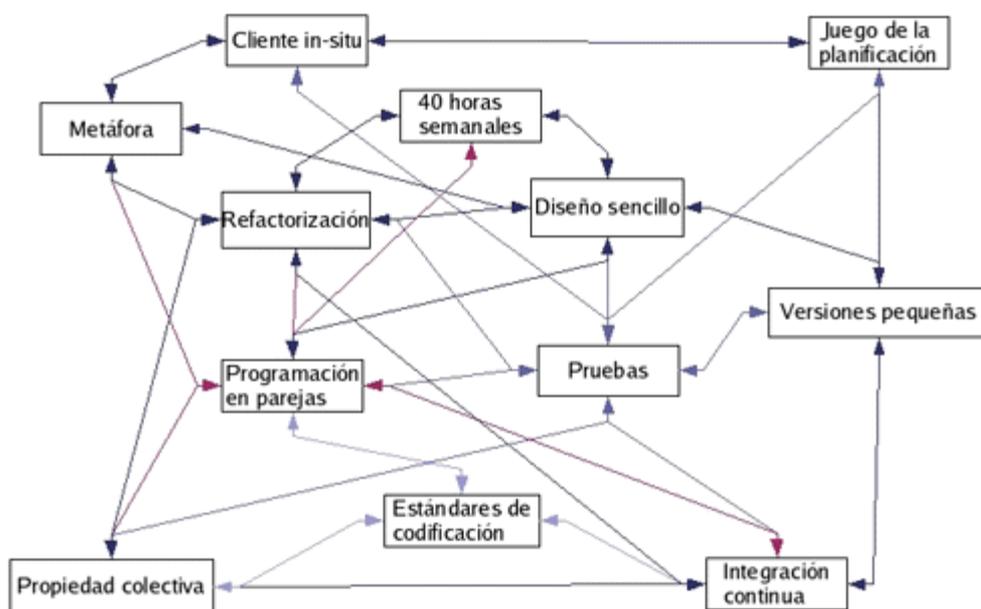
Gráfica 2.1: Planificación y Retroalimentación en XP¹¹

¹¹ http://vc-on.blogspot.com/2007_09_01_archive.html

Extreme Programming promueve la utilización de ciertas prácticas que mejoran la productividad y eficiencia del código.

- **Equipo Completo:** Todos los miembros de un proyecto pertenecen al mismo equipo, ningún rol es propiedad exclusiva de una persona, no existen especialistas, sino contribuyentes con habilidades especiales.
- **Planificación:** Se realizan pequeñas reuniones frecuentemente en las cuales el equipo expresa que es lo que necesita para poder realizar su trabajo, aquí se escriben las historias de usuario, que representan cada uno de los requerimientos que el cliente tiene para poder concluir el proyecto.
- **Pruebas automatizadas:** La automatización de pruebas es muy importante, ya que es muy probable que por presión de tiempo se trate de saltar las pruebas manuales, lo que puede desembocar en errores en el sistema final.
- **Entregas:** Se entrega el software funcionando y una vez haya pasado las respectivas pruebas al cliente, también se entrega software de forma frecuente a usuarios finales.
- **Diseño Simple:** Se procura analizar el código y refactorarlo de ser necesario para mantener la simplicidad del sistema.

- Programación en Pares: El código se escribe en pareja, con esto se asegura que el software productivo fue revisado por al menos otra persona, lo que genera mejores diseños y mayor estabilidad en el software.
- TDD¹²: Se basa en el desarrollo guiado por pruebas, lo cual implica que existen pruebas unitarias para cada parte de funcionalidad significativa del sistema.
- Propiedad Colectiva de Código: Cualquier programador puede mejorar cualquier fragmento de código así no sea creado por el. Con esto se tiene una mejor calidad de código y menores defectos.



Gráfica 2.2: Prácticas XP¹³

¹² <http://www.agiledata.org/essays/tdd.html>

Al tratarse el presente proyecto de un trabajo de innovación que requiere mucha investigación, los requerimientos del sistema son muy susceptibles a sufrir cambios y alteraciones, razón por la cual Extreme Programming encaja perfectamente en este escenario respondiendo de manera muy efectiva ante esta posibilidad. Por otro lado al ser la comunicación con el cliente una de sus prácticas fundamentales, mitiga los posibles riesgos que puedan surgir en el proyecto significativamente.

2.1.3 Software Prototyping

Software Prototyping es una metodología de desarrollo en la cual se crean prototipos de aplicaciones de software, los cuales son versiones incompletas del programa que está siendo desarrollado. La idea detrás de este tipo de desarrollo es permitir a los interesados en el software que tengan una idea del diseño que se está proponiendo desde etapas tempranas del ciclo de vida del mismo y puedan ser fuente de retroalimentación del proyecto. Un prototipo por lo general simula algunos aspectos del software y comparándolo con la versión final, estos pueden ser completamente distintos.

El desarrollo por prototipos permite a los ingenieros de software tener una visión clara de los recursos que están siendo invertidos en el proyecto, y si

¹³ Fuente: <http://www.cyta.com.ar/ta0502/v5n2a1.htm>

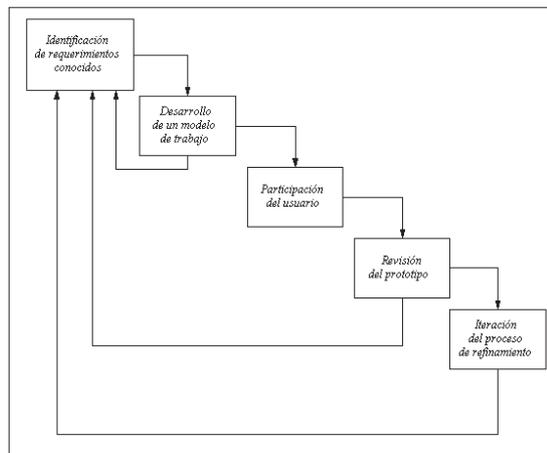
es que los tiempos de entrega van a ser cumplidos con éxito. El proceso de desarrollar un prototipo envuelve los siguientes pasos.

Identificar requerimientos básicos: Determinar los requerimientos básicos incluyendo entradas y salidas de información. Detalles como la seguridad pueden ser ignorados en esta etapa.

Desarrollo de un prototipo inicial: El prototipo inicial es desarrollado el cual únicamente incluye interfaces de usuario.

Revisión: Los clientes y usuarios finales, examinan el prototipo y dan retroalimentación a los desarrolladores, incluyendo cambios y adiciones.

Revisar y mejorar el prototipo: Utilizando las especificaciones y comentarios que dio el cliente acerca del prototipo, este puede ser mejorado. Sin embargo una negociación previa de si los nuevos requerimientos entran dentro del alcance del contrato y producto puede ser necesaria. Si lo cambios son efectivamente introducidos, los pasos 3 y 4 vuelven a ser repetidos.



Gráfica 2.3: Modelo de Desarrollo por Prototipos¹⁴

Prototipos Horizontales

Un prototipo de tipo horizontal muestra una visión mucho más general del sistema o subsistema, concentrado sus esfuerzos en la interacción con el usuario más que en detalles de bajo nivel.

- Confirma los requerimientos de interfaz gráfica del sistema y el alcance del mismo.
- Demuestra una versión muy preliminar del sistema lo cual puede ser útil para encontrar financiamiento para el proyecto.
- Provee un estimado de cuanto será el tiempo de desarrollo, el costo y el esfuerzo que se necesitara para terminarlo.

¹⁴ Fuente: http://www.ub.edu.ar/catedras/ingenieria/mids_web/prototyp/estrdes.htm

Prototipos Verticales

Un prototipo vertical es una elaboración más completa de un subsistema o función. Es muy útil para obtener información detallada de los requerimientos de dicha funcionalidad, con los siguientes beneficios:

- Optimización del diseño de la base de datos
- Obtener información en los volúmenes de datos que serán manejados, interfaces del sistema requeridas, tamaño de los requerimientos de networking y necesidades de performance.

Desventajas del Desarrollo por Prototipos

Si es que no es usado de manera correcta, este tipo de desarrollo también puede tener sus desventajas.

- Análisis insuficiente: El enfocarse en prototipo puede distraer a los programadores del análisis completo del sistema, pasando por alto mejores soluciones o especificaciones incompletas.
- Confusión entre el prototipo y el sistema final: Muchas veces los usuarios tienden a pensar que un prototipo, el cual está hecho para ser luego cambiado drásticamente, es realmente como quedará el sistema final. Esto puede llevar a confusión en cuanto a performance y detalles como seguridad.

- Excesivo tiempo de desarrollo de un prototipo: Por lo general un prototipo está hecho para ser desechado en comparación a la versión final, es por esta razón que si los desarrolladores pierden mucho tiempo en el desarrollo de los mismos, su productividad se verá altamente penalizada en un futuro.
- Costo de desarrollar un prototipo: El costo de desarrollar un prototipo puede ser alto ya que envuelve programadores que deberán trabajar en el mismo.

El desarrollo por prototipos es mucho más beneficioso para sistemas que tengan mucha interacción con los usuarios, además es una excelente herramienta para diseñar buenas interfaces gráficas y obtención iterativa de requerimientos del usuario.

El uso de Software Prototyping en el presente proyecto es fundamental, ya que al tratarse de un trabajo de innovación, muchas veces los requerimientos del proyecto no son claros, y se debe ir ajustando el sistema conforme vaya apareciendo nueva funcionalidad y se investiguen nuevas formas de desarrollo. Por otro lado, para ofrecer la mejor usabilidad, diseño y experiencia de usuario se debe cambiar al sistema conforme el cliente crea necesario.

Es por estas razones que gracias a la creación rápida de prototipos, se obtiene una visión mucho más clara de lo que el cliente en realidad desea, y

en tempranas etapas del ciclo de vida del software, por lo que se mitiga el riesgo de cambio considerablemente.

2.2 Content Management Systems

2.2.1 Introducción

Un sistema manejador de contenidos o CMS es un sistema de gestión de sitios Web e intranets que integra herramientas para crear y gestionar (añadir, editar) contenido en tiempo real sin necesidad de código de programación, cuyo objetivo es estructurar y facilitar la creación, administración, distribución, publicación y disponibilidad de la información.

Su característica principal es la gran cantidad de funcionalidad que se pueden agregar a través de extensiones y complementos (galerías de imágenes, administradores de encuestas, gestores de formularios, entre otros).

Se puede considerar a un CMS como un marco o esqueleto pre estructurado del sitio web, con características básicas de usabilidad, visualización y gestión de su contenido. Con lo cual permite la creación, almacenamiento y administración de información de forma dinámica a través de una interfaz web a la cual se tiene acceso a través del internet, con estos se reduce el costo de mantenimiento del portal con respecto a un sitio web estático.

Un CMS permite a la empresa tener autonomía total sobre el contenido y la evolución de su presencia en Internet, prescindiendo de la ayuda de terceras personas o empresas especializadas para el mantenimiento rutinario. En muchos casos ni siquiera se necesita de un empleado dedicado al sitio web (web master), debido a que cada miembro del equipo puede gestionar su propio contenido, reduciendo los recursos humanos necesarios y el costo de la creación, la contribución y el mantenimiento de contenidos. La habilidad requerida para trabajar con un CMS no va mucho más allá de los conocimientos necesarios para la utilización de un editor de texto simple.

La presentación de un sitio web creado con un CMS es personalizable gracias a la utilización de temas y diseños que pueden ser fácilmente modificados e inclusive pueden ser encontrados de manera gratuita en el internet.

Como ejemplo de CMS se tiene a WordPress que es un sistema CMS Open Source hecho en PHP, con el que se puede crear blogs o portales de alta calidad con características de gestión cada vez más completas. Otros ejemplos son Liferay, Joomla, Drupal.

2.2.2 Ventajas de un CMS

El uso de un CMS brinda varias ventajas, muchas de las cuales se han ido incluyendo con el pasar de los años basados en las necesidades actuales de

los usuarios y mejoras que la propia comunidad de usuarios ha ido sugiriendo.

- Fácil de actualizar y mantener: Gracias a sus capacidades de administración y configuración, ajustar un sitio o actualizarlo es muy fácil cuando el portal ha sido establecido correctamente. No hay necesidad de saber HTML, ya que los CMS de calidad están distribuidos de tal forma que sus opciones son reconocibles intuitivamente y poseen gran documentación en internet.
- Motores de Búsqueda: La optimización de un sitio para los motores de búsqueda es una fase importante en el desarrollo de un sitio web. Con la ayuda de módulos, extensiones y complementos, junto con una gran investigación de palabras clave, hacen que el sitio aparezca en la primera página de los motores de búsqueda.
- Libre y de código abierto: Cuando un CMS es de carácter libre y su código open source no tienen costo alguno, no empiezan a cobrar cuando el sitio alcanza un tamaño específico, ni bloquean parte de su funcionalidad a los usuarios.
- Plugins¹⁵: Existen una gran cantidad de plugins y complementos para agregar funcionalidad específica al sitio web. Banners de propaganda, encuestas, formularios, mapas. La gran mayoría de las necesidades

¹⁵ http://en.wikipedia.org/wiki/Plug-in_%28computing%29

que pueda tener el administrador del sitio ya se encuentran implementadas a través de plugins.

- Seguridad: Al ser sistemas que son utilizados por muchísimos usuarios a nivel global, su seguridad es mejorada día tras día, con lo cual se ahorra muchos costos que implicaría el tener una persona dedicada a tratar temas de seguridad del sitio. También brinda herramientas para administrar la base de datos y hacer copias de seguridad de la misma.
- Control de Revisiones: Existen CMS en los cuales se puede guardar revisiones de las páginas que se está editando, de tal forma que se puede volver a un punto anterior cada vez que el administrador lo requiera, también brinda cierto grado de auditoría al saber quien realizó los cambios y cuando los hizo efectivos.

Gestión de Usuarios: Los CMS proveen sistemas completos de control de usuarios y roles, con lo que los administradores simplemente escogen que partes del sitio web quieren mostrar e inclusive llegan a ofrecer permisos de modificación y edición completamente personalizados.

2.2.3 Drupal

Drupal es un sistema manejador de contenido open source. Lo que significa que su código está disponible de manera libre para cualquier persona que quiera hacer uso del mismo y modificarlo.

De igual manera significa que no se debe pagar por la utilización de Drupal, al ser su licencia de carácter GNU/GPL. Esto es conveniente para los usuarios ya que no deben desarrollar todo el sitio por su cuenta, además se puede llegar a tomar ventaja de documentación existente y contribuida por miles de usuarios que utilizan Drupal.

Al ser un sistema de gestión de contenidos es un software que facilita la creación, organización, manipulación y eliminación de la información representada en forma de imágenes, documentos, guiones y texto plano.

La funcionalidad de Drupal puede ser extendida a través del uso de módulos contribuidos por la comunidad o desarrollados por los usuarios, también la apariencia y comportamiento de Drupal pueden ser alterados por dichos módulos. A la fecha de Julio de 2012, hay cerca de 17.200 plugins libres listos para ser utilizados.

A pesar de que Drupal ofrece una interfaz de programación sofisticada para los desarrolladores, no es necesario tener conocimientos de programación para la instalación de sitio web básico y su administración.

Al existir gran variedad de CMS tales como WordPress, Joomla, Liferay, Plone, Magento, entre otros, Drupal sobresale al presentar ciertas ventajas que benefician al usuarios.

Primero existen 3 puntos que deben discutirse al momento de considerar el uso de un CMS y si resulta ser la tecnología que podrá cumplir nuestras necesidades.

- **Confiable y Robusto:** ¿Existen errores en el código? ¿Afectará al sitio, se debe añadir parches o actualizaciones para arreglar el código defectuoso?
- **Eficiencia:** ¿El software utiliza los recursos del servidor de manera adecuada? ¿Es propenso a tener problemas de concurrencia o velocidad desde el inicio?
- **Flexibilidad:** ¿Si se cambia de opinión acerca de lo que se quiere para el sitio, será capaz de implementar los cambios sin volver a hacer todo desde cero?

Drupal ha sido cuidadosamente elaborado y bien diseñado con el paso de los años. De hecho, los puntos anteriormente mencionados se toman tan en serio por los desarrolladores de Drupal que están escritas en su conjunto de principios y prácticas base, disponibles en <http://drupal.org/node/21945>.

Segundo, desde el punto de vista del administrador, para escoger un sistema CMS se debe tener en cuenta los siguientes puntos.

- Fácil de configurar y ejecutar: ¿Se puede empezar a crear un sitio con el mínimo de esfuerzo? ¿Se tiene que aprender acerca de otras tecnologías antes de estar en condiciones de utilizar el CMS?
- Intuitivo: Una vez que haya decidido comenzar a utilizar el CMS, ¿va a ser fácil de aprender cosas nuevas? ¿Si no se tienen conocimientos técnicos altos, se va a tener problemas para administrar el sitio?
- Flexible y fácil de extender: Se puede crear un sitio único, sofisticado e innovador utilizando el CMS?
- Seguridad: ¿Páginas web hechas con el CMS han sido utilizadas con éxito en aplicaciones del mundo real? ¿Los errores de seguridad conocidos son solucionados de forma rápida y con regularidad?

Todos estos puntos son precisamente temas por los cuales Drupal es reconocido. Además, Drupal es la plataforma más robusta si se la compara directamente contra sus competidores más cercanos, WordPress y Joomla.

Por otro lado posee excelente documentación, invaluable al momento de crear módulos personalizados, ya que desde su concepción Drupal fue pensado para que desarrolladores puedan crear módulos de forma rápida e intuitiva.

Sin embargo, como punto en contra Drupal no es tan amigable para el administrador como si lo son otros CMS, además, los temas para personalizar la apariencia del sitio claramente se quedan detrás de lo que posee su competencia.

	WordPress	Joomla	Drupal
Facilidad de hosting e instalación	●	●	●
Facilidad de configuración: Sitios Simples	●	○	○
Facilidad de configuración: Sitios Complejos	●	●	○
Facilidad de Uso: Editores de Contenido	●	○	○
Flexibilidad Gráfica	●	●	●
Accesibilidad y optimización para motores de búsqueda	○	●	○
Flexibilidad Estructural	○	○	●
Roles de Usuario y Workflow	○	●	○
Comunidad/Web 2.0	●	○	●
Extensión e Integración	●	●	●
Seguridad	○	○	
Soporte/Fortaleza Comunidad	●	●	●
○ Alguna ○ Solida ● Excelente			

Tabla 2.1: Comparativa WordPress, Joomla, Drupal¹⁶

¹⁶ Fuente: <http://www.idealware.org/reports/2010-os-cms>

Una vez analizados estos puntos se puede concluir que de entre los CMS más grandes y conocidos Drupal es la mejor alternativa gracias a las ventajas que ofrece a los desarrolladores al momento de extender y crear módulos personalizados. Esto es justamente lo que se busca para la funcionalidad del módulo web del presente trabajo de desarrollo.

Es por esa razón que Drupal CMS ha sido elegido como el manejador de contenidos que alojará la parte web del sistema desarrollado en el presente trabajo de investigación.

2.2.3.3 Arquitectura

Drupal está desarrollado en el lenguaje PHP, razón por la cual requiere de un servidor web que sea capaz de procesar páginas que contengan código PHP, también, al ser un sistema gestor de contenido dinámico, es un requisito previo se disponga de una base de datos para instalar la estructura necesaria con la cual funcionará. Por defecto Drupal ofrece soporte de conectividad para las bases de datos MySQL¹⁷ y MariaDB¹⁸, sin embargo es posible adicionar librerías si se requiere instalar el CMS utilizando otras bases de datos.

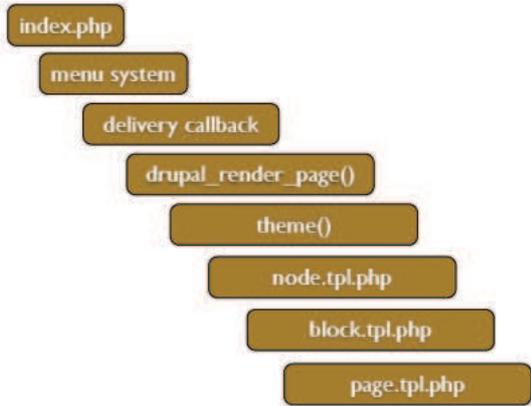
¹⁷ <http://www.mysql.com/>

¹⁸ <http://mariadb.org/>



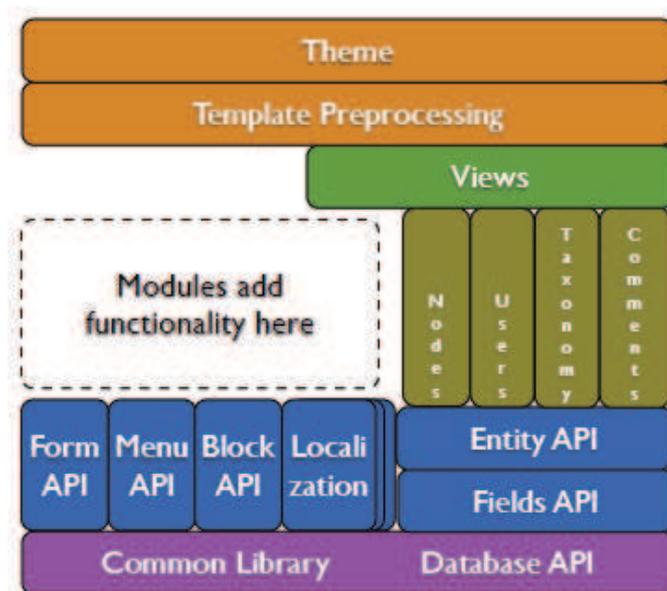
Gráfica 2.4: Arquitectura Física Drupal¹⁹

El proceso de visualización de una página web en Drupal se realiza en forma secuencial, la cual comienza con la petición que el cliente realiza hacia el servidor; seguidamente se invoca el archivo index.php que empezará a invocar funciones de los módulos que sean necesarios, uno tras otro, dependiendo de la URL que el cliente haya solicitado.



Gráfica 2.5: Invocación Página Web en Drupal²⁰

¹⁹ Fuente: http://www.ent.iastate.edu/it/Intro_to_Drupal_7_Architecture.pdf



Gráfica 2.6: Arquitectura Lógica Drupal²¹

2.2.3.4 Módulos

Los módulos son la base de Drupal, de hecho mucho de su poder y robustez se encuentra en sus módulos, que no son más que fragmentos de funcionalidad intercambiables que constituyen la base del sistema o extienden el mismo para llevar a cabo tareas especiales.

Para poder añadir funcionalidad personalizada al sistema, se puede desarrollar un nuevo módulo, para lo cual se crean dos archivos básicos. El primer archivo le informa a Drupal acerca de la existencia del módulo, mas no su código. Y el segundo archivo representa en si el código que ejecutará

²⁰ Fuente: http://www.ent.iastate.edu/it/Intro_to_Drupal_7_Architecture.pdf

²¹ Fuente: http://www.ent.iastate.edu/it/Intro_to_Drupal_7_Architecture.pdf

el módulo, este puede ser tan sencillo como tres líneas de código o incluir complejas funciones y archivos externos.

La creación de un módulo es algo que cualquiera persona con conocimiento básico de PHP puede realizar. Simplemente debe seguir reglas especiales como nombrar funciones de manera específica.



Gráfica 2.7: Estructura básica de un módulo en Drupal²²

El core de Drupal ofrece a los módulos la oportunidad de integrarse en el ciclo de procesamiento de las páginas a través del uso de funciones especiales llamadas hooks. Un ejemplo de esto es la inicialización de una página; al inicio del ciclo de vida de visualización de una página Drupal verifica si algún módulo requiere ser inicializado, para hacerlo escanea los módulos existentes en el sistema y verifica si es que alguno de ellos implementa la función llamada `hook_init()`.

Implementar un hook en Drupal significa declarar una función que siga el patrón de nombramiento `nombre-módulo_init()`. Así por ejemplo si nuestro módulo se llamara `visitas`, la función `hook_init()` sería nombrada `visitas_init()`.

²² Fuente: <http://www.missingubercartmanual.com/How-to-Create-a-Basic-Drupal-Module>

A través de esta función se brinda al módulo la posibilidad de inicializarse y agregar los recursos que necesite durante el transcurso de la petición de la página. Conforme el ciclo de vida avanza Drupal procesa hook tras hook, dando la oportunidad a los módulos de llevar a cabo su funcionalidad.

El sistema de hooks de Drupal es quizá el paso más sencillo y más importante dentro del desarrollo de módulos, que a la final no son más que una colección de funciones hook que Drupal irá llamando cuando así lo requiera.

```
<?php
/*****
 * Implementation of hook_block()
 * @see http://api.drupal.org/api/function/hook\_block/6
 * create a minimal block
 */
function yourModule_block( $op = 'list', $delta = 0, $edit = array() ) {
  switch ( $op ) {
    case 'list':
      // appears on the Blocks configuration page admin/build/block
      $blocks = array();
      $blocks[0]['info'] = t('My Block');
      return $blocks;
    case 'view':
      // content of block; required to have 'subject' and 'content' elements
      $blocks = array();
      $blocks['subject'] = t('<center>My Block Status</center>');
      $blocks['content'] = t('<center>My Block Content</center>');
      return $blocks;
  }
}
```

Gráfica 2.8: Implementación de una función hook²³

²³ Fuente: <http://www.missingubercartmanual.com/How-to-Create-a-Basic-Drupal-Module>

2.3 Imágenes Panorámicas

2.3.1 Introducción

La fotografía panorámica es una técnica de la fotografía, con la cual, usando equipos y software especializado, es posible capturar imágenes con un campo de visión alargado. El término también se aplica a fotografías que han sido recortadas en una relación de aspecto relativamente ancha.

Una imagen que muestra un campo de visión mayor a la del ojo humano, cerca de 160 grados por 75 grados puede ser denominada panorámica. Esto generalmente significa que tiene una relación de aspecto de 2:1 o mayor, teniendo la imagen al menos 2 veces más ancho que su largo.

Algunas de las imágenes panorámicas tienen relaciones de aspecto de 4:1 y algunas veces pueden llegar a tener una relación de 10:1, con campos de visión de hasta 360 grados. Tanto la relación de aspecto como la cobertura de la fotografía son factores importantes en definir una imagen panorámica verdadera.

La fotografía panorámica ha existido por más de un siglo, ya que se puede encontrar ejemplos de fotos panorámicas en 360 grados que datan ²⁴de hace más de 100 años.

²⁴ http://www.archives.gov/exhibits/panoramic_photography/part_2.html

Actualmente la fotografía panorámica ha tomado mucha fuerza gracias a la reducción en costos de la tecnología necesaria para tomar y visualizar este tipo de fotos. Otro aspecto muy importante es que el aumento en la velocidad del internet hace posible que se pueda compartir fotos con otras personas alrededor del mundo. Gracias a esto surgieron nuevas formas de promocionar lugares, eventos, y negocios a través del internet haciendo uso de fotos panorámicas.

2.3.2 Usos

La fotografía panorámica se utiliza ampliamente en las universidades y en la industria de bienes raíces. Las fotos panorámicas permiten al usuario ver el entorno de un lugar de manera remota por lo general a través del internet. En la actualidad una gran variedad de industrias utilizan esta tecnología para ayudar a comercializar sus servicios y productos.

En los últimos años la calidad, usabilidad y accesibilidad de las fotos panorámicas ha mejorado considerablemente, como ejemplo de esto, se tiene ciertos sitios web que permiten al usuario navegar a través de tours virtuales conformados por varias fotos panorámicas simplemente haciendo clic en mapas o planos del piso integrados en ellas.

Para la mayoría de los negocios, una visita virtual debe ser accesible desde todas partes, por lo cual, es lógico pensar que la solución principal es un viaje virtual mostrado a través de la web.

Por otro lado, una visita virtual útil no está sólo conformada por una serie de fotografías panorámicas. Una mejor experiencia se puede obtener mediante la visualización de una gran variedad de elementos como videos, textos, imágenes fijas y contenido interactivo en la web. Un estudio realizado por el Grupo PEW Research²⁵ mostró que más de 5 millones de estadounidenses vieron visitas virtuales todos los días en el año 2004 y que es probable que 108 millones de estadounidenses tuvieran acceso a visitas virtuales para agosto de 2010.

Las visitas virtuales son muy populares en la industria de bienes raíces. Ya que ayudan a mostrar a los clientes el lugar en el cual están interesados sin necesidad de que lo visiten físicamente, esto trae muchas ventajas, especialmente cuando los clientes están por mudarse a una nueva localización a kilómetros de su ubicación actual.

2.3.3 Tipos

Existen varios tipos de fotos panorámicas al igual que varias palabras utilizadas para describirlas, sin embargo se las puede dividir en los siguientes grupos principales.

²⁵ <http://pewresearch.org/>

Fotos panorámicas planas: Se las conoce como fotografías planas a las fotos convencionales que son vistas tal cual las tomaron, sin ninguna clase de corrección en la perspectiva de las mismas.

Fotos panorámicas cilíndricas: Una foto cilíndrica se visualiza como si estuviera curvada dentro de un cilindro. Si se viera la imagen original sin ninguna corrección, se podría apreciar que tiene curvas a través del eje horizontal.

Fotos panorámicas esféricas: También denominadas imágenes de formato equirectangular, representan exactamente 360 grados en el eje horizontal y 180 grados en el vertical. Estas imágenes se visualizan como si fueran mostradas por un proyector de diapositivas ubicado en el interior de una esfera. De la misma forma que las imágenes cilíndricas, curvas horizontales se podrían apreciar si se tuviera la imagen original sin ningún tipo de corrección, pero con la diferencia que las partes inferior y superior de la imagen parecerían estar aplastadas.

En la práctica cualquier imagen panorámica se puede almacenar como una proyección esférica. Sin embargo, si la imagen es inferior a 360 grados de ancho, debe ser complementada con píxeles vacíos, o como alternativa el campo real de visión horizontal debe ser acortado de tal forma que el software pueda mostrar la imagen de manera correcta.

Fotos panorámicas cúbicas: Las imágenes panorámicas cúbicas se hicieron populares gracias al software QuickTime 5, que introdujo la posibilidad de

panoramas cúbicos en los cuales el espectador podía mirar hacia arriba y hacia abajo en el suelo. A diferencia de las imágenes cilíndricas y esféricas, las imágenes cúbicas son formadas por 6 imágenes separadas que representan cada cara del cubo. El formato cúbico es el más eficaz para obtener imágenes panorámicas con un campo de visión vertical mayor a 140 grados, y además, ya que cada una de las caras del cubo se ve como si fueran una imagen normal son más fáciles de editar o modificar con el software de edición de imágenes.

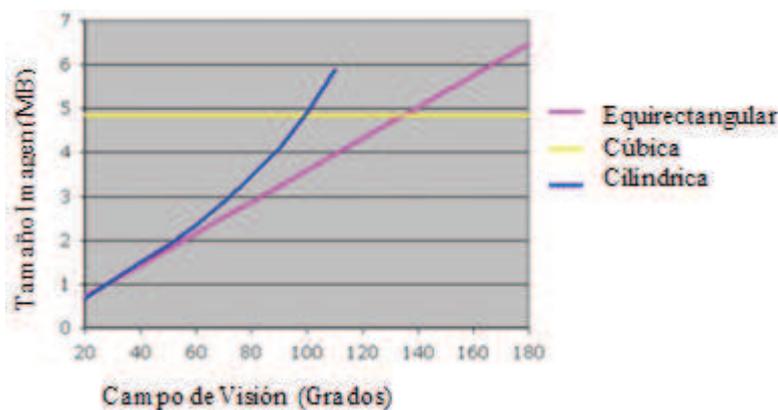


Gráfica 2.9: Tipos de Imágenes Panorámicas²⁶

Un panorama esférico se puede representar mediante una imagen equirectangular para crear una esfera, o lo mismo se puede lograr si la

²⁶ Fuente: <http://www.panoguide.com/howto/panoramas/types.jsp>

imagen se convierte en un cubo. Pero el cubo es más eficiente para almacenar la información de la imagen. A continuación se muestra una comparativa del tamaño de la imagen dependiendo del formato que utilice.



Gráfica 2.10: Tamaño imágenes dependiendo de su formato²⁷

Debido a las características de rendimiento y tamaño que ofrecen los panoramas cúbicos se utilizó dicho formato para el presente proyecto, inclusive si se trabaja con archivos panorámicos sin procesar, estos serán convertidos de tal forma que siempre se utilicen panoramas cúbicos para las visitas virtuales.

2.3.4 Equipo

Se puede crear un panorama mediante el uso de cualquier cámara, inclusive una desechable, en realidad no se necesita de ningún otro equipo, sin

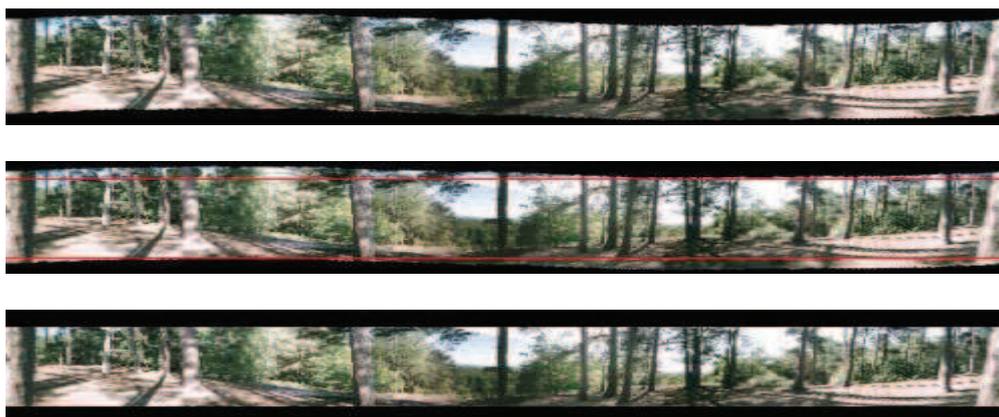
²⁷ Fuente: <http://www.panoguide.com/howto/panoramas/types.jsp>

embargo es muy probable que no se puedan obtener panoramas de buena calidad al no utilizar equipo especializado. Con el uso de aditamentos, se puede tomar fotos sin mayores complicaciones y sin perder detalles de la imagen.

Trípode

Mediante el uso de un trípode se puede asegurar que la cámara gira alrededor de un punto fijo y que se mantiene en un nivel constante. Es importante comprobar que la cámara quede firmemente bloqueada por el cabezal del trípode y que no se tambalee.

Si no se usa un trípode o se falla al momento de mantener el nivel de la cámara constante es posible que al momento de trasladar las imágenes al software de procesamiento, se tenga que recortarlas o reducir su campo de visión.



Gráfica 2.11: Pérdida de campo de visión en la foto²⁸

²⁸ Fuente: <http://www.panoguide.com/howto/panoramas/gear.jsp>

Nivel de burbuja

Un nivel de burbuja es una herramienta muy útil que se puede comprar por un costo muy bajo. Existen niveles diseñados específicamente para encajar en la base de las cámaras. Con lo cual se asegura que se mantiene un nivel constante al momento de tomar fotos.

Cabezal panorámico

Un cabezal panorámico es un accesorio adicional que se pone en la parte superior del trípode antes de conectar la cámara. Permite posicionar la cámara de manera que a medida que se desplaza evita el efecto “parallax²⁹”. Además de esto, los cabezales ayudan a rotar la cámara de manera mucho más sencilla.



Gráfica 2.12: Cabezal Panorámico³⁰

²⁹ <http://en.wikipedia.org/wiki/Parallax>

³⁰ Fuente: <http://www.red-door.co.uk/pages/productpages/nodal-ninja-panorama-head-ultimate-M1-S.html>

Lente gran angular

Con la ayuda de un lente gran angular no se necesita tomar tantas fotos para crear un panorama de 360 grados. Sin embargo, hay que tomar en cuenta que muchos programas son solo compatibles con cierto tipo de lentes. Razón por la cual se debe investigar si el software con el que se trabajará soporta el lente gran angular utilizado.



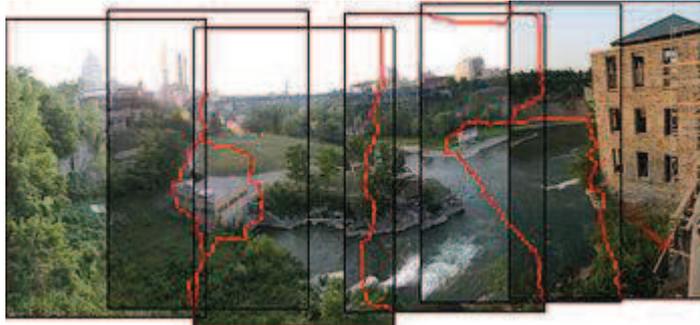
Gráfica 2.13: Lente gran angular³¹

2.3.5 Stitching

El Stitching es el proceso de combinar múltiples imágenes fotográficas con campos de visión superpuestos para producir un panorama segmentado o imagen de alta resolución. El stitching por lo general se realiza con el uso un software especializado aunque algunas cámaras lo realizan internamente en su memoria.

³¹ Fuente: http://es.wikipedia.org/wiki/Objetivo_gran_angular

Este proceso se puede dividir en 3 partes: registro de imágenes, calibración y optimización de las imágenes. Durante el registro se busca un set de imágenes que tengan características similares, luego se las alinea para que existan diferencias mínimas entre píxeles superpuestos.



Gráfica 2.14: Stitching de imágenes³²

Al momento de calibrar las imágenes se trata de minimizar las diferencias entre un modelo ideal de campo de visión, y el campo de las fotos que se fueron combinando, en este punto se corrigen defectos ópticos como distorsiones, diferencias entre imágenes, aberraciones cromáticas, entre otras.

Por último durante la optimización de imágenes, se altera la imagen resultante basándose en lo encontrado en las fases anteriores, y se aplican diferentes filtros a las fotos para alcanzar la mejor calidad posible.

³² Fuente: http://en.wikipedia.org/wiki/Image_stitching

Software

Existe software especializado para el proceso de Stitching entre los cuales se puede enumerar los siguientes programas:

- Autostitch
- Hugin
- Panorama Maker
- PTGui
- Panorama Tools
- Microsoft Image Editor
- Stitcher CleVR

Existen muchos otros programas que también pueden unir imágenes múltiples aun cuando no sea su objetivo primario el crear fotografías panorámicas, un ejemplo popular es el conocido programa Photoshop³³ de Adobe Systems, que incluye una herramienta conocida como Photomerge.

2.3.6 Krpano

2.3.6.1 Introducción

Krpano Viewer es un componente de software de alto rendimiento que permite visualizar toda clase de imágenes panorámicas y visitas virtuales interactivas. El visor está disponible como aplicación de Flash y HTML5.

³³ <http://www.adobe.com/es/products/photoshop.html>

El visor está diseñado para que su uso sea posible a través de un navegador en el caso de computadores de escritorio (Windows, Mac, Linux) o en dispositivos móviles y tablets (iPhone, iPad, Android).

Además del visor krpano incluye un set de herramientas llamadas “krpano tools”, las cuales ayudan a preparar las imágenes panorámicas para ser visualizadas y utilizadas. Gracias al uso de estas herramientas el crear un panorama o visita virtual es posible de forma muy rápida y sencilla.

Características

Flash y HTML5: El visor de krpano está disponible como una aplicación de flash o de HTML5. Ambos aplicativos tienen y utilizan las mismas interfaces para controlar los panoramas y su presentación.

- Alto performance y renderizado de alta calidad: El renderizado con alta calidad de imágenes y su performance son una de las características más importantes del desarrollo de krpano. Adicional a esto, el visor se mantiene lo más pequeño y eficiente como sea posible.
- Altamente personalizable: Casi todo es personalizable en el visor de krpano. Existen muchas opciones por defecto, sin embargo la adición de nueva funcionalidad y configuración personalizada es posible en caso de ser necesaria.

Krpano tiene un sistema de scripting basado en XML muy flexible y dinámico. Para construir interfaces de usuario simplemente basta con incluir y organizar imágenes dentro de la pantalla del visor. En caso de que fuera necesaria una personalización más profunda del visor es posible incluir plugins externos hechos en flash o HTML5.

- Amplia gama de formatos de imágenes panorámicas y archivos:
 - Imágenes panorámicas cúbicas.
 - Imágenes panorámicas esféricas (360x180 grados).
 - Imágenes panorámicas cilíndricas.
 - Panoramas parciales esféricos o cilíndricos.
 - Imágenes planas.
 - Archivos cúbicos QTVR (. mov).
 - Imágenes Zoomify
 - Videos panorámicos (esféricos, cilíndricos, parciales).
 - Archivos JPEG (*. Jpg, *. Jpeg).
 - Archivos TIFF / BigTIFF (*. Tif, *. Tiff, *. BTF, *. TF8, *. Bigtiff).
 - Documentos de Photoshop (*. Psd, *. Psb).
 - Kolor RAW (*. Kro).

- **Multi-Resolución:** Cargar imágenes muy grandes como por ejemplo en el tamaño de giga píxeles, no es posible hoy en día ya que se necesitaría demasiada capacidad de procesamiento y una cantidad de memoria muy grande como para un aplicativo web. Para resolver este problema krpano utiliza un sistema de carga de multi resoluciones, donde las imágenes de gran tamaño se convierten a varios niveles de resolución, los cuales son divididos en imágenes más pequeñas. Cada una de estas piezas es cargada bajo demanda por el visor, con este procedimiento se logra ver imágenes grandes muy rápido y sin mucho consumo de memoria.
- **Visitas Virtuales/Hotspots/Scripting:** Mediante el uso de varias imágenes panorámicas, hotspots, mapas y thumbnails, es posible crear visitas virtuales muy complejas.

2.3.6.2 Licencia

El visor de krpano y sus herramientas pueden ser descargadas para su utilización en un ambiente de pruebas sin costo alguno, sin embargo si se quiere implementar el visor en un ambiente final de producción se debe pagar una licencia, caso contrario aparecerán marcas de agua dentro de las fotos panorámicas.

Además de esto, para que dispositivos móviles puedan ver los panoramas en HTML 5, se debe adquirir una licencia adicional específica para estos dispositivos. Así, en total Krpano requiere de 3 licencias, las 2 primeras (visor y herramientas) tienen un costo de 90 euros, y la licencia adicional para iPhone y iPad que tiene un costo de 39 euros.

Licencia	Descripción	Costo (Dólares)
Krpano	Licencia necesaria para el visor flash de krpano y la utilización de krpano Tools	\$110.6
Krpano HTML5/iPhone/iPad	Licencia para poder utilizar el visor HTML5 de krpano	\$47.93

Tabla 2.2: Licencias Krpano

2.3.6.3 Justificación

A pesar de ser una herramienta con licencia comercial, Krpano resulta ser la mejor opción si se quiere desarrollar visitas virtuales de forma personalizada. Como primer punto a su favor, la documentación que posee es excelente, se puede encontrar referencias a cada una de las funciones disponibles dentro del visor y su descripción en la página web oficial de krpano.

Como segundo punto a su favor, sus capacidades de scripting dinámico hacen que sea muy fácil implementar funcionalidad sobre el visor en forma programática, llegando inclusive a ofrecer la posibilidad de interactuar con sitios web gracias a funciones JavaScript que ofrece.

Por último, es el único visor disponible a la fecha que utiliza HTML5 para su funcionamiento, única opción para poder mostrar visitas virtuales dentro de dispositivos como el iPhone y el iPad que no poseen Flash.

Es por estas razones que Krpano se convierte en la mejor opción para desarrollar el sistema de visitas virtuales, que requiere de un visor muy eficiente que pueda interactuar con los administradores y usuarios finales del sitio web.

2.4 Realidad Aumentada

2.4.1 Definición

Realidad aumentada o AR, en su forma más sencilla se puede definir como el arte de sobreponer contenidos generados por medio de un computador sobre una imagen o vista directa del mundo que nos rodea.

Es, literalmente, mejorar o aumentar el entorno a nuestro alrededor. El ejemplo más frecuente del uso de realidad aumentada y con el que la mayoría de personas está familiarizado es el análisis de los deportes.

En la transmisión de un partido de fútbol, la realidad estaría representada por las imágenes del juego en sí, las cuales se aumentan con flechas o dibujos que ayudan a ver datos que pueden ser de interés al espectador, como por ejemplo, la distancia del jugador al arco en un tiro libre.

Por otro lado cuando se analiza una jugada que está fuera de juego, los comentaristas se apoyan de líneas trazadas en una capa gráfica adicional que por supuesto no existe en el campo real.

AR trata de aumentar el medio ambiente del mundo real con información virtual tratando de mejorar los sentidos y habilidades de la gente. AR mezcla personajes virtuales con el mundo real. Se pueden identificar 3 escenas comunes en una escena AR, la primera es la combinación de lo real y lo virtual, la segunda añade interactividad en tiempo real, y por último se pueden registrar escenas 3D en el mundo real.

2.4.2 Hardware Requerido

En la práctica existen muchísimas formas de aplicar realidad aumentada, sin embargo debido a que su uso aumentó exponencialmente gracias al

desarrollo de dispositivos móviles, se listan los requisitos necesarios en cuanto a hardware para que sea posible aumentar la realidad usando un dispositivo móvil.

Cámara

El primero y más obvio de los requerimientos de hardware es una cámara, a través de la cual se captura un video de lo que el usuario se encuentra visualizando en ese instante.

Hay dos formas de construir aplicaciones de realidad aumentada sobre las capacidades de video de un dispositivo móvil. En primer lugar, se puede examinar activamente los elementos del video capturados en sí, a través de técnicas especiales de reconocimiento de patrones y de la rama de “Computer Vision³⁴” en general.

O bien, se puede utilizar el video capturado como fondo para la aplicación, ignorando completamente los contenidos del mismo. El segundo enfoque es muy utilizado en aplicaciones de realidad aumentada ya que el costo de procesamiento involucrado con inspeccionar el video es por lo general muy alto.

³⁴ http://en.wikipedia.org/wiki/Computer_vision

GPS

El GPS es necesario únicamente si se va a desarrollar aplicaciones de realidad aumentada basadas en localización. Esto significa que existirán puntos de interés sobre los cuales se conoce de antemano su ubicación en coordenadas geográficas, latitud y longitud.

En este caso es necesario que se sepa en qué punto de la tierra se encuentra el dispositivo, lo cual se puede hacer apoyándose en el GPS del o tecnologías alternativas no tan eficaces, como por ejemplo, triangulación de torres de señal celular.

Acelerómetro

El acelerómetro es un sensor que se encarga de medir la aceleración relativa del dispositivo comparada con una aceleración en caída libre de referencia. Pueden medir la magnitud y la dirección de aceleración, y ser usados para saber en qué orientación se encuentra el dispositivo.

Las aplicaciones de realidad aumentada se ven beneficiadas con el uso del acelerómetro para acomodar el campo de visión relativo a la orientación del dispositivo.

En el caso que la orientación cambie es primordial acomodar los elementos agregados a la pantalla, manteniendo una experiencia de usuario estable.

Cuando se trata de mediciones de aceleración, se debe tener en cuenta que el acelerómetro mide justamente la aceleración lineal del dispositivo. Cuando está en reposo (en cualquier orientación), las mediciones representan la fuerza de gravedad que actúa sobre el dispositivo, y corresponden a su balanceo y orientación (en las direcciones X, Y por lo menos). Pero mientras está en movimiento, el acelerómetro representa la aceleración de la gravedad, además de la aceleración del propio dispositivo en relación con su estado de reposo.

Giroscopio

Un giroscopio se puede utilizar para medir, o mantener, la orientación de un dispositivo. A diferencia del acelerómetro que mide la aceleración lineal del dispositivo, un giroscopio mide la orientación directa.

Cuando se menciona al giroscopio, la mayoría de la gente piensa inmediatamente en los giroscopios mecánicos, que son esencialmente, un disco giratorio que montado dentro de aros múltiples le permite tomar cualquier orientación. Sin embargo, el giroscopio interno de los dispositivos móviles está basado en MEMS ³⁵(o más bien tres giroscopios MEMS independientes).

³⁵ http://en.wikipedia.org/wiki/Microelectromechanical_systems

Las aplicaciones de realidad aumentada se benefician del uso de un giroscopio ya que cuando este se combina con el acelerómetro, los desarrolladores de aplicaciones pueden medir el movimiento del dispositivo en 6 ejes, arriba, abajo, izquierda, derecha, adelante y hacia atrás.

Brújula Digital (Magnetómetro)

El magnetómetro mide la intensidad del campo magnético que rodea al dispositivo. Si existiera ausencia de campos magnéticos locales que afecten al dispositivo, esta medida corresponderá al campo magnético de la Tierra, permitiendo determinar la orientación con respecto al Polo Norte geomagnético.

La orientación geomagnética y la verdadera orientación del dispositivo, en relación con el Polo Norte geográfico, puede variar ampliamente (por varias decenas de grados, dependiendo de la ubicación del mismo). Sin embargo obteniendo ambos datos se pueden hacer correcciones matemáticas para obtener información eficaz.

En las aplicaciones de realidad aumentada, especialmente las que son basadas en geo localización, el conocer estos datos es primordial para saber hacia qué dirección se está dirigiendo el dispositivo. Norte, Sur, Este u Oeste, además permite realizar cálculos para determinar la distancia y ubicación relativa con respecto a determinados puntos de interés.



Gráfica 2.15: Hardware Realidad Aumentada³⁶

2.4.3 Aplicaciones

AR tiene muchas aplicaciones en áreas que se pueden beneficiar de la utilización de la tecnología de realidad aumentada. Al principio AR tenía un enfoque militar e industrial, pero pronto se introdujo para el uso comercial y en la industria del entretenimiento.

Tareas de apoyo

Tareas complejas, como montaje, mantenimiento y cirugía pueden ser facilitadas mediante la inserción de etiquetas con instrucciones especiales en el campo de visión, dependiendo del instrumento o parte con la cual se esté trabajando. Como ejemplo se tiene a Volkswagen que utiliza AR para analizar, comparar y verificar las diferencias entre partes.

³⁶ Fuente: http://www.societyofrobots.com/sensors_accelerometer.shtml

<http://en.wikipedia.org/wiki/Gyroscope>

Medicina

Un ejemplo interesante es la vista virtual de rayos X tomados en una tomografía previa sobre el paciente, ayudando al médico a ver diferencias, mejoras y tomar decisiones basándose en las imágenes.

Navegación

El ejemplo clásico de la utilización de realidad aumentada se lo tiene en los mapas, donde el destino como la ruta para llegar a un punto se marca sobre la cámara del dispositivo, ayudando al usuario a orientarse de mejor manera.

Militares y servicios de emergencia

Durante el combate, AR proporciona información útil a los soldados acerca de su ubicación, cuantos enemigos lo rodean e inclusive se puede detectar a un enemigo que el soldado no vio.

Arte

AR puede ayudar a crear arte en tiempo real, como son pinturas, dibujo y modelado, también se puede utilizar como herramienta valiosa para la restauración, brindando una pre visualización de la obra completa.

Turismo y visitas turísticas

La realidad aumentada se usa mucho en aplicaciones turísticas, donde se resalta información de los sitios más importantes y puede brindar una

conexión con acontecimientos históricos. AR también puede proporcionar ubicación de museos y monumentos.

Tiendas y Negocios

Caja digital de Lego: Lego tuvo la idea de ubicar un quiosco dentro de su tienda donde los niños pueden llegar con la caja que están considerando comprar, mantenerla frente a una cámara, y seguidamente se le muestra en una pantalla la imagen del set completamente reconstruido.

- Zugara: La tienda Zugara utiliza su aplicación espejo mágico para permitir al comprador ponerse delante de una webcam y probar diferentes estilos de ropa. Inclusive tienen opciones para interactuar con el usuario y sus redes sociales.
- FoodTracer: Este proyecto utiliza el reconocimiento de imágenes en realidad aumentada para ofrecer a los compradores de comestibles más información sobre los alimentos que están escogiendo.

2.4.4 Servicios Realidad Aumentada

Existen aplicaciones de realidad aumentada que ganaron mucha popularidad desde su introducción, y que extendieron su alcance hasta proveer de servicios gratuitos y SDK para el desarrollo en plataformas móviles. De esta forma se puede encontrar 3 servicios principales de realidad aumentada basada en localización. Wikitude, Layar y Junaio.

Wikitude

Wikitude fue introducida como una aplicación de realidad aumentada en el año 2008, siendo desarrollada por la compañía Austriaca Mobilizy que luego tomó el nombre de Wikitude. Fue una de las aplicaciones pioneras en dispositivos móviles que usaba la localización y mostraba información acerca de los alrededores de los usuarios.

El contenido que muestra la aplicación de Wikitude es generada por los propios usuarios. Los cuales pueden añadir información a través de una interface web ya sea en formato KML ³⁷o ARML ³⁸.

Es posible crear servicios web definidos por el usuario para integrar información existente con Wikitude, sin embargo esto requiere que el servicio se registre en su página y también debe acogerse a estándares de traslado de datos que tiene el servicio de Wikitude.

Layar

Layar es otra compañía que ofrece servicios de búsqueda mediante realidad aumentada usando primariamente la localización. Tienen de igual forma un aplicativo móvil multiplataforma y proveen kits de desarrollo para integrar Layar con aplicaciones personalizadas.

³⁷ <http://es.wikipedia.org/wiki/KML>

³⁸ <http://openarm1.org/wikitude4.html>

El contenido de esta aplicación es generado por lo que ellos llaman “layers” que no son más que servicios web en formato REST enviando información geo localizada de los puntos de interés próximos al usuario. Estos servicios son desarrollados por compañías ajenas a Layar usando una API pública. Sin embargo Layar se hace responsable de la validación de estos servicios y su proceso de publicación.

Junaio

Junaio es un servicio muy similar a los anteriores, sin embargo a diferencia de ellos trata de que su información sea más eficaz a través del uso de marcadores LLA³⁹, simplemente agregan información de altitud que servirá para una mejor visualización cuando se use la realidad aumentada.



Gráfica 2.16: Navegadores de AR basados en geo localización⁴⁰

A pesar de que existen estos servicios libres para incorporar AR en aplicaciones, el problema principal es que los desarrolladores se atan a su

³⁹ <http://www.junaio.com/develop/docs/documentation/general/location-based/>

⁴⁰ Fuente: <http://en.wikipedia.org/wiki/Wikitudo>

servicio y siempre tendrán que pasar por los servidores web de estas compañías. Tampoco se puede cambiar mucho el formato en el cual se muestran los puntos de interés ni incorporar capas completamente personalizadas a los mismos.

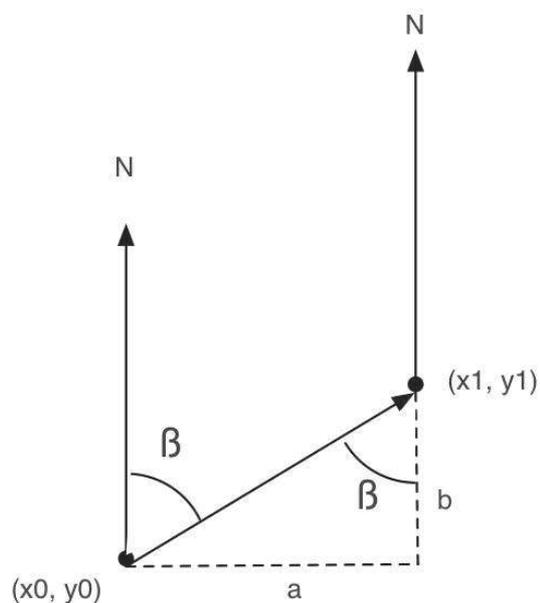
Son servicios ampliamente recomendados si las necesidades del usuario no son muy elevadas, como por ejemplo mostrar marcadores simples en AR. Sin embargo si se quiere un mayor nivel de personalización se tendrá que usar alternativas como motores de AR especializados.

2.4.5 Teoría de AR basada en geo localización

Se podría discutir mucho si basarse en geo localización para agregar elementos adicionales a los usuarios de las aplicaciones es efectivamente realidad aumentada, después de todo no se reconoce ningún tipo de patrón y no se agrega nada nuevo a lo que ya se sabe de antemano. Muchas personas simplemente lo consideran otra forma de presentar la realidad, aunque como se mencionó esto no es una opinión generalizada.

De todos modos, cuando se basa en geo localización, el principio es un simple ejercicio matemático en el cual se forma un plano y se reconoce sus puntos con la ayuda de los sensores del dispositivo móvil. Por supuesto que el modelo que se expone a continuación puede complicarse mucho conforme se aumenten elementos como la altitud y orientación.

Las matemáticas detrás de este tipo de aplicaciones se muestran en el siguiente gráfico:



$$\beta = \arctan(a, b)$$

Gráfica 2.17: Principio Matemático de la AR geo localizada⁴¹

Donde el punto (x_0, y_0) representa la posición del dispositivo móvil en la tierra (latitud, longitud), mientras que el punto (x_1, y_1) representa la posición del objeto que se desea mostrar en la pantalla a través de la cámara.

Por lo general, se llama a este objeto como un POI ⁴²(Punto de Interés). Se pueden representar muchos POI dentro la pantalla siendo el mismo principio.

⁴¹ Fuente: <http://www.invasivecode.com/blog/archives/1435>

Dado que por lo general la distancia entre el POI y la ubicación del dispositivo es mucho menor que la distancia comprendida entre el Polo Norte con el POI y el dispositivo, se puede asumir existen dos líneas paralelas que conectan dichos puntos con el Polo Norte.

Cuando dos líneas paralelas son intersecadas por una tercera recta, los dos ángulos β mostrados en la gráfica 2.17 son exactamente iguales. El ángulo β representa la desviación desde el Polo Norte de la vista del dispositivo hacia el punto de interés.

Si se conoce de antemano la ubicación GPS del dispositivo y la ubicación del punto de interés, se puede calcular el ángulo β . Si se utiliza fórmulas trigonométricas se sabe que para un triángulo rectángulo $\beta = \arctan(a, b)$.

Pero por otro lado, β también es el ángulo proporcionado por un magnetómetro cuando el punto de interés está exactamente en el centro de la pantalla del dispositivo, (cuando se mira a través de la cámara).

⁴² http://www.pois-gps.com.ar/?page_id=2552/tecnologia-gps/%C2%BFque-es-un-poi-point-of-interest-o-pdi-punto-de-interes/

Por lo tanto, la ecuación matemática ya está resuelta. Cuando el ángulo β calculado coincide con el ángulo proporcionado por el magnetómetro, se sabe que el objeto está en el centro de la pantalla.

Ahora suponiendo que se dispone de un iPhone, el mismo que está en orientación vertical frente al usuario, la coordenada x del punto de interés en pantalla tendrá que ser $x = 320/2 = 160\text{px}$. Para simplificar el modelo, se supone que la coordenada y es siempre fija. Por lo tanto, $y = 480/2 = 240\text{px}$. Usando un poco de matemáticas adicionales se podría usar la altitud del dispositivo para mover el punto de interés arriba o abajo en la pantalla.

Obviamente, para lograr que todo sea más realista, se necesita mover el POI de derecha a izquierda en la pantalla con una desviación proporcional a la diferencia entre el ángulo β y el ángulo proporcionado por el magnetómetro.

Por otro lado, suponiendo que la cámara del iPhone que se toma como ejemplo puede ver un campo de visión de 60° (apertura), entonces la coordenada en x del punto de interés será $x = 320/2 + \partial * 320 / (60/2)$, donde ∂ es la diferencia el ángulo β encontrado y la orientación del magnetómetro mientras se desplaza el dispositivo.

2.4.5.1 Operaciones matemáticas con coordenadas

Cuando se habla de coordenadas geográficas se parte de una fórmula fundamental que ayudará a deducir y resolver problemas, como la distancia

entre 2 coordenadas, encontrar el ángulo de desviación entre una ubicación de origen y el destino, o encontrar una coordenada partiendo de un origen y una distancia establecida. Esta es la fórmula Haversine como se conoce en el idioma inglés.

Haversine ⁴³ es una ecuación de suma importancia para la navegación, ya que ayuda a encontrar la distancia de gran-círculo ⁴⁴ entre 2 puntos de una esfera dados su latitud y longitud. Haversine está definida como:

$$\text{havrsin}(\theta) = \sin(\theta/2)^2 = \frac{1 - \cos(\theta)}{2}$$

Partiendo de esta fórmula base y reemplazando datos, se puede llegar a la fórmula de distancia entre 2 coordenadas:

$$\begin{aligned} d &= 2r \arcsin \left(\sqrt{\text{havrsin}(\phi_2 - \phi_1) + \cos(\phi_1) \cos(\phi_2) \text{havrsin}(\psi_2 - \psi_1)} \right) \\ &= 2r \arcsin \left(\sqrt{\sin^2 \left(\frac{\phi_2 - \phi_1}{2} \right) + \cos(\phi_1) \cos(\phi_2) \sin^2 \left(\frac{\psi_2 - \psi_1}{2} \right)} \right) \end{aligned}$$

Como se mencionó partiendo de esta fórmula se puede resolver una cantidad de problemas comunes cuando se trata con localizaciones, por lo cual es aplicable a muchísimas ramas, siendo la realidad aumentada una de ellas.

⁴³ http://es.wikipedia.org/wiki/F%C3%B3rmula_del_Haversine

⁴⁴ http://es.wikipedia.org/wiki/Gran_c%C3%ADrculo

2.5 Desarrollo Móvil

2.5.1 Introducción

El desarrollo de aplicaciones móviles es un área que simplemente ha ido en crecimiento los últimos años, es indudable que con la aparición del iPhone y seguidamente el App Store⁴⁵, el desarrollo móvil se convirtió en un negocio muchísimo más atractivo para programadores alrededor del mundo. Por otro lado el boom que existe en cuanto a Smartphone y Tablets hace pensar que precisamente en este campo se encuentra gran parte del desarrollo de software.

Existen varias formas de desarrollo móvil, desde aplicaciones nativas a versiones amigables con entornos móviles de páginas web enteras.

En el caso de cada plataforma de desarrollo móvil, tienen integrado un IDE⁴⁶ que provee al programador de herramientas para escribir, probar e instalar aplicaciones dentro de los dispositivos físicos sobre los cuales corre dicha plataforma.

El entorno web ha querido apoderarse del desarrollo de aplicaciones móviles, sin embargo no ha logrado ni siquiera igualar al rendimiento que se obtiene

⁴⁵ http://en.wikipedia.org/wiki/App_Store

⁴⁶ http://en.wikipedia.org/wiki/Integrated_development_environment

por parte de una aplicación nativa. Aunque por otro lado es la única forma de desarrollo en la cual no se tiene que preocupar si es que se instala en una plataforma u otra.

Actualmente existen dos plataformas que son los líderes indiscutibles dentro del desarrollo móvil, Apple iOS y Google Android.

2.5.2 iOS

2.5.2.1 Requerimientos

Para poder desarrollar sobre plataforma iOS se debe utilizar el SDK⁴⁷ que provee Apple de manera gratuita, entre sus requerimientos se tiene que el SDK únicamente corre sobre Mac OS X, por lo que se necesita tener al menos un computador Mac con un sistema operativo Mac OS X Lion (10.7) instalado y actualizado a su última versión. Los requisitos de hardware implícitos para este sistema operativo son los mismos para el SDK y una memoria de 4GB como mínimo.

2.5.2.2 Herramientas

Apple distribuye el iOS SDK para poder desarrollar aplicaciones iOS. En él se provee a los desarrolladores con todas las librerías y dependencias necesarias para poder crear código que pueda correr en un sistema iOS.

⁴⁷ http://en.wikipedia.org/wiki/Software_development_kit

iOS SDK incluye un conjunto de herramientas y utilidades que se usan para desarrollar aplicaciones en iPhone, iPad y aplicaciones de escritorio en Mac OS X. Entre estas herramientas se tiene:

- Xcode 4: Xcode es el conjunto completo de herramientas para el desarrollo de aplicaciones iOS y OS X, en esta versión las herramientas se rediseñaron para ser más rápidas y más fáciles de usar.
- iOS Simulator: El simulador de iOS ejecuta una aplicación en la medida de lo posible como si estuviera dentro de un verdadero dispositivo iOS. Gracias a que es muy rápido para iniciar y depurar las aplicaciones, se convierte en el instrumento de pruebas perfecto para que el desarrollador se asegure que su interfaz funciona en la manera que desea, como por ejemplo acceso a internet y cambios de orientación cuando el teléfono rota. Inclusive se puede simular gestos táctiles con el ratón.
- Interface Builder: Permite crear interfaces gráficas para las aplicaciones iOS sin necesidad de escribir una sola línea de código. A partir de la versión 4 de Xcode, Interface Builder fue integrado completamente dentro del IDE, ya que antes se distribuía como una aplicación individual.

- Compilador Apple LLVM: Compatibilidad con aplicaciones C/C++, Objective-C y nuevas funcionalidades como ARC⁴⁸.
- Debugger⁴⁹ gráfico: Al momento de hacer debug, ya sea en un el simulador o en el dispositivo físico conectado a través de USB, se pueden ver los valores que las variables tienen durante la ejecución directamente en pantalla.
- Análisis Estático: Ayuda a encontrar bugs⁵⁰ en el código antes de que este sea compilado, dando reportes de bugs potenciales que sería prácticamente imposible de replicar.
- Instrumentos para XCode: Conjunto de instrumentos que permiten llevar estadísticas de consumo de memoria, performance gráfico y de procesamiento y herramientas para la optimización de aplicaciones.

El iOS SDK se puede descargar de forma gratuita del sitio de Apple, pero para instalar aplicaciones en un dispositivo real se requiere un registro previo como "Apple Developer". El cual tiene un costo de \$99 dólares al año en el caso de personas individuales o \$299 al año en el caso de empresas.

⁴⁸ <http://clang.llvm.org/docs/AutomaticReferenceCounting.html>

⁴⁹ <http://en.wikipedia.org/wiki/Debugging>

⁵⁰ http://en.wikipedia.org/wiki/Software_bug

Tipo Licencia	Audiencia	Costo	Descripción
iOS Developer Program	Personas Individuales	\$99 al año	Para programadores individuales que crearán aplicaciones y las distribuirán en el App Store
iOS Developer Program	Compañías	\$99 al año	Para compañías con un equipo de desarrollo que crearán aplicaciones iOS para distribuir las en el App Store
iOS Developer Enterprise Program	Compañías	\$299 al año	Para compañías que estarán creando aplicaciones iOS para distribución interna
iOS Developer University Program	Universidades	Gratuito	Para instituciones de educación superior que busquen introducir desarrollo iOS dentro de su currículum

Tabla 2.3: Licencias Desarrollo iOS

2.5.3 Android

2.5.3.1 Requerimientos

Google provee del Android SDK para poder codificar aplicaciones que corran sobre el sistema operativo Android, los requerimientos necesarios son los siguientes:

- Windows XP (32-bit), Vista (32- or 64-bit), o Windows 7 (32- or 64-bit)
- Mac OS X 10.5.8 o superior (x86)
- Linux (testado en Ubuntu Linux, Lucid Lynx)
 - Librería GNU C (glibc) 2.7 o superior
 - Ubuntu Linux, versión 8.04 o superior
 - Si se instala en una distribución de 64-bit, esta debe ser capaz de correr aplicaciones en 32-bit
 - Eclipse 3.6.2 (Helios) o superior
 - Eclipse JDT plugin
 - JDK 6 (JRE por sí solo no es suficiente)
 - Android Development Tools plugin

2.5.3.2 Herramientas

Como herramienta principal en Android se tiene el plugin Android Developer Tools (ADT) que ofrece un entorno de desarrollo a nivel profesional para la creación de aplicaciones Android. Se trata de un IDE completo para Java con características avanzadas con las cuales el desarrollador puede construir, probar, depurar, e instalar aplicaciones en un dispositivo Android.

Entre las herramientas que incluye el plugin ADT se tiene:

- **Completo IDE para Java:** Refactorización, ayuda con errores en el código, navegación integrada entre Java y recursos XML Android. Ofrece editores especializados de XML para Android y un conjunto de herramientas de análisis estático, con lo cual los desarrolladores pueden detectar problemas de rendimiento.
- **Diseñadores de Interfaz Gráfica:** El plugin ADT incluye editores especializados con los cuales se puede crear interfaces de usuario simplemente arrastrando y soltando componentes. También se puede visualizar nuestro diseño sobre tablets, teléfonos y otros dispositivos.
- **Instalación en dispositivos físicos:** ADT ayuda a instalar aplicaciones en cualquier dispositivo comercial que corra el sistema operativo Android, de igual forma conecta el debugger con el dispositivo real para que se pueda probar y depurar la aplicación en tiempo real.

- Simulador de dispositivos: Se pueden crear simuladores personalizados para probar diferentes tamaños de pantalla, teclados y otros componentes de hardware. Además de esto utiliza el hardware del computador para simular hardware avanzado, como cámara, sensores o inclusive telefonía.

CAPÍTULO 3: DISEÑO

3.1 Historias de Usuario

3.1.1 Introducción

A continuación se listan las historias de usuario que fueron recopiladas a lo largo del desarrollo del sistema, las cuales definen los requerimientos que solicitó el auspiciante del presente trabajo de tesis, Ing. Geovanny Raura, director del Centro de Innovación y Desarrollo Empresarial CIDE conforme evolucionaba el proyecto.

Historia de usuario	
Número: 1	Usuario: Ing. Geovanny Raura
Nombre: Administración Visitas Virtuales	
Prioridad en el negocio (Alta/Media/Baja):	Alta
Modificación Número:	1
Descripción: Como administrador, quiero poder gestionar las visitas virtuales, crear nuevas, editar y eliminar las existentes, para evitar la configuración de las mismas de forma manual.	

ESPACIO EN BLANCO INTENCIONAL

Historia de usuario	
Número: 2	Usuario: Ing. Geovanny Raura
Nombre: Facilidad manejo de contenido	
Prioridad en el negocio (Alta/Media/Baja):	Media
Modificación número	1
Descripción: Como administrador, se desea que pueda subir nuevas fotos panorámicas y acomodarlas dentro de las visitas virtuales, para que se pueda tener una facilidad de manejo al estructurar las visitas virtuales.	

Historia de usuario	
Número: 3	Usuario: Ing. Geovanny Raura
Nombre: Geo localización	
Prioridad en el negocio (Alta/Media/Baja):	Alta
Modificación de historia de usuario numero	1
Descripción: Se busca que la aplicación tenga una funcionalidad tal que exista una relación directa entre panoramas y puntos dentro del mapa geográfico.	

ESPACIO EN BLANCO INTENCIONAL

Historia de usuario	
Número: 4	Usuario: Ing. Geovanny Raura
Nombre: Mapas Interactivos	
Prioridad en el negocio (Alta/Media/Baja):	Media
Modificación de historia de usuario numero	1
<p>Descripción:</p> <p>Como administrador, quiero poder hacer clic dentro de un mapa y que se agregue un marcador, el cual representará al panorama ubicado en la localización deseada, esto facilitará la relación de puntos geográficos (latitud, longitud) con imágenes panorámicas.</p>	

Historia de usuario	
Número: 5	Usuario: Ing. Geovanny Raura
Nombre: Multiplataforma	
Prioridad en el negocio (Alta/Media/Baja):	Media
Modificación de historia de usuario numero	1
<p>Descripción:</p> <p>Como usuario se quiere ver las visitas virtuales ya sea desde un browser o un dispositivo móvil (Android, iPhone), de tal forma que se pueda entregar una experiencia de usuario muy buena y la aplicación quede lista para responder a avances tecnológicos en el campo de movilidad.</p>	

ESPACIO EN BLANCO INTENCIONAL

Historia de usuario	
Número: 6	Usuario: Ing. Geovanny Raura
Nombre: Tour Virtual	
Prioridad en el negocio (Alta/Media/Baja):	Alta
Modificación de historia de usuario numero	1
Descripción: Como usuario quiero poder navegar a través de los panoramas de la visita virtual, observando información dentro de cada uno de ellos, esto para que la aplicación transmita de manera efectiva información al público en general.	

Historia de usuario	
Número: 7	Usuario: Ing. Geovanny Raura
Nombre: Integración con CMS	
Prioridad en el negocio (Alta/Media/Baja):	Alta
Modificación de historia de usuario numero	1
Descripción: Como administrador quiero que el módulo web desarrollado se integre con el sistema CMS Drupal, para poder usar dentro de la página web del CIDE de la ESPE.	

ESPACIO EN BLANCO INTENCIONAL

Historia de usuario	
Número: 8	Usuario: Ing. Geovanny Raura
Nombre: Uso de XML para Visitas Virtuales	
Prioridad en el negocio (Alta/Media/Baja):	Alta
Modificación de historia de usuario numero	1
<p>Descripción:</p> <p>Como administrador, quiero que las visitas virtuales además creen en archivos XML para que puedan ser visualizados en el visor Krpano, así lograr que sea más flexible el sistema y configurable por el administrador.</p>	

Historia de usuario	
Número: 9	Usuario: Ing. Geovanny Raura
Nombre: XML accesibles en el CMS	
Prioridad en el negocio (Alta/Media/Baja):	Baja
Modificación de historia de usuario numero	1
<p>Descripción:</p> <p>Como administrador, quiero que los archivos XML sean accesibles dentro de la estructura de archivos que tenga el CMS Drupal, esto para poder analizarlos en caso sea necesario.</p>	

ESPACIO EN BLANCO INTENCIONAL

Historia de usuario	
Número: 10	Usuario: Ing. Geovanny Raura
Nombre: Distribución de las opciones	
Prioridad en el negocio (Alta/Media/Baja):	Baja
Modificación de historia de usuario numero	1
Descripción: Como administrador, quiero que cada opción disponible del sistema tenga su pestaña individual para que sea más fácil su administración.	

Historia de usuario	
Número: 11	Usuario: Ing. Geovanny Raura
Nombre: Movilidad Marcadores	
Prioridad en el negocio (Alta/Media/Baja):	Media
Modificación de historia de usuario numero	1
Descripción: Como administrador se quiere que los panoramas se puedan reorganizar con funcionalidad de arrastrar y soltar para que sea más fácil y cómoda su reubicación dentro de los mapas.	

ESPACIO EN BLANCO INTENCIONAL

Historia de usuario	
Número: 12	Usuario: Ing. Geovanny Raura
Nombre: Tiempo de Carga Visitas	
Prioridad en el negocio (Alta/Media/Baja):	Media
Modificación de historia de usuario numero	1
Descripción: Como usuario se quiere que el tiempo que le tome cargar los panoramas no sea muy grande, en todo caso al menos que se simule una carga previa con imágenes de menor calidad, esto para mejor la experiencia de uso del sistema.	

Historia de usuario	
Número: 13	Usuario: Ing. Geovanny Raura
Nombre: Agregar fotos panorámicas	
Prioridad en el negocio (Alta/Media/Baja):	Alta
Modificación de historia de usuario numero	1
Descripción: Como administrador, debo poder subir fotos en formato original y el sistema debe procesarlos para dejarlos en un formato adecuado para el visor.	

ESPACIO EN BLANCO INTENCIONAL

Historia de usuario	
Número: 14	Usuario: Ing. Geovanny Raura
Nombre: Pre visualización de configuración visitas virtuales	
Prioridad en el negocio (Alta/Media/Baja):	Media
Modificación de historia de usuario numero	1
<p>Descripción:</p> <p>Como administrador, se debe poder ver el resultado preliminar de lo que se está configurando en cada una de las opciones, esto para tener una percepción previa de lo que verá el usuario al terminar la visita virtual.</p>	

Historia de usuario	
Número: 15	Usuario: Ing. Geovanny Raura
Nombre: Creación de Hotspots Dinámica	
Prioridad en el negocio (Alta/Media/Baja):	Alta
Modificación de historia de usuario numero	1
<p>Descripción:</p> <p>Como administrador, debe existir la capacidad de hacer clic sobre cualquier sección dentro del visor de la visita virtual, lo cual agregará un nuevo hotspot en el que podremos agregar información adicional.</p>	

ESPACIO EN BLANCO INTENCIONAL

Historia de usuario	
Número: 16	Usuario: Ing. Geovanny Raura
Nombre: Geo localización de hotspots	
Prioridad en el negocio (Alta/Media/Baja):	Alta
Modificación de historia de usuario numero	1
Descripción: Cuando se agregue un nuevo hotspot a las visitas, este debe quedar de igual forma geo localizado, para que pueda ser visualizado por los posibles clientes externos del sistema.	

Historia de usuario	
Número: 17	Usuario: Ing. Geovanny Raura
Nombre: Interacción Mapas Visor	
Prioridad en el negocio (Alta/Media/Baja):	Media
Modificación de historia de usuario numero	1
Descripción: Como administrador, quiero poder hacer clic sobre un marcador ubicado en el mapa y el sistema mostrará dentro del visor de visitas virtuales la foto panorámica relacionada al marcador seleccionado.	

ESPACIO EN BLANCO INTENCIONAL

Historia de usuario	
Número: 18	Usuario: Ing. Geovanny Raura
Nombre: Opciones en Página	
Prioridad en el negocio (Alta/Media/Baja):	Baja
Modificación de historia de usuario numero	1
Descripción: Cada opción debe quedar representada por una página dentro del CMS, esto para que exista la posibilidad de agregar links a las opciones y sea más fácil la configuración del módulo dentro del CMS.	

Historia de usuario	
Número: 19	Usuario: Ing. Geovanny Raura
Nombre: Crear thumbnails para panoramas	
Prioridad en el negocio (Alta/Media/Baja):	Media
Modificación de historia de usuario numero	1
Descripción: Como administrador quiero poder recortar parte de la foto panorámica subida para utilizarla como vista previa de la visita virtual, para que se pueda escoger que foto usar en otras partes del sistema.	

ESPACIO EN BLANCO INTENCIONAL

Historia de usuario	
Número: 20	Usuario: Ing. Geovanny Raura
Nombre: Selección thumbnail para visita virtual	
Prioridad en el negocio (Alta/Media/Baja):	Media
Modificación de historia de usuario numero	1
<p>Descripción:</p> <p>Como administrador, quiero exista una opción donde pueda decidir si el thumbnail del panorama se usará como vista previa de la visita virtual.</p>	

Historia de usuario	
Número: 21	Usuario: Ing. Geovanny Raura
Nombre: Drag and Drop Hotspots	
Prioridad en el negocio (Alta/Media/Baja):	Media
Modificación de historia de usuario numero	1
<p>Descripción:</p> <p>Como administrador quiero poder agregar los hotspot haciendo drag and drop dentro del visor de visitas virtuales, esto es para evitar errores al momento de hacer clic dentro de la visita virtual.</p>	

ESPACIO EN BLANCO INTENCIONAL

Historia de usuario	
Número: 22	Usuario: Ing. Geovanny Raura
Nombre: Imagen opcional para los hotspots	
Prioridad en el negocio (Alta/Media/Baja):	Media
Modificación de historia de usuario numero	1
Descripción: Como administrador, quiero agregar una imagen opcional a la información descriptiva de los hotspot, esto para dar más funcionalidad a la información expuesta por el mismo.	

Historia de usuario	
Número: 23	Usuario: Ing. Geovanny Raura
Nombre: Galería de Visitas Virtuales	
Prioridad en el negocio (Alta/Media/Baja):	Alta
Modificación de historia de usuario numero	1
Descripción: Como usuario, debo poder ver todas las Visitas Virtuales que se crearon en el sistema en forma de galería, esto servirá para que como usuario pueda ubicarme mejor dentro del sistema.	

ESPACIO EN BLANCO INTENCIONAL

Historia de usuario	
Número: 24	Usuario: Ing. Geovanny Raura
Nombre: Página por Visitas Virtual	
Prioridad en el negocio (Alta/Media/Baja):	Alta
Modificación de historia de usuario numero	1
Descripción:	
<p>Como administrador quiero que cada visita virtual tenga su página correspondiente, con esto se facilitará mostrar las visitas virtuales desde menús, hipervínculos entre otros.</p>	

Historia de usuario	
Número: 25	Usuario: Ing. Geovanny Raura
Nombre: Selección de marcador en los mapas	
Prioridad en el negocio (Alta/Media/Baja):	Baja
Modificación de historia de usuario numero	1
Descripción:	
<p>Como administrador quiero que cuando seleccione un marcador dentro del mapa, este cambie de color o icono facilitando el manejo del sistema al administrador.</p>	

ESPACIO EN BLANCO INTENCIONAL

Historia de usuario	
Número: 26	Usuario: Ing. Geovanny Raura
Nombre: Campo de visión del dispositivo móvil	
Prioridad en el negocio (Alta/Media/Baja):	Media
Modificación de historia de usuario numero	1
Descripción: Como usuario del aplicativo móvil quiero escoger hasta que distancia visualizaré los hotspots que estén cerca de mi ubicación, esto para que sea más fácil ver las localizaciones cercanas.	

Historia de usuario	
Número: 27	Usuario: Ing. Geovanny Raura
Nombre: Visualizar hotspots a través de la cámara	
Prioridad en el negocio (Alta/Media/Baja):	Alta
Modificación de historia de usuario numero	1
Descripción: Como usuario del aplicativo móvil, se quieren ver los hotspots configurados en el sistema web, a través de la cámara del teléfono o tablet, para ubicarme de manera más fácil cuando este cerca de las visitas virtuales.	

ESPACIO EN BLANCO INTENCIONAL

Historia de usuario	
Número: 28	Usuario: Ing. Geovanny Raura
Nombre: Actualización automática en el dispositivo móvil	
Prioridad en el negocio (Alta/Media/Baja):	Alta
Modificación de historia de usuario numero	1
Descripción: Como usuario del aplicativo móvil, la información de las visitas virtuales debe ser actualizada constantemente pero sin que interfiera con la utilización de la aplicación, esto para mantener la experiencia de usuario intacta.	

Historia de usuario	
Número: 29	Usuario: Ing. Geovanny Raura
Nombre: Mapa dentro del dispositivo móvil	
Prioridad en el negocio (Alta/Media/Baja):	Media
Modificación de historia de usuario numero	1
Descripción: Como usuario del aplicativo móvil debo poder ver los hotspot cercanos dentro de un mapa, para que sea más fácil la ubicación del usuario al usar el aplicativo.	

ESPACIO EN BLANCO INTENCIONAL

Historia de usuario	
Número: 30	Usuario: Ing. Geovanny Raura
Nombre: Información adicional de hotspots	
Prioridad en el negocio (Alta/Media/Baja):	Media
Modificación de historia de usuario numero	1
<p>Descripción:</p> <p>Como usuario del aplicativo móvil quiero ver información adicional de los hotspots ya sea haga clic dentro de los datos de la cámara o en el mapa del aplicativo.</p>	

3.1.2 Definición Prioridades

Luego de analizar las historias de usuario y los requisitos involucrados en cada una de ellas se define el siguiente listado de prioridades.

Prioridad Alta
<ul style="list-style-type: none"> 1. Administración visitas virtuales 3. Geo localización 6. Tour virtual 7. Integración con CMS 8. Uso XML para visitas virtuales 13. Agregar fotos panorámicas 15. Creación de hotspots dinámica 16. Geo localización de hotspots 23. Galería de visitas virtuales 24. Página por visita virtual 27. Visualizar hotspots a través de la cámara 28. Actualización automática en el dispositivo móvil

Prioridad Media

2. Facilidad manejo de contenido
4. Mapas interactivos
5. Multiplataforma
11. Movilidad Marcadores
12. Tiempo de Carga Visitas
14. Pre visualización de configuración de visitas virtuales
17. Interacción Mapas Visor
19. Crear thumbnails para panoramas
20. Selección thumbnail para visita virtual
21. Drag and drop Hotspots
22. Imágen opcional para los hotspots
26. Campo de visión del dispositivo móvil
29. Mapa dentro del dispositivo móvil
30. Información adicional de hotspots

Prioridad Baja

9. XML accesibles en el CMS
10. Distribución de las opciones
18. Opciones en página
25. Selección de marcador en los mapas

3.1.3 Definición de Iteraciones

En el presente proyecto de investigación se definieron tres iteraciones principales, cada una de las cuales tiene asignado su conjunto de historias de usuario a ser implementadas. En la primera iteración se busca obtener el primer prototipo funcional, el cual permitirá medir si la funcionalidad del sistema está siendo desarrollada correctamente y de acuerdo a la visión de las personas interesadas en el proyecto.

En la segunda iteración se busca obtener un producto más maduro que implemente casi todos los requerimientos propuestos, y por último, en la

tercera iteración se harán los ajustes y cambios necesarios para obtener un sistema completamente funcional, además se finalizará el desarrollo del aplicativo móvil que complementa al sistema web.

Primera Iteración

1. Administración visitas virtuales
2. Facilidad manejo de contenido
3. Geo localización
4. Mapas interactivos
7. Integración con CMS
8. Uso XML para visitas virtuales
9. XML accesibles en el CMS
10. Distribución de las opciones
11. Movilidad marcadores
13. Agregar fotos panorámicas
17. Interacción mapas visor
25. Selección de marcador en los mapas

Segunda Iteración

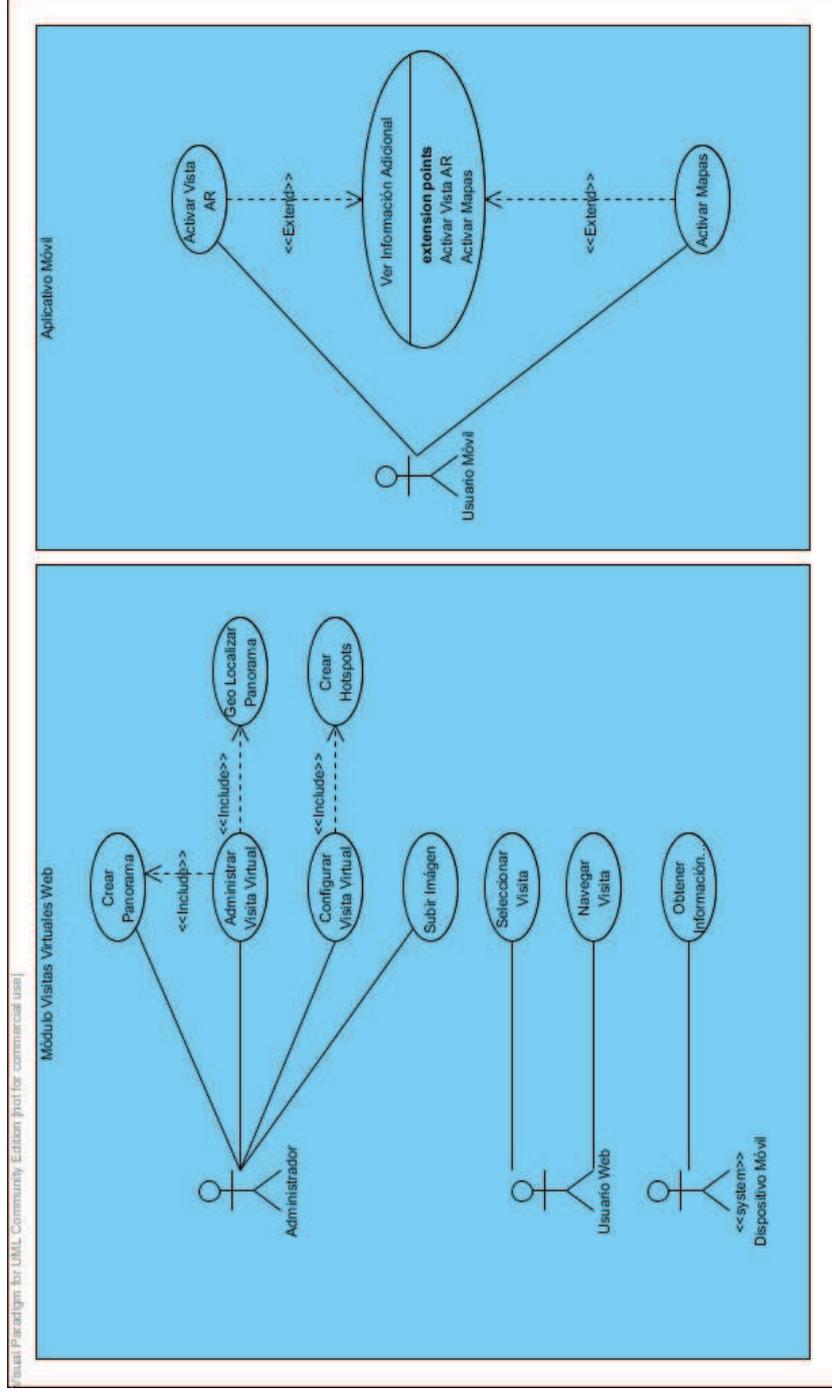
5. Multiplataforma
12. Tiempo de carga visitas
14. Pre visualización de configuración de visitas virtuales
15. Creación de hotspots dinámico
16. Geo localización de hotspots
18. Opciones en página
19. Crear thumbnails para panoramas
20. Selección thumbnail para visita virtual
21. Drag and drop hotspots
22. Imágen opcional para los hotspots
23. Galería de visitas virtuales
24. Página por visita virtual
28. Actualización automática en el dispositivo móvil
30. Información adicional de hotspots

Tercera Iteración

6. Tour virtual
27. Visualizar hotspots a través de la cámara
26. Campo de visión del dispositivo móvil
29. Mapa dentro del dispositivo móvil

3.2 Casos de Uso

3.2.1 Diagrama de casos de uso



3.2.2 Descripción de casos de uso

Nombre:	Crear Panorama	
Identificador:	VW-01	
Actor:	Administrador	
Descripción:	Permite subir una nueva foto panorámica	
Precondiciones:	Usuario debe tener acceso al tipo de contenido panorama	
Postcondiciones:	Imagen panorámica queda procesada en un formato más liviano y lista para usar	
Flujo Principal:	ACTOR	SISTEMA
	<ol style="list-style-type: none"> 1. Administrador decide crear un nuevo panorama dentro de los tipos de contenido. 2. Administrador ingresa un nombre para el panorama y los datos comunes a un tipo de contenido. 3. Administrador escoge el archivo de foto panorámica que desea subir a través de un dialogo de selección de archivo. 4. Administrador indica que quiere guardar el nuevo tipo de contenido Panorama. 	<ol style="list-style-type: none"> 5. El sistema verifica que el archivo no exceda los límites de tamaño permitidos. 6. El sistema muestra pre visualización de la foto que acaba de subir. 7. El sistema procesa la imagen convirtiéndola a formato cúbico. 8. El sistema informa que el nuevo tipo de contenido fue agregado a la base de datos.

<p>Flujo Alternativo 1: La imagen excede los límites permitidos por el sistema</p>	ACTOR	SISTEMA
		<p>1.4. El sistema determina que la foto seleccionada tiene un tamaño superior al permitido.</p> <p>1.5. El sistema informa al usuario que la imagen que seleccionó no es válida.</p>
<p>Flujo Alternativo 2: Usuario quiere recortar parte de la imagen</p>	ACTOR	SISTEMA
	<p>2.6. Administrador escoge una parte de la foto que quiere presentar como vista previa del panorama.</p>	<p>2.7. El sistema crea un nuevo archivo con la parte seleccionada por el usuario.</p> <p>2.8. El caso de uso continúa con el flujo normal en el paso 7.</p>

ESPACIO EN BLANCO INTENCIONAL

Nombre:	Geo Localizar Panorama	
Identificador:	VW-02	
Actor:	Administrador	
Descripción:	Permite asociar una imagen panorámica con un punto geográfico	
Precondiciones:	Usuario debe estar autenticado como administrador y tener acceso a módulo de creación de visitas virtuales	
Postcondiciones:	Se obtiene la posición geográfica del panorama y se relaciona con un objeto del sistema	
Flujo Principal:	ACTOR	SISTEMA
	<ol style="list-style-type: none"> 1. Administrador decide ubicar un nuevo panorama dentro del mapa de Google Maps. 2. Administrador hace clic en el punto donde desea ubicar el panorama. 	<ol style="list-style-type: none"> 3. El sistema crea un nuevo marcador y lo ubica dentro del mapa. 4. El sistema crea un nuevo panorama y muestra el formulario con los datos relacionados al marcador. 5. El sistema muestra el nuevo marcador seleccionado con distinto icono dentro del mapa.

Nombre:	Administrar Visita Virtual	
Identificador:	VW-03	
Actor:	Administrador	
Descripción:	Permite gestionar una visita virtual	
Precondiciones:	Usuario debe estar autenticado en el sistema como administrador y tener acceso al menú de creación de visitas virtuales	
Postcondiciones:	Se guarda la información de la visita virtual en la base de datos y se generan los archivos para configuración en el visor de visitas virtuales	
Flujo Principal:	ACTOR	SISTEMA
	<ol style="list-style-type: none"> 1. Administrador decide crear una nueva visita virtual. 2. Administrador selecciona la opción para crear nueva visita virtual. 6. Administrador llena formularios con datos requeridos por cada uno de los panoramas ingresados. 7. Administrador hace clic en el botón de guardar la visita virtual. 	<ol style="list-style-type: none"> 3. El sistema crea una nueva visita virtual en y muestra un formulario para llenar información del mismo. 4. Se ejecuta el caso de uso VW-02. 5. El sistema muestra una lista de las fotos panorámicas existentes en el sistema. 8. El sistema verifica que los datos ingresados sean válidos. 9. El sistema guarda la información de la visita virtual en la base de datos.

		<p>10. El sistema genera los archivos XML necesarios para pre visualizar la navegación de los panoramas.</p> <p>11. El sistema informa al usuario que la visita virtual se creó exitosamente.</p> <p>12. El sistema renderiza una sección para mostrar la visita virtual creada hasta el momento.</p>
	ACTOR	SISTEMA
<p>Flujo Alternativo 1: Se escoge una visita virtual existente</p>	<p>1.2. El usuario escoge una visita virtual de entre las visitas disponibles.</p>	<p>1.3. El sistema trae todos los datos relacionados con la visita virtual.</p> <p>1.4. El sistema muestra un formulario editable con la información obtenida.</p> <p>1.5. El caso de uso continúa con su flujo normal en el punto 4.</p>

	ACTOR	SISTEMA
Flujo Alternativo 2: No existen imágenes panorámicas en el sistema		<p>2.5. El sistema muestra una lista vacía de imágenes panorámicas.</p> <p>2.6. Se ejecuta el caso de uso VW-01.</p>
Flujo Alternativo 3: Datos no válidos		<p>3.8. El sistema encuentra que uno de los datos ingresados no es válido.</p> <p>3.9. Sistema muestra mensaje de error al usuario.</p> <p>3.10. El caso de uso continúa con su flujo normal en el punto 6.</p>

ESPACIO EN BLANCO INTENCIONAL

Nombre:	Crear Hotspots	
Identificador:	VW-04	
Actor:	Administrador	
Descripción:	Permite agregar un nuevo hotspot dentro de la visita virtual	
Precondiciones:	Usuario debe estar autenticado correctamente como administrador y tener acceso al módulo de configuración de visitas virtuales.	
Postcondiciones:	Los hotspots quedan relacionados a un panorama y geo localizados.	
Flujo Principal:	ACTOR	SISTEMA
	<ol style="list-style-type: none"> 1. Administrador selecciona un marcador en el mapa dentro de la opción de configuración de visitas virtuales. 4. El usuario arrastra un nuevo marcador y lo posiciona en el punto donde se desee crear el hotspot. 	<ol style="list-style-type: none"> 2. El sistema muestra la foto panorámica renderizada por el visor de visitas virtuales, que representa la ubicación seleccionada. 3. El sistema muestra un control para poder crear nuevos hotspots dentro del panorama. 5. El sistema calcula la posición geográfica que el hotspot ocupará basado en la distancia y grados de desplazamiento del visor. 6. El sistema crea el hotspot en y muestra un formulario para llenar información relacionada al mismo.

Nombre:	Subir Imágenes	
Identificador:	VW-05	
Actor:	Administrador	
Descripción:	Permite subir una imagen al servidor para usarla dentro de la configuración de visitas virtuales	
Precondiciones:	Usuario debe estar autenticado como administrador y tener acceso a módulo de configuración de visitas virtuales	
Postcondiciones:	Los hotspots quedan relacionados a un panorama y geo localizados	
Flujo Principal:	ACTOR	SISTEMA
	<ol style="list-style-type: none"> 1. Administrador desea subir una imagen para usarla como descripción del hotspot. 2. Administrador selecciona el archivo que desea subir. 	<ol style="list-style-type: none"> 3. El sistema verifica que el archivo no exceda los límites de tamaño. 4. El sistema guarda el archivo. 5. El sistema relaciona el archivo subido con el hotspots seleccionado y muestra la imagen subida.

ESPACIO EN BLANCO INTENCIONAL

Nombre:	Configurar Visita Virtual	
Identificador:	VW-06	
Actor:	Administrador	
Descripción:	Permite configurar una visita virtual	
Precondiciones:	Debe existir al menos una visita virtual creada en el sistema	
Postcondiciones:	Se guarda la configuración de la visita virtual en la base de datos y se genera el archivo XML final que será usado por el visor para mostrar la visita virtual a los usuarios	
Flujo Principal:	ACTOR	SISTEMA
	<ol style="list-style-type: none"> 1. Administrador selecciona la visita virtual que desea configurar. 4. El usuario ingresa los datos relacionados al hotspot creado. 5. El usuario informa que quiere guardar la configuración de la visita virtual. 	<ol style="list-style-type: none"> 2. El sistema trae los datos relacionados con la visita virtual escogida. 3. Se ejecuta el caso de uso VW-04. 6. El sistema verifica que los datos ingresados sean válidos. 7. El sistema guarda la información de configuración en la base de datos. 8. El sistema genera los archivos XML necesarios para que usuarios puedan ver la visita virtual finalizada. 9. El sistema informa al usuario que la visita virtual se configuró exitosamente.

	ACTOR	SISTEMA
Flujo Alternativo 1: Relacionar foto con el hotspot	<p>1.5. El usuario desea agregar una foto relacionada con el hotspot.</p> <p>1.6. Se ejecuta el caso de uso VW-05.</p>	<p>1.7. El sistema muestra un formulario editable con la información obtenida.</p> <p>1.8. El caso de uso continúa con su flujo normal en el punto 4.</p>
	ACTOR	SISTEMA
Flujo Alternativo 2: Datos no válidos		<p>2.8. El sistema encontró que uno de los datos ingresados no es válido.</p> <p>2.9. Sistema muestra mensaje de error al usuario.</p> <p>2.10. El caso de uso continúa con su flujo normal en el punto 6.</p>

ESPACIO EN BLANCO INTENCIONAL

Nombre:	Escoger Visita Virtual	
Identificador:	VW-07	
Actor:	Usuario Web	
Descripción:	Permite al usuario escoger una visita virtual de entre las existentes en el sistema	
Precondiciones:	Debe existir al menos una visita virtual creada en el sistema	
Postcondiciones:	Usuario debe tener acceso a la visualización de Visitas Virtuales	
Flujo Principal:	ACTOR	SISTEMA
	1. Usuario escoge la opción de galería de visitas virtuales. 5. Usuario escoge la visita virtual que desea ver.	2. El sistema verifica que existan visitas virtuales configuradas con anterioridad. 3. El sistema busca las imágenes thumbnail que fueron elegidas como pre visualización de la visita virtual. 4. El sistema muestra las visitas virtuales existentes en forma de galería. 6. El sistema re direcciona a la página correspondiente de la visita virtual.

	ACTOR	SISTEMA
Flujo Alternativo 1: No existe imagen thumbnail seleccionada		<p>1.3. El sistema determina que no existe foto thumbnail para usar en la galería de Visitas Virtuales.</p> <p>1.4. El sistema escoge el primer panorama que encuentra dentro de la visita y utiliza la imagen thumbnail que le corresponde.</p> <p>1.5. Se continúa con el flujo normal en el punto 4.</p>

ESPACIO EN BLANCO INTENCIONAL

Nombre:	Navegar Visita Virtual	
Identificador:	VW-08	
Actor:	Usuario Web	
Descripción:	Permite al usuario navegar dentro de las Visitas Virtuales	
Precondiciones:	Usuario debe tener acceso a la visualización de Visitas Virtuales.	
Flujo Principal:	ACTOR	SISTEMA
	<ol style="list-style-type: none"> 1. Usuario entra en la pantalla de visualización de visita virtual. 4. Usuario hace clic en hotspot de navegación entre panoramas. 	<ol style="list-style-type: none"> 2. El sistema muestra la visita virtual con sus respectivos panoramas. 3. El sistema muestra la ubicación geográfica del panorama y la descripción del mismo. 5. El sistema muestra la visualización del panorama seleccionado. 6. El sistema muestra información actualizada de la ubicación geográfica del panorama y su descripción.

ESPACIO EN BLANCO INTENCIONAL

Nombre:	Obtener Información Hotspots	
Identificador:	VW-09	
Actor:	Dispositivo Móvil	
Descripción:	Permite que un sistema externo pueda obtener datos de los hotspots de las visitas virtuales	
Flujo Principal:	ACTOR	SISTEMA
	<ol style="list-style-type: none"> 1. El caso de uso inicia cuando un sistema cliente hace una petición al servidor de información de los casos de uso. 	<ol style="list-style-type: none"> 2. El sistema procesa los datos de localización y rango de visión enviados por el cliente. 3. El sistema obtiene la información de los hotspots existentes en la base de datos. 4. El sistema calcula que la distancia entre la localización del dispositivo y el hotspot esté dentro del rango de visión. 5. El sistema devuelve la información de los hotspots codificada.

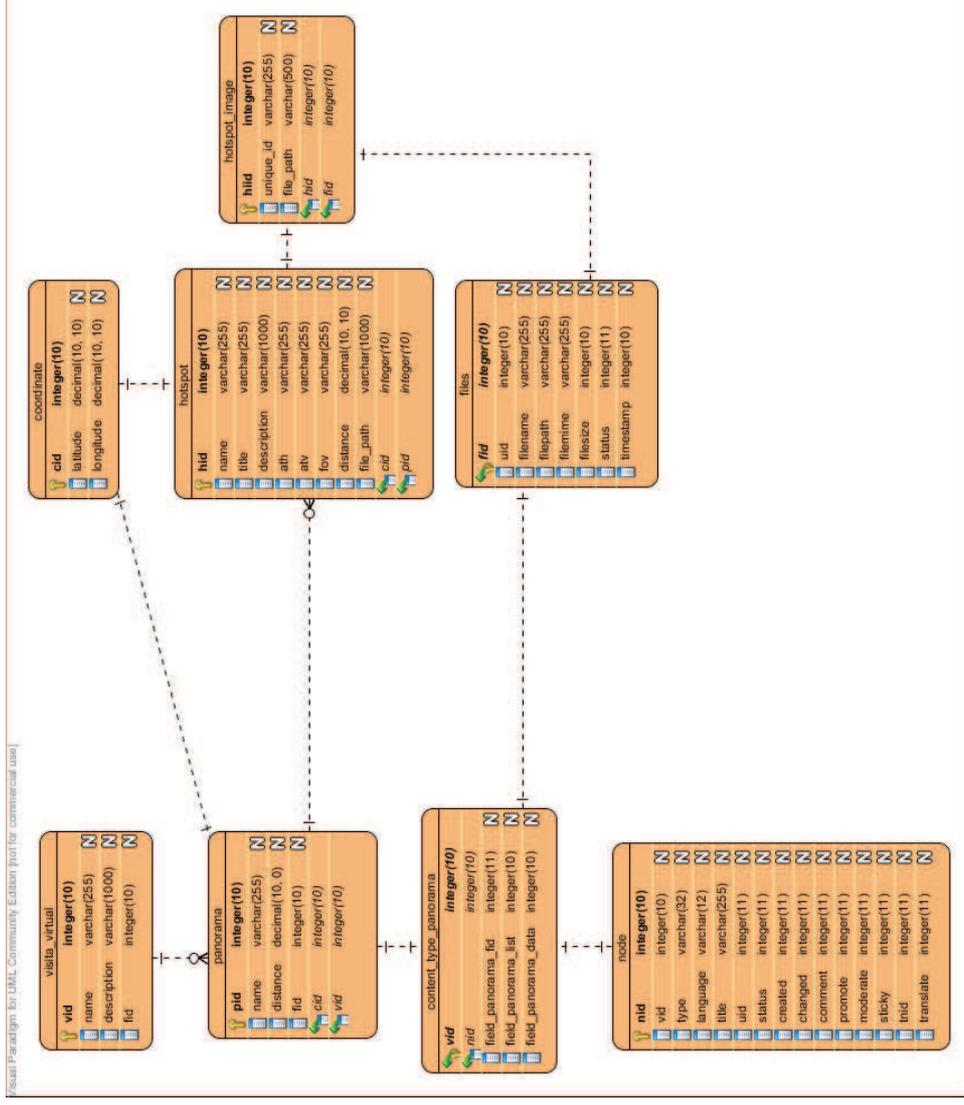
ESPACIO EN BLANCO INTENCIONAL

Nombre:	Activar Vista AR	
Identificador:	VM-01	
Actor:	Usuario Móvil	
Descripción:	Configura el dispositivo móvil para mostrar una captura de la cámara del dispositivo móvil	
Flujo Principal:	ACTOR	SISTEMA
	<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el usuario escoge el tab de vista en AR. 6. El usuario empieza desplazarse y rotar con el dispositivo móvil 	<ol style="list-style-type: none"> 2. El sistema accede a la información de la cámara del dispositivo móvil. 3. El sistema muestra la captura de la cámara en la pantalla del dispositivo. 4. Se ejecuta el caso de uso VW-09. 5. El sistema procesa los datos de hotspot recibidos. 7. El sistema calcula la información de localización del usuario para mostrar hotspots cercanos. 8. El sistema muestra una capa de visualización de información básica del hotspot sobre la captura de la cámara.

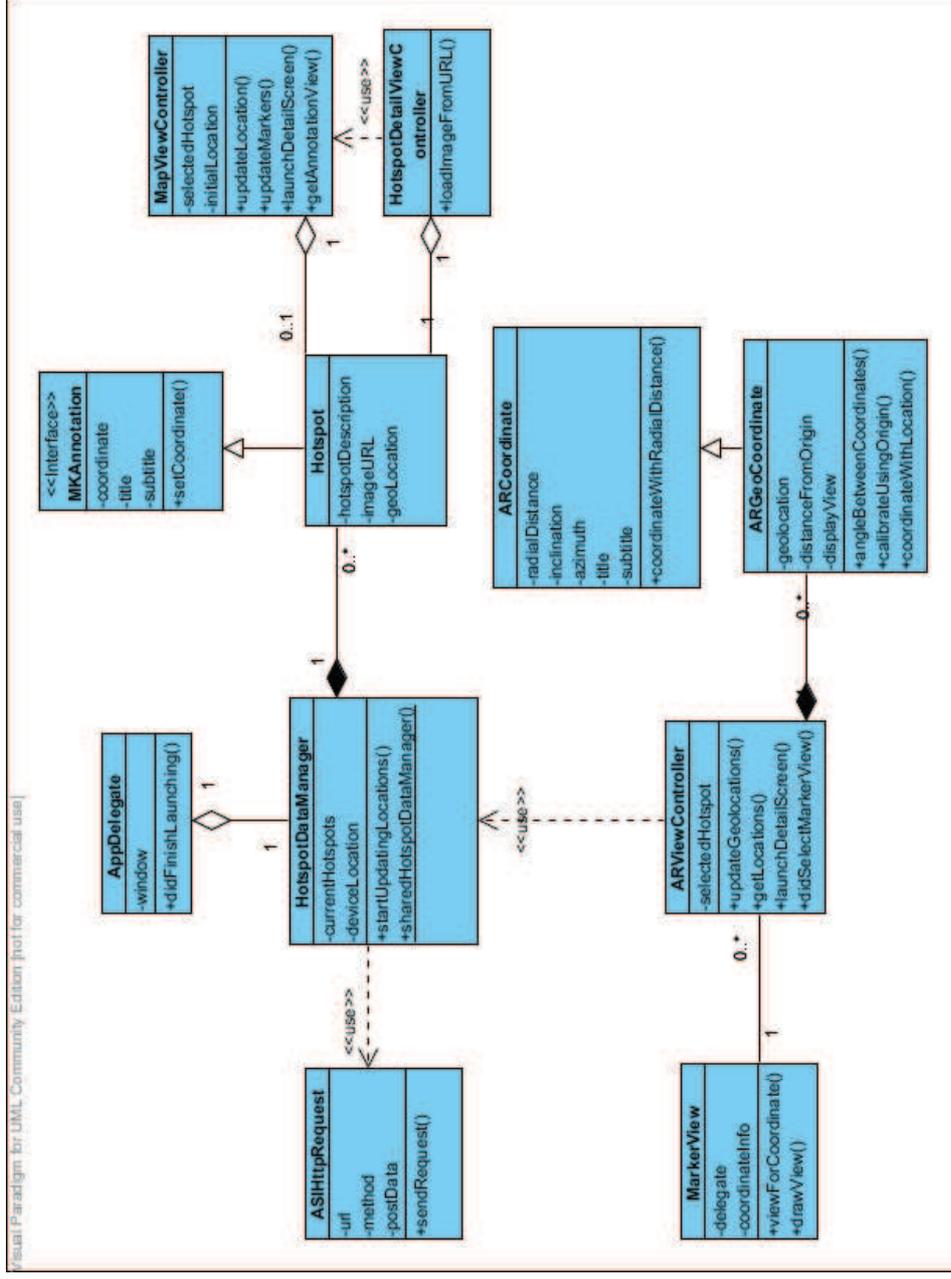
Nombre:	Activar Vista Mapas	
Identificador:	VM-02	
Actor:	Usuario Móvil	
Descripción:	Configura el dispositivo móvil para mostrar un mapa con la información de hotspots disponible	
Flujo Principal:	ACTOR	SISTEMA
	<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el usuario escoge el tab de vista en mapas. 	<ol style="list-style-type: none"> 2. El sistema muestra un mapa en la pantalla del dispositivo. 3. El sistema muestra la captura de la cámara en la pantalla del dispositivo. 4. El sistema accede a información de localización del dispositivo y muestra un marcador en la ubicación actual del usuario. 5. Se ejecuta el caso de uso VW-09. 6. El sistema procesa los datos de los hotspots recibidos y ubica marcadores en cada localización geográfica de los hotspots.

Nombre:	Ver información adicional	
Identificador:	VM-03	
Actor:	Usuario Móvil	
Descripción:	Muestra información adicional del hotspot seleccionado.	
Flujo Principal:	ACTOR	SISTEMA
	<ol style="list-style-type: none"> 1. Usuario hace clic en un marcador del mapa o en la capa añadida a la cámara, en los casos de uso VM-01 o VM-02. 	<ol style="list-style-type: none"> 2. El sistema muestra una pantalla con el título del hotspot seleccionado. 3. El sistema descarga los datos de la imagen descriptiva del hotspot. 4. El sistema muestra la imagen y descripción del hotspot.

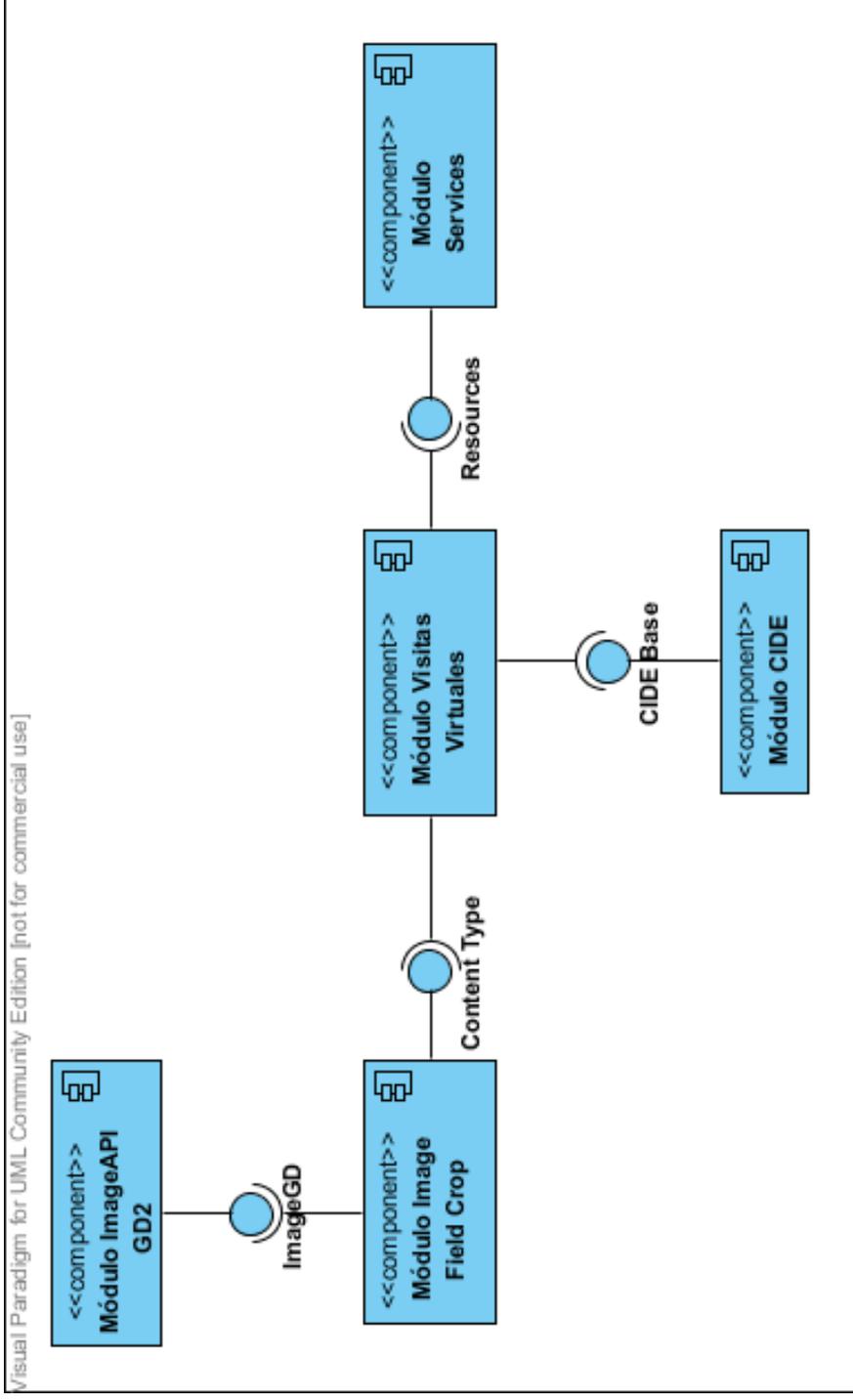
3.3 Diagrama de Entidad Relación



3.4 Diagrama de Clases del Aplicativo Móvil



3.5 Diagrama de Componentes del Módulo Web

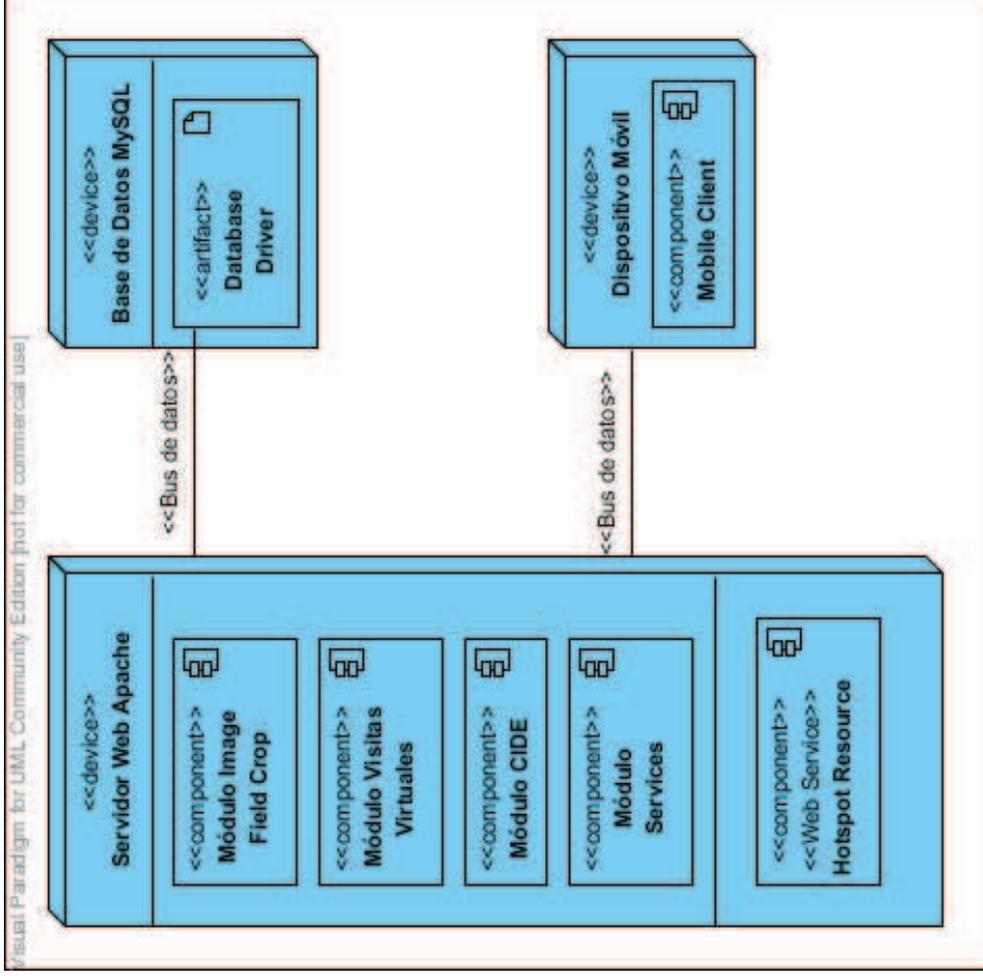


El presente diagrama de componentes muestra la interacción entre los módulos principales del sistema Web de Visitas Virtuales y las dependencias entre ellos. El módulo principal Visitas Virtuales depende directamente de tres diferentes componentes.

En primer lugar depende del componente Image Field Crop, el cual implementa toda la funcionalidad referente al procesamiento de imágenes panorámicas. En segundo lugar tiene dependencia sobre el módulo Services, el cual permite exponer servicios web de tipo REST dentro del CMS Drupal, esta funcionalidad es utilizada dentro del proyecto para el traslado de información acerca de los hotspots configurados dentro de las visitas virtuales. En tercer lugar el módulo de Visitas Virtuales depende del componente CIDE BASE, módulo que contiene un conjunto de funciones utilitarias usadas a lo largo del presente proyecto.

Por último y como se puede observar en el diagrama, el módulo Image Field Crop tiene una dependencia sobre el componente Image API GD2, el cual integra Drupal con la librería de PHP GD2, la cual permite trabajar con archivos de imágenes en forma programática.

3.6 Diagrama de Despliegue



El presente diagrama de despliegue muestra cómo se encuentra distribuido el sistema en forma física y en qué parte irán cada uno de los componentes que lo conforman. Consta de tres partes principales, el servidor Web Apache, la base de datos MySQL y un cliente móvil.

Dentro del servidor irá el módulo principal de visitas virtuales conjuntamente con todas las dependencias necesarias para su correcto funcionamiento. También irá un módulo de servicios web que permite exponer información a clientes externos al servidor.

El sistema web almacenará la información en una base de datos MySQL, comunicándose a la misma a través de un bus de datos, mismo que puede ser una conexión física directa, una conexión a través de internet u otro medio que crea conveniente el administrador del sistema. Todos los detalles de conectividad como protocolos, puertos o seguridad son manejados por el driver de conexión con la base de datos, componente que está presente dentro del servidor de base de datos.

El cliente móvil obtiene información del sistema web a través de un bus de datos, que en la práctica generalmente es representado por una conexión a internet. A través de este bus, el cliente móvil se comunica con los servicios web expuestos por el servidor, procesa los datos obtenidos y finalmente los muestra al usuario del aplicativo móvil.

CAPÍTULO 4: DESARROLLO

4.1 Desarrollo de Módulos del Sistema

El código fuente del módulo web tanto como el aplicativo móvil se encuentran adjuntos en el CD que acompaña al presente trabajo de tesis.

Manual de instalación y usuario se pueden encontrar en la sección de anexos.

4.2 Descripción del API del Módulo Web

Los siguientes puntos están dirigidos a desarrolladores de sitios web que deseen realizar mantenimiento o aumentar funcionalidad al módulo web del sistema de información. Ofrece una introducción al uso de las dependencias y funciones principales disponibles dentro del sistema.

4.2.1 Estándares de Codificación

Los estándares de codificación utilizados para el desarrollo del módulo web son los mismos usados por la comunidad de Drupal en el desarrollo de sus módulos. Mismos que se pueden encontrar en el siguiente sitio web <http://drupal.org/coding-standards#functdecl>.

Estos estándares definen aspectos básicos acerca de cómo se debe estructurar el código dentro de un módulo en Drupal, tales como espaciados, operadores, estructuras de control, argumentos a funciones, comentarios, entre otros.

Por otro lado para el diseño de la base de datos, también se tuvo en cuenta las convenciones de diseño utilizadas en la comunidad de Drupal, las cuales se pueden encontrar en el siguiente enlace <http://drupal.org/node/2497>.

Estas convenciones definen como se debe nombrar las tablas, los tipos de datos recomendados y como ejecutar código SQL dentro de un módulo Drupal de tal forma que se eviten posibles agujeros de seguridad.

4.2.2 Descripción de Módulos

El módulo web del sistema de información tiene dependencias en módulos previamente creados y compartidos a la comunidad Drupal para su uso, mismos que pueden ser encontrados y descargados en la siguiente dirección <http://drupal.org/project/modules>.

Nombre del Módulo: CIDE Base	Versión: 1.0
Descripción: Este módulo contiene un conjunto de funciones utilitarias que proveen funcionalidad comúnmente utilizada en todos los módulos, como por ejemplo subir y guardar archivos, consultas a la base de datos, validaciones, entre otros.	

Nombre del Módulo: Chaos tool suite (ctools)	Versión: 6.x-1.9
<p>Descripción: Este módulo representa un conjunto de APIs y herramientas para mejorar la experiencia de desarrollo sobre Drupal. También contiene un sub módulo muy importante para manejar paneles y páginas. Por el momento incluye las siguientes herramientas: -Plugins, exportables, funciones AJAX, herramientas para trabajo con forms, caché de objetos, diálogos modales, herramientas para CSS.</p>	

Nombre del Módulo: File Field	Versión: 6.x-3.10
<p>Descripción: Provee de un widget para subir archivos universal basado en CCK. Es una alternativa robusta al módulo Upload que viene con el core de Drupal y es indispensable para usuarios que quieran trabajar con un largo numero de archivos.</p>	

Nombre del Módulo: Content Construction Kit CCK	Versión: 6.x-2.9
<p>Descripción: Este módulo permite crear tipos de contenidos personalizados, es decir en terminología Drupal, nos ayuda con la creación de "Content Types", permite añadir nuevos campos a los nodos y definir el formulario que se utilizará para generar el tipo de contenido.</p>	

Nombre del Módulo: Imagefield Crop	Versión: 6.x-1.0
<p>Descripción: Este módulo provee de un widget para que el usuario pueda recortar una parte de la imagen luego de haberla subido al servidor. La imagen se presenta dentro de un área para recortarla, en la cual existe un recuadro que el usuario puede redimensionar y mover.</p>	

Nombre del Módulo: Image API	Versión: 6.x-1.10
Descripción: Este módulo provee de un conjunto de funciones adicionales a las del core de Drupal, con las que se procesará las imágenes, soporta las librerías de PHP GD2 e ImageMagick.	

Nombre del Módulo: Services	Versión: 6.x-3.1
Descripción: Este módulo provee una solución estandarizada para integrar aplicaciones externas con Drupal. Soporta servicios web realizados en las siguientes interfaces: REST, XMLRPC, JSON, JSON-RPC, SOAP, AMF, etc.	

4.2.3 Descripción de Módulo de Visitas Virtuales

4.2.3.1 Listado de Archivos

A continuación se muestran todos los archivos que conforman el módulo de visitas virtuales, conjuntamente con una descripción del mismo.

- **hotspot_resource.inc:** Implementa las funciones que serán utilizadas para enviar la información de los hotspots a través de un servicio REST.
- **visitas_virtuales.info:** Archivo básico de creación de módulos que describe la versión, archivos y dependencias hacia terceros que tiene el módulo de Visitas Virtuales.

- **visitas_virtuales.install:** Contiene funciones que serán llamadas al momento de instalar el módulo de Visitas Virtuales desde el sitio web, como por ejemplo creación de tablas y creación de tipos de contenidos especiales.
- **visitas_virtuales.module:** Contiene toda la implementación de funcionalidad del módulo de visitas virtuales, hooks, declaración de formularios, plantillas, llamadas AJAX, entre otros.
- **/includes/admin_panoramas.inc:** Contiene funciones de manejo de archivos panorámicos e integración con servicios externos.
- **visitas-virtuales-configuracion-form.tpl.php:** Plantilla de página personalizada para la página de configuración de visitas virtuales.
- **visitas-virtuales-creacion-form.tpl.php:** Plantilla de página personalizada para la página de creación de visitas virtuales.
- **visitas-virtuales-galeria-form.tpl.php:** Plantilla de página personalizada para la página de galería de visitas virtuales.
- **visitas-virtuales-visor-form.tpl.php:** Plantilla de página personalizada para la página de visualización de visitas virtuales.

4.2.3.2 Descripción de Funciones

hotspot_resource.inc

- `_visitas_virtuales_get_hotspots()`

Descripción: Función que obtiene los hotspots del sistemas basándose en los valores de localización y rango enviados por el cliente en su petición al servidor.

Valor de Retorno: Retorna un arreglo con todos los hotspots que calificaron de acuerdo al criterio de búsqueda

- `_visitas_virtuales_resource_access($op = 'view', $args = array())`

Descripción: Determina si el cliente que realiza la petición al servicio web del módulo tiene acceso a los servicios.

Parámetros:

- `$op`: La operación que va a ser ejecutada sobre el servicio web
- `$args`: Argumentos adicionales que se pasan a la función

Valor de Retorno: Retorna un valor booleano que representa si el acceso es otorgado.

visitas_virtuales.install

- `visitas_virtuales_schema()`

Descripción: Función que define las tablas, columnas y restricciones que serán creadas conjuntamente con la instalación del módulo de Visitas Virtuales.

Valor de Retorno: Retorna un arreglo con la definición del esquema a crear en la base de datos.

- visitas_virtuales_install()

Descripción: Función llamada al momento de instalar el módulo de Visitas Virtuales, permite ejecutar código personalizado por parte del módulo en este instante.

- visitas_virtuales_uninstall()

Descripción: Función llamada al momento de desinstalar el módulo de Visitas Virtuales, permite ejecutar código personalizado por parte del módulo en este instante.

visitas_virtuales.module

- form_visita_virtual_creacion_nueva()

Descripción: Función callback que se llama cuando el usuario hace clic en el botón de crear nueva visita virtual, valida que se pueda crear una nueva visita virtual para visualizar en el sistema.

- form_visita_virtual_creacion_eliminar()

Descripción: Función callback que se llama cuando el usuario hace click en el botón de eliminar visita virtual, se encarga de remover la visita virtual de memoria y de la base de datos.

ESPACIO EN BLANCO INTENCIONAL

- create_panorama_content_html(\$panorama)

Descripción: Crea el contenido informativo HTML que irá en el visor de visitas virtuales, utilizando los datos del panorama seleccionado.

Parámetros:

- \$panorama: El panorama del cual se creará el HTML de información

Valor de Retorno: Retorna una variable con el contenido HTML creado.

- form_panorama_info_callback()

Descripción: Función ejecutada en los callbacks AJAX que realiza el formulario de información del panorama, ubica los nuevos valores del formulario en el panorama seleccionado.

- form_hotspot_info_callback()

Descripción: Función ejecutada en los callbacks AJAX que realiza el formulario de información de hotspots, ubica los nuevos valores del formulario en el hotspot seleccionado.

- panorama_thumbnail_callback()

Descripción: Función ejecutada cuando el usuario desea utilizar un panorama como thumbnail de la visita virtual, esta función deselecciona cualquier thumbnail previamente usado, y deja seleccionado el nuevo valor.

- panorama_selected_callback(\$mode)

Descripción: Esta función utiliza los datos de localización enviados por el mapa del módulo de visitas virtuales, si encuentra un panorama previamente creado con esas coordenadas lo selecciona, caso contrario crea uno nuevo con esos datos.

Parámetros:

- \$mode: Modo en el cual se ha seleccionado un marcador, puede ser 'configuración' o 'creación'

Valor de Retorno: Los datos del panorama seleccionado o creado en formato JSON para su consumo por el cliente.

- panorama_deleted_callback()

Descripción: Función ejecutada cuando el usuario hace clic en el botón de eliminar panorama, quita el panorama seleccionado de memoria y lo ubica en un arreglo especial de panoramas que se eliminarán cuando se guarde la visita virtual.

- current_hotspot_callback()

Descripción: Callback utilizado por las funciones JavaScript de configuración de visitas virtuales.

Valor de Retorno: Devuelve el hotspot seleccionado actualmente en formato JSON para su consumo por el cliente.

– hotspot_deleted_callback()

Descripción: Función ejecutada cuando el usuario hace clic en el botón de eliminar hotspot, quita el hotspot seleccionado de memoria y lo ubica en un array especial de hotspot que se eliminarán cuando se guarde la visita virtual.

– visita_virtual_selected_callback()

Descripción: Función ejecutada cuando el usuario selecciona una nueva visita virtual, carga los datos de todos sus panoramas relacionados y los ubica en memoria.

– panoramas_json_callback()

Descripción: Callback utilizado por las funciones javascript de creación y configuración de visitas virtuales.

Valor de Retorno: Devuelve una cadena de caracteres con los datos de todos los panoramas en formato JSON relacionadas a la visita virtual seleccionada actualmente.

– hotspot_json_callback()

Descripción: Callback utilizado por las funciones javascript de configuración de visitas virtuales.

Valor de Retorno: Devuelve una cadena de caracteres con los datos de todos los hotspot en formato JSON relacionados al panorama seleccionado actualmente.

– create_visita_virtual()

Descripción: Crea una nueva visita virtual con valores por defecto.

Valor de Retorno: Devuelve un arreglo con valores por defecto de una visita virtual.

– create_hotspot(\$name,\$ath,\$atv,\$fov)

Descripción: Crea un arreglo representando a un hotspot, con su coordenada geográfica calculada basándose en el panorama seleccionado actualmente y los valores enviados a la función.

Parámetros:

- \$name: Nombre identificador único del hotspot.
- \$ath: Posición horizontal esférica donde se encuentra el hotspot dentro de la imagen panorámica.
- \$atv: Posición vertical esférica donde se encuentra el hotspot dentro de la imagen panorámica.
- \$fov: Campo de visión en el momento que se creó el hotspot con el cual se calcula distancia del origen del panorama.

Valor de Retorno: Devuelve un arreglo con valores por defecto y calculados para el hotspot.

ESPACIO EN BLANCO INTENCIONAL

– create_panorama(\$center)

Descripción: Crea un arreglo con valores por defecto de un panorama y lo asocia a un arreglo con su localización geográfica.

Parámetros:

- \$center: Arreglo que contiene los valores de localización geográfica, bajo el nombre de latitud y longitud.

Valor de Retorno: Devuelve un arreglo con valores por defecto y localización geográfica de un panorama.

– find_visita_virtual(\$name)

Descripción: Busca una visita virtual de entre las existentes en memoria basándose en el nombre enviado, si la encuentra, ubica el índice que le corresponde dentro del arreglo de visitas virtuales a una variable en sesión.

Parámetros:

- \$name: Nombre de la visita virtual.

– find_panorama(\$center)

Descripción: Busca un panorama de entre los existentes en memoria basándose en su centro geográfico, si lo encuentra, ubica el índice que le corresponde dentro del arreglo de panoramas a una variable en sesión.

Parámetros:

- \$center: Arreglo que contiene los valores geográficos de la visita virtual, latitud y longitud.

– find_hotspot(\$name)

Descripción: Busca un hotspot de entre los existentes en memoria basándose en su identificador único, si lo encuentra, ubica el índice que le corresponde dentro del arreglo de hotspot a una variable en sesión.

Parámetros:

– \$name: Nombre identificador único del hotspot.

– load_panoramas_for_visita_virtual(\$vid)

Descripción: Busca los panoramas existente en la base de datos para determinada visita virtual y los ubica en sesión.

Parámetros:

– \$vid: Llave primaria de la visita virtual en la base de datos.

– load_hotspot_for_panorama(\$pid)

Descripción: Busca los hotspots existente en la base de datos para determinado panorama y los ubica en sesión.

Parámetros:

– \$pid: Llave primaria del panorama en la base de datos.

– update_hotspot_db(\$hotspot)

Descripción: Actualiza un registro en la base de datos con los valores del hotspot enviado.

Parámetros:

– \$hotspot: Arreglo con los datos del hotspot que se quiere actualizar.

– save_hotspot_db(\$hotspot)

Descripción: Crea un nuevo registro en la base de datos con los valores del hotspot enviado.

Parámetros:

– \$hotspot: Arreglo con los datos del hotspot que se quiere guardar.

Valor de Retorno: Arreglo que contiene los valores de llave primaria que fueron creados para las tablas hotspot y coordinate.

– delete_hotspot_db(\$hotspot)

Descripción: Elimina el registro correspondiente al hotspot enviado de la base de datos.

Parámetros:

– \$hotspot: Arreglo con los datos del hotspot que se quiere eliminar.

– save_panorama_db(\$panorama)

Descripción: Crea un nuevo registro en la base de datos con los valores del panorama enviado.

Parámetros:

– \$panorama: Arreglo con los datos del panorama que se quiere guardar.

Valor de Retorno: Arreglo que contiene los valores de llave primaria que fueron creados para las tablas panorama y coordinate.

– update_panorama_db(\$panorama)

Descripción: Actualiza un registro en la base de datos con los valores del panorama enviado.

Parámetros:

- \$panorama: Arreglo con los datos del panorama que se quiere actualizar.

– delete_panorama_db(\$panorama)

Descripción: Elimina el registro correspondiente al panorama enviado de la base de datos.

Parámetros:

- \$panorama: Arreglo con los datos del panorama que se quiere eliminar.

– create_xml_configuracion_visita()

Descripción: Genera el XML de configuración de visitas virtuales usado por el visor de Krpano utilizando todos los valores de panoramas y hotspots que se encuentran en sesión.

– create_xml_creacion_visita()

Descripción: Genera el XML de creación de visitas virtuales usado por el visor de Krpano utilizando todos los valores de panoramas y hotspots que se encuentran en sesión.

– create_xml_visor_visita()

Descripción: Genera el XML usado por el visor de Krpano para renderizar el tour virtual concluido, con sus respectivos panoramas y hotspots.

– create_file_path_visita_virtual(\$visita_virtual)

Descripción: Genera la ruta física dentro del servidor que será utilizada para guardar los archivos XML de creación, configuración y visualización para cada visita virtual.

Parámetros:

– \$visita_virtual: Arreglo con los datos de la visita virtual.

Valor de Retorno: Un cadena de caracteres representando la ruta del archivo de configuración.

– create_directory_path_visita_virtual(\$visita_virtual)

Descripción: Genera el directorio físico dentro del servidor que será utilizado para guardar los archivos XML de creación, configuración y visualización para cada visita virtual.

Parámetros:

– \$visita_virtual: Arreglo con los datos de la visita virtual.

Valor de Retorno: Una cadena de caracteres representando la ruta del directorio donde se guardarán los archivos de configuración.

- recursive_remove_directory(\$directory)

Descripción: Elimina recursivamente todos los archivos y subdirectorios de la ruta especificada

Parámetros:

- \$directory: Ruta del directorio a eliminar.

- calculate_bearing_between_points(\$first,\$second)

Descripción: Calcula la variación de grados entre dos coordenadas geográficas usando el polo norte como referencia.

Parámetros:

- \$first: Arreglo con los datos de latitud y longitud de la primera coordenada.
- \$second: Arreglo con los datos de latitud y longitud de la segunda coordenada.

Valor de Retorno: Valor de variación en grados entre los dos puntos.

- calculate_distance_between_points(\$first,\$second)

Descripción: Calcula la distancia física entre dos coordenadas geográficas.

Parámetros:

- \$first: Arreglo con los datos de latitud y longitud de la primera coordenada.
- \$second: Arreglo con los datos de latitud y longitud de la segunda coordenada.

Valor de Retorno: La distancia entre los puntos en kilómetros.

- DMS_to_DEC(\$deg,\$min,\$sec)

Descripción: Transforma coordenadas en notación GPS a notación decimal.

Parámetros:

- \$deg: Grados de la coordenada.
- \$min: Minutos de la coordenada.
- \$sec: Segundos de la coordenada.

Valor de Retorno: El valor decimal que representa al punto geográfico.

- DEC_to_DMS(\$dec,\$type)

Descripción: Transforma coordenadas decimales a notación GPS

Parámetros:

- \$dec: Valor decimal de la coordenada.
- \$type: Tipo de coordenada que se calculará, 'latitude' o 'longitude'.

Valor de Retorno: Un arreglo con los datos de la coordenada GPS, grados, minutos y segundos.

ESPACIO EN BLANCO INTENCIONAL

- calculate_krpano_distance(\$fov,\$max_distance)

Descripción: Calcula la distancia en metros, a la cual se agregó un hotspot dentro del visor de krpano, basándose en el centro del panorama y utilizando el valor fov enviado.

Parámetros:

- \$fov: Valor fov en el momento que se agregó un hotspot en el visor.
- \$max_distance: Distancia máxima en metros que se usará para realizar equivalencia con el fov máximo posible en el visor krpano.

Valor de Retorno: Distancia en metros a la que se encuentra el hotspot.

- calculate_coordinate_from_origin(\$lat1,\$lon1,\$degrees,\$lin_distance)

Descripción: Calcula una coordenada final, partiendo de un origen, conociendo los datos de variación en grados y distancia a la que se encuentra.

Parámetros:

- \$lat1: Latitud del origen en notación decimal.
- \$lat2: Latitud del destino en notación decimal.
- \$degrees: Variación en grados desde el polo norte.
- \$lin_distance: Distancia en metros a la que se encuentra el punto final.

Valor de Retorno: Un arreglo con los valores de latitud y longitud del nuevo arreglo.

4.3 Descripción del API del Aplicativo Móvil

4.3.1 Estándares de Codificación

Los estándares de codificación utilizados para el desarrollo del aplicativo móvil son los mismos que son usados para la programación en la API Cocoa⁵¹ creada por Apple Inc. Mismos que se pueden encontrar en el siguiente sitio web

<http://developer.apple.com/library/mac/#documentation/Cocoa/Conceptual/CodingGuidelines/CodingGuidelines.html>.

Estos estándares definen aspectos básicos acerca de cómo se debe estructurar el código dentro de una aplicación Cocoa utilizando el lenguaje de programación Objective-C. iOS en su API utiliza un subset de funciones provenientes de Cocoa, es por esta razón que se estas convenciones se utilizan como base para el desarrollo de aplicativos móviles.

Entre las convenciones utilizadas se tiene formas de nombrar los métodos, funciones, propiedades y tipos de datos sugeridos, y también abreviaciones o acrónimos que se puede llegar a usar dentro del código.

⁵¹ http://en.wikipedia.org/wiki/Cocoa_%28API%29

4.3.2 Listado de Clases

HotspotDataManager

Archivo Declaración: HotspotDataManager.h
Hereda de: NSObject
Protocolos Implementados: CLLocationManagerDelegate

Métodos:

-(void)reloadHotspots:(NSTimer *)timer
Tipo de Método: Instancia
Descripción: Método que se ejecuta cuando los hotspots han sido actualizados desde el internet.
Parámetros: <ul style="list-style-type: none">- timer: objeto de tipo NSTimer inició la actualización de hotspots.

-(void)stopUpdatingHotspots
Tipo de Método: Instancia
Descripción: Detiene la actualización periódica de hotspots.

ESPACIO EN BLANCO INTENCIONAL

-(void)startUpdatingHotspots:(float)timeInterval

Tipo de Método: Instancia

Descripción: Empieza la actualización de hotspots periódica creando un timer con el tiempo especificado.

Parámetros:

- timeInterval: Tiempo de intervalo de actualización de hotspots.

ARViewController

Archivo Declaración: ARViewController.h

Hereda de: UIViewController

Protocolos Implementados:

ARDelegate, ARMarkerDelegate, ARLocationDelegate

Métodos:

-(void)viewDidLoad

Tipo de Método: Instancia

Descripción: Método que inicializa la vista y registra al objeto como observador de notificaciones del sistema.

ESPACIO EN BLANCO INTENCIONAL

- (void)updateGeoLocations:(NSNotification *)notification
Tipo de Método: Instancia
Descripción: Actualiza las sub vistas del controlador con los datos actualizados de hotspots.
Parámetros: <ul style="list-style-type: none"> - notification: Objeto de tipo NSNotification con los datos de un evento generado por otra clase o el propio sistema.

Propiedades:

Nombre	Tipo	Descripción
arController	AugmentedRealityController	Controlador que muestra las imágenes que van siendo capturadas por la cámara en pantalla.
infoView	UIView	Vista que mostrará información adicional de los hotspots, cuando se haga tap sobre estos.

MarkerView

Archivo Declaración: MarkerView.h
Hereda de: UIView

ESPACIO EN BLANCO INTENCIONAL

Métodos:

- (id)initForCoordinate:(ARGeoCoordinate *)coordinate withDelegate:(id<ARMarkerDelegate>) aDelegate
Tipo de Método: Instancia
Descripción: Crea una vista de forma programática con los datos enviados en el parámetro coordinate.
Valor de Retorno: Objeto de tipo MarkerView creado con los parámetros enviados.

Propiedades:

Nombre	Tipo	Descripción
coordinateInfo	ARGeoCoordinate	Coordenada con los datos de localización geográfica del hotspots que son usados para crear la vista.
delegate	ARMarkerDelegate	Objeto delegado sobre el cual se ejecutarán métodos cuando se produzcan eventos en la vista.

ARGeoCoordinate

Archivo Declaración: ARGeoCoordinate.h
Hereda de: ARCoordinate

Métodos:

+ (ARGeoCoordinate *)coordinateWithLocation:(CLLocation *)location fromOrigin:(CLLocation *)origin
Tipo de Método: Clase
Descripción: Crea un objeto de tipo ARGeoCoordinate, calculando datos usados por la clase a partir de un origen y una localización geográfica.
Parámetros: <ul style="list-style-type: none">– location: Objeto de tipo CLLocation con los datos de latitud y longitud de la coordenada.– origin: Objeto de tipo CLLocation con los datos de latitud y longitud de la coordenada de origen.
Valor de Retorno: Objeto ARGeoCoordinate creado.

- (float)angleFromCoordinate:(CLLocationCoordinate2D)first toCoordinate:(CLLocationCoordinate2D)second
Tipo de Método: Instancia
Descripción: Calcula el ángulo, a partir del polo norte en el que se encuentra una coordenada, a partir del origen especificado.
Parámetros: <ul style="list-style-type: none">– first: Objeto de tipo CLLocationCoordinate2D con los datos de latitud y longitud de la coordenada.– second: Objeto de tipo CLLocationCoordinate2D con los datos de latitud y longitud de la coordenada de origen.
Valor de Retorno: Ángulo en grados calculado.

Propiedades:

Nombre	Tipo	Descripción
geoLocation	CLLocation	Coordenada con los datos de localización geográfica de la coordenada.
displayView	ARMarkerDelegate	Vista que se utilizará para mostrar los datos de la coordenada.

ARCoordinate

Archivo Declaración: ARCoordinate.h

Hereda de: NSObject

Métodos:

- (BOOL) isEqualToCoordinate:(ARCoordinate *) otherCoordinate

Tipo de Método: Instancia

Descripción: Compara la igualdad del objeto con el enviado en el parámetro.

Parámetros:

- otherCoordinate: Objeto contra el cual se realizará la comparación.

Valor de Retorno: Variable booleana indicando si los 2 objetos son iguales.

ESPACIO EN BLANCO INTENCIONAL

<p>+ (ARCoordinate*)coordinateWithRadialDistance: (double)newRadialDistance</p> <p>inclination:(double)newInclination azimuth:(double)newAzimuth</p>
<p>Tipo de Método: Clase</p>
<p>Descripción: Crea un nuevo objeto con los datos de distancia e inclinación enviados en los parámetros.</p>
<p>Parámetros:</p> <ul style="list-style-type: none"> – newRadialDistance: Distancia a la que se encuentra la coordenada desde un origen. – newInclination: Inclinación a la que se encuentra la coordenada. – azimuth: Variación de grados a la que se encuentra la coordenada desde un origen.
<p>Valor de Retorno: Objeto de tipo ARCoordinate creado.</p>

Propiedades:

Nombre	Tipo	Descripción
title	NSString	Texto que describe un título de la coordenada.
subtitle	NSString	Texto que describe un subtítulo para la coordenada.
radialDistance	double	Distancia radial de la coordenada a partir de un origen.
inclination	double	Inclinación de la coordenada.
azimuth	double	Variación en grados a la que se encuentra la coordenada desde un origen.

Hotspot

Archivo Declaración: Hotspot.h
Hereda de: NSObject
Protocolos Implementados: MKAnnotation

Métodos:

-(id)initWithTitle:(NSString *)title description : (NSString *) hotspotDescription latitude : (NSNumber *)latitude longitude : (NSNumber *)longitude;
Tipo de Método: Instancia
Descripción: Inicializa un nuevo objeto de tipo Hotspot con los parámetros enviados.
Parámetros: <ul style="list-style-type: none">- title: Título del Hotspot.- Description: Descripción del Hotspot.- Latitude: Latitud del Hotspot.- Longitude: Longitud del Hotspot.
Valor de Retorno: Objeto de tipo Hotspot inicializado.

Propiedades:

Nombre	Tipo	Descripción
title	NSString	Título de la coordenada.
hotspotDescription	NSString	Texto con detalles del hotspot.
imageUrl	NSString	Texto que contiene la URL de la cual se obtiene la imagen.

HotspotMapViewController

Archivo Declaración: HotspotMapViewController.h

Hereda de: UIViewController

Protocolos Implementados: MKMapViewDelegate

Métodos:

- (void)updateLocations:(NSNotification *)notification

Tipo de Método: Instancia

Descripción: Toma los hotspots que fueron actualizados desde internet y enviados a través de una notificación y los guarda dentro de una propiedad del objeto.

Parámetros:

- notification: Objeto de tipo NSNotification en el cual se encuentran los datos de los hotspots actualizados.

-(void)updateGeoLocations:(NSMutableArray *)hotspots

Tipo de Método: Instancia

Descripción: Quita los marcadores que existían previamente en el mapa y los actualiza de acuerdo a los nuevos hotspots que se obtuvieron desde internet.

Parámetros:

- hotspots: Arreglo con los objetos de tipo Hotspot actualizados.

-(void)mapView:(MKMapView*)mapView annotationView:(MKAnnotationView*)view calloutAccessoryControlTapped:(UIControl *)control
Tipo de Método: Instancia
Descripción: Muestra la vista de detalles de un hotspot cuando el mismo ha sido seleccionado.
Parámetros: <ul style="list-style-type: none"> - mapView: Vista de Mapa sobre el cual está ubicado el marcador. - view: Vista de detalle del marcador que se seleccionó. - control: Botón que fue seleccionado dentro del marcador.

4.4 Pruebas de Aceptación

Prueba de Aceptación	
ID	T0001
Descripción	El administrador agrega una visita virtual, la edita y guardar toda su información en la base de datos.
Inicialización	Eliminar todos los datos de la tabla visita_virtual en la base de datos.
Instrucciones	<ol style="list-style-type: none"> 1. Hacer clic en el botón Nueva Visita Virtual 2. En el campo de texto Nombre de la visita virtual, ingresar "Tour de Prueba" 3. En el campo de texto Descripción de la visita virtual, ingresar "Texto descriptivo de la visita virtual"

	<ol style="list-style-type: none"> 4. En el mapa hacer clic en un punto del mismo 5. En la caja de texto Nombre del panorama ingresar "Bar" 6. En la caja de texto Distancia ingresar el valor "100" 7. Escoger una selección en el campo Nombre del archivo de la foto panorámica 8. Hacer clic en el botón guardar visita virtual
<p>Resultados Esperados</p>	<p>Tabla visita_virtual</p> <ul style="list-style-type: none"> - Se debió haber creado un nuevo registro en esta tabla con el nombre de la visita y su descripción. <p>Tabla panorama</p> <ul style="list-style-type: none"> - Se debió haber creado un registro por cada panorama ubicado en el mapa, con toda la información ingresada en los campos de la pantalla. <p>Archivo XML</p> <ul style="list-style-type: none"> - Se debió haber generado 2 archivos XML dentro de la carpeta visitas dentro del módulo web. El primero con el nombre "nombre_visita_creacion.xml" y el segundo "nombre_visita_configuracion.xml". <p>Pre visualización Visita Virtual</p> <ul style="list-style-type: none"> - Se debió haber mostrado cada foto panorámica escogida y 1 hotspots por cada panorama dentro de la visita virtual para su navegación.

Prueba de Aceptación	
ID	T0002
Descripción	El administrador sube una nueva foto panorámica y esta se guarda en el sistema para su utilización.
Inicialización	Eliminar todos los tipos de contenido panorama existentes.
Instrucciones	<ol style="list-style-type: none"> 1. Hacer clic en el link "Create Content" 2. Seleccionar la opción Panorama 3. En el campo Imagen Panorámico seleccionar la imagen que se va a subir. 4. Hacer clic en el botón guardar 5. Ir a la página de creación de visitas virtuales 6. Hacer clic en el botón Nueva Visita Virtual 7. Agregar un panorama haciendo clic en cualquier punto del mapa 8. Seleccionar el archivo de imagen panorama que fue subido en la lista desplegada en el campo Nombre del archivo de la foto panorámica.
Resultados Esperados	El nombre del archivo que fue subido en el tipo de contenido panorama, aparece en la lista de selección "Nombre del archivo de la foto panorámica" para su utilización.

Prueba de Aceptación	
ID	T0003
Descripción	El administrador agrega un panorama a la visita virtual geo localizando el punto donde lo ubicó.
Inicialización	Eliminar todos los registros de la tabla panorama.
Instrucciones	<ol style="list-style-type: none"> 1. Crear una nueva visita virtual 2. Agregar un marcador al mapa 3. En el campo Imagen Panorámico seleccionar la imagen que se va a subir. 4. Hacer clic en el botón guardar 5. Ir a la página de creación de visitas virtuales 6. Hacer clic en el botón Nueva Visita Virtual 7. Agregar un panorama haciendo clic en cualquier punto del mapa 8. Seleccionar el archivo de imagen panorama que fue subido en la lista desplegada en el campo Nombre del archivo de la foto panorámica.
Resultados Esperados	El nombre del archivo que fue subido en el tipo de contenido panorama, aparece en la lista de selección "Nombre del archivo de la foto panorámica" para su utilización.

ESPACIO EN BLANCO INTENCIONAL

Prueba de Aceptación	
ID	T0004
Descripción	El administrador hace clic dentro del mapa y aparece icono de marcador en un punto seleccionado.
Instrucciones	<ol style="list-style-type: none"> 1. Crear una nueva visita virtual 2. En el mapa hacer clic en cualquier punto
Resultados Esperados	Un icono de marcador habrá aparecido en el punto donde se hizo clic representado el panorama que ingresará el usuario.

Prueba de Aceptación	
ID	T0005
Descripción	El usuario ve el tour virtual desde una máquina y desde su iPhone
Inicialización	<p>Crear una nueva visita virtual</p> <p>Configurar una visita virtual</p>
Instrucciones	<ol style="list-style-type: none"> 1. Acceder a la opción de galerías de visitas virtuales desde un browser 2. Escoger algunas de las visitas virtuales mostradas en la galería 3. Copiar el URL actual de la ventana del navegador 4. Acceder al navegador del iPhone (Safari) y poner en la dirección la URL copiada 5. Navegar a través de la visita virtual
Resultados Esperados	En ambos casos se podrá navegar a través de la visita virtual y será visible en la misma forma desde el browser y desde el iPhone.

Prueba de Aceptación	
ID	T0006
Descripción	Se crean archivos XML para el visor de Krpano
Inicialización	Crear una nueva visita virtual Configurar una visita virtual
Instrucciones	<ol style="list-style-type: none"> 1. Acceder al directorio donde se encuentra instalada el módulo de visitas virtuales 2. Entrar en la carpeta visitas 3. Entrar a una carpeta existente dentro del directorio visitas 4. Verificar la existencia de 3 archivos XML y que sean modificables.
Resultados Esperados	Dentro del módulo de visitas virtuales en la carpeta visitas se creara una carpeta por cada visita virtual existente en el sistema y dentro de ella abran 3 archivos XML, cada uno con los sufijos, _creacion.xml, _configuracion.xml y _visor.xml.

Prueba de Aceptación	
ID	T0007
Descripción	El usuario se autentica e ingresa al módulo de visitas virtuales
Instrucciones	<ol style="list-style-type: none"> 1. Acceder a la página principal del portal web 2. Ingresar su usuario y contraseña 3. Si tiene permisos de administrador podrá ver una opción llamada Visitas Virtuales

	<ol style="list-style-type: none"> 4. Ingresar a dicha opción 5. Hacer clic en el tab de Configuración Visitas Virtuales 6. Hacer clic en el tab de Galeria de Visitas Virtuales 7. Hacer clic en el tab de Creación de Visitas Virtuales
Resultados Esperados	En la pantalla principal del módulo de visitas virtuales, aparecen 1 tab por cada una de las opciones ofrecidas por el sistema. Hacer clic en cada una cargará la página correspondiente

Prueba de Aceptación	
ID	T0008
Descripción	El administrador reubica los marcadores del mapa
Instrucciones	<ol style="list-style-type: none"> 1. Crear una nueva visita virtual 2. En el mapa hacer clic en cualquier punto 3. Hacer clic sobre el botón Guardar Visita Virtual 4. Hacer clic sobre el marcador agregado y arrastrarlo a una nueva ubicación 5. Agregar otro marcador haciendo clic en el mapa 6. Arrastrar el nuevo marcador a la posición deseada
Resultados Esperados	<p>Tabla Coordinate</p> <ul style="list-style-type: none"> - Se habrá creado un registro por cada marcador ubicado en el panorama, la latitud y longitud registrada en la base de datos será la posición final en la que se encuentran los marcadores, inclusive luego de arrastrarlos

Prueba de Aceptación	
ID	T0009
Descripción	El administrador configura una visita virtual.
Inicialización	<p>Eliminar todos los datos de la tabla visita_virtual en la base de datos.</p> <p>Crear una nueva visita virtual.</p>
Instrucciones	<ol style="list-style-type: none"> 1. Seleccionar el tab Configuración Visitas Virtuales 2. Seleccionar una visitas virtual de la lista desplegable campo Nombre de la visita virtual: 3. Hacer clic en cualquier marcador dentro del mapa 4. Arrastrar un marcador desde el control existente en la parte superior derecha del visor de visitas virtuales a una posición dentro del visor 5. En la caja de texto Titulo del hotspots ingresar "Hotspot de prueba" 6. En la caja de texto Descripcion hotspots ingresar "Texto Descriptivo del hotspot" 7. Hacer clic en el botón Guardar Configuración Visita Virtual
Resultados Esperados	<p>Tabla hotspot</p> <ul style="list-style-type: none"> - Se habrá creado un registro por cada hotspot ingresado por el administrador, con toda la información referente al mismo. <p>Tabla coordinate</p>

	<ul style="list-style-type: none"> - Se habrá creado un registro por cada hotspots ingresado por el administrador, que representa la posición geográfica del hotspots referente al panorama en el cual se lo ubicó. <p>Archivo XML</p> <ul style="list-style-type: none"> - El archivo final XML con el sufijo _visor.xml debió haberse generado en la carpeta visitas dentro del módulo de visitas virtuales
--	---

Prueba de Aceptación	
ID	T0010
Descripción	El administrador sube una nueva foto panorámica y escoge una parte para usarla como thumbnail.
Inicialización	Eliminar todos los tipos de contenido panorama existentes.
Instrucciones	<ol style="list-style-type: none"> 1. Hacer clic en el link "Create Content" 2. Seleccionar la opción Panorama 3. En el campo Imagen Panoramico seleccionar la imagen que se va a subir. 4. Hacer clic en el botón Upload 5. Redimensionar el cuadro de selección de thumbnail al tamaño deseado 6. Hacer clic en el botón Guardar
Resultados Esperados	Se procesa la foto panorámica y se crea un nuevo archivo de imagen jpeg, que contiene la selección realizada.

Prueba de Aceptación	
ID	T0011
Descripción	El administrador sube una imagen para un hotspot.
Inicialización	<p>Eliminar todos los datos de la tabla visita_virtual en la base de datos.</p> <p>Crear una nueva visita virtual.</p>
Instrucciones	<ol style="list-style-type: none"> 1. Seleccionar el tab Configuración Visitas Virtuales 2. Seleccionar una visitas virtual de la lista desplegable campo Nombre de la visita virtual 3. Hacer clic en cualquier marcador dentro del mapa 4. Arrastrar un marcador desde el control existente en la parte superior derecha del visor de visitas virtuales a una posición dentro del visor 5. En la caja de texto Imagen el archivo de imagen que se quiere subir y hacer clic en el botón subir imagen
Resultados Esperados	<ul style="list-style-type: none"> - Se creará un nuevo archivo en la estructura pública de Drupal con los contenidos de la imagen subida por el usuario. <p>Tabla hotspot</p> <ul style="list-style-type: none"> - En el registro correspondiente al hotspot de esta tabla, se habrá agregado una referencia a la ruta donde se encuentra el archivo que fue subido al servidor.

Prueba de Aceptación	
ID	T0013
Descripción	El usuario escoge el campo de visión del dispositivo móvil.
Instrucciones	<ol style="list-style-type: none"> 1. Ir a la sección settings de la aplicación 2. Seleccionar la opción de campo de visión 3. Mover el Slider a un distancia de 100 metros 4. Abrir el aplicativo
Resultados Esperados	El ajuste de campo de visión tendrá como resultado que los hotspots que se encuentren a más de 100 metros del dispositivo móvil ya no se podrán visualizar.

Prueba de Aceptación	
ID	T0014
Descripción	Visualizar hotspots a través de la cámara
Instrucciones	<ol style="list-style-type: none"> 1. Abrir el aplicativo móvil 2. Seleccionar el tab de vista AR 3. Movilizarse a un punto dentro del campo de visión de hotspots 4. Seleccionar el recuadro mostrado en la pantalla
Resultados Esperados	Conforme el dispositivo se acerca a la ubicación física del hotspot un recuadro se le muestra con el nombre del mismo y la distancia a la que se encuentra, el hacer clic sobre este recuadro muestra más información del hotspot en una nueva pantalla.

Prueba de Aceptación	
ID	T0015
Descripción	Visualizar hotspots a través del mapa
Instrucciones	<ol style="list-style-type: none"> 1. Abrir el aplicativo móvil 2. Seleccionar el tab de mapas 3. Esperar a que la ubicación del dispositivo sea encontrada totalmente a través del GPS 4. Seleccionar uno de los marcadores que se agregaron a la pantalla 5. Hacer clic en el botón derecho que apareció al seleccionar el marcador
Resultados Esperados	Se mostrará la ubicación de cada hotspot cercano al dispositivo dentro del mapa, al hacer clic sobre ellos se muestra más información del hotspot en una nueva pantalla.

CAPÍTULO 5

5.1 CONCLUSIONES

Una vez finalizado el desarrollo del módulo web tanto como el aplicativo móvil para la administración y visualización de visitas virtuales, se pudo observar que los objetivos planteados en el presente proyecto de tesis se cumplieron con éxito, con lo que se pudo llegar a las siguientes conclusiones.

- La utilización de recorridos virtuales es ideal para mostrar información institucional a los usuarios interesados en conocer o visitar el campus gracias a la interactividad e inmersión que ofrecen.
- La utilización del CMS Drupal, reduce significativamente el tiempo y esfuerzo requeridos para implementar un nuevo módulo, ya que libera al desarrollador de detalles adicionales a la lógica de negocio como son seguridad, estructura y administración del sitio web.
- El software visor de imágenes panorámicas Krpano, agiliza la construcción de recorridos virtuales, al ofrecer una interfaz común de desarrollo tanto para dispositivos móviles como para browsers y computadores de escritorio.

- Krpano ofrece un conjunto de herramientas adicional muy útiles para el procesamiento de imágenes panorámicas, con las cuales es posible generar recorridos virtuales muy rápidamente.
- La incorporación de geo localización ofrece la base sobre la cual se pueden integrar tecnologías como son los recorridos virtuales y realidad aumentada.
- El desarrollo en las plataformas móviles iOS y Android, es realmente muy rápido, ya que sus herramientas liberan al desarrollador de implementar detalles como la interfaz gráfica y concentrarse completamente en la lógica de negocio.
- La utilización de realidad aumentada es una herramienta invaluable para que los usuarios puedan conocer un posible punto de interés ya que proporciona la ubicación exacta y distancia a la que se encuentran dichos lugares.
- La utilización de un servicio de mapas como Google Maps es fundamental para incrementar la experiencia de usuario, gracias a fotos satelitales, identificación de rutas y geo localización que este ofrece.
- El uso de una herramienta CASE es fundamental en un proyecto que utilice metodologías de desarrollo ágil, ya que permitirá hacer

modificaciones sobre los diagramas del sistema de manera sencilla y sin necesidad de volver a dibujar o reordenar los mismos.

- La base de datos diseñada probó ser efectiva para el almacenamiento de información, sin necesidad de relaciones complejas ni procedimientos adicionales dentro del motor de base de datos.

5.2 RECOMENDACIONES

- A pesar de la excelente documentación que posee Drupal para el desarrollo de módulos, su curva de aprendizaje es un poco elevada, por lo que se aconseja estar familiarizado con la administración de este CMS y tener conocimientos en el lenguaje de programación PHP para poder empezar con el desarrollo de un módulo.
- Probar el aplicativo móvil finalizado en un dispositivo real es fundamental para su correcto funcionamiento, ya que el simulador aunque representa una gran ayuda, no brinda una simulación exacta de la memoria y procesador del dispositivo.
- Aunque existe la posibilidad de mostrar imágenes panorámicas en formatos sin procesamiento previo, es altamente recomendado optimizarlas a través del uso de herramientas externas, esto debido a que el tiempo de descarga y el uso de recursos en los browsers de los clientes puede reducir la velocidad del aplicativo considerablemente.
- Mostrar mucha información dentro de un recorrido virtual puede provocar confusión entre los usuarios del sistema, por lo que se recomienda tratar de simplificar la información expuesta con el uso de hipervínculos o contenido multimedia como fotos y videos.
- La utilización de la base de datos MySQL es ampliamente recomendada para un sistema que no tiene un manejo de

transacciones crítico, como por ejemplo un sistema de información, esto gracias a que es muy liviano, confiable y tiene soporte para la gran mayoría de lenguajes de programación.

- La utilización de una herramienta CASE compatible con UML 2.0 ahorra mucho tiempo de diseño al incorporar estereotipos y relaciones que sería necesario dibujarlas manualmente en otros casos.

BIBLIOGRAFÍA

- <http://www.reynardthomson.com/what-is-prototyping.html>
- http://en.wikipedia.org/wiki/Software_prototyping
- <http://peekwebdesign.com.au/cms-made-easy/>
- Mercer, David. Drupal 7. 2010, Packt Publishing
- http://www.ent.iastate.edu/it/Intro_to_Drupal_7_Architecture.pdf
- <http://www.missingubercartmanual.com/How-to-Create-a-Basic-Drupal-Module>
- Butcher, M., Garfield, L., Wilkins, J., Farina, M., Rickard, K. & Dunlap, G. Drupal 7 Module Development. 2010, Packt Publishing
- http://en.wikipedia.org/wiki/Virtual_tour#Applications
- <http://www.panoguide.com/howto/panoramas/types.jsp>
- http://en.wikipedia.org/wiki/Image_stitching
- <http://www.krpano.com>
- Roche, Kyle. Pro iOS 5 Augmented Reality. 2011, Apress.
- <http://www.pocket-lint.com/news/38795/what-is-augmented-reality-ar>
- <http://answers.oreilly.com/topic/1751-mobile-accelerometers-and-gyroscopes-explained/>

- <http://www.math.ksu.edu/~dbski/writings/haversine.pdf>
- <http://www.movable-type.co.uk/scripts/latlong.html>
- <http://www.invasivecode.com/blog/archives/1435>
- <https://developer.apple.com/technologies/tools/>
- <https://developer.apple.com/programs/which-program/>
- <http://developer.android.com/tools/index.html>